

# **Animerede spørgeskemaer for sikkerhedsbevidsthed**

Theo Andersen

Kongens Lyngby 2007

Technical University of Denmark  
Informatics and Mathematical Modelling  
Building 321, DK-2800 Kongens Lyngby, Denmark  
Phone +45 45253351, Fax +45 45882673  
[reception@imm.dtu.dk](mailto:reception@imm.dtu.dk)  
[www.imm.dtu.dk](http://www.imm.dtu.dk)

# Resumé

---

Brugen af simple tekstbaserede spørgeskemaer til undersøgelse af folks viden ved komplekse tekniske emner er problemfyldt, da folk ofte simpelthen ikke forstår emnerne beskrevet i spørgsmålene. Derfor kan det i disse tilfælde muligvis kun være muligt at udtrække folks viden ved at bruge forbedrede spørgeskemaer som præsenterer dem for de praktiske scenarier på den måde de er vant til. Dette emne bliver i denne rapport undersøgt med fokus på IT sikkerhedsbevidsthed.

Idet en computer abstraherer dens indre teknikker grafisk for brugeren, har spørgeskemaer inden for sikkerhedsbevidsthed det samme problem. Løsningen er muligvis at benytte animerede spørgeskemaer, hvilket præsenterer brugerne for den samme grafiske abstraktion som vist på computeren.

I denne rapport bliver IT sikkerhed samt den nuværende metodik inden for spørgeskemaer undersøgt for at beskrive forbedrede metodiker til udvikling af spørgeskemaer. Derved undersøges det endvidere hvorledes en realisering af animerede spørgeskemaer i programmet Adobe Captivate kan benyttes, og fra dette beskrives udviklingen af et sikkerhedssite, hvorved brugernes reaktioner til de animerede spørgeskemaer samt sikkerhedsbevidsthed er analyseret. Til sidst beskrives optimering af spørgeskemaudvikling ved hjælp af spørgeskemastyring.

De afsluttende bemærkninger konkluderer at animerede spørgeskemaer gør det muligt at opfange brugernes reaktioner meget bedre end de simple, da de fremviser problemstillingen på en måde brugere direkte kan sammenligne med deres egne erfaringer. Desuden er Captivate blevet fundet til at være et udmærket værktøj til konstruktion af animerede spørgeskemaer, hvis man udelukkende benytter det til den animerede del af spørgeskemaerne.



# Abstract

---

The use of simple questionnaire for investigating people's knowledge relating to technical topics is problematic, as the general public often simply does not understand the topics described. It may therefore only be possible to extract people's knowledge by using enhanced questionnaires, which present them to the practical scenario's in the way their used to. This subject is in this report investigated with focus on IT security awareness.

Being that a computer abstracts its inner techniques graphically for the user; questionnaires for security awareness have the same problems. The solution may possibly be to use animated questionnaires, which present the users to the same graphical abstraction as shown on their computers.

In this report IT security and present methodology for questioners is examined to describe enhanced methodologies for developing questionnaires. It's thereby also investigated how a possible realization of animated questionnaires in the program Adobe Captivate is usable and from this the development of a security website is described, in which the user's reactions are analyzed. Lastly the optimization of questionnaire development is described, by using questionnaire management.

The finishing statements conclude that animated questionnaires does make it possible to record the users reactions much better than the simple, because they display the issue in a way the user can compare directly to their own experiences. In addition Captivate has been found to be a fairly good tool to the construction of animated questionnaires, if one uses it exclusively for the animated part of the questionnaires.



# Forord

---

Dette eksamensprojekt er blevet udarbejdet hos instituttet for Informatik og Matematisk Modellering, ved Danmarks Tekniske Universitet som del af et større projekt i hvilket IMM arbejder i samarbejde med DK-CERT og DPU, for at undersøge IT sikkerhedsbevidstheden blandt den generelle befolkning i Danmark.

Mange tak til min vejleder Robin Sharp for hans hjælp med dette projekt og til CERT for deres hjælp med Captivate spørgeskemaerne. Specielt mange tak til min kæreste Ciara, for at være tålmodig nok til at lytte til animerede spørgeskemaer gennem de sidste fem måneder.

Charlottenlund, Juli 2007

Theo Andersen, s002025





# Indhold

---

<b>Resumé</b>	<b>i</b>
<b>Abstract</b>	<b>iii</b>
<b>Forord</b>	<b>v</b>
<b>1 Indledning</b>	<b>1</b>
1.1 Problemformulering . . . . .	2
1.2 Rapporten . . . . .	2
<b>2 Computer sikkerhed</b>	<b>5</b>
2.1 Angreb udefra . . . . .	6
2.2 Angreb indefra . . . . .	8
2.3 Generelt . . . . .	9
2.4 Opsummering . . . . .	11
<b>3 Nuværende metodik</b>	<b>13</b>
3.1 ENISA Eksempel . . . . .	14
3.2 Udviklingsmetodik . . . . .	15
3.3 Simple Spørgeskemaer . . . . .	16
3.4 Opsummering . . . . .	17
<b>4 Forbedret metodik</b>	<b>19</b>
4.1 Forbedret Spørgeskema . . . . .	19
4.2 Adobe Captivate . . . . .	21
4.3 Phishing Test . . . . .	27
4.4 Opsummering . . . . .	36

<b>5 Website for Sikkerhedsbevidsthed</b>	<b>39</b>
5.1 Kravspecifikation . . . . .	40
5.2 Iterationsplan . . . . .	40
5.3 Iteration 1 . . . . .	42
5.4 Iteration 2 . . . . .	58
5.5 Brugeranalyse . . . . .	71
5.6 Opsummering . . . . .	82
<b>6 Spørgeskemastyring</b>	<b>85</b>
6.1 Kampagne . . . . .	86
6.2 Spørgskemakampagneoprettelse . . . . .	88
6.3 Opsummering . . . . .	90
<b>7 Konklusion</b>	<b>91</b>
<b>A Kildekode</b>	<b>93</b>
A.1 ENISA - Simpelt Spørgeskema . . . . .	93
A.2 ENISA - Forbedret Spørgeskema . . . . .	94
A.3 Wireless Example . . . . .	96
A.4 Phishing Example . . . . .	98
A.5 Iteration 1 - Website for sikkerhedsbevidsthed . . . . .	118
A.6 Iteration 2 - Website for sikkerhedsbevidsthed . . . . .	152

# Indledning

---

Idet internettets muligheder udvider sig og computere nu om dage er hvermands eje er der en enorm grobund for sikkerhedstrusler. Ved at computere kan kontakte hinanden ud over hele verden på et brøkdels af et sekund, kan andre også få kontakt den anden vej med samme lethed. Dette øger risikoen for sikkerhedstrusler til et punkt hvor det er vigtigt at computerbrugeren sætter sig ind i den sikkerhed der skal til for at sikre sin computer.

Sikkerhedsbevidsthed eller 'Security Awareness' på engelsk, er en betegnelse for hvad man forstår ved sikkerhed og hvor højt man prioriterer det. Idet udviklingen inden for computere går så hurtigt nu om dage, kan normale brugere have svært ved at følge med. Således bliver det lettere for ondsindede personer at få kontrol over brugernes computere, eller endnu værre få stjålet følsomme personlige oplysninger som cpr-nummer, hvilket kan lede til identitetsforfalskning og meget andet.

Et fornuftigt niveau af sikkerhedsbevidsthed hos computerbrugeren, vil ikke blot forbedre sikkerheden og sænke risikoen for angreb på brugerens computer, men desuden også forbedre sikkerheden på arbejdspladsen, da færdighederne der gør sig gældende er de samme på alle computere.

Computere er for os i grunden et værktøj til at abstrahere meget komplekse og tekniske opgaver ned til den mindst mulige abstrahering, hvorved opgavens parametre opfyldes samtidig med at opgaven bliver stillet så enkelt som muligt. De fleste brugere kender således oftest kun til det yderste lag af abstraktion

præsenteret af computeren, værende den grafiske fremstilling og kender derfor ofte ikke til de funktioner computeren udfører bag ved, men kun til konsekvenserne.

På grund af dette kan mange brugere ikke forstå simple tekstbaserede spørgsmål, da de er vant til at få præsenteret langt de fleste valg grafisk. For at ”få alle med” kan det derfor være nødvendigt at forbedre spørgeskemaerne grafisk, sådan at man kan analysere brugernes viden i relation til scenarier de kan genkende fra deres eget miljø. Derved vil brugerne blive præsenteret for det samme niveau af abstraktion de er vant til, hvilket muligvis vil forbedre deres evne til at kunne forstå og svare på spørgsmål med teknisk baggrund, som med sikkerhedsbevidshed.

## 1.1 Problemformulering

Målet for dette eksamensprojekt kan summeres op til følgende:

**”Undersøg problemerne ved spørgeskemaer med fokus på sikkerhedsbevidsthed, samt muligheden for at løse dette ved brug af animerede spørgeskemaer”**

Der vil endvidere blive undersøgt følgende:

- Hvorvidt programmet Adobe Captivate kan bruges til udvikling af animerede spørgeskemaer (som forespurgt af CERT).
- Muligheden for at benytte styring af spørgeskemakampanjer, for at optimere udviklingen.

Til dette vil en internetside blive udviklet, for at teste brugernes reaktioner.

## 1.2 Rapporten

Denne rapports struktur er som følger:

**Kapitel 2:** De grundlæggende aspekter af computersikkerhed, samt sikkerhedsbevidstheden der kræves for at opretholde sikkerheden på sin computer.

**Kapitel 3:** Den nuværende metodik der benyttes til at undersøge folks viden om sikkerhedsbevidsthed.

**Kapitel 4:** De metoder der kan forbedre den nuværende metodik. Heri bliver forbedrede, samt animerede spørgeskemaer lavet i Adobe Captivate, beskrevet.

**Kapitel 5:** Udvikling af en webløsning, til en forbedret undersøgelse af sikkerhedsbevidsthed samt de animerede spørgeskemaer virkning, og en efterfølgende analyse af de registrerede data.

**Kapitel 6:** Beskrivelse af et spørgeskemastyringsmodul.

**Kapitel 7:** Konklusion.



# Computer sikkerhed

---

At undgå sikkerhedsproblemer kræver viden om hvordan man indstiller sin computer, samt sig selv, på at være bevidste overfor potentielle sikkerhedstrusler. Dette kræver at man kender til de måder computeren kan blive angrebet på og hvordan man forebygger dem.

Computer sikkerhed [10] kan deles op i følgende tre aspekter:

**Fortrolighed:** Kun betroede personer kan tilgå fortroligt materiale.

**Integritet:** Et emne er hvad det ser ud til og ikke er ændret eller mangelfuldt.

**Rådighed:** Et emne er til rådighed når du har brug for det.

Dette er de mål, der prøves at opnås ved computersikkerhed.

Disse vil modvirke hinanden, som typisk ved at et højt niveau af fortrolighed eller integritet kan være på bekostning af rådigheden. F.eks. opholder en computer der er slukket aspektet for fortrolighed, men slet ikke rådighed da man så ikke selv kan benytte den. Derfor indeholder alt computersikkerhed en slags balance der skal opvejes inde for den valgte form, for at den passer til brugerens behov. Man vil sjældent benytte et så højt sikkerhedsniveau som praktisk muligt, da den maksimale udnyttelse af et aspekt så stærkt vil modvirke et andet.

## 2.1 Angreb udefra

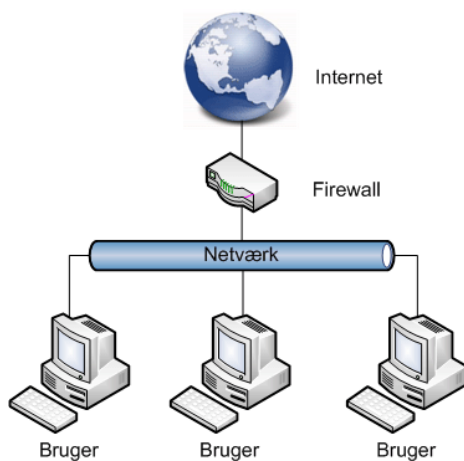
En side af computersikkerhed kan defineres som samlingen af trusler der kan angribe computeren udefra, som ved hvad der kan gøres for at modvirke hackerangreb<sup>1</sup>.

En grund til at hackere i disse dage er svære for myndighederne at spore og retsforfølge, er at de ofte slører sig over computere i flere nationer eller kontinenter med vidt forskellige lovgivninger inden for området.

### 2.1.1 Firewall

Computere kontakter andre computere over et netværk, abstraheret via porte. Idet der findes mange porte, mange reserverede til specifikke programmer og protokoller, er det derfor vigtigt at kunne lukke de porte man ikke benytter, for at begrænse indtrængningsmulighederne gennem netværket. Dette gør en firewall.

En firewall er et filter, beregnet til at blokere for alle de netværks-porte der ikke benyttes hvilket minimerer de veje hackere kan benytte til at kontakte computeren og forebygger derved angreb udefra.



Figur 2.1: Firewall i større netværk

På store netværk som på universiteter og i firmaer, er der næsten altid en dedikeret hardware-firewall (til tider sammenbygget med en router), der filtrerer

<sup>1</sup>En hacker er i denne rapport generelt defineret som en ondsindet person der forsøger at ulovligt tilgå andres computere for sin egen vindings skyld.



alt trafik ind og ud af netværket (illustreret på figur 2.1), men i små hjemmenetværk kan det betale sig at benytte en personlig programbaseret firewall.

Mange personlige firewalls er fra starten sat op til at blokere de fleste porte, og kører så i baggrunden på computeren og lytter, således at man får en besked når firewallen for første gang blokerer en pakke, sådan at man kan vælge om det blokerede program er lovligt eller ej.

### 2.1.2 Adgangskoder

Mange brugere glemmer hvor vigtigt det er at have gode adgangskoder og det er kendt at nogle endda benytter sig af meget nemme koder, som husdyrs navne eller tegnkombinationer fra tastaturet (som 'asdf'). Dette gør det rigtig nemt for ondsindede personer som hackere at bryde ind på computeren, eller tilgå andre personlige informationer tilgængelige fra Internettet.

Hackere benytter små programmer designet til at gætte sig frem til brugeres adgangskoder og på den måde bryde ind i andre folks computere. Disse programmer er i grunden simple programmer, som afprøver strengkombinationer på en givet login. Typisk bliver navne og ord først prøvet fra en intern database (som en ordbog), for derefter at gå til at prøve alle tegnkombinationer igennem (populært kaldet 'brute force'). Den første metode udføres ret hurtigt, hvor den anden tager mere tid. Dette især hvis man har et kodeord med et rimeligt antal karakterer.

For at gøre det så svært som muligt for en hacker at gætte en adgangskode, skal denne helst opfylde så mange som muligt af de følgende krav [2]:

- Være på mindst 8 tegn
- Bruge både små og store bogstaver
- Brug specieltegn og tal
- Ikke have noget med brugernavnet at gøre
- Ikke være et kendt ord eller navn
- Ikke være en genbrugt adgangskode

Idet alle nu om dage har mange forskellige kodeord, kan det ikke udelukkes at et par af disse regler bliver overtrådt. Dette blot for at brugeren kan huske kodeordet uden at være nød til at skrive det ned, hvilket så vil udgøre en sikkerhedsrisiko skulle dette blive tabt eller stjålet. Dog skal det gøres klart at det anbefales at overholde så mange af punkterne som muligt, for at holde sine adgangskoder mest effektive.

Dette er et eksempel på hvordan en bruger vil vægte rådigheden lidt højere, dvs. hvor nemt det er at tilgå entiteten, og derved lade fortroligheden falde.

### 2.1.3 Trådløst Internet

Trådløse netværk i hjemmet er en god måde at gøre det muligt at gå på nettet lige meget hvor man er i huset, men gør det også muligt for ubudne gæster at kigge med på de pakker der bliver transmitteret gennem luften.

Alle trådløse accesspoints har en indbygget mulighed for kryptering og ved at kryptere og derved kode den information der bliver sendt over netværket, kan man forhindre at andre lytter med eller benytter netværket. Dette er en meget simpel måde for brugere at undgå at andre misbruger deres forbindelse, eller de transmitterede informationer.

## 2.2 Angreb indefra

En anden side af computersikkerhed vedrører det program der findes som kan skade computeren eller åbne bagdøre til at tillade angreb udefra.

### 2.2.1 Virus

En virus er et computerprogram som kopierer sig selv og har typisk den effekt at den ødelægger computeren, eller åbner sikkerhedshuller sådan at hackere har nemmere ved at angribe computeren. Det faktum af den spreder sig gør at mange computere hurtigt bliver smittet, og at den derved også kan være svær at finde. Ordet virus kommer fra den biologiske virus hvis kendetegn også er at den spreder sig.

En af de mest kendte virusser var 'ILOVEYOU'. Denne virus virkede ved at man fik en email hvorved man blev snydt til at starte et vedhæftet program, der så startede virusen som så sendte sig selv videre til alle på den inficerede computers addresseliste, overskrev filer med bestemte filendelser, samt lavede en række sikkerhedshuller med mere.

Den normale metode til at undgå virusser er at benytte et anti-virus program som genkender programmets signatur.

### 2.2.2 Adware og Spyware

Idet der findes så mange gratis programmer der kan hentes og installeres fra internettet, er der også begyndt at komme megen Adware [13] og Spyware [14].

Disse er små ekstraprogrammer der følger med et installeret program og derved installerer uventet indhold ud over det brugeren har bedt om. Ved Adware bliver et ekstra program installeret sammen med det der installeres, typisk som en slags annoncering, der tjener penge ind til det originale produkts ejere. Problemet med Adware er at det ofte tager sig i form af Spyware med hvilket brugerens informationer og aktioner bliver overvåget og optaget ulovligt. Der-til kan Spyware også blive benyttet til at overtage computeren eller indeholde en form for virus, som typisk en trojansk virus der åbner bagdøre ind i systemet.

Det er typisk de små gratis programmer man kan hente fra uofficielt udseende sider der indeholder Adware og Spyware og for at undgå dem kan man holde øje med hvad man henter og installerer fra nettet. Desuden findes der gratis scannere som finder og sletter dem (antispyware eller adaware programmer).

## 2.3 Generelt

Indtil nu er sikkerhedsproblemer i relation til specifikke angreb blevet beskrevet, men mange sikkerhedsproblemer har rod i den måde brugeren bruger computeren. Brugere kan, hvis de ikke er bevidste nok, komme til at installere ondsindet software, eller på anden måde selv forårsage sikkerhedsproblemer, hvilket gør at sikkerhedsforebyggende software ikke altid kan fange problemet.

På grund af internettets enorme kommunikationsstyrke er det derved også let at distribuere fælder, til f.eks. at lokke brugere til at installere en virus på deres computer. Disse problemer skyldes brugerens egen uvidenhed, ved at han selv ubevidst kommer til at tillade ondsindet software til hans computer. I disse situationer vil sikkerhedsprogrammer som virusscannere ikke altid være i stand til at opdage truslen, da truslen selv er blevet ”lukket ind” og det er derfor vigtigt at brugere er bevidste overfor de trusler der findes og tager sig de forbehold der skal til for at bevare sikkerheden på sin computer.

### 2.3.1 Opdateringer

Som bekendt kører alt hardware og software i versioner, sådan at det fortløbende kan rettes ind og forbedres funktions og sikkerhedsmæssigt. For at opholde sikkerhedsniveauet på en computer, er det derfor nødvendigt at opdatere alle sikkerhedsrelevante programmer engang imellem. De vigtigste af disse er operativ systemet og anti-virus scannere.

Ved at mange brugere ikke forstår sig nok på computere, får de ikke opdateret dem og kører derved håbløst udaterede versioner, hvilket gør det meget nemt for hackere og virusser at virke på computeren.

Et generelt problem er specielt at visse ondsindede personer ydnytter store producenters log til at finde huller i software. Da det typisk tager nogle dage for selv mere erfarne computerbrugere, fra en opdatering bliver udsendt til den bliver installeret på deres computere, kan disse personer udnytte denne tid til at lave et angreb specificeret ud fra den log der beskriver rettelsen af programmet. Dette taler især for automatiske opdateringer som de fleste programmer har, sådan at deres versioner kan blive opdateret så hurtigt som muligt.

### 2.3.2 Internet

Som beskrevet kan brugere selv udvidende komme til at hente og installere programmer der kan indeholde virus. Derudover findes der også internetsider der forsøger at lokke brugeren til at give sit brugernavn og kodeord på falsk grundlag. Denne form for svindel kaldes Phishing [5].

Phishing foregår ved at svindleren konstruerer et scenarie hvorved han angiver sig for at være en anden, for derved at få lokket fortrolige informationer som brugernavn og adgangskode, ud af offeret.

En af de mest udbredte phishing metoder er på internetsider, hvor svindleren udarbejder en webside der ligner en kendt side udefra, og så derpå snyder brugere til at indtaste deres brugernavn og kodeord, der så blive udnyttet af svindlerne. Det svarer på en måde til at en svindler laver en falsk bank-automat, der hugger både kort og pinkode når en person bruger den. Dette ville sjældent ske og meget hurtigt blive opdaget i virkeligheden, men på internettet er det så nemt at kopiere grafisk indhold, at der findes en del af denne slags svindel.

En af de nemmeste måder at afsløre en phished webside er at kigge oppe på adressebaren og se om adressen virkelig passer. Derudover skal man også sikre sig at protokollen er af den sikre variant 'https' og at siden er certifieret, hvilket er angivet af en lås.

Hvad angår handel over internettet, som homebanking, er disses ansvarlige meget opmærksomme på dette område og denne slags forbindelser bliver normalt stærkt beskyttet. Samtidigt er der oprettet flere slags tekniske og juridiske sikkerhedsforanstaltninger som gør at brugeren kan føle sig sikker med at handle på nettet. Kryptering benyttes desuden til at oprette digitale signaturer, som det offentlige f.eks. benytter til at identificere borgere således at disse kan tilgå skatte informationer sikkert over internettet. Dog forhindrer alt dette ikke andre i at svindle brugeren ved at lave en phished webside.

### 2.3.3 Email

Idet at en email ikke koster noget af sende, er det en meget nem distributionskilde for uønskede reklamer og andet af mere alvorlig karakter som f.eks. phishing.

Det meste phishingsvindel hænger sammen med emails, ved at der først sendes en forfalsket phished email til brugeren, med et link til et phished side. Fra denne mail bliver brugeren så ledt hen på en forkert internetadresse til en phished internetside. Mange firmaer fortæller deres kunder at de aldrig vil spørge om adgangskoder eller pinkoder over email, for at undgå at deres kunder bliver offer for denne slags svindel.

## 2.4 Opsummering

Den fornødne viden der kræves for at en bruger kan beskytte sig selv og sin computer, kræver flere steder at han forstår lidt mere end det yderste abstraherede lag. Især den generelle sikkerhedsbevidsthed er kritisk, da brugere hurtigt kan træde forkert, hvis de ikke ved bare lidt om de faldgruber der findes.

Ved enhver sikkerhedsovervejelse vil en bruger veje de tre sikkerhedsaspekter op og vælge derefter. Og da mennesker normalt er meget vanepregede, vil brugere nok have en tendens til at vælge en sikkerhedskonfiguration der ikke sænker deres brug af computeren.

Hvis den generelle bruger skal overbevises til at tilegne sig disse informationer og brugsmønstre om sikkerhedsbevidsthed, skal de helt kunne se det som 'god logik' samtidigt med at de føler deres arbejdsrutiner ikke bliver synderlig sinkede af dem.



## Nuværende metodik

---

Spørgeskemaet er en meget benyttet metode ved brugeranalyser. Benyttet på internettet er de nemme at distribuere og opsamling af responsdata foregår automatisk. Normalt benyttes et simpelt tekstbaseret spørgeskema, som består af en række tekstspørgsmål i hvilke brugeren kan afkrydse sine svar. For at spørgeskemaet skal virke bedst muligt, er det vigtigt at alle modtagere nemt kan gå det igennem og svare på spørgsmålene, uden at synes at have spildt unødigt tid på det. Alle spørgsmålene skal derfor være så klart formuleret som muligt, og det skal være nemt for modtageren at finde det korrekte svar uden at tænke videre over det. Dette er også typisk situationen, da de fleste brugeranalyser omhandler brugsmønstre eller meninger som brugeren nemt kan relatere til. Dog kan der fremkomme komplikationer ved brugeranalyser der omhandler tekniske emner som ved sikkerhedsbevidsthed, da ikke alle brugere nødvendigvis kan relatere til emnerne, hvilket så gør at disse brugere ikke forstår spørgsmålene og ikke kan besvare spørgeskemaet rigtigt. Disse problemer kan forklares ved at analysere nærmere hvordan den normale bruger benytter sin computer.

En computer abstraherer som tidligere beskrevet komplekse teknikker således at mange lag af protokoller og teknikker bliver abstraheret til et enkelt grafisk interface, hvori brugeren med så få klik som muligt kan udføre den pågældende funktion. Idéen med dette er at brugeren ikke behøver at bekymre sig om de indre teknikker, eller forstå dem, og derved kan koncentrere sig om helheden i hvad programmet gør. En bruger behøver f.eks. ikke at vide hvordan en email-klient sender og modtager mails på det tekniske plan, men behøver blot vide hvad han skal gøre for at sende og modtage sine mails. På denne måde kræver

computeren så lidt som muligt af brugeren sådan at flere brugere kan bruge den. Dette gør at brugere til tider har deres eget billede af hvad der egentlig sker i en funktion og dette et grafisk abstraheret billede, idet de f.eks. ikke nødvendigvis forstår at de fører en synkroniserings transaktion med deres emailserver, men blot at de trykker på send/modtag -knappen, hvorved de rigtige personer får deres mails. Nogle brugere kan endda blive forvirrede og usikre, hvis det grafiske design af deres program ændrer sig lidt ved en opdatering.

Denne forskel i forståelsen af computerens funktioner leder til hypotesen at mange brugere ikke forstår begreber beskrevet med tekst i et spørgeskema og derved ikke vil kunne svare på det, eller endnu værre vil svare forkert. For at få så mange som muligt af brugerne til at forstå en brugeranalyse korrekt, kan det blive nødvendigt at udvide spørgeskemaerne grafisk, således at brugerne får vist den grafiske repræsentation de er vant til, så vidt muligt. Det gælder om at kunne relatere til det brugerne kender og få præsenteret/formuleret spørgsmålene på en nem og forståelig måde. Dette forbedrede spørgeskema kaldes et animeret spørgeskema og vil blive beskrevet i det næste kapitel.

### 3.1 ENISA Eksempel

ENISA er “The European Network and Information Security Agency” [9] hvilket rådgiver og assisterer EUs medlemsstater i udviklingen af awareness-programmer og hjælper med at rådgive sikker brug af it-teknologier. I ENISA rapporten “How to Raise Information Security Awareness” [4] er det beskrevet hvordan man kan opbygge og håndtere en awareness kampagne. Deri er der også et eksempel på hvordan et således spørgeskema kunne se ud. Figur 3.1 viser spørgsmål syv fra ENISA spørgeskemaet.

7. Do you have any of the following in place to protect your computer and electronic data?

Please indicate all that apply.

- a.  Anti-virus software that is updated regularly
- b.  Firewall
- c.  Anti-spam filter
- d.  Good password practices
- e.  Process of regular backup of data
- f.  Up-to-date Internet browser with encryption
- g.  Others please indicate \_\_\_\_\_

Figur 3.1: ENISA - Spørgeskema

Selve spørgeskemaet er meget generelt, og indeholder kun tekst-spørgsmål.

I spørgsmålet bliver der refereret til tekniske emner som firewall or anti-spam



filtre, hvilke er navne ikke alle brugere kender til. Det er helt forståeligt at der i et spørgeskema for sikkerhedsbevidsthed bliver spurgt om f.eks. kryptering, men det vil mest sandsynligt ikke være alle brugere der forstår hvad dette er. I at man kun kan vælge at afkrydse om man har installeret teknologien eller ej, kan det af de udtrukne data desuden ikke skelnes om en bruger ikke har installeret teknologien, eller ikke forstår hvad teknologien er. Hvis en bruger ikke afkrydser 'firewall' kan det ikke siges om brugeren ikke ved hvad en firewall er, eller om han godt ved hvad det er men har valgt ikke at bruge en. Dette er en stor usikkerhed ved ENISA spørgeskemaet og vil lede til misvisende statistikker, hvor det vil forsage at statistikken læner sig mere mod at brugere ikke er interesseret i at beskytte sig.

Selvom de tekniske emner i spørgeskemaet fra ENISA er emner der er nødvendige at spørge om i en sådan forespørgsel, bliver man nød til at analysere hvordan man stiller spørgsmålet således at så mange brugere som muligt forstår det og sådan at de forskellige valg ikke er tvetydige som i det viste eksempel. Hvis spørgsmålene er tvetydige kan det ødelægge meningen i brugeranalysen og statistikken bliver misvisende.

I dette tilfælde vil det være gavnligt at udvide spørgsmålet med en ekstra række afkrydsningsfelter, hvorfra brugeren kan vælge 'Ved ikke'. På denne måde vil man både få udryddet tvetydigheden i at en bruger der ikke har installeret teknologien ikke afkrydser og man får desuden fanget brugere som ikke ved hvad teknologien er. Det vil også forbedre besvarelserne hvis man tilføjer indlæring til spørgsmålet. En beskrivelse af hver teknologi vil få folk der godt kender teknologien men ikke navnet til at forstå spørgsmålet. Dog skal dette så præsenteres på en måde så det ikke forvirrer brugeren unødvendigt.

## 3.2 Udviklingsmetodik

Ved udvikling af spørgeskemaer på internettet, er realiseringen mulig på flere måder. Disse måder begrænser alle spørgeskemaernes udformning på hver deres måde. Efterfølgende beskriver kort nogle af de mest benyttede metoder.

**Hyper Text Markup Language** (html) er et script-agtig sprog som er kernen af alle internetsider. Disse sider beskriver opstillingen af tekst og billeder på internettet. Næsten alle andre metoder inden for webudvikling generer html-indhold eller er omkranset af en htmlside. Html-sider beskriver statisk indhold godt og præcist, men har ikke direkte megen mulighed for mere animeret grafisk indhold.

**Hypertext Preprocessor** (PHP) er et opensource serverside scriptsprog, der har en syntaks der minder meget om Java og C. PHP skrives inline i html

og udføres før html-siden tolkes, på serverens side. PHP-sider kan derved gøre html-sider meget dynamiske og giver mulighed for database tilgang. Et andet meget lignende scriptsprog er ASP eller Active Server Pages.

**Flash** er et vektor-animations software, som indlejres i html-sider og danner visuelt forbedrede internetsider med stærke medietilslutninger. Denne vektorbaserede grafik er ideel på internettet da den fylder meget lidt, men gør det et større arbejde at gøre siderne mere dynamiske, da indholdet så er beskrevet vektorbaseret og ikke af tekst.

**Java eller .NET** er blandt alternativerne til scriptsprog som PHP og benyttes især til større og mere komplicerede websider eller webservices . Disse beskriver ikke det grafiske indhold af en webside, men former ligesom scriptsprogene det dynamiske som forbindelsen til database mm.. Dog performer disse meget bedre i præstationstunge sammenhæng, men dette er sjældent nødvendigt i spørgeskemasammenhæng.

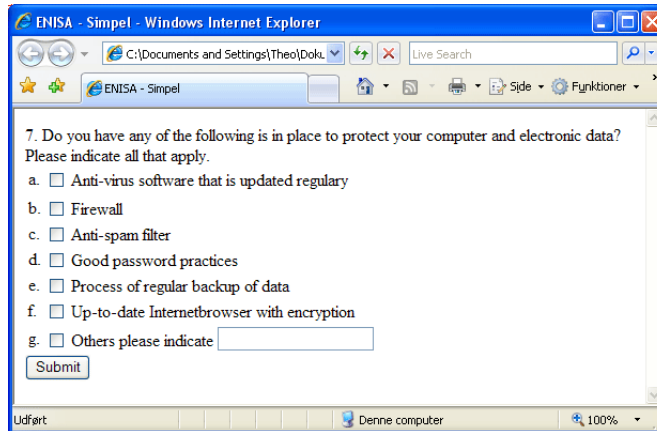
Langt de fleste internetsider fungerer ved at der bliver dannet en html-side med eller uden vektor-baseret grafik, som så beskriver sidens udseende for brugeren. Dette er næsten altid html eller flash. Ud over disse har alle teknologierne til fælles at de vises på en internetbrowser på en computer. Dette gør at man skal have størrelsen af spørgeskemaet i betragtning, samt browserens fortolkning af den valgte teknologi, da de forskellige browsers fortolkere ikke altid betragter siderne ens.

### 3.3 Simple Spørgeskemaer

Figur 3.2 et eksempel på en realisering af ENISA spørgeskemaets spørgsmål 7 vist i en browser, og er implementeret i den mest basale form muligt med html.

I en sådan realisering ville siden (bilag A.1.1 - enisasimpel.html) således være sat simpelt op grafisk, javascript ville validere formdata og der ville være benyttet en form for server side scripting (som php) til at lægge data op i databasen.

Til forbedring af et spørgeskema som dette er de vigtigste aspekter de grundlæggende, som hvad man spørger om og hvordan brugeren skal svare. Som det blev beskrevet i analysen af ENISA spørgeskemaet var et vigtigt punkt at brugere ikke kunne markere om de ikke kendte til en af valgene og ydermere kunne det også være gavnligt med mulighed for forklaring af teknologierne.



Figur 3.2: Simpelt Spørgeskema - ENISA Spørgsmål 7

## 3.4 Opsummering

Simple spørgeskemaer kun indeholdende tekst er ikke altid optimale til udførelse af brugeranalyse, når emnet er teknisk som med sikkerhedsbevidsthed. Hvis man ikke sørger for at analysere og udforme spørgsmålene grundigt, risikerer man at mange af brugerne ikke forstår spørgsmålene, selv om de måske egentlig benytter teknikken. I ENISA eksemplet kunne det endda ses at det viste spørgsmål ikke skelnede mellem folk der ikke benyttede en teknik, eller ikke vidste hvad svaret var, hvilket ville forsage en forvirrende statistik.

Spørgeskemaer udviklet til internettet tillader nem distribution og opfangning af data. Men man bør udnytte internettets muligheder og optimere og forbedre spørgeskemaerne så vidt muligt, for at få så mange brugere med som muligt og for at få præciseret spørgeskemaet således at eventuelle tvetydigheder er udryddet i den endelige version. Til dette er animerede spørgeskemaer muligvis overlegne til de tekstbaserede simple, da disse kan visuelt kan præsenteres så det ligner det miljø brugeren er vant til.

Når man udvikler et spørgeskema er det nødvendigt at analysere det grundigt. Hvordan spørgsmålet stilles og hvilke muligheder brugeren har til at vælge sit svar, har stor betydning for hvordan de opfangede meninger former sig.

Spørgeskemaer skal optimalt overholde disse målsætninger:

- Kræve så lidt tid af brugeren som muligt, mens der stadigvæk bliver opfanget de fornødne oplysninger.
- Ikke indeholde tvetydigheder og være så intuitivt og nemt forståeligt at

brugeren ikke behøver at tænke, men blot svare.

Det næste kapitel vil omhandle mulighederne for at forbedre spørgeskemaer ud fra disse punkter.

# Forbedret metodik

---

Mange spørgeskemaer bliver ikke analyseret ordentligt, hvilket kan resultere i tvetydige spørgsmål og utilstrækkelig identifikation af de værktøjer der skal til for at få præsenteret spørgsmålene på den mest intuitivt forståelige måde.

Analyseringen af et spørgeskema er vigtigt hvis det skal virke optimalt. Måden spørgsmålene bliver præsenteret på betyder alt for om brugerne virkelig forstår spørgsmålene rigtigt eller ej. Det gælder om at finde den bedste måde at vise og formulere spørgsmålene på, sådan at det kræver så lidt af brugeren som muligt, samtidigt med at spørgsmålene er nemme at forstå. I flere tilfælde kan det være en fordel at tilføje billeder, eller mere avancerede animerede medier til spørgeskemaet i de tilfælde hvor spørgsmålene omfatter tekniske emner, som på en computer ofte er abstraheret grafisk for brugeren.

## 4.1 Forbedret Spørgeskema

I sidste kapitel blev en simpel version af Enisa spørgeskemaets spørgsmål syv beskrevet. Jeg har til dette afsnit forbedret spørgsmålet, hvilket er vist i en browser på figur 4.1.

I sidste afsnit blev det pointeret at en forbedring i form af en ekstra række afkrydsningsfelter, vil udvide spørgsmålet til også at opfange forskellen mellem om brugeren ikke forstår spørgsmålet/teknologien, eller om brugerne har valgt ikke at have teknologien installeret. Under udvikling af denne forbedrede version

ENISA - Forbedret - Windows Internet Explorer

file:///C:/Documents%20and%20Setti Live Search

ENISA - Forbedret

7. Do you have any of the following is in place to protect your computer and electronic data?  
Please indicate all that apply.

	Don't know	Yes	No	
a.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Anti-virus software that is updated regularly
b.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Firewall
c.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Anti-spam software that blocks unused entries to the computer, thereby increasing the computer's security on the network
d.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Good password management
e.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Process of regular backup of data
f.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Up-to-date Internet browser with encryption
g.			<input type="checkbox"/>	Others please indicate <input type="text"/>

Submit

Figur 4.1: forbedret Spørgeskema - ENISA Spørgsmål 7

kom det dog for øje at det var endnu smartere at tilføje to ekstra rækker, som så bliver sat op så man enten kan vælge: 'Ved ikke', 'Ja' eller 'Nej'. På denne måde bliver brugeren nød til at svare på en af dem for hvert spørgsmål og man får så en af de tre svarmuligheder, for hvert svar. Denne forbedring gør spørgsmålet klarere for brugeren og eliminerer tvetydighederne fra den simple version.

For at hjælpe brugere der ikke helt forstår de tekniske termer, er der tilføjet popup-bokse, hvilket kommer frem når man holder musen under visse nøgleord, markeret med en understregning. Boksene er små tekstforklaringer sat til ikke at være synlige, på nær når man holder musen over et nøgleord. Denne slags skjult information, er særdeles nyttig da den som udgangspunkt er skjult for brugeren, sådan at designet holdes overskueligt. Dertil er det nemt at tilgå som en slags 'hint' hvis man har brug for det.

Med disse forbedringer til Enisa spørgsmålet, vil spørgsmålet opfange mere korrekte informationer, uden så mange tvetydigheder som før. Og det vil om muligt fange flere brugere, idet de kan få lidt hjælp til begreberne på en nem og overskuelig måde.

Det kan ses at små ændringer kan forbedre sådanne spørgeskemaer meget, især hvis man genanalyserer spørgsmålet og tænker fra brugerens side. Hvis brugere kan komme i tvivl ved et spørgsmål kan det gøre de opfangede data inkonsistente og man kan derved ikke helt regne med de statistikker man får ud af det.

## 4.2 Adobe Captivate

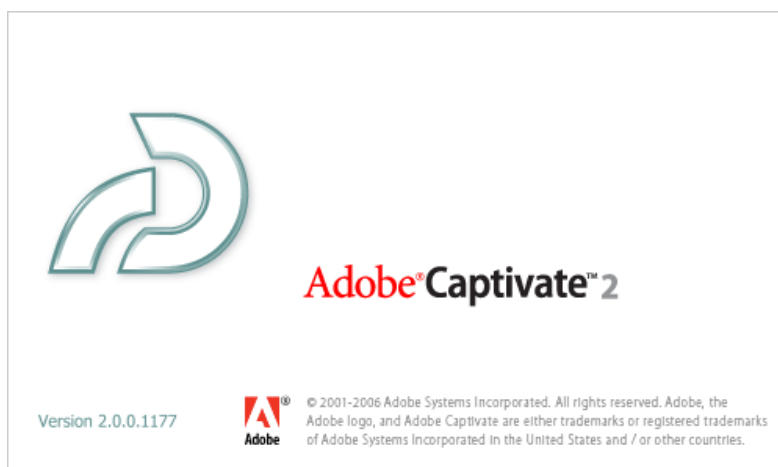
*”Rapidly create engaging learning experiences”*

Sådan lyder overskriften på hjemmesiden for Adobe Captivate 2. Captivate [1] er et program designet til at lave indlæringsvideoer til internettet og kan optage skærbilleder af programmer, således at disse kan benyttes til at konstruere en interaktiv video der fremviser et program. Man kan dertil tilføje små text-quiz’er, ekstra knapper, tekst felter og andre småting til at skræddersy indlæringsvideoen.

CERT (det danske Computer Emergency Response Team) er i forbindelse med et sikkerhedsbevidstheds projekt, interesseret i brugbarheden af Adobe Captivate til udvikling af animerede spørgeskemaer. Captivate eksemplerne i dette kapitel er derved blevet fremlagt for dem og udviklet med deres input.

Captivate er især interessant da dets opbygning tillader hurtig optagelse af skærbilleder, hvilket kan benyttes til opbygning af animerede spørgeskemaer. Dog vil det ved udvikling af animerede spørgeskemaer muligvis begrænse funktionaliteten at Captivate egentlig er designet til at lave brugerindlæringsvideoer.

I denne sektion vil det derfor blive undersøgt hvorledes man kan opsætte Captivate, for at konstruere interaktive spørgeskemaer beregnet til at teste brugere over internettet.



Figur 4.2: Adobe Captivate 2 - Splash Screen

### 4.2.1 Funktionalitet

Captivate er et af Adobe's mange webudviklings-programmer og er bygget ovenpå deres populære flash-teknologi, som benyttes på internettet til at konstruere vektorbaseret animeret grafik. Et Captivate projekt bliver udsendt som en swf-fil (shockwave flash) som så direkte kan indlejres i en web-side. I og med at flash er så udbredt på nettet, understøttes det af stort set alle browsere.

Captivate er bygget op omkring slides, ligesom i Microsofts Powerpoint. Når man går i record-mode, minimeres Captivate og man kan optage et skærmbillede ved at trykke på printscreen-knappen. Hver gang man trykker hører man en klik-lyd og billedet bliver derved lagt ind som et ekstra slide i ens projekt. Der er også mulighed for at vælge at slide-størrelsen kan være ens med et programs størrelse, sådan at den kun tager et billede af et specifikt program.

Når man har optaget sine billeder kan man tilføje ekstra slides, knapper, tekst og derved tilføje interaktion til billederne, således at det færdige produkt kan opføre sig tilsvarende til det rigtige program. På denne måde kan man simulere et program, med mulighed for tilføjede instrukser og benytte dette til at lave et animeret spørgeskema.

Captivate er lavet til ikke-udviklere, og reklamerer også for at man ikke behøver at kunne programmere for at benytte det. Dette betyder desværre også at næsten alle af Captivate's funktioner, kun kan vælges med musen og at man ikke selv har mulighed for at programmere nogle andre slags interaktioner<sup>1</sup> end dem der udbydes. Idet projektet bliver et indlejret flash-element på en webside, kan man som det eneste programmerbare vælge at udføre javascript ved en interaktion i videoen. Det javascript der køres, køres så på websiden således at det svarer til at dette javascript lå på derpå.

### 4.2.2 Proof of Concept

For at vise at Captivate kan benyttes til at lave et animeret spørgeskema og at man kan udtrække respons-data, vises her et eksempel på hvordan et spørgeskema konstrueres. Dette eksempel vil være mit *proof of concept*, altså mit bevis på at Captivate kan bruges til at konstruere animerede spørgeskemaer og eksempel på hvordan man bygger det i Captivate.

Til eksemplet har jeg valgt et scenario hvor brugeren skal vælge mellem forskellige tilgængelige forbindelser på et trådløs netværk, for derved at identificere det mest sikre netværk. Dette vil blive lavet ved at optage screenshots fra Windows' eget program til at finde og vælge trådløse netværk. Det vil blive vist hvordan

---

<sup>1</sup>I Adobe's flash findes der f.eks. *Actionscript*, som gør det muligt at programmere interaktioner på script-form.

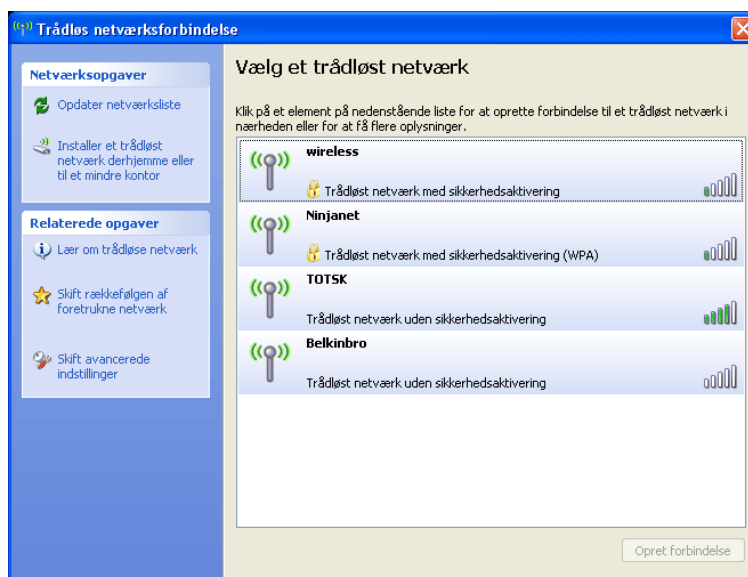


man får bygget projektet og tilpasset det, således at de opfangede data bliver hentet ud af videoen og ind i et form-kald. Således bliver eksemplet koncentreret om Captivate.

Eksemplet kan ses på adressen: <http://wireless.tre-ben.dk/>.

### 4.2.2.1 Optagning

Når Captivate startes, vælges 'Record or Create a new project' hvorefter det kan vælges hvorvidt projektstørrelsen skal passe med det program der skal optages, eller om det skal være en anden størrelse. Idet vi gerne udelukkende vil optage handlinger i et enkelt programvindue, vælger vi at tilpasse den til en applikation og trykker 'Ok'.



Figur 4.3: Adobe Captivate - Screenshot uden interaktioner

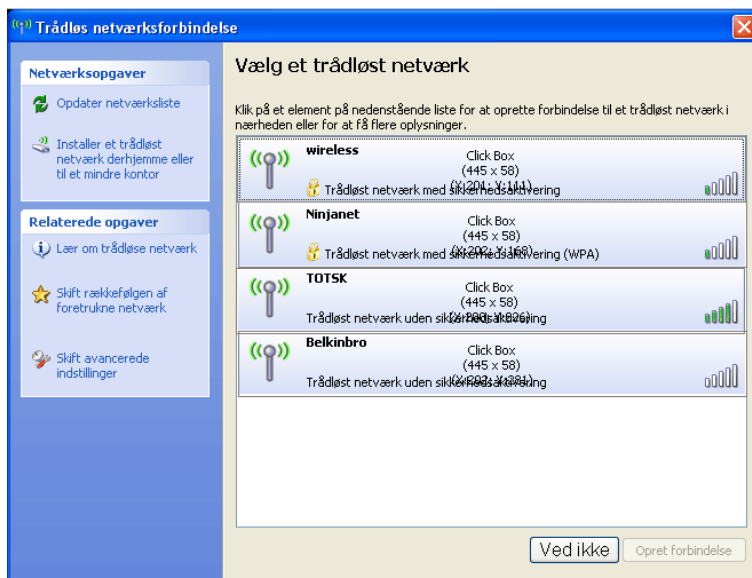
Herefter kommer der en rød-kasse frem, som angiver optagelsesområdet og man kan fra en dialogboks vælge hvilket program man vil optage. Når man fortsætter minimeres Captivate og man kan optage ved at trykke på *printscreen* og stoppe ved at trykke på *pause*.

Captivate finder ofte selv ud af hvornår skærbilledet skifter og optager derved et nyt billede. Captivate finder også ud af hvor man har trykket og laver derved et interaktionsfelt.

Figur 4.3 viser et screenshot af det trådløse scenarie, som optaget af Captivate. Billedet er af det slide hvor brugeren har mulighed for at trykke på hver forbindelse. I næste slide vil den valgte forbindelse have en blå baggrund og derfra kan brugeren vælge at markere en anden forbindelse, eller trykke 'Opret forbindelse'.

#### 4.2.2.2 Tilretning

Når alle de nødvendige screenshots er optaget kan det ses at Captivate har lagt dem alle ind i hver deres slide i den rækkefølge man optog dem. De gange man har klikket i programmet er også optaget, som indsatte interaktioner.



Figur 4.4: Adobe Captivate - Screenshot med interaktioner

Nu tilføjer man så startslides med information, interaktioner og knapper, samt indstiller alle interaktioner til at gå til de korrekte slides. For forståelsens skyld er der i videoen tilføjet start og slut slides. På figur 4.4 kan man se det samme slide som før, men nu med alle nødvendige interaktioner sat på. Man kan se at der er stiplede kasser der markerer hver forbindelse og der er tilføjet en 'Ved ikke' knap til brugere som ikke er sikre på svaret.

For at få animationen til at virke som programmet, har jeg lavet interaktionerne sådan at man kan blive ved med at vælge de forskellige forbindelser, så de bliver markeret blå. På denne måde virker scenariet fuldstændig ligesom det

egentlige program.

En oversigt over de forskellige slides kan ses på figur 4.5. Captivate genererer selv et diagram over interaktionerne, men dette bliver meget hurtigt helt uoverskueligt når man laver tilbagegående interaktioner.

1. Indledende Overskrift
2. Søger efter netværk
3. Oversigt over netværk
4. Netværk 1 valgt
5. Netværk 2 valgt
6. Netværk 3 valgt
7. Netværk 4 valgt
8. Testen er færdig

Figur 4.5: Adobe Captivate - Oversigt over slides

De første tre slides går til deres efterfølgende slide og slides fire til syv går til hinanden eller med interaktionerne for 'opret forbindelse' og 'ved ikke' til det sidste slide. På denne måde får man illusionen af at der søges og findes fire netværk som man kan vælge imellem og derved vælge en forbindelse, eller angive at man ikke ved hvilken.

### 4.2.2.3 Opfangning

Nu er selve videoen lavet og hvis man vælger 'publish' laves en flash fil sammen med en webside der kører videoen, således at man kan prøve det og se hvordan det virker. Dog mangler der nu som det sidste at få lagret interaktionsinfo fra slidene ud af flash-filmen.

Ved at gå ind under 'Project + Quiz Manager' og sætte et hak i "Enable reporting for this project", kan man sætte Captivate til at opfange interaktionsdata. Captivate har en række reporterings metoder som man kan vælge til at eksportere sine data, som næsten alle er beregnet til at ramme dedikerede rapporteringsprogrammer, lige på nær én mulighed. Ved at vælge at rapportere som en email, vil videoen og websiden automatisk blive sat op til at sende interaktionsdataene som semicolon-separeret tekst til den definerede email. Dette kræver dog at der er blevet sat et score-slide til ved slutningen af videoen. Denne kan sættes under 'Quiz Settings' sektionen af 'Quiz Manager'.

Det score-slide der bliver oprettet derved er ikke optimalt til videoen, da denne bliver oprettet som svar til en quiz hvilket vi ikke er interesseret i. Men ved at ændre og slette nogle af felterne, fandt jeg ud af at jeg kunne få den til at ligne en takke-slide i stedet. Denne metode betyder dog at brugeren på det sidste slide, skal trykke på en knap for at køre det javascript som fra websiden sender emailen. Ved at indsætte et ekstra slut-slide før det sidste, kan man benytte et javascript som automatisk bliver udført når brugeren når til slidet og på den måde udsende emailen uden at brugeren behøver at trykke på en knap.

Opfangning af data med email er ikke særlig nyttigt da interaktionsdataene så ikke automatisk bliver lagret. Dog kunne det javascript der sendte emailen ændres, således at den funktion der bliver kørt på websiden ikke sender en email men submitter en form til en anden side, hvilken så kan være en 'handler' der overfører relevante data til en database. Derved bliver alle interaktionsdataene samlet sammen og sendt til en javascriptfunktion på html-siden, hvilket i stedet for at sende en email, sender dataene via et simpelt form-kald til en anden webseite til at opsamle dataene til en database eller lignende.

Dette kan ses i følgende kildekode udklip fra websiden, som indeholder scenariet (appendix [A.3.1](#)):

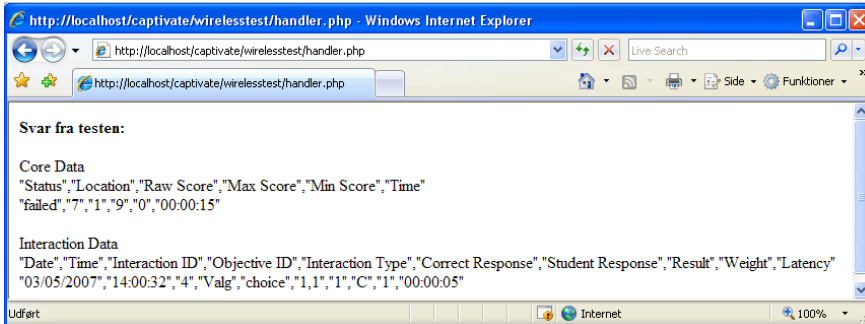
```
function sendMail() {
    document.emailForm.elements["Results"].value = "\r" + gstrEmailBody;
    document.emailForm.submit();
}
</script>

<form name="emailForm" action="handler.php" method="POST">
<input name="Results" type="hidden">
</form>
```

Interaktionsdata bliver lagt ind i det skjulte felt i formen og formen bliver så sendt til filen 'handler.php' (appendix [A.3.2](#)) som håndterer informationerne og overfører dem til en database eller andet.

Captivate er som standard sat op til at opfange alle interaktionsdata, men da vi har mange interaktioner som egentlig er til for at simulere et rigtigt program, vil vi gerne have dem udeladt af de opfangede. Dette gøres ved at gå ind under 'Project + Advanced Interaction' og fjerne hakket i checkboxene på de interaktioner der ikke skal med. Med dette gjort ser de opfangede interaktionsdata ud som vist på figur [4.6](#).

Af relevante data kan ses at brugeren har valgt forbindelse nummer fire. Captivates quiz-funktioner vil prøve at bedømme interaktionerne som bestået/ikke bestået og opfange responstider, men dette kan vi ignorere da vi er kun interesseret i hvad brugeren trykkede på. Ved håndtering af de opfangede data vil



Figur 4.6: Adobe Captivate - Screenshot af opfanget data

man så filtrere Captivates formatering fra, således at man kan lagre interaktionerne for sig.

Hermed er det vist at det er muligt at oprette animerede spørgeskemaer i Captivate, selv om det ikke oprindeligt er designet til det. Ydermere er det muligt at tilrette det konstruerede sådan at de lagrede interaktionsdata ikke er begrænset til et bestemt rapporteringsystem eller en email, men kan sendes til en webside via en form hvorfra der er mange mulige måder at lagre informationerne på.

## 4.3 Phishing Test

Med det vist at det er muligt at benytte Captivate til at konstruere visuelt mere animerede spørgeskemaer, vil jeg nu beskrive et udvidet eksempel hvori Captivate bliver brugt fuldt ud til at bygge en ”Phishing Test” som skal teste brugerens viden inden for phishing og internet svindel.

Dette eksempel er ment som en komplet test af Captivate, til udvikling af animerede spørgeskemaer. Da jeg i sidste test fik konkluderet at Captivate kan benyttes, vil jeg i denne test gerne identificere eventuelle svagheder og styrker der findes ved fuld brug af Captivate til opbygning af spørgeskemaer.

Phishing testen kan ses på siden <http://phishing.tre-ben.dk/>.

### 4.3.1 Analyse

Phishing<sup>2</sup> omhandler falske emails der linker internetsider, hvilke ligner rigtige sider og derved forsøger at få lokket følsomme oplysninger som CPR-nummer

<sup>2</sup>Phishing er beskrevet i afsnit 2.3.2 - ”Computer Sikkerhed”.

eller brugernavn og kodeord fra brugeren, hvilke derfra kan misbruges af gerningsmændene.

For at undgå phishing, skal man kunne genkende falske internetadresser både som link i sine emails og i sin browser, desuden skal man være klar over at man først har en sikker forbindelse når der findes både en lukket lås, samt 'https' i starten af internetadressen. Låsen ikonificerer et certifikat, hvilken man yderligere kan kontrollere ved at klikke på låsen. Ofte kan man i en phished email gennemskue om den er phished eller ej ved formuleringen, men ellers er den eneste måde, at kontrollere om linket refererer til en adresse der i orden. Afsenderadressen i en email kan nemt forfalskes, så denne kan ikke benyttes til kontrol.

Brugere skal gerne være bevidste på forskellen mellem de viste oplysninger fra browseren, og dem fra internetsiderne som browseren viser. Research [5] viser at nogle brugere endda kan blive snydt af internetsider, der viser et billede af en lås på sine internetsider.

Testen skal kunne undersøge disse forhold og desuden indeholde indlæring ved at forklare den rigtige fremgang til spørgsmålene til sidst, som værende motivationen for brugerne.

### 4.3.2 Design

Denne test vil blive delt op i to spørgeskemaer; en for emails og en for internetsider. Brugerne skal vises eksempler på rigtige samt forfalskede emails og internetsider og skal så kunne svare om de vil stole på disse. Hvis de svarer 'nej' skal de yderligere begrunde hvorfor de ikke stoler på siden/emailen, for at få opfanget hvad der gjorde udslaget.

Idet vi grafisk skal vise internetsider og emails, der normalt ville fylde hele skærmen, på en mindre del af skærmens størrelse, sammen med en form for input som knapper, må vi nødvendigvis minimere det nødvendige der skal vises, for at spørgsmålet kan forstås.

Hvert spørgeskema deles op i følgende slides, beskrevet på figur 4.7:

Første slide indeholder en overskrift og det næste noget tekst hvor der står beskrevet de forudsætninger brugeren skal svare ud fra. Teksten forklarer at man skal forestille sig at man har klikket på et link og skal til at logge ind på en side, herved skal man markere om man i en normal situation ville stole på den.

De efterfølgende fem slides derefter er spørgsmålene og når testen er færdig får man vist svarene, hvorved man vil få forklaret sine svar samt det korrekte svar til hvert spørgsmål. Hvis man svarer 'nej' til et spørgsmål får man vist

1. Indledende overskrift
2. Indledende beskrivelse
3. Spørgsmål 1
4. Spørgsmål 2
5. Spørgsmål 3
6. Spørgsmål 4
7. Spørgsmål 5
8. Testen er færdig
9. Svar fra test

Figur 4.7: Adobe Captivate - Oversigt over slides

yderligere et slide, hvorved brugeren bedes uddybe hvorfor.

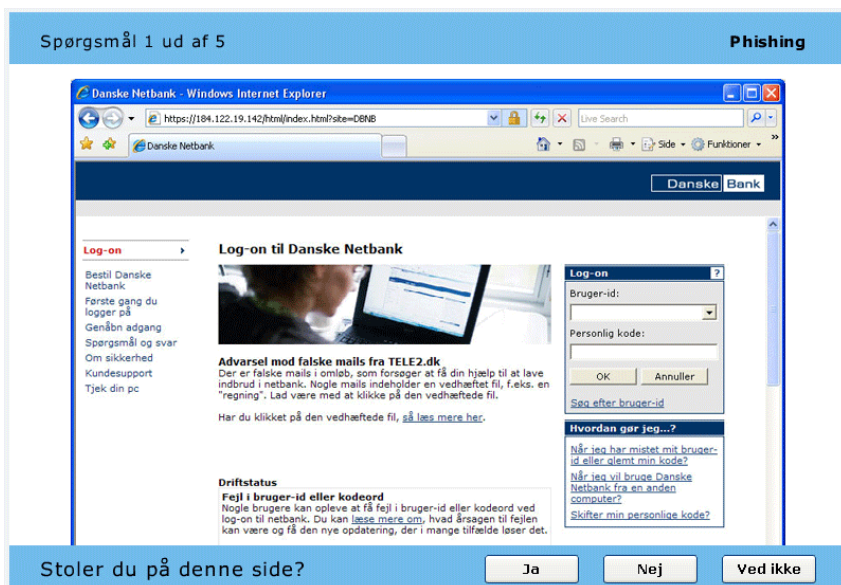
Svarsiden vil ikke være lavet i Captivate, da Captivate ikke kan håndtere informationer, men kun optage dem. Derfor vil denne være lavet som en php-forbedret html-side.

### 4.3.3 Implementering

Implementeringen er delt op i en del der beskriver det lavet i Captivate og en del der beskriver svarsiden lavet i html og php.

Det visuelle design af spørgeskemaerne består af et simpelt baggrundsbillede lavet i Adobe Photoshop og har en blå bjælke øverst og nederst, sådan at overskrifter og knapper virker naturligt afgrænset fra indholdet i midten. Størrelsen af dette billede afgør størrelsen af spørgeskemaet, som er sat til at kunne være i en browser på en computer med en opløsning på mindst 800 x 600 pixels. Det grafiske design kan ses på figur 4.8.

Designet er konstrueret til at være simpelt og overskueligt. Hver type overskrift er det samme sted på hvert slide og de to spørgeskemaer er lavet med fuldstændigt samme design og opbygning således at dette har en genkendelseeffekt for en bruger der allerede har prøvet den ene. Dette er for at underbygge den tidligere hypotese om at brugere er vanepregede og finder sig bedst tilpas i velkendte situationer.



Figur 4.8: Phishing Internet - Spørgsmål 1 ud af 5

#### 4.3.3.1 Captivate del

I Captivate skelnes der mellem slide-typer som har forskellige egenskaber. Her skaber især slide-typen *question slide*, som bruges hver gang man skal lave spørgeskemaelementer som afkrydsningsfelter problemer. Problemet med disse slides er at de ikke kan kopieres og hver gang man skal bruge et *question slide* må man oprette et nyt og tilrette alt så det passer til de forrige. Dette er grund til meget tidsspilde når man som i tilfældet her skal bruge fem ens i hvert spørgeskema. Figur 4.9 viser både det standard oprettede *question slide* og det tilrettede slide, brugt i phishing testen.

Man kan spekulere på om folkene bag Captivate muligvis har nægtet kopiering af *question slides*, fordi de vil undgå rod med kopierede slides der bliver kopieret samme dataid, men problemet sænker udviklingstiden betydeligt selvom det kun er et mindre problem.

I en del af Captivates editor kan man se størrelse og position af den ting man har markeret. Denne egenskab benyttede jeg meget når jeg ville være sikker på at en overskrift var stillet samme sted som i de foregående slides, eller andre steder hvor Captivate ikke lige fik kopieret alt med.





The image shows a split-screen interface for a multiple-choice question. The left side is a white box with a black border containing the question and five radio button options (A-E). The right side is a blue box with a white background containing the question in Danish and five checkbox options (A-E). At the bottom of the white box are 'Clear', 'Skip', and 'Submit' buttons. At the bottom of the blue box is a 'Videre' button.

**Multiple choice**

Type the question here

- A) type the answer here
- B) type the answer here
- C) type the answer here
- D) type the answer here
- E) type the answer here

Clear Skip Submit

Hvad fik dig til at svare nej?

- A) Adressen var forkert
- B) Der manglede 'https' i adressen
- C) Der manglede en lås
- D) Siden virkede forkert
- E) Ved ikke

Videre

Figur 4.9: Phishing Internet - Question Slide

### *Screenshots*

Billederne benyttet til spørgsmålene blev, som vist på figur 4.8, ændret således at kun fyldte en del af billedet, for at gøre plads til knapper og tekst. Dette viste dog at Captivate ved optagning af screenshots blev oprettet med en sort farve som baggrund, hvilket kunne ses i kanterne af billederne. Dette var ikke synligt når man fyldte et slide helt ud med det opfangede billede, men især synligt når billedet blev gjort mindre. For at undgå dette udtrak jeg de optagede screenshots til billedredigeringsprogrammet Photoshop, og ændrede dem således at kanterne havde en gennemsigtig baggrund.

Selvom dette er et lille grafisk problem, kan det have konsekvenser da det enten unødigt kan forvirre brugerne eller foresage ekstra udviklingstid, da alle billeder der ikke fylder et helt slide så skal rettes til.

### *Animering af certifikat*

Angående internet-spørgsmålene blev det med Cert diskuteret om det var muligt at animere certifikatet, således at brugerne kunne trykke på låsen og få beskrivelsen af certifikatet frem, for at undersøge om dette var korrekt. Dette skulle så nødvendigvis implementeres ved at have et hurtigt skift mellem to næsten ens spørgeskemaslides.

Idet at Captivate så fast skelner mellem forskellige slides, kunne det dog desværre ikke lade sig gøre at få skift mellem slidesne til at virke så godt at det gav illusionen af at den viste web-side virkede som den ville i virkeligheden. Idet at slidesne ikke udelukkende bestod af et billede, insisterede Captivate på at lave en blød skiftende overgang mellem slides hvilket brød illusionen.

Konklusionen må desværre være at hvis man vil have mere visuelt animerede sider i Captivate, skal billederne stå alene i slidet.

### Opsummering

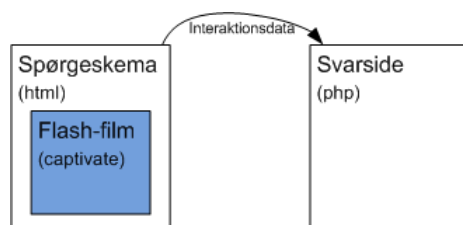
Opbygningen af spørgeskemaet gik rimelig godt. Jeg fik de ting med jeg gerne ville have og det gør ikke så meget at man ikke kan kontrollere certifikatet da dette så holder testen mere enkel.

Udviklingstiden føler jeg dog kunne have været hurtigere, hvis Captivate ikke var et så insisterende ikke udvikler-værktøj. Det at visse slides ikke kunne kopieres og andre småting ikke altid kunne kopieres med gjorde at jeg måtte gentage mange ting samt kigge tilbage for at se positioner på overskrifter og andet som skulle gå igen, for at disse ikke flyttede sig når man kom til næste slide. Yderligere kunne det mærkes begyndte Captivate at blive langsomt jo flere slides og billeder jeg fik med, og ændringer samt flytninger begyndte at halte lidt, dette selvom computeren der udvikles på er fra dette år og rimelig hurtig.

En mere optimal brugsfordeling mellem Captivate og normal html, ville være at lade så meget som muligt ligge på html-delen og bruge Captivate til de visuelle ting, som skal være animerede. Hvis denne metode var brugt på denne test ville det således kun være billedet af internetsiden/emailen, som var et indlejret flash-dokument, og designet og knapperne udenfor ville være html. Denne metode ville lade så meget som muligt være kode, og derved fuldt genbrugeligt og nemmere at udvikle, samtidigt med at man stadigvæk ville have den animerede del.

#### 4.3.3.2 Html del

Html-filen som indeholder og kører den Captivate producerede flash-fil (email.htm - bilag A.4.3 og internet.htm - bilag A.4.2), blev ligesom i wireless eksemplet modificeret således at data blev sendt via en html-form til en handler-side (inethandler.php - bilag A.4.4 og mailhandler.php - bilag A.4.5). Dette er vist på figur 4.10.



Figur 4.10: Spørgeskema og handler

Handleren udtrækker ved hjælp af *php* de nødvendige brugerinput, lægger dem op i en database samt konstruerer svarsiden hvori brugeren kan se sine svar

sammen med en beskrivelse af det korrekte svar på spørgsmålet.

Phishing	
PK	id
	q1
	q2
	q3
	q4
	q5

Figur 4.11: Phishing - Database tabel

Udtrækkene af dataene sker ved brug af to php funktioner; en der først deler hele teksten op i strenge, og en der derefter deler hver strenge op i dele, således at de korrekte brugerinput kan udtrækkes. Disse bliver derefter lagt op i to tabeller i en simpel database (en for hvert spørgeskema), med det simple formål at gemme testene fra de to spørgsmål. Tabelstrukturen kan ses på figur 4.11 og er en simpel struktur med én primærnøgle 'id', som er et tal der automatisk bliver talt en op hver gang der indsættes en ny række. Resten i tabellen er de opfangede brugerinfo.

Benyttelsen af databasen i dette eksempel er tilføjet hurtigt, da denne sektion er mere orienteret på Captivate's muligheder og sammenspillet mellem dettes nære omgivelser. Derfor vil der i en sådan opbygget tabelstruktur højest sandsynligt forekomme duplikater, hvor den samme person har prøvet det samme spørgeskema to gange. Hvis man skulle forbedre dette, ville det være nødvendigt at tilføje en 'bruger', og oprettelse samt login af denne, hvilket ville være uden for dette underprojekts omfang.

Spørgsmål 1 ud af 5 - Danske netbank

**Forkert!**

De svarede: **Nej**

- Adressen var forkert
- Der manglede 'https' i adressen
- Der manglede en lås

Adressen på denne side viser en ip-adresse istedet for Danske NetBanks egen adresse.

Pga. dette kan man ikke være sikker på om den viste side virkelig er Danske NetBank og det frarådes derfor at stole på denne side og indtaste nogen form for brugerinformation på den.

Figur 4.12: Phishing Internet - Svarside

Svarsiden indeholder et afskåret billede der viser det vigtige fra spørgsmålet.

I denne kan adressen ses i internet-spørgeskemaet og den del af emailen der indeholder det falske link i email-spørgeskemaet. Til venstre for dette kan man se om man har svaret korrekt eller forkert, og ens valg står markeret med grønt eller rødt; for rigtigt og forkert. Neden for dette kan man yderligere læse en kort beskrivelse af spørgsmålet, som indeholder beskrivelse af den korrekte besvarelse.

Figur 4.12 viser svaret for første spørgsmål i internet-spørgeskemaet. Svarene er stillet enkelt og overskueligt op og de kritiske emner i billedet er markeret med en rød cirkel for at gøre svarsiden så intuitiv og forståelig som muligt.

#### *Opsummering*

Med dette er et udviklet af et animeret spørgeskema for henholdsvis phishede emails og internetsider færdig.

Fokus har med disse spørgeskemaer været på forståelsen, og jeg har gjort meget ud af at gøre spørgsmålene så intuitive som muligt. Captivate har sine fordele ved muligheden for animerede dele konstrueret fra optagede screenshots, men det kræves at man kender Captivate godt, før at man kan få tilpasset de udviklede film til at være animerede spørgeskemaer.

### 4.3.4 Forskningens Døgn

I forbindelse med Forskningens Døgn, som er en årlig landsdækkende begivenhed hvor en række virksomheder og universiteter tager ud for at vise hvad de arbejder med, tog Robin Sharp og jeg til Lyngby Kulturhus for at vise phishing testen og fortælle om svindel på Internettet.

Selve testens opsætning og design virkede ret godt. Ingen virkede i tvivl om hvor man skulle trykke for at komme videre eller om positioneringen af informationerne. Test-computeren var en lille bærbar Dell-pc, hvorpå man også kunne se at størrelsen af testen var lige præcist så stor som den kunne være, for at det hele kunne passe på skærmen.

Idet den computeren var lånt fra instituttet var den derved nyopsat, hvilket gjorde at transitionen fra Captivate til handler-siden ikke virkede første gang. Dette fordi en nyopsat windowscomputer har et par indbyggede sikkerhedsadvarsler som kommer frem ved første opstart. Når man første gang havde klikket 'ja' til disse, forblev disse dog væk.

De fleste vi snakkede med forstod godt, eller havde hørt om sikkerhedsproblemer ved Internettet. Alligevel havde de ikke taget sig megen tid til at sætte sig ind i hvad man skulle gøre for at undgå dem. Under halvdelen af testpersonerne

kendte til funktionen af låsen, og disse forstod kun at den “gerne skulle være tilstede”. De personer der havde mest problemer med testen, var personer som ikke kunne relatere til eksemplerne.

Som det virker er nogle internetbrugere af den ældre generation meget vaneprægede, og mest holder sig til de samme 5-10 sider på internettet, som f.eks. en web-mail, amazon.com og en netbank. Denne type brugere kunne typisk ikke forholde sig til om Paypal-siden (som var helt korrekt og sikker) var til at stole på eller ej, fordi de ikke kendte til den. Dette er et aspekt som er svært at gøre noget ved, da man ikke direkte kan forudse hvilke sider brugerne er vant til og brugerne gerne skulle kunne identificere om en side de ikke havde set før var sikker eller ej.

Det kom til min forståelse at svarsidens logik måske var en anelse for strikt. Idet logikken er lavet således at man får 'forkert' med mindre man svare alt nødvendigt korrekt, får brugerne en et for negativt svar med hvilke de måske kunne miste modet lidt. En bedre metodik for disse ville være at ændre logikken således at man får 'korrekt' hvis man blot svarer ja/nej rigtigt. Derunder skal undersvarene stadigvæk markeres med farvekoderne således at man kan se at ens svar kun delvist er korrekt.

På denne måde holder man en slags positivt ladet svarindstilling hvilket ikke tager modet fra testpersonerne og muligvis leder dem til at blive mere interesseret i emnet, frem for at blive fornærmede. Brugere har en meget større chance for at blive interesserede i et emne, hvis de føler at de ikke er helt håbløse til det. På denne måde kan man holde illusionen lidt for dem, samtidigt med at man stadigvæk holder tråden i testen, og får forklaret til dem hvordan de kan blive endnu bedre. Man kan så stadigvæk holde statistikken fuldstændig kritisk når man vurderer hvad personerne samlet i statistikken reelt kan.

En anden mindre design forbedring ville være at skille svarsiden op i en side for hvert spørgeskema, på denne måde får man delt informationerne mere op, sådan at det er nemmere at overskue og testpersonerne så vil have lettere ved at overskue og læse svarteksten. Desuden kunne det muligvis være interessant af have det som standard i denne slags tests, at tilføje et “Lærte du noget af denne test?” -spørgsmål efter den sidste svarside.

Statistisk set var det ikke mange informationer vi fik opfanget selvom vi fik mere indblik i brugernes opfattelser af phishing, ved samtaler med dem på dagen. Derfor vil jeg ikke gå igennem statistikkerne af de opfangede phishing data her, men vente til der er blevet opfanget lidt flere informationer. Dette vil så blive beskrevet i det efterfølgende kapitel.

Forskningens døgn gav gennem samtalerne med forskellige bruger et syn på

hvordan de bruger internettet især. En ældre herre skulle næsten overbevises om at phishing virkelig var et problem da han kun var vant til at besøge et begrænset antal internetsider.

Det blev især bekræftet at brugerne er meget vaneprægede og ydermere også er konkurrenceprægede. Brugerne virkede som om de synes det var pinligt hvis de havde svaret størstedelen forkert og nogle undskyldte endda allerede før de begyndte testen.

Stortset alle af brugerne forstod spørgeskemaerne med det samme, selv om de ikke nødvendigvis vidste meget om phishing, hvilket tyder på at de konstruerede animerede spørgeskemaer har fordele i at gøre spørgsmålene forståelige.

## 4.4 Opsummering

Forbedring af spørgeskemaer omhandler for det meste analysen. Det handler om at gennemtænke situationen sådan at spørgeskemaet holder en intuitiv rød tråd og holdes præcist og klart at forstå, samtidigt med at man får bedt om de nødvendige informationer. I forbindelse med tekniske emner som sikkerhedsbevidsthed, kan det være en fordel at benytte et værktøj som Captivate til opbygningen af spørgeskemaet, da man så derved nemt kan få inkluderet interaktive animerede dele hvilke 'emulerer' et rigtigt program. Således har brugeren har meget nemmere ved at visualisere spørgsmålet for sig.

I forbindelse med forskningens døgn fik jeg igennem samtaler med brugere forstået at brugerne kan være meget vaneprægede. Nogle af brugerne stolede endda ikke på internetsider, som ellers var ganske officielle og meget benyttede websider, fordi de ikke var en af de websider de normalt besøgte. Det skal dertil huskes, at ikke alle brugere benytter computeren som tiltænkt, men måske kun benytter en del af den. Derved får de nogle brugsvaner man måske ikke helt havde tiltænkt. Det er vigtigt at være opmærksom brugernes vaner når man bygger spørgeskemaer, da det oftest er dem man forsøger at trække ud.

Adobe Captivate gør det muligt nemt at bygge animationer af programmer, for på den måde at kunne inkludere animationer i spørgeskemaer. Man kan dog godt mærke at Captivate ikke er designet til dette formål. Ved benyttelse af Captivate til animerede spørgeskemaer vil det derfor anbefales at man kun benytter Captivate til den animerede del af spørgeskemaet, og derfra indlejrer dette i en html-side, hvorfra man kan holde alt de udenomlæggende informationer i html, hvilket håndterer dette meget bedre end Captivate.

Ved udvikling af spørgeskemaer kan det ses at det kræver mere viden og tid at lave de forbedrede mere grafiske spørgeskemaer. Man må dertil vurdere til hvert spørgeskema hvor meget tid og hvor mange ressourcer man vil bruge. Det

er en stor fordel at holde spørgeskemaer under samme emne/gruppe med samme størrelse og design, på denne måde sparer man både udviklingstid og man holder udseendet ens for brugeren.

For at få så mange brugere som muligt, kan man med fordel at tilføje et brugermotiv. Til sikkerhedsbevidstheds spørgeskemaer, har jeg fundet ud af at det passer godt at tilføje indlæring i slutningen af hvert spørgeskema. På denne måde kan man opfordre brugere til at prøve spørgeskemaet, imod at de kan lære mere om emnet.





# Website for Sikkerhedsbevidsthed

---

I dette kapitel beskrives udviklingen af et online database-orienteret website, der har til formål at tilbyde spørgeskemaer inden for sikkerhedsbevidsthed. Brugere vil kunne prøve spørgeskemaer som vil teste deres viden og samtidigt tilbyde indlæring inden for emnet, hvilket vil virke som sidens motiv. På denne måde vil brugernes tendenser blive optaget gennem spørgeskemaerne samtidigt med at brugerne føler de lærer noget af det.

Det skal igennem de opfangede brugerinformationer vurderes hvor vidt animerede spørgeskemaer er en decideret forbedring over de simple og hvordan brugerne henholder sig til disse.

Siden vil blive opbygget som *php*-forbedret *html*, bygget op omkring en *mysql*-database. Use-case modellen fra UML [7] vil blive benyttet til analysen, men andet UML vil ikke blive anvendt, da web-sider som udgangspunkt er stateless<sup>1</sup> og ikke objektorienterede.

Den følgende kravspecifikation vil specificere opgaven i detaljer.

---

<sup>1</sup>Med 'stateless' menes det, at en enkelt html-side som udgangspunkt ikke ved hvor den er i forhold til andre dokumenter på websiden.

## 5.1 Kravspecifikation

Der ønskes udviklet et online sikkerhedsbevidstheds site, der skal gøre det muligt for brugere at teste deres viden om computersikkerhed gennem animerede spørgeskemaer. På denne måde vil brugernes tendenser blive opfanget og lagres for at kunne vises statistisk.

Sidens motiv skal være indlæring lært gennem spørgeskemaerne, for på den måde at trække brugere til siden ("Vil du lære mere om computersikkerhed?").

Underfunktioner som systemet skal indeholde:

- Brugeroprettelse og login, for at forhindre duplikeret data.
- Statistik

Siden skal opbygges med henblik på brugbarhed og intuitivitet. Således skal siden se professionel ud, og derved gerne have et look-and-feel som andre sider, så brugerne på en måde føler de kender siden når de først ser den. Derudover skal det også være meget nemt for brugerne at oprette sig. Det underliggende database-design skal bygges med henblik på udvidelser og udtrækning til statistik.

### 5.1.1 Identifikation af use-cases

En use-case er en funktion udført på systemet af en bruger af systemet kaldet en aktør. Dette system benyttes af følgende tre aktører:

**Anonym Bruger** er brugeren som blot har surfet ind på siden, uden at have logget ind. Denne bruger kan kun se forsiden af websiden og kan derfra ikke prøve spørgeskemaer men kun oprette en ny bruger.

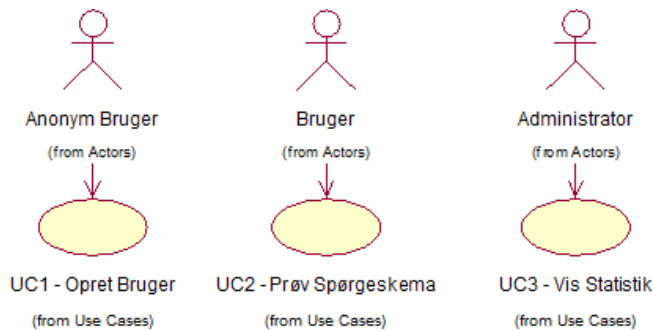
**Brugeren** kan, efter login, afprøve spørgeskemaer.

**Administratoren** kan, når logget ind, se statistik over besvarelserne.

For at kunne identificere hver bruger bliver det nødvendigt at brugere skal oprette sig i systemet for at få tilgang til spørgeskemaerne. Derfor kan den anonyme bruger kun se forsiden og derfra vælge at oprette sig som bruger. Først når brugeren er oprettet kan han tilgå alle de spørgeskemaer han ønsker. Use-case diagrammet på figur 5.1 viser sammenhængen mellem aktører og use-cases.

## 5.2 Iterationsplan

Udvikling af systemet er delt op i iterationer. Hver iteration indeholder hele udviklingen af et afgrænset område sådan at systemet udvikles fuldstændigt i



Figur 5.1: Use-case diagram

mindre dele af gangen, hele tiden sådan at man bygger videre på en foregående udviklet og fuldt virkende testet version af systemet. I slutningen af en iteration vil det blive defineret hvad der vil blive udviklet i den næste.

Den første iteration omgrænser den mest grundlæggende del af systemet, for at det er funktionelt. Dette indbefatter derved også det grundlæggende design af web-siden samt opbygningen af data modellen. Af spørgeskemaer vil der her kun blive benyttet de to jeg udviklede i sidste kapitel (wireless samt phishing).

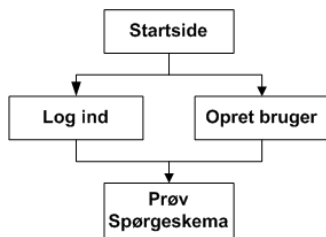
Use-casene i den første iteration er disse:

- UC1 - Opret Bruger
- UC2 - Prøv Spørgeskema (kun wireless og phishing)

Således vil iteration 1 udgøre en designet webside med en fuldt fungerende database, hvor brugeren kan oprette sig samt prøve spørgeskemaerne.

## 5.3 Iteration 1

Systemstrukturen af af websiden der bliver udviklet i denne iteration er vist på figur 5.2.



Figur 5.2: Struktur over website

Fra startside kan der enten logges på med et gyldigt brugernavn og kodeord, eller man kan oprette en ny bruger. Først når man er logget ind kan man benytte spørgeskemaerne.

Dette er den basale struktur af systemet som vi blive udviklet i denne iteration. Som beskrevet i kravspecifikationen vil der blive lagt vægt på at siden forbliver enkel og nem at bruge.

### 5.3.1 Use-case model

Use-case et og to, vil her blive beskrevet på use-case form. Idet use-casene er så simple vil system sekvens diagrammer ikke blive beskrevet.

#### 5.3.1.1 UC1 - Opret Bruger

Denne use-case omhandler opretningen af en bruger i systemet. Det er vigtigt at denne virker og er meget let af udføre.

<i>Mål:</i>	At oprette en bruger i systemet
<i>Type:</i>	Hovedfunktion
<i>Forudsætninger:</i>	Ingen
<i>Succes tilstand:</i>	Brugeren bliver oprettet som bruger i systemet
<i>Fiasko tilstand:</i>	Brugeren bliver ikke oprettet
<i>Primær aktør:</i>	Den anonyme bruger
<i>Startbegivenhed:</i>	Den anonyme bruger vælger at oprette sig i systemet
<i>Prioritet:</i>	Høj
<i>Afviklingstid:</i>	Kort
<i>Hyppighed:</i>	Mellem

**Success scenarie:**

1. Brugeren vælger at oprette sig som bruger
2. Brugeren indskriver nogle generelle oplysninger
3. Brugeren bliver oprettet som bruger i systemet

**Fiasko scenarie:**

1. Brugeren vælger at oprette sig som bruger
2. Brugeren indskriver nogle generelle oplysninger
3. Brugeren bliver ikke oprettet i systemet, og kan ikke prøve spørgeskemaer.

For at få præciseret statistik fra spørgeskemaerne, er der tilføjet nogle generelle spørgsmål når en bruger opretter sig. Ved at have disse informationer om hver person der prøver spørgeskemaerne, vil statistikken kunne relateres til køn, alder osv.. Informationer skal som udgangspunkt ikke være helt personlige info som cpr-nummer, adresse eller email, og det skal helst ikke tage særlig lang tid at oprette sig som bruger.

**5.3.1.2 UC2 - Prøv spørgeskema**

Denne use-case omhandler afviklingen af spørgeskemaer på siden. Spørgeskema afviklingen er hjertet af siden.

<i>Mål:</i>	At afvikle et spørgeskema
<i>Type:</i>	Hovedfunktion
<i>Forudsætninger:</i>	Brugeren er logget ind
<i>Succes tilstand:</i>	Brugeren får prøvet et spørgeskema
<i>Fiasko tilstand:</i>	Et spørgeskema kan ikke afprøves
<i>Primær aktør:</i>	Brugeren
<i>Starthbegivenhed:</i>	Brugeren vælger et spørgeskema
<i>Prioritet:</i>	Høj
<i>Afviklingstid:</i>	Kort
<i>Hyppighed:</i>	Ofte

**Success scenarie:**

1. Brugeren vælger et spørgeskema
2. Spørgeskemaet bliver afviklet for brugeren
3. Brugeren returnerer til spørgeskema oversigt

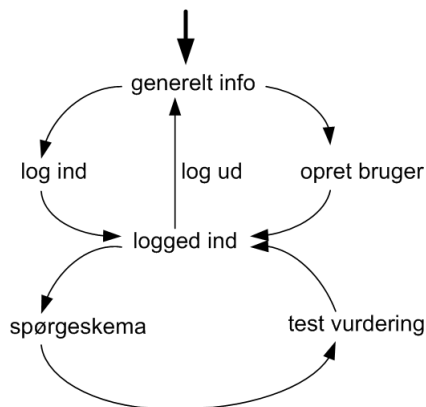
**Fiasko scenarie:**

1. Brugeren vælger et spørgeskema
2. Spørgeskemaet bliver ikke afviklet for brugeren
3. Brugeren returnerer til spørgeskema oversigten (i bedste fald)

### 5.3.2 Design

Den første iterations' sidestruktur er som allerede beskrevet rimelig enkel. En anonym bruger skal oprette sig for at komme til at prøve spørgeskemaer. De dele der skal bestemmes før implementeringen kan påbegyndes, er en opbygning af databasens struktur, samt et design af brugergrænsefladen.

Strukturen for websiden er vist på figur 5.3. Som det kan ses kommer man først frem til en forside med noget generelt info, hvorfra man kan vælge at oprette sig som bruger eller logge ind på siden. Oprettelsen af brugere er meget enkel, da dens formål er at sikre af samme person ikke prøver det samme spørgeskema et unødigt antal af gange, og samtidigt at gøre det muligt at optage nogle generelle personinformationer som alder, køn og computerfærdighed. Disse data kan være gavnlige senere ved oprettelse af statistikker. Når en bruger er logget ind kommer han frem til en side hvor der kan vælges mellem spørgeskemaer og der ved at holde musen over spørgeskemanavne kan ses en beskrivelse af hvad det enkelte spørgeskema går ud på. Når et spørgeskema bliver valgt, bliver dette vist, hvorefter brugeren bliver bedt svare om han fik noget ud af spørgeskemaet, før han bliver returneret til startside for brugere der er logget ind.

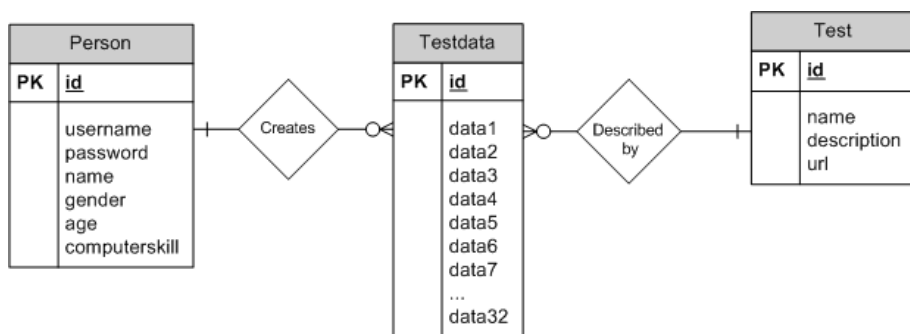


Figur 5.3: Side struktur

Overordnet skal det være nemt for en bruger at oprette sig samt prøve og lære af spørgeskemaerne.

### 5.3.2.1 Data model

Systemets database tager vare på de optagne data fra spørgeskemaerne og de oprettede brugere. For at beskrive denne har der lavet en ER-model [3], som er vist på figur 5.4.



Figur 5.4: ER diagram

Hele databasen er centreret omkring tabellen *Testdata* som opfanger brugerdata fra alle spørgeskemaer på websiden. De forskellige test er beskrevet af tabellen *Test* og den person hvilket testdata tilhører, er beskrevet i tabellen *Person*. Relationerne kan ses med forgreningen ved at en person kan have mange rækker

*Testdata*, men en række i *Testdata* skal have kun en tilknyttet person. Herved passer også netop en beskrivelse af en test muligt med mange rækker i *Testdata*. Derfor skal der altså være oprettet en beskrivelse af testen og en person som laver testen, for at testen kan passe ind i modellen.

Ved at holde testdata separat for testbeskrivelse kan alle disse data holdes i en enkelt tabel og dets beskrivelser kan nemt senere udbygges hvis det er nødvendigt.

Modellen opfylder betingelserne for tredje normalform ved at alle tabeller har en primærnøgle og ingen gentagne felter. Der er desuden ingen sammensatte primærnøgler og ingen attributter der ikke er direkte afhængige af primærnøglen.

### 5.3.2.2 Brugerflade design

Brugerfladen af en webside som denne skal lægge meget vægt på at siden ser intuitiv og overskueligt ud, for at få så mange brugere med som muligt. Til dette er den første beslutning der skal tages, størrelsen af websiden.

Brugeres computere benytter forskellige opløsninger, hvilket vil have indvirkning på hvordan siden ser ud på deres browsere. Ved brug af internetbrowsere bliver noget af denne opløsning dog benyttet af browseren selv til adressefelt, knapper mm.. Derfor er den optimale størrelse af en webside også betinget af hvilken browser der bruges. W3schools [12] har statistikker over generelt benyttede opløsninger og browser benyttelse. Disse statistikker vil jeg benytte til at finde den optimale webstørrelse.

#### Display Resolution

The current trend is that more and more computers are using a screen size of 1024x768 pixels or more:

<b>2007</b>	<b>Higher</b>	<b>1024x768</b>	<b>800x600</b>	<b>640x480</b>	<b>Unknown</b>
January	26%	54%	14%	0%	6%
<b>2006</b>					
July	19%	58%	17%	0%	6%
January	17%	57%	20%	0%	6%

Figur 5.5: Opløsnings Statistik

Figur 5.5 viser en statistik over benyttelsen af skærmopløsninger. Det kan ses at de mest benyttede opløsninger er 1024x768 eller større. 50% bruger stadigvæk 1024x768 men denne størrelse bliver mindre med tiden da folk rykker til endnu større opløsninger. Den mindre opløsning 800x600 bruges af omkring 14%, men



er stærkt på vej ned.

Idet vi skal bruge websiden til grafik, som fylder meget på skærmen, vil jeg vælge at tilpasse mine websider til 1024x768. De sidste nedafgående 10% vil så skulle benytte scrollbarne for at kunne se websiden helt. Da browseren tager noget af pladsen, og da der desuden skal være plads til overskrift, design, og menu på websiden ville 800x600 simpelthen være for lidt for at evt. mere medie-krævende Captivate spørgeskemaer ville kunne være der.

**Browser Statistics Month by Month**

2007	IE7	IE6	IE5	Fx	Moz	S	O
April	19.2%	37.3%	1.7%	32.9%	1.3%	1.7%	1.7%
March	18.0%	38.7%	2.0%	31.8%	1.3%	1.7%	1.6%
February	16.4%	39.8%	2.5%	31.2%	1.4%	1.7%	1.5%
January	13.3%	42.3%	3.0%	31.0%	1.5%	1.7%	1.5%

Figur 5.6: Browser Statistik

For at finde ud af den optimale størrelse af websiden, må jeg også kigge på benyttede browsere og deri finde ud af hvilke browsere jeg vil understøtte. Forskellige browsere har forskellige måder at tolke html-sider og hvis man ikke understøtter og tester sine sider i dem, kan man risikere at siderne ikke bliver vist ordentligt.

På figur 5.6 kan det ses at Internet Explorer 6, som er den der kom fra start med Windows XP, er den mest benyttede internetbrowser for tiden. Dog er dens brug på vej ned, pga. Internet Explorer 7 som kommer med det nyudsendte Windows Vista og Firefox. Windows XP maskiner, vil også med automatiske opdateringer sat på, opgradere Internet Explorer til version 7. Mozilla, Safari og Opera er dog næsten helt ubenyttede med under 2% hvorimod Firefox holder omkring en trediedel af brugerene. Derfor vil jeg vælge at understøtte Internet Explorer 6 og 7, samt Firefox.

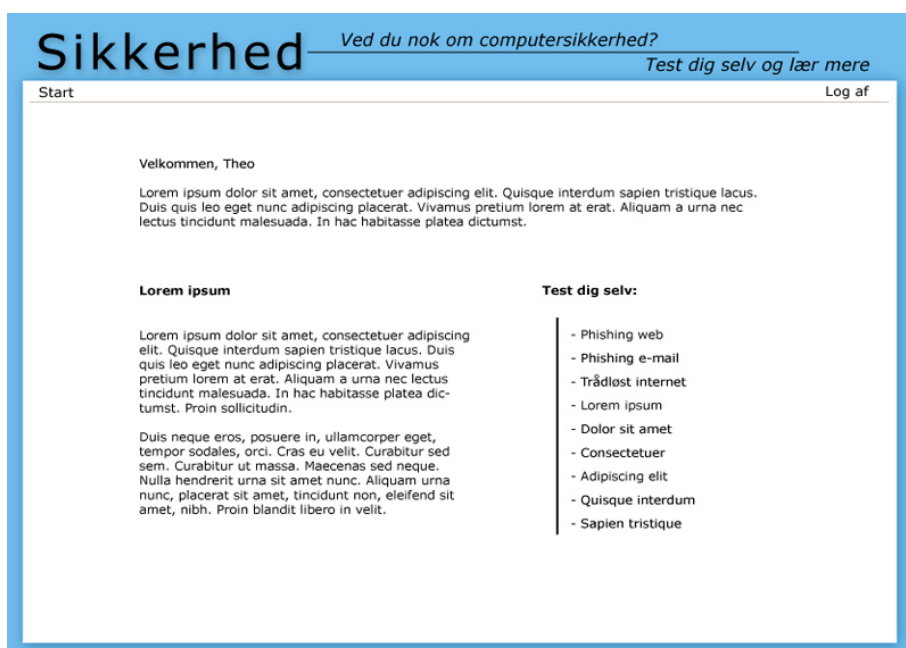
	1024x768	800x600
Internet Explorer 7	1000x620	780x435
Internet Explorer 6	995x590	770x420
Firefox	1000x600	780x430

Tabel 5.1: Brugbare browser opløsninger (pixels)

Som før sagt, benytter browserne noget af skærmens opløsning til adressebar, knapper mm.. Derfor er den brugbare opløsning forskellig fra browser til browser, da de har hvert deres layout. På figur 5.1 er vist en liste over de 'rigtige'

opløsninger fra hver browser. Den er lavet ved at tage et screenshot af hver browser i en opløsning, og derefter måle den brugbare opløsning i et billedbehandlingsprogram.

Det kan ses at IE6 bruger lidt mere af skærmen end de andre, derfor har den, den mindste brugbare opløsning hvilket vil bestemme den opløsning designet skal laves ud fra. Ved 1024x768 er denne størrelse 995x590 pixels. Idet der altid kan scrolles ned i en browser uden at det ser u hensigtsmæssigt ud vil det vigtige være at designet overholder de 995 pixels i bredden.



Figur 5.7: Brugerflade design

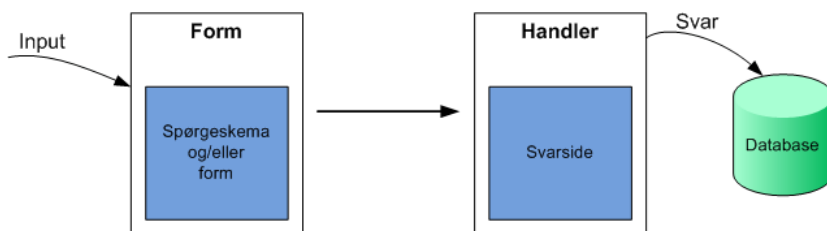
Figur 5.7 viser designet af brugerfladen. Designet er foreløbigt lavet som et billede i illustrator, og det viste indhold er designet af siden brugeren ser når vedkommende er logget ind. Der er benyttet mumletekst fra [www.lipsum.com](http://www.lipsum.com) til at fylde de steder hvor teksten ikke er bestemt endnu. F.eks. skal tekstfeltet til venstre virke som en beskrivelse af et spørgeskema, som vises når man holder musen over et spørgeskema til højre.

Selve designet skulle forestille et ark papir, der ligger på en blå baggrund. Dette design er meget simpelt, for at holde siden enkel og overskuelig.

### 5.3.3 Implementering

Jeg vil til denne side benytte mit eget domæne [www.tre-ben.dk](http://www.tre-ben.dk), hvilket vil sige at siden kommer til at være tilgængelig fra <http://sikkerhed.tre-ben.dk> eller <http://www.tre-ben.dk/sikkerhed>. Siden tilbyder *php* samt database tilgang via *mysql*.

Siden vil blive bygget med *XHTML* (af standarden *transitional*), *JavaScript*, *Cascading Style Sheets* samt *php* til det mere dynamiske som databasetilgangen.



Figur 5.8: Typisk struktur af form og handler

Figur 5.8 viser sammenhængen mellem form og handler hvilket bliver benyttet i implementeringen. Form siden er den side hvor spørgeskemaet eller andet forminput er. Disse bliver så ved submit bliver sendt til en handler side som håndterer informationerne ved at vise et svar for brugeren og lægge oplysningerne op i databasen.

Implementeringsrækkefølgen er som følger:

- Database
- Brugerflade
- Login håndtering
- Opret Bruger (UC1)
- Afprøv Spørgeskema (UC2)

Først vil ER-modellen blive omdannet til tabeller og sql-sætninger, således at databasen er klar til at blive benyttet af de omkringliggende dele af systemet. Derefter vil brugerfladen blive implementeret efterfulgt af use-case 1 og 3 som specificeret tidligere.

Idet internet-sider er stateless og ikke objektorienterede vil udviklingen af denne

side være orienteret omkring html-siderne som beskriver det grafiske, og dets php-tilknytning til databasen.

### 5.3.3.1 Database

For at siden kan implementeres helt skal databasen først implementeres. Derfor er der fra ER-modellen blevet identificeret tre tabeller: *test*, *person* og *testdata*. Associationerne er blevet omdannet til to fremmednøgler: *person\_id* og *test\_id*, som begge ligger i *testdata* og refererer til henholdsvis den person der ejer testdata og den test den tilhører.

Alle primære nøgler er lavet som heltal, der automatisk tæller en op når en ny række bliver indsat. Dette er både for enkelthedens skyld og fordi computere arbejder bedst med tal, og derfor bruger færre kræfter på at sammenligne disse. Nu er dette ikke så vigtigt i vores tilfælde da denne database ikke skal håndtere ret mange personer på samme tid. Sql-sætninger der repræsenterer oprettelsen af disse tabeller findes i bilag [A.5.1](#).

Med de to associationer kan alle testdata forbindes til en person og en test-beskrivelse, og de optællende primærnøgler sørger for at hver ny række bliver oprettet med et unikt id.

Med den grundlæggende datamodel oprettet fysisk, kan resten af siden nu blive opbygget og testet med denne.

Person	Test	Testdata
<u>id</u> username password name gender age computerskill	<u>id</u> name description url	<u>id</u> person_id test_id data1 data2 ... data32

Figur 5.9: Tabeller i databasen

Figur [5.9](#) viser tabellerne i den fysiske database. De primære nøgler er understregede og fremmednøglerne er skrevet som tabellen de refererer til, efterfulgt af en understreg og derefter feltnavnet.

### 5.3.3.2 Brugerflade

I design-afsnittet blev den generelle brugerflade bestemt og et billede af den blev oprettet i Adobe Illustrator. Måden jeg har implementeret dette på, er at jeg har klippet billederne ud af designet fra Illustrator og sat dem ind i tabeller i en html-side. Illustrator indeholder en funktion hvor man kan vælge hvordan siden skal klippes ud og så sætter den det selv ind i tabeller i en html-side. Udklippet kan ses på figur 5.10.



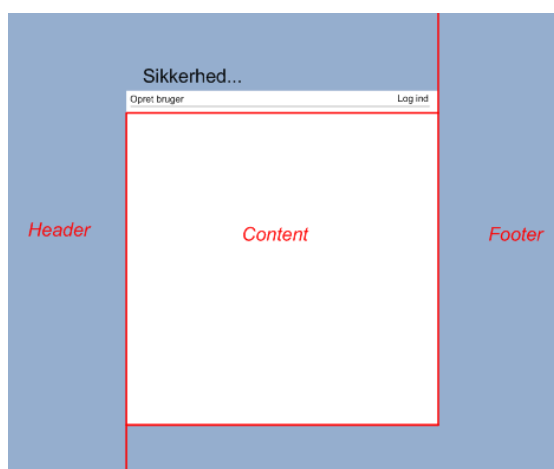
Figur 5.10: Brugerflade design - udklippet

Med den genererede html-fil med tabeller fra Illustrator, har jeg fjernet og tilpasset tabellerne sådan at unødvendige billeder ikke er med, og skal hentes af brugeren hver gang siden hentes. Jeg har desuden fjernet det hvide midterbillede, således at der kan fyldes normalt indhold ind. De to billeder, der bestemmer skyggerne i hver side af designet, er blevet rettet til så de kun er 1 pixel høje, og derved er sat ind således at de gentager sig selv alt efter hvor højt designet er. På denne måde kan selve designets højde bestemmes efter hvad der bliver fyldt ind i det hvide felt, og resten af designet passer sig ind vertikalt efter dette og ser stadigvæk godt ud.

Med brugerflade-designet lavet så det så godt ud i browserne, har jeg delt det op i en *header* (bilag A.5.2) og en *footer* (bilag A.5.3) -side. På denne måde kan

jeg ved hjælp af php-funktionen `import()`, inkludere headeren i starten af en ny side, og derefter footeren i slutningen, sådan at designet hele tiden er det samme, og jeg kan holde ændringer af header og footer et enkelt sted, for alle siderne.

Opdelingen af brugerfladen i header og footer er illustreret på figur 5.11. Det kan ses at det første der bliver tegnet i browseren er headeren, derefter kommer indholdet af siden hvorefter footeren til sidst bliver tegnet. Dette er en nem metode til at holde koden separat, sådan at man kan koncentrere sig om indholdet af siderne, og kan ændre det overordnede design et enkelt sted hvis man vil.



Figur 5.11: Brugerflade - Opdeling i header og footer

Jeg har valgt at lade brugerfladen være lavet på den måde at brugeren egentlig henter hele designet hver gang han går til en ny side. Alternative løsninger med et fast design (iframe, frames), lader ikke siden være af en dynamisk højde, hvilket jeg har brug for. Den dynamiske højde jeg får i mit design, tillader at siden kan blive af variabel højde alt efter hvor meget jeg putter ind i den, og jeg slipper for en scrollbar, der fylder midt i siden og ikke passer ind i designet.

Desuden er der som forklaret gjort en indsats, for at selve designet er så enkelt og pladsbesparende, at brugeren ikke vil mærke at billederne bliver genindlæst.

### 5.3.3.3 Login håndtering

Alle spørgeskemaer på siden kan kun tilgås ved at oprette en bruger og logge ind på siden. En bruger bliver oprettet i databasens Person-tabel og disse informationer bliver der sammenlignet med ved login.

Det kodeord, der bliver indtastet i enten *Opret Bruger* formen eller ved login, bliver med det samme krypteret med MD5, hvilket er en envejs krypteringsalgoritme. På denne måde ligger alle kodeordene krypteret inde i databasen og ved login bliver de krypterede strenge sammenlignet, for at validere en bruger.

Når en bruger logger ind bliver der oprettet en browser session, hvilken indeholder id-nummeret på brugerens række i databasen. Alle sider kontrollerer på denne sessionsnøglen allerede i headeren hvorved der kan tages stilling til om brugeren har adgang til den pågældende side, eller skal dirigeres automatisk videre til en 'ingen adgang' side, inden det forespurgte indhold bliver vist. Dette sørger for at holde internetsidernes state, sådan at siderne kan huske brugeren indtil browseren bliver lukket, sessionen timer ud, eller sessionen bliver lukket.

Login håndteringen (logind.php i bilag A.5.9 og logindh.php i bilag A.5.10) på dette website er rimelig simpel og opfylder fint for dets formål, hvilket er at tvinge brugerne til at logge ind sådan at de opfangede testdata kan refereres til personer.

#### 5.3.3.4 Opret Bruger

Sektionen til oprettelse af en ny bruger kan ses på figur 5.12. Denne er konstrueret til at have en kort brugstid sådan at brugere nemt kan oprette sig.



The screenshot shows a web form titled "Opret bruger" (Create user) on a page with a blue header. The header contains the word "Sikkerhed" in large white text, followed by the text "Ved du nok om computersikkerhed?" and a link "Test dig selv og lær mere". Below the header, there are two links: "Opret bruger" and "Log ind". The form itself is titled "Opret bruger" and contains the following fields and options:

- Brugernavn: [text input]
- Kodeord: [text input]
- Gentag kodeord: [text input]
- Fuldt navn: [text input]
- Alder: [text input]
- Køn:  Mand  Kvinde
- Computerfærdighed:  Dårlig  Mellem  God
- Send [button]

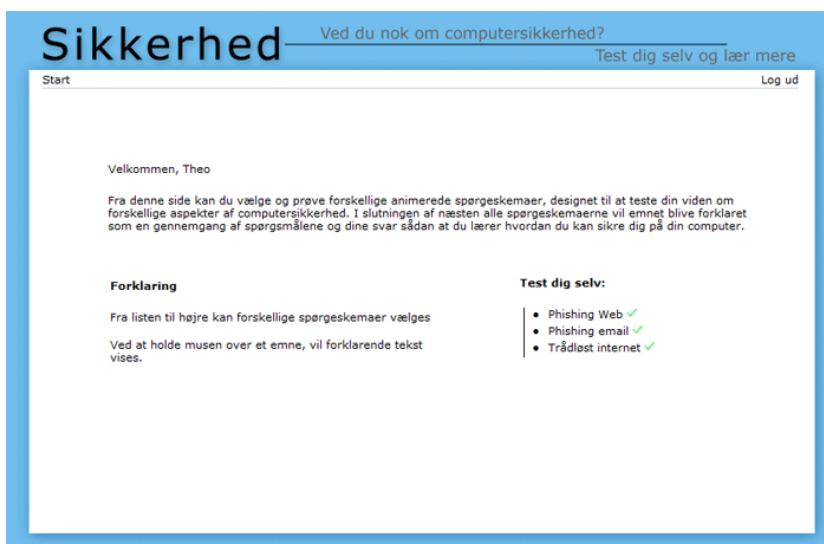
Figur 5.12: Opret Bruger

Der er som tidligere forklaret ingen krav om email, men brugere skal specificere alder, køn samt computerfærdighed i tre grader. Alle felter bliver javascript valideret, således at brugeren får en besked hvis han indtaster ikke valide informationer. Dette sørger også for at holde konsistensen af data i databasen oppe.

Html-formen på siden (opretbruger.php bilag A.5.12) sender informationerne videre til dens handler side (opretbrugerh.php bilag A.5.13) hvilken kontrollerer om det valgte brugernavn er unikt og derefter indsætter informationerne på en ny række i person-tabellen.

### 5.3.3.5 Afprøv Spørgeskema

Med brugeren logget ind kommer han ind på en side hvor han kan se de mulige spørgeskemaer og vælge disse. Denne side hedder start.php (bilag A.5.8) og kan ses på figur 5.13.



Figur 5.13: Hovedside for brugere der er logget ind

På denne side bliver brugeren budt velkommen og den generelle målsætning for siden forklares igen. På siden kan brugeren vælge mellem de mulige spørgeskemaer og når han holder musen over linket, vises der i stedet for teksten til venstre en forklaring af hvad spørgeskemaet går ud på.

Funktionaliteten ved at lade teksten blive ændret automatisk uden at genhente websiden, er lavet ved at lade teksten være et enkeltlinje tekstfelt til overskrif-



ten og et flerlinjet tekstfelt til brødteksten, og så style disse således at de ligner resten af teksten. En javascriptfunktion med en enkelt switch forgrening bestemmer, alt efter hvilket id den får som parameter, hvilket spørgeskema teksten skal referere til. Antallet af valg i funktionen er bestemt fra databasen, i og med at denne bliver genereret dynamisk derfra via. php.

Herudover er det også blevet lavet således, at der vises et lille hak ude ved siden af spørgeskemalinket hvis brugeren allerede har udfyldt denne før. Hvis dette er tilfældet, kan brugeren godt prøve spørgeskemaet igen, men dennes informationer vil ikke blive lagt op i databasen, sådan at databasen holdes sammenlignelig på personniveau.



Figur 5.14: Phishing Internet spørgeskemaet tilpasset til websiden

De to phishing spørgeskemaer er sammen med det mindre trådløse blevet tilføjet som valgbare spørgeskemaer på siden. Idet phishing spørgeskemaerne oprindeligt var designet til at være alene på en webside, er de blevet ændret således at de har en mørkeblå kant udenom indholdet. Dette er for at separere spørgeskemaet fra sidens omkringliggende design så det ikke ser rodet ud. På figur 5.14 kan det ene af spørgeskemaerne ses som det ser ud tilpasset til websiden.

For sammenhængens skyld blev det tilføjet at det sidste felt i testdata-tabellen (data32), skulle holde teksten 'OK' for en færdig oploadet tekst. Dette er i tilfælde af at spørgeskemaer bliver delt op og oploadet til databasen i flere træk,

hvilket til tider kan være nemmere end at hive form-informationer med over flere sider.

I slutningen af hver test, bliver brugeren sendt videre til en slutside, hvorpå han bliver spurgt om han fik noget ud af testen. Denne side (testslut.php bilag A.5.15) er en enkelt lille html form som virker som sin egen handlerside, sådan at den ved forsendelse af formen sender informationerne til sig selv, som så opdaterer rækken med enten et JA eller et NEJ. For at kunne virke sender testsiden id-nummeret på den række der blev indsat i tabellen, således at siden kan opdatere den. Meningen med denne forespørgsel var at få statistik om hvorvidt brugeren vidste indlæringsinformationerne i forvejen eller ej.

### 5.3.4 Test

Systemet der er blevet udviklet i denne iteration, er løbende blevet testet, så jeg vil i denne sektion udføre en systemtest, ved at oprette en ny bruger og gå alle funktioner igennem. Vægten vil blive lagt på helheden og hvordan de forskellige dele virker sammen.

Testen af delene vil virke ved først at teste javascriptvalideringen af formene og derefter kontrollere i databasen om informationerne blev gemt korrekt.

#### Brugeroprettelse og login

Valideringen af *Opret bruger* virker som den skal, og man får først lov til at fortsætte når alle felter er udfyldt. Dog er de tre grader af computerfærdighed; Dårlig, Mellem og God, nok ikke helt pædagogiske nok. Brugere kan ikke lide at stemple sig selv med teksten 'dårlig' og en normal computerbruger kunne godt finde på at stemple sig som 'god'. Derfor har jeg ændret de tre tekstangivelser til henholdsvis: Uerfaren, Erfaren og Ekspert.

Ved at trykke på 'send' kom beskeden "Bruger 'Theo' er allerede oprettet" frem, da jeg allerede havde oprettet denne bruger. Ved at gå tilbage og ændre brugernavnet blev brugeren oprettet og den nyoprettede bruger blev automatisk logget ind og videresendt til startsiden.

Ved undersøgelse på de opfangede testdata, fandt jeg ud af at personens id i databasen ikke blev indsat ordentligt hvilket gjorde at der blot stod '0' istedet for. Dette blev nemt rettet ved database *inserts* i spørgeskemaernets handler, da de var kommet til at kalde en forkert funktion.

Til forsiden blev der tilføjet en startknap, for at brugere nemmere skulle kunne finde ud af hvordan de kom igang. Og der blev tilføjet kontaktinfo, til eventuelle spørgsmål.

Også login formen blev testet og virkede som den skulle ved at logge en bruger ind, når der blev skrevet de rigtige brugeroplysninger.

Brugeroprettelsen og login virker og har en kort brugstid hvilket er godt. Endvidere er brugeroprettelsen så tilstrækkelig lille, at der godt kunne udvides med flere brugeroplysningerne hvis det blev fundet nødvendigt. En mulig udvidelse kunne være at tilføje popup beskrivelser til de tre computerfærdigheder, således at brugere i tvivl kunne læse en mere nøjagtig beskrivelse af dem. Dette blev dog pga. tidsbegrænsning ikke udviklet i denne iteration.

### Login håndtering

Login håndteringen og dets brug af sessions blev kontrolleret ved først at skrive url'en til beskyttede sider og se hvilken side der blev loadet. Disse virkede alle som de skulle og beskyttede sider blev videresendt til *ingen adgang* siden.

Sessionen blev desuden testet til at forblive slettet når browseren blev lukket, som den skulle.

### Afprøv Spørgeskema

De tre spørgeskemaer blev løbet igennem en ekstra gang, og de opfangede testdata blev kontrolleret i databasen til at passe. De små hakker i spørgeskema-listen virkede som de skulle, men en lille mulig tilrettelse blev set ved de skiftende spørgeskemabeskrivelser.

Idet der ved de skiftende spørgeskemabeskrivelser bruges *onmouseout* lige når brugeren trækker musen væk fra et link, kan teksten virke lidt skiftende når man tager musen fra et link til et andet. Dette har jeg ændret sådan at tekstbeskrivelsen først bliver sat tilbage når musen kommer over en af de andre felter på siden. Således skifter beskrivelserne ikke frem og tilbage.

### Browsertest

Under implementeringen er det udviklede blevet løbende testet hvilket også gælder de tre understøttede browsere. Siden er implementeret i Internet Explorer 7, samt med et vågent øje på Firefox og Internet Explorer 6.

Ved at gå alle siderne igennem i de to andre browsere kunne det yderligere bekræftes at siderne vises som de skal i alle tre browsere.

## 5.3.5 Opsummering

Den første iteration blev udviklet uden store problemer og indeholder en virkende basal del af systemet.

Iterationen var dog mere omfattende end planlagt hvilket delvist skyldes at der var mange detaljer som med fordel kunne tilføjes, for at forbedre forståeligheden af siden. Denne slags detaljer tog dog tid at få tilføjet og tilpasset de tre browsere.

Siden opfylder XHTML standarden, og er testet i Internet Explorer 6 og 7 samt Mozilla Firefox, hvilke repræsenterer størstedelen af de benyttede browsere. Brugerfladen er designet til at være intuitiv og indbydende og systemet er opbygget så det er nemt at udvide.

Det kan mærkes at det tager ekstra tid at holde siden intuitiv og klar, da dette kræver at der bruges tid på mange små detaljer. Disse små detaljer bidrager dog til et helhedsbillede af en klar, gennemtænkt og overskuelig side, hvilket i sidste ende vil sørge for at brugerne nemmere forstår siden og at der bliver trukket flere brugere til den.

### Iterationsplan

Den næste iteration omfatter følgende udvidelse af systemet:

- UC3 - Vis Statistik

Denne use-case vil blive udviklet sammen med en udvidelse af spørgeskemaerne i UC2 - Afprøv Spørgeskema, for at tilføje flere tilbudte spørgeskemaer.

Med denne iteration udviklet, vil websiden være fuldt udviklet og opfylde kravene fra kravspecifikationen.

## 5.4 Iteration 2

I denne iteration vil statistikmodulet blive udviklet og *UC2 - Afprøv Spørgeskema* vil blive udvidet. Med dette vil websitet for sikkerhedsbevidsthed være fuldt udviklet til at teste og udtrække statistik om brugeres bevidsthed inden for sikkerhed. Herefter vil afsnittet 'Brugeranalyse' beskrive en analyse af de opfangede svar.

### 5.4.1 Use-case model

Denne iteration indeholder kun én use-case, nemlig *UC3 - Vis Statistik*.

#### 5.4.1.1 UC3 - Vis Statistik

Denne use-case omhandler visningen af statistik og viser svarfordelingen af de udførte tests.

<i>Mål:</i>	At få vist en statistik over testoplysningerne
<i>Type:</i>	Hovedfunktion
<i>Forudsætninger:</i>	Der er oplysninger at lave statistik over
<i>Succes tilstand:</i>	Et øjebliksbillede af statistikken bliver vist
<i>Fiasko tilstand:</i>	Statistikken bliver ikke vist
<i>Primær aktør:</i>	Administratoren
<i>Startbegivenhed:</i>	Administratoren åbner statistik siden
<i>Prioritet:</i>	Høj
<i>Afviklingstid:</i>	Kort
<i>Hyppighed:</i>	Sjælden

**Success scenarie:**

1. Administratoren åbner statistik siden
2. Statistikkerne bliver vist for administratoren

**Fiasko scenarie:**

1. Administratoren åbner statistik siden
2. Statistikkerne bliver ikke vist for administratoren

Statistikkerne viser balancen af brugernes reaktioner og bliver dannet ud fra de optagne oplysninger på det tidspunkt de er kaldt.

## 5.4.2 Design

Dette afsnit beskriver designet af udvidelserne til det udviklede i Iteration 1.

Udvidelsen af spørgeskemadelen vil være en kodeords test, samt tilføjelse af generelle tekstspørgsmål til phishing web testen.

### 5.4.2.1 Test kodeord spørgeskema

Dette spørgeskema er en udvidelse af *UC2 - Prøv Spørgeskema* og omhandler en enkelt side, hvori brugeren kan indtaste sit kodeord og få vurderet hvor sikkert det er.

I sikkerhedskapitlet i starten af denne rapport blev der beskrevet flere regler som et godt kodeord helst skal overholde for at være sikkert.

Disse regler er at kodeordet skal:

- Være på mindst 8 tegn

- Bruge både små og store bogstaver
- Bruge specieltegn og tal
- Ikke have noget med brugernavnet at gøre
- Ikke være et kendt ord eller navn
- Ikke være en genbrugt adgangskode

Disse regler skal på en eller anden vis, kobles til en vurderingsmekanisme, sådan at der ud fra reglerne kan vises en form for bedømmelse af hvor sikkert kodeordet er og hvad der eventuelt kan gøres for at forbedre det.

De to sidste regler kan der umiddelbart ikke gøres noget ved, da jeg ikke har nogen direkte mulighed for at sammenligne tekststrengene med en ordbog automatisk og jeg kan heller ikke kontrollere om det indtastede kodeord ligner en anden af brugernes koder. Dog kan resten af reglerne godt danne grundlag for en kodeordstest.

Vurderingsmekanismen jeg vil benytte til at bestemme hvor sikkert et kodeord er, er bestemt ud fra en skala fra 0 til 100, hvor 100 er et godt sikret kodeord. På denne skala vil 0-40 points repræsentere et dårligt kodeord, 41-70 et mellem, og 71-100 et godt kodeord. Disse inddelinger vil grafisk blive repræsenteret af farverne rød, gul og grøn, sådan at vurderingen af kodeordet viser sig intuitivt.

Ved at regne lidt på problemstillingen har jeg udviklet et pointsystem, som vil vurdere kodeordet på en enkel måde, hvor fokus er på at resultatet er forståeligt for brugere.

Denne vurdering vil foregå i følgende trin:

1. Længden af kodeordet er det mest kritiske og bliver bedømt efter følgende ligning:  $Point = (Length \cdot 15) - 50$ , hvor resultatet ikke kan blive større end 70 eller mindre end nul. De 70 points er for en længde på otte karakterer, og der gives ikke yderligere points for længere kodeord i denne test, da længere kodeord tit kan føre til at man ikke kan huske dem.
2. Hvis der er både små og store bogstaver i kodeordet tildes der 15 points og ellers nul.
3. Hvis der findes specieltegn eller tal i kodeordet tildes der også 15 point og ellers nul.

Denne vurderingsmekanisme vurderer længden af kodeordet som vigtigere end specialtegn og forskel på bogstavstørrelsen alene.

Når spørgeskemaet har udført vurderingen, skal der vises nogle tip til hvordan brugeren yderligere kan sikre sit kodeord. Desuden skal der blandt disse tip også stå de ting der ikke kunne beregnes frem til, som at kodeordet helst ikke må være genbrugt eller en del af det brugernavn den af parret med.

Denne metode er grundlæggende udviklet for at matche de grundlæggende krav for kodeord der blev beskrevet i det første kapitel.

En mere præcis mekanisme ville være at specificere en ligning som udregne hvor mange forsøg en algoritme ved brug af "brute force" skulle igennem, for at gætte det indtastede kodeord. Dette ved at beregne alle kombinationer for et bestemt karaktersæt. Dog var dette til denne opgave unødigt præcist, da jeg følte det var nok at benytte de førbeskrevne regler til mekanismen, for at give et vurderet svar uden at bruge for lang tid på udviklingen af spørgeskemaet.

#### 5.4.2.2 Phishing Web

For yderligere at hjælpe undersøgelsen om forskellen i forståelse mellem normale tekstbaserede og forbedrede animerede -spørgeskemaer. Vil der blive tilføjet et normalt spørgeskema til Phishing Web spørgeskemaet. Disse spørgsmål vil blive vist før det nuværende Phishing Web spørgeskema når man vælger det fra listen og vil omhandle generelle emner inden for phishing.

Idet phishing er et grafisk betinget emne, bestemt af aspekter som om der er en lås det rigtige sted på skærmen, kan phishing-spørgsmål som lavet i Captivate versionen, simpelthen ikke laves på tekstformat uden at simplificere problemstillingen. I en tekstversion af et phishing spørgeskema vil brugeren nødvendigvis få beskrevet de korrekte svar for at kunne vælge dem. På denne måde vil brugeren så få serveret en blanding af ukorrekte og korrekte muligheder, hvor brugeren i et animeret spørgeskema selv må se på billedet for at bestemme de mulige svar. Derved kan det allerede vurderes at en form for forbedret animeret spørgeskema for phishing, hvor brugeren bliver stillet frem for et simuleret praktisk scenarie, er en nødvendighed for at brugerne skal kunne få tilstrækkeligt ud af spørgeskemaet og for at brugeren vurderes korrekt.

Derfor vil disse tilføjede tekstspørgsmål omhandle mere generelle aspekter inden for phishing, og vil ydermere have til formål at være en slags 'teaser' for det egentlige phishing spørgeskema, hvori brugeren kan udlede svarene til spørgsmålene fra spørgeskemaets indledende tekst og spørgeskemaforklaringerne til slut. En phishing-quiz fra Microsoft internetside [8] er blevet brugt som inspiration til spørgsmålene.

Spørgsmålene vil være stillet som følger:

### Hvad er phishing på internettet?

1. Internetsider der forsøger at give dig virus, eller andre ubehageligheder.
2. Internetsider der giver sig ud for at være en andre sider for at lokke personlige oplysninger, så som kodeord eller kreditkort oplysninger, ud af dig.
3. Internetsider der giver sig ud for at være en andre sider for at overtage din computer og benytte den til at udføre kriminelle handlinger.
4. En metode hvorved firmaer benyttet internettet til at prøve at 'fiske' nye kunder til sig.
5. Ved ikke.

### Det er sikkert at indtaste personlige oplysninger eller bankoplysninger i et pop-up-vindue

1. Rigtig
2. Forkert
3. Ved ikke

Det første spørgsmål er generelt til om brugeren forstår hvad phishing går ud på og indeholder svarmuligheder der er ment til at forvirre brugeren, for at brugeren ikke alt for let skal kunne gætte sig til det rigtige svar.

Det næste spørgsmål refererer med vilje til noget grafisk og indeholder derfor også en "Ved ikke" -svarmulighed, da det ikke er sikkert at brugeren kan visualisere problemstillingen. Dette spørgsmål er et godt eksempel på en situation hvor et animeret spørgeskema ville være mere præcist, da brugeren så ikke ville skulle forestille sig situationen selv. Det vil derfor være interessant at se hvordan brugerne svarer på dette spørgsmål.

Disse spørgsmål er med vilje blevet valgt således at de ikke afslører for meget af de mere præcise krav for at identificere phishing. Dette er for at brugerne igennem disse tekstspørgsmål ikke bliver indlært for hurtigt og derved kan benytte det lærte til at klare sig bedre i det animerede spørgeskema.

#### 5.4.2.3 Statistik

Statistikken bliver dannet når den kaldes og er derfor lavet ud fra et øjebliksbillede af databasen. Derfor er det vigtigt at der i statistikken bliver vist tidspunkt og generelle informationer omkring databasen, så som antal af rækker i de forskellige tabeller. Disse generelle informationer beskriver således versionen af det viste øjebliksbillede.



Idet hvert spørgeskema er bygget fundamentalt forskelligt op hvad angår svar, kan der ikke umiddelbart bygges nogen autogenerering af denne statistik som vil være tilstrækkelig sigende. Derfor må der i statistikken forekomme sektioner med statistik specielt udviklet til hvert spørgeskema.

Brugerfladen af statistik-siden behøver ikke ligne den overordnede side, men skal være overskuelig og printbar.

Følgende generelle informationer beskriver øjebliksbilledet af databasen, for hvilket statistikken er dannet ud fra:

- Dato og klokkeslet
- Antal personer
- Antal rækker af testdata
- Antal spørgeskemaer

Herudover skal det være muligt at indsnævre brugergruppen for hvilken resten af statistikken bliver dannet ud fra. På den måde vil det være muligt at få statistikken til at blive vist for en ønsket brugergruppe. Dette kan laves ved at have form-felter, i hvilket man kan vælge hvilke brugergrupper man vil have med i statistikken. Det skal altid være muligt at se tallene på ændringerne når man ændrer brugergruppen, således at disse ændringer kan udføres så hurtigt og let som muligt.

Statistikken skal være opdelt i spørgeskemaer og vise hvor stor procentdel af den valgte brugergruppe der svarer rigtig og forkert på de individuelle spørgsmål samt samlet for spørgeskemaet.

### 5.4.3 Implementering

Denne sektion beskriver implementeringen af iteration 2. Alle ændrede eller oprettede filer er blevet tilføjet som et nyt bilag.

#### 5.4.3.1 Test Kodeord Spørgeskema

Kodeords testen er ligesom de andre lavet med en formside og en handlerside (password.php - bilag A.6.2 og passwordh.php - bilag A.6.3). Formsiden kan ses på figur 5.15.

Den første side er som vist på figuren en simpel form, hvorved brugeren kan indtaste et kodeord. Kodeordet bliver ved submit valideret af et javascript (linje 145 i script.js - bilag A.6.5) som regner pointene ud og sender resultatet til



**Sikkerhed** Ved du nok om computersikkerhed? [Test dig selv og lær mere](#)

Start Log ud

**Hvor sikkert er dit kodeord?**

Hvor sikkert et kodeord er, er bestemt af hvor nemt det er at gætte.

Test dig selv og skriv dit kodeord i feltet nedenfor og tryk på 'test' for at få en vurdering af hvor sikkert dit kodeord er, samt med foreslag til hvordan du kan forbedre.

Dit kodeord vil ikke blive gemt, og kun hvor sikkert det blev vurderes registreres til statistik.

**Test dit kodeord:**

Figur 5.15: Test kodeord - password.php

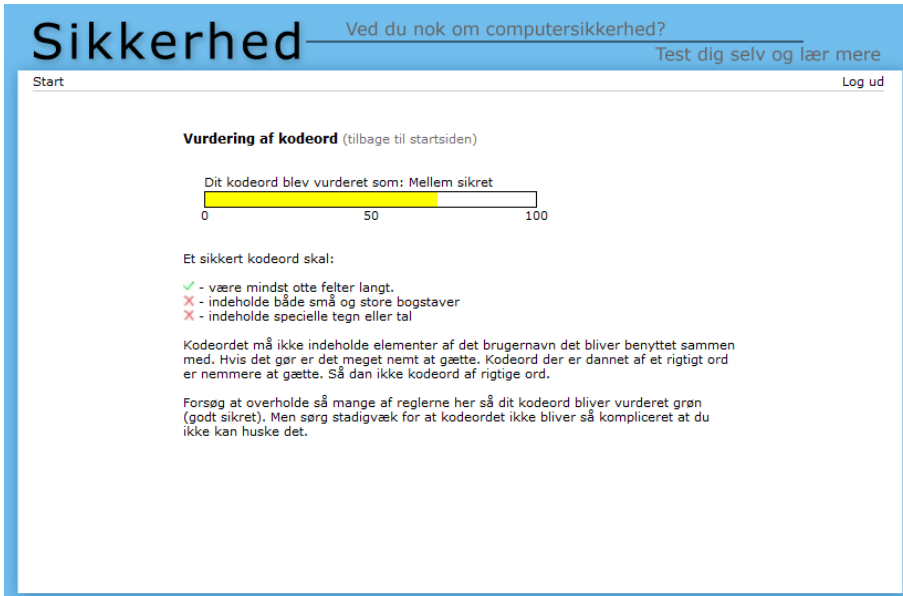
handleren.

Pointene bliver regnet ud som beskrevet i designet, og det udregnede bliver delt op i fire strenge som sendes videre og repræsenterer følgende: *point*, *længde*, *både store og små bogstaver*, *har tal eller specielle tegn*. Disse fire strenge bliver sat ind i skjulte formelementer, hvilket er elementer der ikke kan ses på formsiden men godt kan tilgås fra handlersiden.

Handleren er vist på figur 5.16 og oversætter vurderingen til tekst og grafik. En vurderingsbar viser hvor godt sikret kodeordet er ved at repræsentere vurderingen med farver. Baren er lavet som en tabel, der bliver stilet med en farvet blok begyndende fra venstre side, dynamisk bestemt i *php*. Dårligt sikrede kodeord bliver repræsenteret med en rød farve, mellem med gul og et godt med grøn.

Under baren kan brugeren så yderligere se hvilke af de tre regler for længde, store og små bogstaver samt tal og specialtegn kodeordet overholdt, og der står beskrevet generelle tips om kodeordssikkerhed.

Dette spørgeskema er igen lavet sådan at kun det første forsøg lagres i databasen og brugeren bliver efterfølgende adspurgt om testen var nyttig. Dette er igen for at undgå at lagre unødigt data, hvis en bruger skulle finde på at taste



Figur 5.16: Test kodeord handler - passwordh.php

en masse opfundne kodeord ind for at se hvordan mekanismen virker. Dog kan det ikke undgås at der også mistes et par rigtige kodeord på denne måde.

### 5.4.3.2 Phishing Web

De to tekstspørgsmål der udvider Phishing Web spørgeskemaet, er implementeret således at man kommer til dem når man vælger 'Phishing Web' fra oversigts-siden, og så derefter kommer til den egentlige Captivate animerede phishingtest når man trykker 'Videre'. Figur 5.17 viser hvordan tekstspørgsmålene ser ud i en browser.

For at få de valgte svar fra tekstspørgsmålene med i databasen, bliver de i stedet for at blive indsat med det samme, lagt over i gemte form-elementer på phishing-siden (internet.php bilag A.6.9), således at de sendes med derfra også, og derfra kan opdateres sammen med de andre informationer på handler-siden (inethandler.php bilag A.6.10). Kildeteksten til tekstspørgsmålene kan ses i *inetintro.php*, bilag A.6.11.

### 5.4.3.3 Statistik

Statistik-siden er implementeret i to php-sider *statistik.php* (bilag A.6.12) og *statdata.php* (bilag A.6.13).



Figur 5.17: Tekst spørgsmål til Phishing Web - inetintro.php

Siden ligger på <http://sikkerhed.tre-ben.dk/statistik/statistik.php> og er ikke refereret til fra hovedsiden.

Statistikken er opbygget ved at have de generelle personinfo øverst i form-elementer, således at disse løbende kan ændres for at indskrænke/udvide de rammer spørgeskemastatistikken bliver dannet ud fra. For at implementere dette så nemt for brugeren som muligt, er AJAX (Asynchronous Javascript And XML) [11] blevet benyttet.

AJAX er en Javascript metode, der gør det muligt asynkront at kalde en anden side inde fra en allerede hentet html-side, uden at skulle kalde en ny side. Ved brug af dynamisk html og javascript kan en side hentes ind, bestemt ud fra normale javascriptevents på den pågældende side. På denne måde kan man dynamisk opdatere eller ændre en side, mens den bliver vist for brugeren, uden at skulle skifte til en ny side.

Statistikken er implementeret således at *statistik.php* indeholder de generelle informationer, vist med form-elementer. Når disse ændres af brugeren, kaldes et event til en javascriptfunktion, som så ved brug af AJAX kalder og henter *statdata.php* ind og putter dennes indhold ind *statistik.php*. Dette gøres hver gang form-elementernes indhold bliver ændret, og sørger derved for at opdatere

statistikken efter de valgte afgrænsninger.

Visningen af en del af statistikken kan ses på figur 5.18 og en mere detaljeret forklaring af statistikens ajax-modul er beskrevet i bilag A.6.1.

#### Statistik for sikkerhed.tre-ben.dk

Kriterier		Done
Alder:	<input type="text" value="0"/> til <input type="text" value="99"/>	
Køn:	<input checked="" type="checkbox"/> Mand <input checked="" type="checkbox"/> Kvinde	
Computerfærdighed:	<input checked="" type="checkbox"/> Uerfaren <input checked="" type="checkbox"/> Erfaren <input checked="" type="checkbox"/> Ekspert	
Valgt statistik: 6 / 6 personer i 13 / 13 testrækker fra 5 spørgeskemaer		12/6-2007 22:40:14
Adgangskoder		3 / 3
Point (avg):	51.6	
Længde (avg):	8.33 tegn	
Længde > 7:	66.6 %	Testen hjalp 66.6 %
Phishing - Web		3 / 3
Hvad er phishing?	A: 33.3 %, B: 33.3 %, C: 33.3 %, D: 0 %, E: 0 %	
Pop-up vindue, sikkert?	A: 33.3 %, B: 33.3 %, C: 33.3 %	
1) Danskenetbank:	Ja: 0 %, Nej: 100 %, Ved ikke: 0 %, Korrekt: 66.6 %	
2) Tastselv skat:	Ja: 33.3 %, Nej: 33.3 %, Ved ikke: 33.3 %, Korrekt: 33.3 %	
3) Paypal:	Ja: 100 %, Nej: 0 %, Ved ikke: 0 %, Korrekt: 100 %	
4) Apple store:	Ja: 0 %, Nej: 100 %, Ved ikke: 0 %, Korrekt: 66.6 %	
5) Jyske bank:	Ja: 33.3 %, Nej: 33.3 %, Ved ikke: 33.3 %, Korrekt: 33.3 %	
Samlet (avg):	Korrekt: 3 / 5	Testen hjalp 100 %

Figur 5.18: Statistik

Statistikken for spørgeskemaerne har hver deres blok, og er implementeret hver for sig med funktioner tilføjet til *classes.php* (bilag A.6.4). Øverst i højre hjørne af hver blok kan man se hvor mange statistik-svar der er vist ud fra de valgte afgrænsninger, ved siden af antallet for det pågældende spørgeskema i alt. De viste statistikker indeholder en hjælpende popup-tekst som vises når man holder musen over overskriften, for at huske en på detaljerne for de forskellige spørgeskemaer.

Til phishing statistikkerne blev de pågældende handler sider fra spørgeskemaerne ændret, således at disse også indsætter hvor mange svar der blevet besvaret rigtigt for hver bruger, i databasen. Dette gøres for at skære ned på udregningerne til hver af statistikens opdateringer, således at alle spørgeskemaer ikke skal vurderes på ny, hver gang siden opdateret.

Denne realiseringsmetode, med benyttelse af AJAX, gør at siden virker meget intuitiv og let, da statistikken er holdt på en vist side, der blot bliver opdateret når afgrænsningerne bliver ændret. Et alternativ kunne have været at implementere dette med et iframe, hvilken er et internt frame, som indeholder en anden side (i dette tilfælde så *statdata.php*). Iframes er dog lidt mere kluntede at arbejde med og kan heller ikke dynamisk skaleres i størrelsen, hvilket gjorde at jeg valgte AJAX.

Brugen af AJAX gør det muligt at tilføje ekstra funktionalitet til websider, som får siderne til at være meget mindre statiske end ellers. Dette gør siderne mere intuitive, overskuelige og nemmere at bruge, dog med den bekostning at det tager ekstra udviklingstid at udvikle. Desuden bliver siderne kodemæssigt lidt mindre overskuelige, da hver side så ikke nødvendigvis repræsenterer en 'state'.

Til spørgeskemaer kan det derfor godt betale sig at sætte sig ind i mulighederne af AJAX<sup>2</sup>, som her beskrevet. Da de tilbyder muligheder for interaktivitetsforbedring på linje med popup info-bokse som tidligere beskrevet.

#### 5.4.4 Test

Systemet, der er blevet udviklet i denne iteration er løbende blevet testet. Derfor vil jeg i denne sektion beskrive en systemtest, ved at oprette en ny bruger og gå alle funktionerne fra iterationen igennem. Vægten vil på samme måde som i sidste iteration blive lagt på helheden og på hvordan de forskellige dele virker sammen.

Testen af delene vil virke ved først at teste javascript valideringen af formene og derefter kontrollere i databasen om informationerne blev gemt korrekt.

##### Test kodeord spørgeskema

Kodeordstesten virker som den skal og er meget nem at benytte. Det enkelte indtastningsfelt sender kodeordet til handlersiden, hvorpå den grafiske vurdering meget intuitivt med forklaringer viser hvor sikkert kodeordet er. I databasen bliver vurderingen af det første kodeord brugeren tester desuden lagt ind fint.

##### Phishing Web

Tekstspørgeskema udvidelsen af Phishing Web spørgeskemaet virker godt og javascript valideringen sørger for at de fornødne felter bliver afkrydset før resultaterne bliver sendt videre. I databasen bliver de udvidede spørgsmål lagt ind sammen med dem fra den første iteration.

---

<sup>2</sup>Andet eksempel på brug af AJAX: <http://www.google.com/webhp?complete=1&hl=en>

### Statistik

Statistikken opdateres hurtigt på trods af de mange php funktioner og database opdateringer der skal kaldes. Teksten der ændres til 'Updating' når der bliver hentet en opdatering til statistikken, gør det meget intuitivt klart hvornår opdateringen er i gang. Selv ved at prøve at fremprovokere cache-fejl, kunne disse ikke findes.

### Browsertest

Da der i denne iteration bruges layoutet fra den første iteration, blev de fleste fejl på tværs af browsere fundet under implementeringen og der blev under denne test kun fundet en enkelt browserfejl.

Ved testningen af phishing spørgeskemaerne i Internet Explorer 6 viste det sig pludselig af handleren nogle gange ikke viste svarene. Denne fejl viste sig dog kun nogle gange, selv med samme input i Captivate delen.

Efter at køre en meget intensiv test på denne blev det kontrolleret at Captivate delen virkede og javascriptet sendte de rigtige informationer over til handleren. Da jeg så undersøgte handleren i Internet Explorer 6, side om side med Internet Explorer 7, viste det sig at Captivate i IE6 ikke altid havde startinformationerne med, men nogle gange kun de sidste. Derfor ville PHP-delen der forkorter de omkringværende karakterer så gøre dette forkert og inputene ville ikke blive fundet. Dette blev løst ved at ændre handleren sådan at den undersøgt om det første af informationerne var med eller ej, således at den kunne håndtere strenge, og få fat i de rigtige informationer lige meget om starten var med eller ej. Denne fejl var særdeles svær at finde, især da det var en fejl i Captivate hvis indre funktioner ikke kendes.

### Opsummering

Da denne iteration udelukkende omhandler udvidelser til det allerede udviklede system, var der kun en enkelt større fejl at finde, der ikke var fundet under implementeringen og det udviklede virker i alle tre understøttede browsere efter hensigten.

#### 5.4.5 Opsummering

Hele siden er nu med testen af Iteration 2 færdigudviklet og er testet til at virke efter hensigten.

Der er brugt en del tid på at få siden til at se ud og virke så intuitivt som muligt, så brugerne nemt skal kunne svare på spørgeskemaerne.

Siden består nu af et brugersystem, der er meget nemt at benytte, samt fire spørgeskemaer inden for phishing, kodeord og trådløst internet. Det bliver i den næste sektion interessant at se om brugeranalysen understøtter disse forhåbninger.



## 5.5 Brugeranalyse

For at teste siden og få brugerrespons har jeg udsendt den både til en række venner samt udsendt den på musikhuset Aage Jensens brugerforum [6]. Den sidste gruppe udviste megen interesse for sikkerhedsbevidsthed og gav et overraskende stort respons. Derved er der på en lille måned blevet oprettet 230 brugere og testet 600 spørgeskemaer.

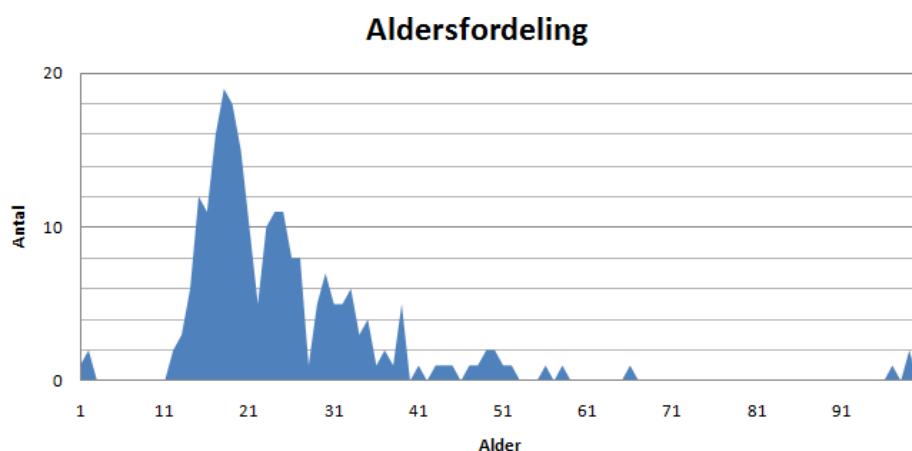
I denne sektion vil statistikkerne blive analyseret, for at få undersøgt hvorledes tendensen af brugernes viden inden for sikkerhedsbevidsthed er. Derudfra vil analysen også forsøge at give et svar på hvor godt de animerede spørgeskemaer virker.

### 5.5.1 Statistik

De 230 personer som har prøvet sikkerhedssiden har dannet en stor mængde responsdata, repræsenteret på statistiksiden. I denne sektion vil dette blive analyseret, startende med brugerfordelingen.

#### 5.5.1.1 Brugerfordeling

Idet størstedelen af brugerne også er brugere fra musikhusets forum, vil brugerne mest sandsynligt være yngre og mere teknisk kyndige end normale brugere, da musikere ofte er teknikinteresserede. Figur 5.19 viser et diagram over aldersfordelingen for brugerne i systemet.



Figur 5.19: Aldersfordeling

Som det kan ses er størstedelen af brugerne mellem 10 og 30 år gamle, hvilket

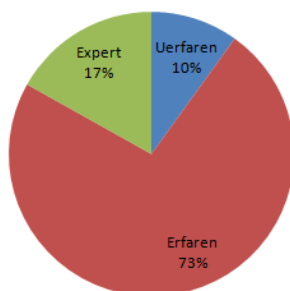
betyder at de sandsynligvis er vokset op med computere og derfor er rimelig vant til at benytte dem.



Figur 5.20: Kønsfordeling

Kønsfordelingen fra figur 5.20, viser at næsten alle af brugerne er mænd, hvilket også passer sammen med at de fleste brugere på musikhusets forum er mænd. Dette behøver ikke have nogen stor indvirkning på besvarelserne, men mænd er normalt mere teknisk interesseret i computere end kvinder og har derved ofte mere erfaring med dem.

### Fordeling over computererfaring



Figur 5.21: Fordeling over computererfaring

Af fordelingen over computererfaring som er vist på figur 5.21, er det vist at størstedelen af brugerne har specificeret dem selv som erfaren. Dette passer sam-

men med de tidligere beskrivelser, om at de oprettede brugere består af lidt mere teknisk orienterede brugere.

Brugerfordelingen er ret snæver og består for det meste er yngre mænd med hang til teknik. Dog må man så forvente at disse formodentlig vil vide mere om computersikkerhed end den gennemsnitlige bruger. Da disse brugere sandsynligvis er vokset op med computere og fra deres brug på musikhusets forum nok bruger internettet dagligt, er de velkendt med internettet og kender derfor nok til flere af de nævnte sikkerhedsaspekter.

Den snævre brugergruppe gør at det ikke giver megen mening at benytte værktøjerne til brugerafgrænsning. Derfor vil analysen af spørgeskemastatistikkerne alle være for hele den opfangede brugergruppe. At der er oprettet så mange personer gør dog at der vil dannes et rimelig godt billede for brugergruppen, da der med så mange personer er meget mindre risiko for tilfældigheder som er ødelæggende for statistikken.

De næste dele vil beskrive statistikkerne for spørgeskemaerne i detaljer. Til dette vil de forskellige dele af statistikkerne blive delt op, vist og forklaret for sig.

### 5.5.1.2 Spørgsmål 1 - Phishing Web

Det første spørgeskema omhandlede phishede internetsider, hvorved brugeren skulle svare på om han ville stole på en vist side eller ej. Som det kan ses øverst til højre på figur 5.22 fik dette spørgeskema 176 besvarelser.

Phishing - Web					176 / 176
Hvad er phishing?	A: 1.7 %	B: 78. %	C: 2.8 %	D: 5.1 %	E: 11. %
Pop-up vindue, sikkert?	A: 7.3 %	B: 87. %	C: 5.1 %		

Figur 5.22: Phishing Web - Tekstbaseret del

Inden brugeren kom til phishing spørgeskemaet, blev han bedt om at besvare to tekstbaserede spørgsmål. Det første værende forskellige forklaringer til hvad phishing er og det sidste en tekstforklaring om hvorvidt brugeren ville stole på at indtaste følsomme oplysninger i et popupvindue. De korrekte svar til begge spørgsmål er B, hvilket i det sidste betyder at man svarer 'nej'.

Som det kan ses svarer fire ud af fem rigtigt på hvad phishing er og en anelse flere har korrekt svaret 'nej' på det sidste. Idet brugeren i det første kan se alle muligheder, da dette er et multiple-choice spørgsmål, kan han derved muligvis gætte sig frem til det mest sandsynlige svar ved at læse dem alle igennem. Derfor vil det nok ikke helt være fire ud af fem personer der virkelig ved hvad phishing

er.

Det samme er også til dels gældende i det andet spørgsmål, da det nok kan tænkes at det oftere er korrekt at svare forsigtigt i et spørgeskema designet med sikkerhed som hovedtema.

Dette er ulempen ved multiple-choice spørgsmål. Ved at man kan se alle muligheder kan det i nogle tilfælde være muligt at gætte sig frem til den korrekte ved simpel logik. En hypotese vil være at det ikke vil være muligt at gøre dette i de animerede spørgeskemaer, da man dertil selv skal udlede mulighederne fra det grafiske og alle mulige svar derved ikke står beskrevet direkte. Dette vil muligvis kunne udledes fra de efterfølgende animerede spørgeskemaer.

1) DanskeNetbank: Ja: 39. %,	Nej: 58. %,	Ved ikke: 1.7 %,	Korrekt: 38. %
------------------------------	-------------	------------------	----------------

Figur 5.23: Phishing Web - Spørgsmål 1

Figur 5.23 viser svar-fordelingen fra det første animerede spørgsmål, hvilket var et billede af Danske Banks Netbank som der ikke kan stoles på, da adressen er udformet som en IP-adresse.

Det skal bemærkes at der fra phishing statistikkerne er vist både et procenttal for alle der svarede 'nej', samt for alle der korrekt svarede 'nej' og fik korrekt beskrevet hvad problemet ved siden var (vist ved 'korrekt').

Det kan ses at lidt over halvdelen har svaret 'nej' til dette spørgsmål, men at kun 38% har kunne beskrive hvad det var der var forkert ved siden. Dette viser at kun lidt over en tredjedel af brugerene har kunne identificere problemet med ip-adressen.

2) Tastselv skat: Ja: 82. %,	Nej: 15. %,	Ved ikke: 1.7 %,	Korrekt: 5.6 %
------------------------------	-------------	------------------	----------------

Figur 5.24: Phishing Web - Spørgsmål 2

Figur 5.24 viser fordelingen fra det andet spørgsmål, som var en side fra Tastselv Skat, hvor der manglede en lås og det korrekte svar derved var 'nej'.

Det kan her ses at hele 82% forkert har svaret 'ja', hvilket er langt størstedelen af brugerne. Kun 15% har svaret nej og endnu færre har korrekt afkrydset at låsen manglede.

Forklaringen til dette må ligge i at brugerne stoler på Skat. De stoler så blindt at de glemmer vigtige aspekter så snart de ser grafikken og instinktivt svarer 'ja'.

Dette spørgsmål viser at brugere vurderer en Internet side som en helhed, og sjældent skelner mellem browserens egne informationer og den side browseren viser, hvilket gør dem meget sårbare over for phishede sider.

Dette spørgsmål viser desuden styrken af et animeret spørgeskema versus et simpelt. Faktummet at siden er vist grafisk fremfor beskrevet via tekst, gør at brugerne genkender logoet og grafikken fra Skat, og derved svarer derudfra som de ville hvis de selv som normalt blev præsenteret for siden. Dette viser også at hypotesen fra før er sand og at de animerede spørgeskemaer undgår usikkerheden i multiple-choice spørgsmål, da det korrekte svar så ikke kan gættes så tydeligt som ved et tekstbaseret spørgeskema.

Det kan ses at det for at få den rigtige reaktion fra brugeren virkelig er meget vigtigt at vise problemstillinger i spørgeskemaer, som brugeren vil få dem præsenteret i en praktisk situation, for at få samme reaktion fra brugeren. Måden brugeren reagerer på hvis præsenteret for noget visuelt, i forhold til den samme situation beskrevet i tekst er ikke ens, og man risikerer at få en statistik der er misvisende samt i dette tilfælde måske for let.

3) Paypal: Ja: 83. %, Nej: 15. %, Ved ikke: 1.1 %, Korrekt: 83. %

Figur 5.25: Phishing Web - Spørgsmål 3

Figur 5.25 viser fordelingen for det tredje spørgsmål, hvilket var en paypal-side som ikke var phished.

Denne side svarede 83% korrekt på.

4) Apple store: Ja: 73. %, Nej: 23. %, Ved ikke: 2.2 %, Korrekt: 14. %

Figur 5.26: Phishing Web - Spørgsmål 4

Det fjerde spørgsmål er vist på figur 5.26 og er fordelingen for en apple-store side hvor der hverken er https eller lås og siden kan derfor ikke stoles på. På denne side var der desuden grafisk tilføjet en lås på siden, for at snyde brugerne til at tro at siden var sikker.

Her kommer de samme tendenser som fra skat frem. 73% har svaret at de stoler på siden og har ikke bidt mærke i låsen eller protokollen. Det er ikke helt lige så mange som før, og man kan også se at 23% har svaret nej og kun 14% har svaret korrekt.

Idet Apples computere og især deres IPODs er så populære for tiden, kender

mange fra brugergruppen meget sandsynligt nok Apples hjemmeside, og føler derved at de kan stole på den. De har på grund af dette ikke bidt mærke til at der her sikkerhedsmæssigt mangler endnu mere end på Skat siden, hvilket giver endnu et stort fejludslag i statistikken.

5) Jyske bank: Ja: 32. %, Nej: 65. %, Ved ikke: 1.7 %, Korrekt: 48. %

Figur 5.27: Phishing Web - Spørgsmål 5

Det femte og sidste spørgsmål er vist på figur 5.27 og viser fordelingen for en Jyskenetbank side, hvor 'k'et i adressen er udskiftet med et 'c'.

På denne har halvdelen korrekt svaret 'nej' og identificeret at adressen var korrekt, hvilket viser at i hvert fald halvdelen af brugerne ér bevidste overfor betydningen af adressen.

Samlet (avg): Korrekt: 1.9 / 5 Testen hjalp 57 %

Figur 5.28: Phishing Web - Samlet resultat

Af det samlede resultat kan det ses at de 176 der prøvede dette spørgeskema i gennemsnit fik 1.9 ud af fem rigtige hvilket ikke er meget. Det kan også ses at lidt over halvdelen har markeret at de lærte noget af spørgeskemaet.

Cirka halvdelen af brugerne forstår at adressen har betydning for sikkerheden og kun 5 til 10% forstår virkelig betydningen af lås og https.

I det første tekstbaserede spørgsmål svarede fire ud af fem rigtigt, og virkede til at forstå det grundlæggende i phishing. Dog viser det sig at det alligevel ikke rigtig fanger, da langt de fleste ikke forstår betydningen af protokol og lås og næsten vigtigere ikke forstår at skelne mellem browserens informationer og internetsiden som browseren viser.

Det grafiske i de animerede spørgsmål virker til at have en stor effekt på hvordan brugerne vurderer og svarer på spørgsmålene. Igennem dem kan vi endda få udledt lidt om hvordan brugerne tænker og hvor meget de forstår. Dette er en stærk egenskab ved de animerede spørgeskemaer.

### 5.5.1.3 Spørgsmål 2 - Phishing Email

Dette spørgeskema omhandlede phishing i emails og fik 150 besvarelser som det også kan ses på figur 5.29.

Phishing - Email	150 / 150			
1) Mytravel:	Ja: 68. %,	Nej: 29. %,	Ved ikke: 1.3 %,	Korrekt: 24 %

Figur 5.29: Phishing Mail - Spørgsmål 1

Svarfordelingen for det første spørgsmål er vist på figur 5.29. Dette spørgsmål består af en phished mail der indeholder et forkert link samt beder om upassende oplysninger.

Kun en fjerdedel har korrekt svaret nej og valgt de rigtige begrundelser. De fleste har svaret at de ville stole på mailen. Enten har de fejlsvarende ikke set at linket ikke gik til Mytravels side, ellers har de ikke lagt mærke til at man ligesom i internetbrowsere, kan holde musen over linket for at se den url den refererer til.

2) Paypal:	Ja: 2 %,	Nej: 97. %,	Ved ikke: 0 %,	Korrekt: 83. %
------------	----------	-------------	----------------	----------------

Figur 5.30: Phishing Mail - Spørgsmål 2

Det andet spørgsmåls fordeling er vist på figur 5.30. Dette spørgsmål var med vilje lavet til at være rimelig åbenlyst phished med en tydelig forkert afsender email og et åbenlyst forkert link.

De fleste af brugerne har også gennemskuet dette, da næsten alle har svaret 'nej' og 83% har fået den helt rigtig. Især at afsendermailen var forkert gør mailen meget nem at gennemskue, hvilket nok tæller en god del i det store antal af korrekte svar.

3) Tastselv skat:	Ja: 59. %,	Nej: 38. %,	Ved ikke: 1.3 %,	Korrekt: 19. %
-------------------	------------	-------------	------------------	----------------

Figur 5.31: Phishing Mail - Spørgsmål 3

Fordelingen på figur 5.31 stammer fra det tredje spørgsmål. Denne mail indeholder et forkert link der kan være svært at se da det er et punktum der er udskiftet med en understregning ('.').

38% af brugerene har svaret 'nej' og kun 19% har svaret korrekt på denne. De fleste brugere er mest sandsynligt ikke vant til at kigge på formen af en url og koncentrerer sig mere om de tekstbaserede dele af den. Derfor fanger de her så ikke at urlen er forkert.

4) AOK:	Ja: 88. %,	Nej: 9.3 %,	Ved ikke: 1.3 %,	Korrekt: 88. %
---------	------------	-------------	------------------	----------------

Figur 5.32: Phishing Mail - Spørgsmål 4

Figur 5.32 viser fordelingen for det fjerde spørgsmål. Dette spørgsmål består af en ”Alt om København” mail, der ikke er phished.

Langt de fleste har herved også svaret ’ja’ til at stole på denne mail, hvilket er korrekt.

5) Falck:	Ja: 52 %,	Nej: 46 %,	Ved ikke: 1.3 %,	Korrekt: 34 %
-----------	-----------	------------	------------------	---------------

Figur 5.33: Phishing Mail - Spørgsmål 5

Det sidste og femte spørgsmåls fordeling er vist på figur 5.33. Denne består af en Falck-mail hvilken har flere links, hvori kun det sidste afslører at det er en phished mail.

Den ene halvdel af brugerne har svaret ’ja’ og den anden halvdel ’nej’. Dog har kun 34% svaret ’nej’ og korrekt begrundet hvorfor.

Samlet (avg):	Korrekt: 2.2 / 5	Testen hjalp 67. %
---------------	------------------	--------------------

Figur 5.34: Phishing Mail - Samlet

De 150 brugere der har prøvet dette spørgeskema har i gennemsnit svaret 2.2 ud af fem rigtige. Paypal og AOK mailsne har de fleste fået rigtigt, men de sidste tre har de fleste ikke fanget.

Fejlområdet er delt op i en gruppe af brugere der enten ikke har været nysgerrige nok til at holde musen over linket i mailen, eller ikke var i stand til at tyde om et link var phished eller ej.

Som brugerne også får beskrevet som respons til deres besvarelser, er det kritiske i om en email er phished, linket. I dette spørgeskema havde jeg med vilje ikke forklaret at man kunne holde musen over linket for at se dets rigtige indhold, da jeg ville udtrække hvor mange af brugerne der var nysgerrige nok til at gøre dette. Dette kan nu siges at være aller højest halvdelen, hvilket er det antal der svarede nej til det sidste spørgsmål.

Det kan diskuteres om dette spørgsmåls ’simulering’ af en emailklient er tydelig nok. Men brugerne burde stadigvæk være nysgerrige nok til at undersøge links i emailen, da disse er bestemmende for hvorvidt emailen er phished eller ej. Det



kan til sidst ses at denne test hjalp to tredjedele af brugerne, hvilket betyder at de fleste af brugerne blev gjort opmærksomme på noget de ikke vidste i forvejen.

Aspekter som hvorvidt en bruger undersøger links i en email, er endnu et punkt hvor animerede spørgeskemaer virker godt. Ved at præsentere problemstillingen så vidt muligt som brugerne er vant til, kan vaner udledes, hvilket giver meget mere præcise svar end ved simple spørgeskemaer.

Det er dog svært at nå et design, der passer til alle brugeres email-miljøer, da der findes mange forskellige emailklienter og hver brugerne har hver deres fortolkning af hvordan det egentlig virker.

#### 5.5.1.4 Spørgsmål 3 - Test Adgangskode

Det tredje spørgsmål omhandlede en adgangskodetest hvor brugeren kunne teste et kodeord for at få indblik i hvor nemt det var for en ondsindet person at gætte.

Adgangskoder	153 / 153
Point (avg): 70.9	
Længde (avg): 8.80 tegn	
Længde > 7: 64.7 %	Testen hjalp 18.9 %

Figur 5.35: Adgangskoder

Af de 153 brugere der har prøvet denne test, er det kun omkring 19% der mener at denne test har hjulpet dem, hvilket tyder på at de fleste brugere er opmærksomme på vigtigheden af et godt kodeord.

Det kan dog ses at kun tre fjerdedele af brugerne har et kodeord der er mindst 8 karakterer langt.

#### 5.5.1.5 Spørgsmål 4 - Trådløst

Den sidste test, var et animeret spørgeskema hvori brugeren skulle vælge et sikkert trådløst netværk ud af fire netværk. Denne test blev oprindeligt udviklet som 'proof of concept' for at se om Captivate kunne bruges til at udvikle animerede spørgeskemaer.

Trådløst internet	137 / 137
Med sikkerhedsaktivering: 83. %	
Uden sikkerhedsaktivering: 16. %	Testen hjalp 35. %

Figur 5.36: Trådløst internet

Som det kan ses på figur 5.36 forstod langt størstedelen at vælge et netværk med sikkerhedsaktivering, hvilket også betyder at de forstod spørgeskemaet. Derved har en tredjedel markeret at de blev hjulpet af spørgeskemaet, hvilket tyder på at de fleste ikke helt kendte til alle konsekvenserne af et usikkert netværk, selvom de forstod at det er godt at have et netværk med sikkerhedsaktivering.

## 5.5.2 Feedback

Fra musikhusets forum fik jeg en del respons i form af kommentarer om siden. Langt de fleste af disse kommentarer var positive og nogle også kritiske. I denne sektion vil en del af kommentarerne blive gennemgået for at få indsigt i brugernes reaktion på siden.

*Er det sidste "test dit kodeord" ikke en fælde?  
Når nu man lige er blevet belært om alt det her phishing-halløj...*

Til dette fik brugeren svaret at det nu ikke var en fælde, men at det var korrekt set og han ikke skulle benytte hans rigtige kodeord, men et andet der har samme form istedet.

*En "tilbage til spørgsmålet"-funktion ville være toppen.  
Nogle gang sidder man og kigger på et eller andet, trykker nej,  
og så tænker man: "Hov - var der også noget galt med XXX"*

Dette var egentlig designet sådan med vilje, for at det skulle være så enkelt som muligt for brugeren og for at undgå komplikationer med Captivate. Men det ville nok være mere brugbart med en tilbage-knap fra når man skal uddybe et 'nej'-svar.

*Alle rigtige på nær 1, jeg bruger ikke IE7, så jeg vidste ikke at  
sikkerhedssymbolet (hængelåsen) var flyttet der op...  
Men god side, der helt sikkert kan lære folk, at være meget op-  
mærksomme på hvad det er de laver på nettet*

Emnet med hængelåsen er blevet diskuteret tidligere i denne rapport, men konklusionen er at der nok er et par personer der får en ekstra fejl pga. det, selv om det i grunden Microsofts valg, når nu IE7 er den nye standard for Microsoft platformen. Brugere bør dog være bevidste nok til at lede efter låsen, når de bliver i tvivl om en side er phished. Da brugeren jo fortæller at han godt ved at låsen skal være der, burde han havde været nysgerrig nok til at lede efter låsen et andet sted. Microsoft har netop flyttet låsen hen ved siden af adressebaren, for at denne skulle være nemmere at lægge mærke til.

*Din test er fin, men den påstår jeg har svaret forkert på 2 spørgsmål angående hjemmesider, hvor jeg åbenlyst har svaret korrekt, så jeg ville nok gennemgå resultaterne grundigt inden jeg lagde hovedet på blokken rent statistisk*

Denne brugers besvarelser fik jeg slået op i databasen, og det viste sig at han ikke havde specificeret sine svar tilstrækkeligt. F.eks. havde han svaret "Siden virker forkert" til et sted hvor adressen var forkert. Hvis han godt havde set at det var adressen der var forkert, skulle han have markeret det, da jeg ellers ikke kan tyde om han virkelig at set det.

Denne kommentar er et eksempel på hvor konkurrencemindedede brugere kan være. Mange brugere ser et spørgeskema som en quiz og vil således gøre sig umage på at få så mange rigtige som muligt, selvom dette ikke nødvendigvis repræsenterer deres egentlige kunnen.

*Test 2: Her kræver du, at man rent faktisk går ind på de links. Det sad jeg ikke og tjekkede ud, eftersom det er en test og man ikke kunne trykke på de "falske links", da det er screenshots. Jeg ville bestemt have opdaget, at der var noget galt, når jeg først havnede på de adresser.*

Denne bruger er åbenbart af typen som først trykker på linket i mailen, og så bagefter bestemmer om siden er phished. Igen er det med linksne i phishing-mail spørgeskemaet, et valg for netop at se hvor mange brugere der undersøget dette og denne bruger hører tydeligvis til gruppen der først undersøger det bagefter. Derfor er det også korrekt at han får ekstra fejl i testen, hvor han så muligvis ville have færre i phishing-web testen.

Dette er en konsekvens af designvalget for emailtesten, og er grundet at jeg ikke var interesseret i at afsløre det med linkene, da jeg gerne ville kunne undersøge hvor mange der ville finde ud af dette, og derved hvor mange der var opmærksomme på at linket er det kritiske i en phished mail.

*Jeg synes det er en god test, jeg faldt i en enkelt gang, og var overforsigtig en anden gang, men jeg synes testen var god og at fremføringen og formen var god.*

*Rigtig god test  
Jeg fik ca. halvdelen korrekt og lærte en hel del.*

Som det kan læses er mange brugere vanedyr, som er meget vant til deres egne browsere og emailklienter. Samtidigt er mange brugere og især de mere teknisk vidende brugere, konkurrence mennesker som ikke bryder sig om at

tabe. Dette giver nogle kritiske kommentarer, som er grundet designmæssige valg i spørgeskemaet så som browseren i Phishing-web og visningen af email i Phishing-email. Disse valg er dog nødvendige da det er for stort et arbejde at lave en version for hver brugers computeropsætning. Dette gør desuden statistikken lidt strengere, da det så forventes at brugerne kan se ud over sine egne vaner, og se det der ligger bag spørgsmålene.

Det kan ses at det er en fordel at holde spørgeskemaerne så ens som muligt, da brugerne så vil vænne sig til måden spørgsmålene bliver præsenteret på og derved vil kunne svare mere ud fra deres normale vaner og derved svarer mere præcist.

Generelt er kommentarerne positive. De fleste skriver at siden var meget forståelig og at de var overraskede over hvor meget de alligevel ikke vidste om computer-sikkerhed.

## 5.6 Opsummering

Websitet for sikkerhedsbevidsthed opfylder de i kravspecifikationen opsatte krav ved at det gør det muligt for brugerne at teste deres viden inden for sikkerhedsbevidsthed samtidig med at deres reaktioner optages til at blive vist statistisk. De 230 brugere der oprettede sig, prøvede i alt omkring 600 spørgeskemaer, hvilket viser at man ved at lokke med brugerindlæring inden for sikkerhedsbevidsthed, som var motivet for siden, virkelig trække brugere til.

Udviklingen i iterationer gik godt og delte derved udviklingen op i to logiske dele som blev udviklet fuldstændig for sig og gjorde udviklingen af siden betydeligt mere overskuelig. Vægten blev lagt på at lave siden meget funktionel og intuitivt nem at overskue, hvilket ud fra resultatet af brugeranalysen er vist til at lykkedes.

De mange svar gav et godt billede på hvor godt de 230 brugere forstår sig på de viste emner inden for sikkerhedsbevidsthed. Hvor kodeordtesten og den trådløse test ikke gav så bemærkelsesværdige respons, var de to phishing tests meget populære, især Internet versionen.

I Internetversionen blev resultatet at brugerne i gennemsnit fik 1.9 ud af 5 rigtige, hvilket ikke er meget. Især i Skat-spørgsmålet blev det gjort tydeligt at brugerne ofte ikke skelner mellem det grafiske vist på siden af browseren og browserens egne informationer, hvilket derved viser at de ikke helt forstår laget bag det yderste grafisk abstraherede lag. Denne test viste sig gennem statistik og feedback at være fuldt forståeligt for brugerne og at den var et animeret spørgeskema og ikke et simpelt spørgeskema gjorde at det kunne udtrækkes me-

re præcise brugeranalyser.

Emailversionen fik gennemsnittet 2.2 ud af 5 rigtige, hvilket stadigvæk ikke er meget. I denne var det største problem for brugerne linket. De fleste brugere forstår ikke at det vigtigste punkt for at beslutte om en email er phished eller ej er linket, hvilket derved gav mange fejlsvar.



# Spørgeskemastyring

---

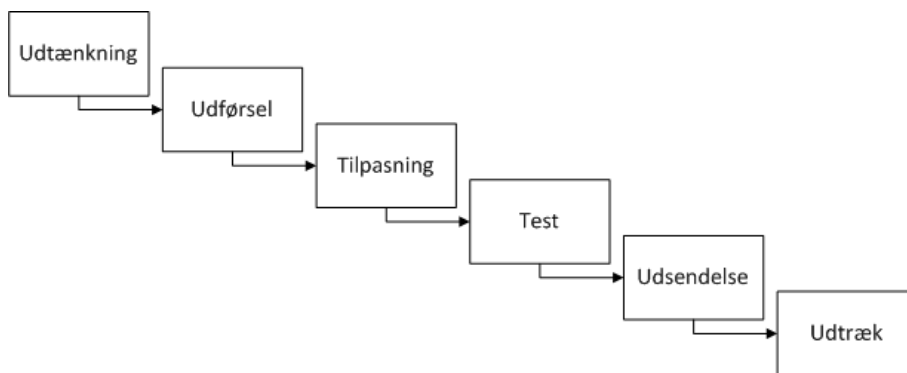
En problemstilling ved kampagner inden for spørgeskemaudsendelse og håndtering er styring af spørgeskemaerne. Idet der benyttes mange spørgeskemaer, kan der hurtigt mistes meget tid på opsætning og håndtering af spørgeskemaerne.

Et modul til håndtering og styring af spørgeskemaer på et administrativt niveau vil kunne minimere udviklingstiden fra et spørgeskema bliver udviklet, til det bliver oprettet på siden og derefter muligvis redigeret. Ved at udvikle en standardmetode for udvikling af spørgeskemaerne kan denne tid væsentlig optimeres og effektivt mindske udviklingstiden samt gøre udvikling af spørgeskemaerne nemmere at overskue.

Dette kapitel vil beskrive teorien bag opbygningen af et modul til spørgeskemastyring beregnet til at optimere udviklingen af spørgeskemaer på internettet. Denne vil dog ikke beskrive implementeringen da det falder ud over denne opgaves omfang.

## 6.1 Kampagne

Tiden fra et spørgeskema bliver udtænkt til statistikkerne udtrækkes kan kaldes en spørgeskemakampagne. Nogle dele af en sådan kampagne kan danne grund for en standardskabelon eller løsning, med det formål at optimere udviklingen af kampagnen. Denne sektion vil beskrive en typisk spørgeskemakampagnes levetid, for at få identificeret hvilke dele der med fordel kan optimeres.



Figur 6.1: Spørgeskema udvikling

Ved at tilpasse de normale udviklingsfaser specifikt til udviklingen af spørgeskemaer kan et diagram bygges, dette er vist på figur 6.1 og består af følgende dele:

### Udtænkning

Den første del indeholder analysen og designet af spørgeskemaet.

Denne del er ikke en oplagt del til optimering, da fokus her mest er på analyse af spørgeskemaets virkning og effekt. Man kan dog benytte en slags skabelon for brugerfladen, sådan at disse grafisk ligner hinanden fra spørgeskema til spørgeskema.

### Udførelse

Den første del af implementeringen hvor der bygges de kerneelementer spørgeskemaet består af, så som: spørgsmål og intern grafik.

Denne del er af samme grunde som den tidligere ikke oplagt til direkte optimering ud over eventuelt brug af en slags skabelon.

### Tilpasning

Den sidste del af implementeringen, hvor alle de interne dele tilpasses det over-



ordnede system og alle databaseforbindelserne skal implementeres.

Denne del har stort optimeringspotentialer, da den gentages og er næsten ens for hvert spørgeskema der bygges. De specifikationer der på baggrund af det overordnede spørgeskemasystemets struktur, skal bestemmes for at tilpasse spørgeskemaet, kan laves som en standardløsning som derved sparer udvikleren for en del tid. Dette vil blive yderligere beskrevet i næste sektion.

### Test

Denne kan ses som en slags sidste del af tilpasningen, hvor spørgeskemaet er oprettet i systemet og bliver testet igennem inden udsendelse.

Ved at tilføje en test-del til spørgeskemastyringen, kan testen muligvis optimeres ved at tilbyde at der nemt kan udelukkes test-data fra databasen uden at skulle fjerne det manuelt før spørgeskemaet sendes ud.

### Udsendelse

Spørgeskemaet er testet og brugere begynder nu at benytte det. I denne fase kan det ske at en form for redigering vil blive nødvendigt, hvis der opdages yderligere problemer.

En mulighed for nemt at redigere hovedemner i et spørgeskema kan muligvis være en fordel hvis det viser sig at der er efterfølgende ændringer, som først giver sig til syne efter at spørgeskemaet er blevet afprøvet af et antal af personer, som f.eks. ved feedback fra brugere. Normalt vil det dog være tilstrækkeligt at dette kan ændres manuelt af udvikleren.

### Udtræk

Spørgeskemaet har kørt den angivne tidsperiode og der bliver nu udtrukket data til statistik.

Statistikudtræk kan være meget tidskrævende og kan optimeres allerede ved analysen, da udtrækkene afhænger direkte af hvordan spørgeskemaet er udtænkt. Derudover kan et forslag være at have en mulighed for at oprette en standardstatistik der så kan udbygges til at passe til spørgeskemaet.

### Opsummering

De steder hvor spørgeskemaet skal passes sammen med det udenomlæggende system er oplagt til optimering. Ved at udarbejde en standardløsning, som skabeloner eller en slags wizard, kan en funktion til at hjælpe udviklingen af spørgeskemaer udvikles. Spørgeskemaer der udvikles gennem en sådan funktion, vil automatisk kunne få tilpasset mange dele til det omlæggende system, hvilket letter udviklingen. En sådan løsning er beskrevet i næste sektion.

## 6.2 Spørgskemakampagneoprettelse

For at udnytte de mulige optimeringer kan der udvikles en slags 'wizard' eller funktion, til brug ved udvikling af spørgeskemakampaner. Herved tages udvikleren igennem flere oprettelses trin der automatiserer delene hvor muligt og ellers hjælper udvikleren til at strukturere kampagnens udvikling.

Dette vil danne en strukturerende skal omkring spørgeskemaet, hvorved kampagnen vil have flere tilstande udover den der repræsenterer det færdige oprettede spørgeskema. På denne måde vil det også være administrativt gavnligt, da det udover de implementeringsmæssige fordele giver et overblik over hvor langt forskellige kampaner er i udviklingen.

Overordnet ville oprettelses-funktionen indeholde følgende funktioner:

**Tilpasning** Når oprettelse af en ny kampagne vælges, uploades og tilpasses filerne således at de passer sammen med det omkringliggende system. En skabelon bliver dannet for handleren, som danner grundlag for implementeringen af handleren til spørgeskemaet, mens det kan testes uden at normale brugere kan se det.

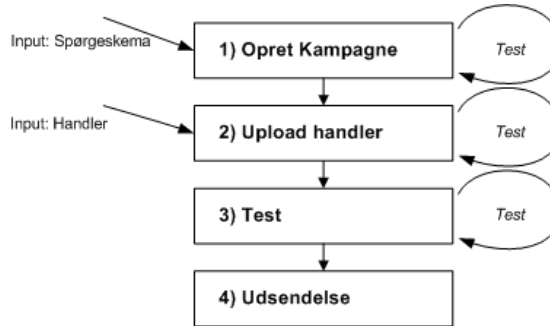
**Test** Spørgeskemaet kan testes mens det ikke er aktivt på hovedsiden, således at en grundig test kan udføres, før spørgeskemaet udsendes.

**Udsendelse** Når en fuldstændig test har vist at spørgeskemaet virker, kan spørgeskemaet nemt gøres tilgængeligt for de normale brugere.

De tre ovenstående tilstande, er overremnerne for status af spørgskemakampagnen og de enkelte trin som funktionen består af, er vist i figur 6.2 samt er beskrevet i de efterfølgende sektioner.

Overordnet vil hvert trin indeholde muligheder for at udvikleren kan teste spørgeskemaet uden at det kan tilgås af andre, mens hvert trin desuden skal godkendes før kampagnen går videre til det næste trin. Det sidste vil sige at udvikleren aktivt selv skal bekræfte at et bestemt trin er færdigt.

Da funktionen helst ikke skal ændre udviklerens normale implementationsgang, skal udvikleren desuden kunne vælge at implementere kampagnen lokalt og derfra uploade spørgeskemaet eller lave ændringer direkte i webinterfacet. På denne måde tvinges udvikleren ikke til at ændre sine metoder og udvikle i webinterfacet.



Figur 6.2: Spørgskemakampagne trin

### 6.2.1 Tilpasning

Beskrivelse af de trin hvori spørgeskemaet tilpasses den overordnede side:

#### 1) Opret nyt spørgeskema

Funktionen starter ved at udvikleren vælger at oprette en ny spørgeskemakampagne i spørgeskemastyrings-systemet. Dertil bliver en ny kampagne oprettet med et givent navn og selve spørgeskemaet kan derefter uploades. Et nyt bibliotek til spørgeskemaet bliver oprettet og hvis spørgeskemaet er et Captivate dokument, vil det html-dokument der indeholder det automatisk blive oprettet. Denne del kunne hvis nødvendigt også tilpasses andre former for spørgeskemaer end Captivate.

Spørgeskemaet er nu tilpasset og ligger det rigtige sted, men mangler handleren. Fra udviklersiden kan spørgeskemaet i denne tilstand godt tilgås, hvor normale brugere ikke kan.

#### 2) Upload af spørgeskemahandler

Næste trin er at oprette handleren til spørgeskemaet. En skabelon for en typisk spørgeskemahandler er konstrueret og vises sådan den egentlige handler nemt kan implementeres og dets viste respons nemt kan testes. På samme måde som før kan denne handler nu uploades og udvikleren kan teste handleren uden at normale brugere kan tilgå den.

### 6.2.2 Test

Beskrivelse af delen hvor kampagnen testes:

### 3) Test

I denne fase er det muligt at teste hele kampagnen og man kan i funktionen kontrollere de opfangne data for at se om det hele passer som det skal. Herved skal det også være muligt at kunne lade bestemte brugere ud over udvikleren teste kampagnen før den endnu udsendes.

## 6.2.3 Udsendelse

Delen hvori spørgeskemaet bliver udsendt:

### 4) Udsendelse

I denne sidste del er spørgeskemaet klart til at blive udsendt. Udvikleren kan herfra vælge at udsende, hvorved de rigtige database-rækker bliver opdateret sådan at brugere nu kan se den nye spørgeskemakampagne.

På denne måde kan det ses hvornår en kampagne er klar til at blive udsendt, således at man kan styre dette tidspunkt.

## 6.3 Opsummering

Generelt kan det godt betale sig at standardisere udviklingen af spørgeskemakampagner. En hjælpefunktion til udvikling af spørgeskemakampagner, sparer udvikleren for en del af arbejdet og strukturerer de udviklingsfaser der skal til for at udvikle spørgeskemaerne. Ved at have de sideløbende test-faciliter gøres udviklingen desuden lettere og samtidigt sørges der også for at de kampagner der er under udvikling ikke kan tilgås af brugerne.

Denne udviklingsmetode vil desuden have den fordel at struktureringen, gør det klart hvor langt de forskellige kampagner er i udviklingen.

En mulig udvidelse til spørgeskemastyringen kunne være en række standard-skabeloner for udvidelse af statistikmodulet, da der også her er flere emner der går igen.

## Konklusion

---

Det er igennem undersøgelserne i dette projekt blevet påvist, at brugere er meget vaneprægede og bedst evner at forholde sig til scenarier, som direkte kan sammenlignes med situationer de er vant til. Idet computere abstraherer de tekniske aspekter grafisk for brugeren, vil et simpelt tekstbaseret spørgeskema ikke være tilstrækkeligt, for at den normale, ikke teknisk vidende bruger kan oversætte dette til de billeder han er vant til. Derfor kan man ved at benytte animerede spørgeskemaer, konstruere scenarier af den samme grafiske abstrahering for brugeren, hvorved han meget nemt vil kunne svare og derved give et meget mere ærligt og præcist billede af hans viden.

Det desuden blevet vist at brugere er meget konkurrenceorienterede. De fleste brugere vil gerne svare så godt på et spørgeskema som muligt, selv om dette ikke nødvendigvis repræsenterer brugerens egentlige evne. Dette kan i sammenhæng med nogle spørgeskemaer påvirke nøjagtigheden i negativ retning, hvis brugeren logisk kan gætte sig frem til det korrekte svar. Af de udviklede spørgeskemaer viste dette sig at være et problem ved nogle af de simple spørgeskemaer, da der ved en tekst beskrivelse nødvendigvis er beskrevet emner, som brugeren ellers ved en grafisk abstrahering selv må analysere sig frem til.

Den udviklede sikkerhedsside virkede bedre end forventet, ved at ende med at indeholde over 200 oprettede personer og 600 spørgeskemaer. Dette tyder på at indlærings-motivet var endnu mere populært end ventet og at fokuseringen på at siden skulle være intuitiv og klart opbygget, var korrekt. Af det mest populære emne phishing, fik den ellers mere teknisk orienterede brugergruppe

i gennemsnit kun 2 ud af 5 rigtige, hvilket i statistikken tydede stærkt på at brugerne ikke var i stand til at skelne mellem browserens viste internetside og dens egne informationer.

Adobe Captivate kan benyttes til udvikling af animerede spørgeskemaer da det gør det nemt at optage screenshots, til at danne animerede interaktive program-simulationer. Det skal dog gøres klart at programmet ikke er udviklet til at bygge animerede spørgeskemaer, og det kan dertil stærkt anbefales at man kun benytter Captivate til de dele som skal være animerede og holder al information udefra i et beskrivende sprog som html eller lignende.

Som beskrevet i kapitlet om spørgeskemastyring, er det muligt at strukturere spørgeskemaer som kampagner og udvikle et modul til struktureret styring af spørgeskemaudvikling. Dette vil både mindske udviklingstiden, og give et godt overblik, da alle spørgeskemakampagner så er struktureret i udviklingsfaser.

Animerede spørgeskemaer er en nødvendighed for at få en sigende statistik om et emne så teknisk som sikkerhedsbevidsthed. Udfordringen ligger i at få det konstruerede scenarie til at kunne relateres bedst muligt, til de visninger brugeren er vant til fra sin egen computer. På denne måde vil brugeren ikke behøve at overveje meningen af spørgsmålet, men blot svare som han ville i en normal situation.

# BILAG A

## Kildekode

---

### A.1 ENISA - Simpelt Spørgeskema

#### A.1.1 enisasimpel.html

```
1 <html>
2 <head>
3   <title>ENISA - Sempel</title>
4   <script type="text/javascript" language="JavaScript">
5     function validate(form)
6     {
7       if((form.a.checked == true) ||
8         (form.b.checked == true) ||
9         (form.c.checked == true) ||
10        (form.d.checked == true) ||
11        (form.e.checked == true) ||
12        (form.f.checked == true) ||
13        (form.g.checked == true))
14       { return true; }
15       else
16       {
17         alert("You must select at least one checkbox to continue!");
18         return false;
19       }
20     }
21   </script>
22 </head>
23 <body>
24 <form action="#" onsubmit="return validate(this);">
25   7. Do you have any of the following is in place to protect your computer and electronic data?<br />
26   Please indicate all that apply.<br />
27   <table>
28   <tr>
29     <td>a.</td>
30     <td><input type="checkbox" name="a" /> Anti-virus software that is updated regularly <br /></td>
```

```

31 <tr>
32 <tr>
33 <td>b.</td>
34 <td><input type="checkbox" name="b" /> Firewall <br /></td>
35 </tr>
36 <tr>
37 <td>c.</td>
38 <td><input type="checkbox" name="c" /> Anti-spam filter <br /></td>
39 </tr>
40 <tr>
41 <td>d.</td>
42 <td><input type="checkbox" name="d" /> Good password practices <br /></td>
43 </tr>
44 <tr>
45 <td>e.</td>
46 <td><input type="checkbox" name="e" /> Process of regular backup of data <br /></td>
47 </tr>
48 </tr>
49 <td>f.</td>
50 <td><input type="checkbox" name="f" /> Up-to-date Internetbrowser with encryption <br /></td>
51 </tr>
52 <tr>
53 <td>g.</td>
54 <td><input type="checkbox" name="g" /> Others please indicate
55 <input type="text" name="gt" /><br /> </td>
56 </tr>
57 </table>
58 <input type="submit" value="Submit" />
59 </form>
60 </body>
61 </html>

```

## A.2 ENISA - Forbedret Spørgeskema

### A.2.1 enisaforbedret.html

```

1 <html>
2 <head>
3 <title>ENISA - Forbedret</title>
4 <script type="text/javascript" language="JavaScript">
5   function validate(form)
6   {
7     if(((form.a[0].checked == true)|| (form.a[1].checked == true)|| (form.a[2].checked == true))&&
8       ((form.b[0].checked == true)|| (form.b[1].checked == true)|| (form.b[2].checked == true))&&
9       ((form.c[0].checked == true)|| (form.c[1].checked == true)|| (form.c[2].checked == true))&&
10      ((form.d[0].checked == true)|| (form.d[1].checked == true)|| (form.d[2].checked == true))&&
11      ((form.e[0].checked == true)|| (form.e[1].checked == true)|| (form.e[2].checked == true))&&
12      ((form.f[0].checked == true)|| (form.f[1].checked == true)|| (form.f[2].checked == true)))
13     { return true; }
14     else
15     {
16       alert("You have not checked all required fields!");
17       return false;
18     }
19   }
20 </script>
21 </head>
22 <style>
23   body table{
24     line-height: 14px;
25   }
26   a.info {

```



```
27     position:relative; /*this is the key*/
28     z-index:24; padding-bottom: 0px;
29     line-height: 15px;
30     color:#000;
31     text-decoration: none;
32     border-bottom: 1px dashed #838383;
33     }
34
35     a.info:hover{z-index:25; }
36
37     a.info span{display: none}
38
39     a.info:hover span{ /*the span will display just on :hover state*/
40     line-height: 18px;
41     display:block; padding-left: 4px; padding-right: 4px;
42     position:absolute; width: 15em;
43     left: 3em; top: 1em;
44     border:1px solid;
45     background-color: #cbcbcb;
46     text-align: center
47     }
48
49
50 </style>
51 <body>
52 <form action="#" onsubmit="return validate(this);">
53     7. Do you have any of the following is in place to protect your computer and electronic data?<br />
54     Please indicate all that apply.<br /><br />
55     <table>
56     <tr>
57         <td></td>
58         <td align="center" style="font-size: 11px; width: 30px;
59             font-style: italic; padding-right: 3px;">Don't know</td>
60         <td align="center" style="font-size: 11px; width: 30px;
61             font-style: italic; padding-right: 3px;">Yes</td>
62         <td align="center" style="font-size: 11px; width: 30px;
63             font-style: italic; padding-right: 3px;">No</td>
64     </tr>
65     <tr>
66         <td>a.</td>
67         <td align="center"><input type="radio" name="a" /></td>
68         <td align="center"><input type="radio" name="a" /></td>
69         <td align="center"><input type="radio" name="a" /></td>
70         <td> <a class="info" href="#">Anti-virus<span>Software that identifies and eliminates
71             malicious software, as virusses and other.
72             </span></a>
73             &nbsp;   <span>software that is updated regulary <br /></span></td>
74     </tr>
75     <tr>
76         <td>b.</td>
77         <td align="center"><input type="radio" name="b" /></td>
78         <td align="center"><input type="radio" name="b" /></td>
79         <td align="center"><input type="radio" name="b" /></td>
80         <td> <a class="info" href="#">Firewall<span>Software that blocks unused entries to the computer,
81             thereby increasing the computer's security on the
82             network</span></a> <br /></td>
83     </tr>
84     <tr>
85         <td>c.</td>
86         <td align="center"><input type="radio" name="c" /></td>
87         <td align="center"><input type="radio" name="c" /> </td>
88         <td align="center"><input type="radio" name="c" /> </td>
89         <td><a class="info" href="#">Anti-spam filter<span>Software that identifies and blocks
90             incomming spam.</span></a>
91         <br /></td>
```



```

10 <!-- Begin
11 var gstrEmailTo = "";
12 var gstrEmailSubject = "";
13 var gstrEmailBody = "";
14 var myVar = 0;
15
16 function appendEmailBody(strBody)
17 {
18     var temp_str = strBody;
19     while(temp_str.indexOf("|") > -1)
20     {
21         temp_str = temp_str.substr(0,temp_str.indexOf("|")) + "\r" +
22             temp_str.substr(temp_str.indexOf("|") + 1)
23     }
24     gstrEmailBody += temp_str + "\r";
25 }
26
27 function sendMail()
28 {
29     alert(gstrEmailBody);
30
31     document.emailForm.elements["Results"].value = "\r" + gstrEmailBody;
32     document.emailForm.submit();
33 }
34
35 function padMail(strAddress, strSubject, strBody)
36 {
37     gstrEmailTo = strAddress;
38     gstrEmailSubject = strSubject;
39     var temp_str = strBody;
40     while(temp_str.indexOf("|") > -1)
41     {
42         temp_str = temp_str.substr(0,temp_str.indexOf("|")) + "\r" +
43             temp_str.substr(temp_str.indexOf("|") + 1)
44     }
45     gstrEmailBody = temp_str + "\r";
46 }
47 // End -->
48 </script>
49
50 <form name="emailForm" action="handler.php" method="POST">
51 <input name="Results" type="hidden">
52 </form>
53 </head>
54
55 <body bgcolor="#FFFFFF">
56 <center>
57 <script type="text/javascript">
58 <!--
59 writeDocument(' <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000"
60 codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0"
61 width="660" height="509" ID="Captivate1">' +
62     '<param name="movie" value="wireless_skin.swf">' +
63     '<param name="quality" value="high">' +
64     '<param name="menu" value="false">' +
65     '<param name="loop" value="0">' +
66     '<embed src="wireless_skin.swf" width="660" height="509" loop="0" quality="high"
67 pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash"
68 menu="false"></embed>' +
69     '</object>');
70 //-->
71 </script>
72 <noscript>
73     <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000"
74 codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0"

```

```
75 width="660" height="509" ID="Captivate1">
76   <param name="movie" value="wireless_skin.swf">
77   <param name="quality" value="high">
78   <param name="menu" value="false">
79   <param name="loop" value="0">
80   <embed src="wireless_skin.swf" width="660" height="509" loop="0" quality="high"
81   pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash"
82   menu="false"></embed>
83 </object>
84 </noscript>
85 </center>
86 </body>
87 </html>
```

### A.3.2 handler.php

```
1 <html>
2   <head>
3     <title>
4     </title>
5   </head>
6   <body>
7   <b>Svar fra testen:</b><br />
8   <?php
9     echo nl2br($_POST['Results']);
10  ?>
11 </body>
12 </html>
```

## A.4 Phishing Example

### A.4.1 index.html

```
1 <html>
2   <head>
3     <title>
4     Phishing - Svindel på internettet
5     </title>
6   </head>
7   <style>
8   body {
9     margin:50px 0px; padding:0px;
10    text-align:center;
11      font-family: Verdana;
12      font-size: 2px;
13    }
14
15  #Content {
16    margin:0px auto;
17    text-align:left;
18  }
19
20  .svar {
21    background-color: white;
22  }
23
24  table td {
25    font-family: Verdana;
26    font-size: 11pt;
27  }
```

```
28
29 h1 {
30   font-family: Verdana;
31   font-size: 23pt;
32   margin-bottom: 0px;
33 }
34
35 h2 {
36   font-family: Verdana;
37   font-size: 14pt;
38   font-weight: normal; margin: 0px;
39 }
40
41
42 h3 {
43   font-family: Verdana;
44   font-weight: normal;
45   padding-top: 15px; padding-left: 10px;
46 }
47 a:link, a:visited { color: blue}
48 a:hover { color: black;}
49 a { font-size: 12pt; }
50
51 </style>
52 <body bgcolor="#F1F4F5" style="margin:0px;margin-top: 12px;">
53
54 <table id="content" width="790" border="0" cellpadding="0" cellspacing="0">
55 <tr>
56   <td colspan="2" align="center"></td>
57 </tr>
58 <tr>
59 <td>
60   <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0">
61     <tr style="height: 49px; background-color: #70bded;">
62       <td colspan="2" align="center"></td>
63     </tr>
64     <tr>
65       <td colspan="2" align="center" style="padding-left: 15px; padding-top: 50px;">
66         <h1 style="color: Black;">Phishing</h1><br />
67         <h2>Svindel på internettet</h2>
68       </td>
69     </tr>
70
71     <tr height="100">
72       <td align="center"><a href="internet.htm">Internet Svindel</a></td>
73       <td align="center"><a href="email.htm">Email Svindel</a></td>
74     </tr>
75
76     <tr>
77       <td colspan="2" align="center"><br /><br />
78       Test dig selv og lær mere ved at klikke på et link ovenfor
79     </td>
80   </tr>
81 <tr><td style="height:175px;"></td></tr>
82 <tr style="height: 41px; background-color: #70bded;">
83   <td colspan="2" align="center">
84     Ved du nok til at undgå internetsvindel?
85 </td>
86 </tr>
87 </table>
88 </td>
89 </tr>
90 </table>
91 </body>
92 </html>
```

## A.4.2 internet.htm

```

1 <!-- saved from url=(0013)about:internet -->
2 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
3 "http://www.w3.org/TR/html4/loose.dtd">
4 <html>
5 <head>
6 <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
7 <title>Phishing - Internet</title>
8 <script src="standard.js" type="text/javascript"></script>
9 <script type="text/javascript">
10 <!-- Begin
11 var gstrEmailTo = "";
12 var gstrEmailSubject = "";
13 var gstrEmailBody = "";
14 var myVar = 0;
15
16 function appendEmailBody(strBody)
17 {
18     var temp_str = strBody;
19     while(temp_str.indexOf("|") > -1)
20     {
21         temp_str = temp_str.substr(0,temp_str.indexOf("|")) + "\r" +
22         temp_str.substr(temp_str.indexOf("|") + 1)
23     }
24     gstrEmailBody += temp_str + "\r";
25 }
26
27 function sendMail()
28 {
29     //alert(gstrEmailBody);
30
31     document.emailForm.elements["Results"].value = "\r" + gstrEmailBody;
32     document.emailForm.submit();
33 }
34
35 function padMail(strAddress, strSubject, strBody)
36 {
37     gstrEmailTo = strAddress;
38     gstrEmailSubject = strSubject;
39     var temp_str = strBody;
40     while(temp_str.indexOf("|") > -1)
41     {
42         temp_str = temp_str.substr(0,temp_str.indexOf("|")) + "\r" +
43         temp_str.substr(temp_str.indexOf("|") + 1)
44     }
45     gstrEmailBody = temp_str + "\r";
46 }
47 // End -->
48 </script>
49
50 <form name="emailForm" action="inethandler.php" method="POST">
51 <input name="Results" type="hidden">
52 </form>
53 </head>
54
55 <body bgcolor="#F1F4F5">
56 <center>
57 <script type="text/javascript">
58 <!--
59 writeDocument(' <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000" '+
60 'codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0" '+
61 'width="790" height="545" ID="Captivatel">'+
62 ' <param name="movie" value="phishingweb.swf">'+
63 ' <param name="quality" value="high">'+

```

```

64 ' <param name="menu" value="false">'+
65 ' <param name="loop" value="0">'+
66 ' <embed src="phishingweb.swf" width="790" height="545" loop="0" quality="high"' +
67 '   pluginspage="http://www.macromedia.com/go/getflashplayer"' +
68 '   type="application/x-shockwave-flash" menu="false"></embed>'+
69 '</object>');
70 //-->
71 </script>
72 <noscript>
73   <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000"
74     codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0"
75     width="790" height="545" ID="Captivate1">
76     <param name="movie" value="phishingweb.swf">
77     <param name="quality" value="high">
78     <param name="menu" value="false">
79     <param name="loop" value="0">
80     <embed src="phishingweb.swf" width="790" height="545" loop="0" quality="high"
81       pluginspage="http://www.macromedia.com/go/getflashplayer"
82       type="application/x-shockwave-flash" menu="false"></embed>
83   </object>
84 </noscript>
85 </center>
86 </body>
87 </html>

```

### A.4.3 email.htm

```

1 <!-- saved from url=(0013)about:internet -->
2 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
3 "http://www.w3.org/TR/html4/loose.dtd">
4 <html>
5 <head>
6 <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
7 <title>Phishing - Email</title>
8 <script src="standard.js" type="text/javascript"></script>
9 <script type="text/javascript">
10 <!-- Begin
11 var gstrEmailTo = "";
12 var gstrEmailSubject = "";
13 var gstrEmailBody = "";
14 var myVar = 0;
15
16 function appendEmailBody(strBody)
17 {
18   var temp_str = strBody;
19   while(temp_str.indexOf("|") > -1)
20   {
21     temp_str = temp_str.substr(0,temp_str.indexOf("|")) +
22     "\r" + temp_str.substr(temp_str.indexOf("|") + 1)
23   }
24   gstrEmailBody += temp_str + "\r";
25 }
26
27 function sendMail()
28 {
29   //alert(gstrEmailBody);
30
31   document.emailForm.elements["Results"].value = "\r" + gstrEmailBody;
32   document.emailForm.submit();
33 }
34
35 function padMail(strAddress, strSubject, strBody)
36 {
37   gstrEmailTo = strAddress;

```

```

38 gstrEmailSubject = strSubject;
39 var temp_str = strBody;
40 while(temp_str.indexOf("|") > -1)
41 {
42     temp_str = temp_str.substr(0,temp_str.indexOf("|")) + "\r" +
43     temp_str.substr(temp_str.indexOf("|") + 1)
44 }
45 gstrEmailBody = temp_str + "\r";
46 }
47 // End -->
48 </script>
49
50 <form name="emailForm" action="mailhandler.php" method="POST">
51 <input name="Results" type="hidden">
52 </form>
53 </head>
54
55
56 <body bgcolor="#F1F4F5">
57 <center>
58 <script type="text/javascript">
59 <!--
60 writeDocument(' <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000" ' +
61 'codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0"' +
62 'width="790" height="545" ID="Captivatel">'+
63 ' <param name="movie" value="phishingmail.swf">'+
64 ' <param name="quality" value="high">'+
65 ' <param name="menu" value="false">'+
66 ' <param name="loop" value="0">'+
67 ' <embed src="phishingmail.swf" width="790" height="545" loop="0" quality="high" '+
68 'pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash"' +
69 ' menu="false"></embed>'+
70 ' </object>');
71 //-->
72 </script>
73 <noscript>
74 <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000"
75 codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0"
76 width="790" height="545" ID="Captivatel">
77 <param name="movie" value="phishingmail.swf">
78 <param name="quality" value="high">
79 <param name="menu" value="false">
80 <param name="loop" value="0">
81 <embed src="phishingmail.swf" width="790" height="545" loop="0" quality="high"
82 pluginspage="http://www.macromedia.com/go/getflashplayer"
83 type="application/x-shockwave-flash" menu="false"></embed>
84 </object>
85 </noscript>
86 </center>
87 </body>
88 </html>

```

#### A.4.4 inethandler.php

```

1 <html>
2 <head>
3 <title>
4 Phishing - Internet - Se dine svar
5 </title>
6 </head>
7 <style>
8 body {
9 margin:50px 0px; padding:0px;
10 text-align:center;

```



```
11     font-family: Verdana;
12     font-size: 2px;
13 }
14
15 #Content {
16     margin:0px auto;
17     text-align:left;
18 }
19
20 .svar {
21     background-color: white;
22 }
23
24 table td {
25     font-family: Verdana;
26     font-size: 11pt;
27 }
28
29 h1 {
30     font-family: Verdana;
31     font-size: 15pt;
32     color: #003466;
33 }
34
35 h3 {
36     font-family: Verdana;
37     font-weight: normal;
38     padding-top: 15px; padding-left: 10px;
39 }
40
41 a:link {
42     color: #7897a1;
43     text-decoration: none;
44     font-size: 11px;
45 }
46 a:visited {
47     color: #7897a1;
48     text-decoration: none;
49     font-size: 11px;
50 }
51 a:hover {
52     text-decoration: none;
53     font-size: 11px;
54     color: blue;
55 }
56
57 </style>
58 <body bgcolor="#F1F4F5">
59 <?php
60     function striptostings( $result )
61     {
62         // strip the start info
63
64         $result = strstr(stripslashes($result),
65             "\",\"Student Response\", \"Result\", \"Weight\", \"Latency\"");
66         $num = strpos($result, "\n");
67         $result = substr($result, $num+1);
68
69         // now $result is only the five lines of results
70         // strip these five lines into an array
71
72         for($i = 0; $i < 4; $i++)
73         {
74             $num = strpos($result, "\n");
75             $arr[$i] = substr($result, 0, $num);
```

```

76     $result = substr($result, $num+1);
77 }
78 $arr[4] = $result;
79
80 return $arr;
81 }
82
83 function stripastring( $string, $print )
84 {
85     // is in 10 parts seperated by "," we need parts 3 and 7
86     $string;
87     $num = strpos($string, "\",\""); // 1
88     $string = substr($string, $num+1);
89     $num = strpos($string, "\",\""); // 2
90     $string = substr($string, $num+1);
91     $num = strpos($string, "\",\""); // 3 (we need this one)
92     $arr[0] = substr($string, 2, $num-2);
93     $string = substr($string, $num+1);
94     $num = strpos($string, "\",\""); // 4
95     $string = substr($string, $num+1);
96     $num = strpos($string, "\",\""); // 5
97     $string = substr($string, $num+1);
98     $num = strpos($string, "\",\""); // 6
99     $string = substr($string, $num+1);
100    $num = strpos($string, "\",\""); // 7 (last one we need)
101    $arr[1] = substr($string, 2, $num-2);
102    $string = substr($string, $num+1);
103
104    if($print == 1)
105    {
106        if($arr[1] != 1)
107            echo $arr[0].".".$arr[1]."<br />";
108        else
109            echo $arr[0]."<br />";
110    }
111    if($arr[1] != 1)
112        return $arr[0].".".$arr[1];
113    else
114        return $arr[0];
115 }
116
117
118 $result = nl2br($_POST['Results']).": til";
119 //echo $result;
120 //echo "<br />Her: <br />";
121
122 $a = striptostrings( $result );
123
124 stripastring( $a[0], 0 );
125 stripastring( $a[1], 0 );
126 stripastring( $a[2], 0 );
127 stripastring( $a[3], 0 );
128 stripastring( $a[4], 0 );
129
130 // Læg alt op i databasen
131
132 mysql_connect("localhost", "tre_ben_dk", "kablooie") or die(mysql_error());
133 mysql_select_db("tre_ben_dk") or die(mysql_error());
134 $result = mysql_query("INSERT INTO phishing_internet (id, q1, q2, q3, q4, q5) VALUES (".
135     "'', ".
136     "'',".stripastring( $a[0], 0 )."' ' ".
137     "'',".stripastring( $a[1], 0 )."' ' ".
138     "'',".stripastring( $a[2], 0 )."' ' ".
139     "'',".stripastring( $a[3], 0 )."' ' ".
140     "'',".stripastring( $a[4], 0 )."' ' ".

```

```

141                                     ");" or die(mysql_error());
142     mysql_close();
143
144 ?>
145 <table id="content" width="790" border="0" cellpadding="0" cellspacing="0">
146 <tr>
147     <td align="center"><h1>Svar fra test</h1></td>
148 </tr>
149 <tr>
150     <td align="right"><a href="index.html">Tilbage til startside</a></td>
151 </tr>
152 <tr>
153 <td>
154 <!-- Start Svarblok ----->
155 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0">
156 <tr style="height: 49px; background-color: #70bde4;">
157     <td colspan="2"><h3>Spørgsmål 1 ud af 5 - Danske netbank</h3></td>
158 </tr>
159 <tr>
160     <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
161     <div style="font-weight: bold; font-color: green;">
162     <?php
163         $r = substr(stripastring( $a[0], 0 ), 2, 1);
164         $r2 = substr(stripastring( $a[0], 0 ), 3);
165         if( ($r == "N") &&
166             (substr_count($r2, "A") > 0) &&
167             (substr_count($r2, "B") == 0) &&
168             (substr_count($r2, "C") == 0) &&
169             (substr_count($r2, "E") == 0))
170             { echo "Korrekt!"; }
171             else
172             { echo "Forkert!"; }
173
174     ?>
175 </div>
176 <br />
177 <?php
178     echo "De svarede: ";
179     if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
180     if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
181     if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
182 ?>
183 <div style="font-size: 8pt; padding-left: 10px;">
184 <?php
185     if(substr_count($r2, "A") > 0)
186     { echo "- <font style=\"color:green\">Adressen var forkert</font><br />"; }
187     if(substr_count($r2, "B") > 0)
188     { echo "- <font style=\"text-decoration: underline; color:red\">".
189         "Der manglede 'https' i adressen</font><br />"; }
190     if(substr_count($r2, "C") > 0)
191     { echo "- <font style=\"text-decoration: underline; color:red\">".
192         "Der manglede en lås</font><br />"; }
193     if(substr_count($r2, "D") > 0)
194     { echo "- <font style=\"color:green\">Siden virkede forkert</font><br />"; }
195     if(substr_count($r2, "E") > 0)
196     { echo "- <font style=\"text-decoration: underline; color:red\">Ved ikke</font><br />"; }
197     ?>
198 </div>
199 </td>
200 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
201 
202 </td>
203 </tr>
204 <tr>
205 <td align="left" colspan="2" style="padding-left: 15px; padding-right: 100px;

```

```

206 padding-bottom: 10px; font-size: 13px;">
207 <br />
208 Adressen på denne side viser en ip-adresse istedet for Danske NetBanks egen adresse.
209 <br /><br />
210 Pga. dette kan man ikke være sikker på om den viste side virkelig er Danske NetBank<br />
211 og det frarådes derfor at stole på denne side og indtaste nogen form for brugerinformation
212 på den.
213 </td>
214 </tr>
215 </table>
216 <!-- Slut Svarblok ----->
217 </td>
218 </tr>
219 <tr style="padding-top: 50px;">
220 <td align="right"><a href="index.html">Tilbage til startside</a></td>
221 </tr>
222 <tr>
223 <td>
224 <!-- Start Svarblok ----->
225 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0">
226 <tr style="height: 49px; background-color: #70bde3;">
227 <td colspan="2"><h3>Spørgsmål 2 ud af 5 - Tastselv Skat</h3></td>
228 </tr>
229 <tr>
230 <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
231 <div style="font-weight: bold; font-color: green;">
232 <?php
233 $r = substr(stripastring( $a[1], 0 ), 2, 1);
234 $r2 = substr(stripastring( $a[1], 0 ), 3);
235 if( ($r == "N") &&
236 (substr_count($r2, "A") == 0) &&
237 (substr_count($r2, "B") == 0) &&
238 (substr_count($r2, "C") > 0) &&
239 (substr_count($r2, "E") == 0))
240 { echo "Korrekt!"; }
241 else
242 { echo "Forkert!"; }
243
244 ?>
245 </div>
246 <br />
247 <?php
248 echo "De svarede: ";
249 if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
250 if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
251 if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
252 ?>
253 <div style="font-size: 8pt; padding-left: 10px;">
254 <?php
255 if(substr_count($r2, "A") > 0)
256 { echo "- <font style=\"text-decoration: underline; color:red;\">".
257 "Adressen var forkert</font><br />"; }
258 if(substr_count($r2, "B") > 0)
259 { echo "- <font style=\"text-decoration: underline; color:red;\">".
260 "Der manglede 'https' i adressen</font><br />"; }
261 if(substr_count($r2, "C") > 0)
262 { echo "- <font style=\"color:green;\">Der manglede en lås</font><br />"; }
263 if(substr_count($r2, "D") > 0)
264 { echo "- <font style=\"color:green;\">Siden virkede forkert</font><br />"; }
265 if(substr_count($r2, "E") > 0)
266 { echo "- <font style=\"text-decoration: underline; color:red;\">Ved ikke</font><br />"; }
267 ?>
268 </div>
269 </td>
270 <td align="right" valign="top" style=" padding-top: 10px; padding-right: 10px;">

```

```

271     
272     </td>
273 </tr>
274 <tr>
275     <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px;
276     padding-bottom: 10px; font-size: 13px;">
277     <br />
278     Denne sides adresse er rigtig nok men låsen mangler.<br /><br />
279     Den lukkede lås sammen med 'https' til forskel for det normale 'http' istarten
280     af adressen, viser at websiden har certifikat og kan sende data sikkert mellem dig og siden.
281     <br />
282     Da låsen ikke er tilstede frarådes det at stole på denne side og indtaste nogen
283     form for brugerinformation på den.<br />
284     <br />
285     Bemærk: På nogle browsere vises låseikonet ikke oppe vedsiden af adressen,
286     men nede på baren under websiden.
287     </td>
288 </tr>
289 </table>
290 <!-- Slut Svarblok ----->
291 </td>
292 </tr>
293 <tr style="padding-top: 50px;">
294     <td align="right"><a href="index.html">Tilbage til startside</a></td>
295 </tr>
296 <tr>
297 <td>
298 <!-- Start Svarblok ----->
299 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0">
300 <tr style="height: 49px; background-color: #70bde4;">
301     <td colspan="2"><h3>Spørgsmål 3 ud af 5 - Paypal</h3></td>
302 </tr>
303 <tr>
304     <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
305     <div style="font-weight: bold; font-color: green;">
306     <?php
307         $r = substr(stripastring( $a[2], 0 ), 2, 1);
308         $r2 = substr(stripastring( $a[2], 0 ), 3);
309         if( ($r == "Y"))
310             { echo "Korrekt!"; }
311         else
312             { echo "Forkert!"; }
313     <?php
314     </div>
315 </td>
316 <td style="width: 510px; padding-left: 15px; padding-top: 10px;">
317     <?php
318     echo "De svarede: ";
319     if($r == "Y") { echo "<font style=\"color:green\">Ja</font><br />"; }
320     if($r == "N") { echo "<font style=\"color:red\">Nej</font><br />"; }
321     if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
322     <?php
323     <div style="font-size: 8pt; padding-left: 10px;">
324     <?php
325     if(substr_count($r2, "A") > 0)
326     { echo "- <font style=\"text-decoration: underline; color:red;\">".
327     "Adressen var forkert</font><br />"; }
328     if(substr_count($r2, "B") > 0)
329     { echo "- <font style=\"text-decoration: underline; color:red;\">".
330     "Der manglede 'https' i adressen</font><br />"; }
331     if(substr_count($r2, "C") > 0)
332     { echo "- <font style=\"text-decoration: underline; color:red;\">".
333     "Der manglede en lås</font><br />"; }
334     if(substr_count($r2, "D") > 0)
335     { echo "- <font style=\"text-decoration: underline; color:red;\">".

```

```

336         "Siden virkede forkert</font><br />"; }
337     if(substr_count($r2, "E") > 0)
338     { echo "- <font style=\"text-decoration: underline; color:red;\">Ved ikke</font><br />"; }
339     ?>
340 </div>
341 </td>
342 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
343     
344 </td>
345 </tr>
346 <tr>
347     <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px;
348     padding-bottom: 10px; font-size: 13px;">
349     <br />
350     Denne side er ikke phished.<br /><br />
351     Siden har 'https' og en lås sådan at vi ved der kan sendes data sikkert mellem os og siden.
352     <br />
353     Derudover kan vi se at adressen passer.<br (>
354     Da disse tre ting er opfyldt er der ingen grund til ikke at stole på siden.
355     </td>
356 </tr>
357 </table>
358 <!-- Slut Svarblok ----->
359 </td>
360 </tr>
361 <tr style="padding-top: 50px;">
362     <td align="right"><a href="index.html">Tilbage til startside</a></td>
363 </tr>
364 <tr>
365 <td>
366 <!-- Start Svarblok ----->
367 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0">
368 <tr style="height: 49px; background-color: #70bde4;">
369     <td colspan="2"><h3>Spørgsmål 4 ud af 5 - Apple store</h3></td>
370 </tr>
371 <tr>
372     <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
373     <div style="font-weight: bold; font-color: green;">
374     <?php
375         $r = substr(stripastring( $a[3], 0 ), 2, 1);
376         $r2 = substr(stripastring( $a[3], 0 ), 3);
377         if( ($r == "N") &&
378             (substr_count($r2, "A") == 0) &&
379             ((substr_count($r2, "B") > 0) || (substr_count($r2, "C") > 0)) &&
380             (substr_count($r2, "E") == 0))
381             { echo "Korrekt!"; }
382             else
383             { echo "Forkert!"; }
384         ?>
385     </div>
386     <br />
387 <?php
388     echo "De svarede: ";
389     if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
390     if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
391     if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
392     ?>
393 <div style="font-size: 8pt; padding-left: 10px;">
394     <?php
395     if(substr_count($r2, "A") > 0)
396     { echo "- <font style=\"text-decoration: underline; color:red;\">".
397     "Adressen var forkert</font><br />"; }
398     if(substr_count($r2, "B") > 0)

```

```

401     { echo "- <font style=\"color:green;\">Der manglede 'https' i adressen</font><br />"; }
402     if(substr_count($r2, "C") > 0)
403     { echo "- <font style=\"color:green;\">Der manglede en lås</font><br />"; }
404     if(substr_count($r2, "D") > 0)
405     { echo "- <font style=\"color:green;\">Siden virkede forkert</font><br />"; }
406     if(substr_count($r2, "E") > 0)
407     { echo "- <font style=\"text-decoration: underline; color:red;\">Ved ikke</font><br />"; }
408     ?>
409     </div>
410 </td>
411 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
412 
413 </td>
414 </tr>
415 <tr>
416 <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px;
417 padding-bottom: 10px; font-size: 13px;">
418 <br />
419 Denne sides adresse er rigtig nok, men låsen mangler sammen med 'https'.<br /><br />
420 Låse eller andre ikoner på selve websiden, holder ingen garantier for at siden
421 virkelig er sikker.<br />
422 Man bør kun stole på browserens egne indikatorer.
423 <br />
424 Da låsen og 'https' ikke er tilstede, frarådes det at stole på denne side og indtaste
425 nogen form for brugerinformation på den.<br />
426 <br />
427 </td>
428 </tr>
429 </table>
430 <!-- Slut Svarblok ----->
431 </td>
432 </tr>
433 <tr style="padding-top: 50px;">
434 <td align="right"><a href="index.html">Tilbage til startside</a></td>
435 </tr>
436 <tr>
437 <td>
438 <!-- Start Svarblok ----->
439 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0">
440 <tr style="height: 49px; background-color: #70bde4;">
441 <td colspan="2"><h3>Spørgsmål 5 ud af 5 - Jyske bank</h3></td>
442 </tr>
443 <tr>
444 <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
445 <div style="font-weight: bold; font-color: green;">
446 <?php
447     $r = substr(stripastring( $a[4], 0 ), 2, 1);
448     $r2 = substr(stripastring( $a[4], 0 ), 3);
449     if( ($r == "N") &&
450         (substr_count($r2, "A") > 0) &&
451         (substr_count($r2, "B") == 0) &&
452         (substr_count($r2, "C") == 0) &&
453         (substr_count($r2, "E") == 0))
454     { echo "Korrekt!"; }
455     else
456     { echo "Forkert!"; }
457
458     ?>
459 </div>
460 <br />
461 <?php
462     echo "De svarede: ";
463     if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
464     if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
465     if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }

```

```

466 ?>
467 <div style="font-size: 8pt; padding-left: 10px;">
468 <?php
469 if(substr_count($r2, "A") > 0)
470 { echo "- <font style=\"color:green;\">Adressen var forkert</font><br />"; }
471 if(substr_count($r2, "B") > 0)
472 { echo "- <font style=\"text-decoration: underline; color:red;\">".
473   "Der manglede 'https' i adressen</font><br />"; }
474 if(substr_count($r2, "C") > 0)
475 { echo "- <font style=\"text-decoration: underline; color:red;\">".
476   "Der manglede en lås</font><br />"; }
477 if(substr_count($r2, "D") > 0) { echo "- <font style=\"color:green;\">".
478   "Siden virkede forkert</font><br />"; }
479 if(substr_count($r2, "E") > 0)
480 { echo "- <font style=\"text-decoration: underline; color:red;\">Ved ikke</font><br />"; }
481 ?>
482 </div>
483 </td>
484 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
485 
486 </td>
487 </tr>
488 <tr>
489 <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px;
490 padding-bottom: 10px; font-size: 13px;">
491 <br />
492 Adressen på denne side er forkert. Bemærk at det sidste 'k' er udskiftet til et 'c'.
493 <br /><br />
494 Så små ændringer som dette kan gøre at siden man ser, rent faktisk er en helt anden.
495 Derfor bør man
496 altid kigge efter når man følger et link (især fra en email) om den peger det rigtige
497 sted hen.<br />
498 <br />
499 Da vi ikke kan være sikre på at denne side virkelig er jyske banks kan det ikke tilrådes
500 at stole på den eller indtaste nogen
501 form for følsomme oplysninger på den.
502 </td>
503 </tr>
504 </table>
505 <!-- Slut Svarblok ----->
506 </td>
507 </tr>
508 </table>
509 </body>
510 </html>

```

### A.4.5 mailhandler.php

```

1 <html>
2 <head>
3 <title>Phishing - Email - Se dine svar
4 </title>
5 </head>
6 <style>
7 body {
8   margin:50px 0px; padding:0px;
9   text-align:center;
10    font-family: Verdana;
11    font-size: 2px;
12 }
13
14
15 #Content {
16   margin:0px auto;

```



```
17 text-align:left;
18 }
19
20 .svar {
21 background-color: white;
22 }
23
24 table td {
25 font-family: Verdana;
26 font-size: 11pt;
27 }
28
29 h1 {
30 font-family: Verdana;
31 font-size: 15pt;
32 color: #003466;
33 }
34
35 h3 {
36 font-family: Verdana;
37 font-weight: normal;
38 padding-top: 15px; padding-left: 10px;
39 }
40
41 a:link {
42 color: #7897a1;
43 text-decoration: none;
44 font-size: 11px;
45 }
46 a:visited {
47 color: #7897a1;
48 text-decoration: none;
49 font-size: 11px;
50 }
51 a:hover {
52 text-decoration: none;
53 font-size: 11px;
54 color: blue;
55 }
56 </style>
57 <body bgcolor="#F1F4F5">
58 <?php
59 function striptostrings( $result )
60 {
61 // strip the start info
62 $result = strstr(stripslashes($result),
63 "\",\"Student Response\", \"Result\", \"Weight\", \"Latency\"");
64 $num = strpos($result, "\n");
65 $result = substr($result, $num+1);
66
67 // now $result is only the five lines of results
68 // strip these five lines into an array
69
70 for($i = 0; $i < 4; $i++)
71 {
72 $num = strpos($result, "\n");
73 $arr[$i] = substr($result, 0, $num);
74 $result = substr($result, $num+1);
75 }
76 $arr[4] = $result;
77
78 return $arr;
79 }
80
81 function stripastring( $string, $print )
```

```

82 {
83     // is in 10 parts separated by "," we need parts 3 and 7
84
85     $num = strpos($string, "\",\""); // 1
86     $string = substr($string, $num+1);
87     $num = strpos($string, "\",\""); // 2
88     $string = substr($string, $num+1);
89     $num = strpos($string, "\",\""); // 3 (we need this one)
90     $arr[0] = substr($string, 2, $num-2);
91     $string = substr($string, $num+1);
92     $num = strpos($string, "\",\""); // 4
93     $string = substr($string, $num+1);
94     $num = strpos($string, "\",\""); // 5
95     $string = substr($string, $num+1);
96     $num = strpos($string, "\",\""); // 6
97     $string = substr($string, $num+1);
98     $num = strpos($string, "\",\""); // 7 (last one we need)
99     $arr[1] = substr($string, 2, $num-2);
100    $string = substr($string, $num+1);
101
102    if($print == 1)
103    {
104        if($arr[1] != 1)
105            echo $arr[0].".".$arr[1]."<br />";
106        else
107            echo $arr[0]."<br />";
108    }
109    if($arr[1] != 1)
110        return $arr[0].".".$arr[1];
111    else
112        return $arr[0];
113 }
114
115 $result = nl2br($_POST['Results']);
116 //echo $result;
117 //echo "<br />Her: <br />";
118
119 $a = striptostings( $result );
120 stripastring( $a[0], 0 );
121 stripastring( $a[1], 0 );
122 stripastring( $a[2], 0 );
123 stripastring( $a[3], 0 );
124 stripastring( $a[4], 0 );
125
126
127 mysql_connect("localhost", "tre_ben_dk", "kablooie") or die(mysql_error());
128 mysql_select_db("tre_ben_dk") or die(mysql_error());
129 $result = mysql_query("INSERT INTO phishing_email (id, q1, q2, q3, q4, q5) VALUES (".
130     "'', ".
131     "'',".stripastring( $a[0], 0 )."' ' ".
132     "'',".stripastring( $a[1], 0 )."' ' ".
133     "'',".stripastring( $a[2], 0 )."' ' ".
134     "'',".stripastring( $a[3], 0 )."' ' ".
135     "'',".stripastring( $a[4], 0 )."' ' ".
136     "');") or die(mysql_error());
137
138     mysql_close();
139
140 ?>
141 <table id="content" width="790" border="0" cellpadding="0" cellspacing="0">
142 <tr>
143 <td align="center"><h1>Svar fra test</h1></td>
144 </tr>
145 <tr>
146 <td align="right"><a href="index.html">Tilbage til startside</a></td>

```



```

212     beskyttede entiteter med ens cpr-nummer.
213
214     </td>
215 </tr>
216 </table>
217 <!-- Slut Svarblok ----->
218 </td>
219 </tr>
220 <tr style="padding-top: 50px;">
221     <td align="right"><a href="index.html">Tilbage til startside</a></td>
222 </tr>
223 <tr>
224 <td>
225 <!-- Start Svarblok ----->
226 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0">
227 <tr style="height: 49px; background-color: #70bde4;">
228     <td colspan="2"><h3>Spørgsmål 2 ud af 5 - Paypal</h3></td>
229 </tr>
230 <tr>
231     <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
232     <div style="font-weight: bold; font-color: green;">
233         <?php
234             $r = substr(stripastring( $a[1], 0 ), 2, 1);
235             $r2 = substr(stripastring( $a[1], 0 ), 3);
236             if( ($r == "N") &&
237                 (substr_count($r2, "B") > 0) &&
238                 (substr_count($r2, "E") == 0))
239             { echo "Korrekt!"; }
240             else
241             { echo "Forkert!"; }
242
243             ?>
244     </div>
245 <br />
246     <?php
247         echo "De svarede: ";
248         if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
249         if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
250         if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
251     ?>
252 <div style="font-size: 8pt; padding-left: 10px;">
253     <?php
254         if(substr_count($r2, "A") > 0)
255         { echo "- <font style=\"color:green\">Forkert afsenderemail</font><br />"; }
256         if(substr_count($r2, "B") > 0)
257         { echo "- <font style=\"color:green\">Et link \"passer\" ikke ind</font><br />"; }
258         if(substr_count($r2, "C") > 0)
259         { echo "- <font style=\"text-decoration: underline; color:red\">".
260             "Der blev bedt om &nbsp;&nbsp;upassende oplysninger</font><br />"; }
261         if(substr_count($r2, "D") > 0)
262         { echo "- <font style=\"color:green\">Emailen virkede forkert</font><br />"; }
263         if(substr_count($r2, "E") > 0)
264         { echo "- <font style=\"text-decoration: underline; color:red\">Ved ikke</font><br />"; }
265     ?>
266 </div>
267 </td>
268 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
269     
271 </td>
272 </tr>
273 <tr>
274 <td align="left" colspan="2" style="padding-left: 15px; padding-right: 100px;
275     padding-bottom: 10px; font-size: 13px;">
276 <br />

```



```

342     style="border-color: #a8a8a8;" />
343 </td>
344 </tr>
345 <tr>
346     <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px;
347     padding-bottom: 10px; font-size: 13px;">
348     <br />
349     Linket i denne email går til: http://tastselv_skat.DK/SKAT.aspx?oID=349844<br /><br />
350
351     Denne mail ser helt rigtig ud, men hvis man kigger efter på linket kan man se at punktummet
352     imellem tastselv og skat
353     er blevet byttet ud med et '_'. Dette gør at linket potentielt slet ikke ender hos SKAT.<br />
354     Hvis man er usikker på om et link virkelig ender det rigtige sted, bør man hvis muligt
355     indtaste adressen i hånden
356     med en adresse man er sikker på og så klikke sig frem til det rigtige sted. <br />
357     I dette tilfælde er den sikre metode at gå ind på www.skat.dk og derfra klikke sig vej
358     hen til tastselv skat. På
359     denne måde er man sikker på at man ikke kommer ind på en phished hjemmeside.<br />
360     Det kan ikke anbefales at stole på denne email.
361 </td>
362 </tr>
363 </table>
364 <!-- Slut Svarblok ----->
365 </td>
366 </tr>
367 <tr style="padding-top: 50px;">
368     <td align="right"><a href="index.html">Tilbage til startside</a></td>
369 </tr>
370 <tr>
371 <td>
372 <!-- Start Svarblok ----->
373 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0">
374 <tr style="height: 49px; background-color: #70bde4;">
375     <td colspan="2"><h3>Spørgsmål 4 ud af 5 - Apple store</h3></td>
376 </tr>
377 <tr>
378     <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
379     <div style="font-weight: bold; font-color: green;">
380     <?php
381         $r = substr(stripastring( $a[3], 0 ), 2, 1);
382         $r2 = substr(stripastring( $a[3], 0 ), 3);
383         if($r == "Y")
384             { echo "Korrekt!"; }
385         else
386             { echo "Forkert!"; }
387
388     ?>
389 </div>
390 <br />
391 <?php
392     echo "De svarede: ";
393     if($r == "Y") { echo "<font style=\"color:green\">Ja</font><br />"; }
394     if($r == "N") { echo "<font style=\"color:red\">Nej</font><br />"; }
395     if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
396 ?>
397 <div style="font-size: 8pt; padding-left: 10px;">
398 <?php
399     if(substr_count($r2, "A") > 0)
400     { echo "- <font style=\"text-decoration: underline; color:red;\">".
401       "Forkert afsenderemail</font><br />"; }
402     if(substr_count($r2, "B") > 0)
403     { echo "- <font style=\"text-decoration: underline; color:red;\">".
404       "Et link \"passer\" ikke ind</font><br />"; }
405     if(substr_count($r2, "C") > 0)
406     { echo "- <font style=\"text-decoration: underline; color:red;\">".

```



```

472     "Forkert afsenderemail</font><br />"; }
473     if(substr_count($r2, "B") > 0)
474     { echo "- <font style=\"color:green;\">Et link \"passer\" ikke ind</font><br />"; }
475     if(substr_count($r2, "C") > 0)
476     { echo "- <font style=\"text-decoration: underline; color:red;\">".
477       "Der blev bedt om &nbsp;&nbsp;&nbsp;upassende oplysninger</font><br />"; }
478     if(substr_count($r2, "D") > 0)
479     { echo "- <font style=\"text-decoration: underline; color:red;\">".
480       "Emailen virkede forkert</font><br />"; }
481     if(substr_count($r2, "E") > 0)
482     { echo "- <font style=\"text-decoration: underline; color:red;\">Ved ikke</font><br />"; }
483     ?>
484 </div>
485 </td>
486 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
487   
489 </td>
490 </tr>
491 <tr>
492   <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px;
493     padding-bottom: 10px; font-size: 13px;">
494     <br />
495     Linket i denne email g r til: http://www.givmigdinadgangskode.dk<br /><br />
496
497     Denne mail ser rigtig ud, og de f rste links g r virkelig til falcks hjemmeside.
498     Men det sidste link der leder til  ndring af ens brugernavn og adgangskode er
499     tydeligvis falsk.<br />
500     Det kan derfor ikke anbefales at stolle p  denne email.
501   </td>
502 </tr>
503 </table>
504 <!-- Slut Svarblok ----->
505 </td>
506 </tr>
507 </table>
508 </body>
509 </html>

```

## A.5 Iteration 1 - Website for sikkerhedsbevidsthed

### A.5.1 Sql s tninger - Create

```

1 CREATE TABLE `sik_test` (
2   `id` INT( 12 ) NOT NULL AUTO_INCREMENT PRIMARY KEY ,
3   `name` VARCHAR( 100 ) NOT NULL ,
4   `description` TEXT NOT NULL
5 );
6
7 CREATE TABLE `sik_person` (
8   `id` INT( 12 ) NOT NULL AUTO_INCREMENT,
9   `username` VARCHAR( 100 ) NOT NULL ,
10  `password` VARCHAR( 100 ) NOT NULL ,
11  `name` VARCHAR( 100 ) NOT NULL ,
12  `gender` VARCHAR( 1 ) NOT NULL ,
13  `age` INT( 3 ) NOT NULL ,
14  `computerskill` VARCHAR( 2 ) NOT NULL ,
15  PRIMARY KEY ( `id` ) ,
16  UNIQUE KEY username (username)
17 );
18
19 CREATE TABLE `sik_testdata` (

```



```

20 `id` INT(12) NOT NULL AUTO_INCREMENT PRIMARY KEY,
21 `person_id` INT(12) NOT NULL, `test_id` INT(12) NOT NULL,
22 `data1` VARCHAR(50) NOT NULL, `data2` VARCHAR(50) NOT NULL,
23 `data3` VARCHAR(50) NOT NULL, `data4` VARCHAR(50) NOT NULL,
24 `data5` VARCHAR(50) NOT NULL, `data6` VARCHAR(50) NOT NULL,
25 `data7` VARCHAR(50) NOT NULL, `data8` VARCHAR(50) NOT NULL,
26 `data9` VARCHAR(50) NOT NULL, `data10` VARCHAR(50) NOT NULL,
27 `data11` VARCHAR(50) NOT NULL, `data12` VARCHAR(50) NOT NULL,
28 `data13` VARCHAR(50) NOT NULL, `data14` VARCHAR(50) NOT NULL,
29 `data15` VARCHAR(50) NOT NULL, `data16` VARCHAR(50) NOT NULL,
30 `data17` VARCHAR(50) NOT NULL, `data18` VARCHAR(50) NOT NULL,
31 `data19` VARCHAR(50) NOT NULL, `data20` VARCHAR(50) NOT NULL,
32 `data21` VARCHAR(50) NOT NULL, `data22` VARCHAR(50) NOT NULL,
33 `data23` VARCHAR(50) NOT NULL, `data24` VARCHAR(50) NOT NULL,
34 `data25` VARCHAR(50) NOT NULL, `data26` VARCHAR(50) NOT NULL,
35 `data27` VARCHAR(50) NOT NULL, `data28` VARCHAR(50) NOT NULL,
36 `data29` VARCHAR(50) NOT NULL, `data30` VARCHAR(50) NOT NULL,
37 `data31` VARCHAR(50) NOT NULL, `data32` VARCHAR(50) NOT NULL,
38 FOREIGN KEY (person_id) REFERENCES sik_person(id),
39 FOREIGN KEY (test_id) REFERENCES sik_test(id)
40 );

```

## A.5.2 header.php

```

1 <?php
2     session_start();
3     header("Cache-control: private"); // IE 6 Fix.
4     require_once 'classes.php';
5     $validuser = check();
6
7 >
8 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
9     "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
10 <html>
11 <head>
12 <title>Web</title>
13 <meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
14 <link href="http://sikkerhed.tre-ben.dk/style.css" rel="stylesheet" type="text/css" />
15 <script src="http://sikkerhed.tre-ben.dk/script.js" type="text/javascript"></script>
16 <?php
17 if($start != "")
18 {
19
20 echo "<script type=\"text/javascript\">\n".
21     "function settext(id)\n".
22     "{\n".
23     "    switch(id) {\n".
24     "        {\n";
25
26         for($i=2; $i <= countgettesttext(); $i++)
27         {
28             $result = gettesttext($i);
29             foreach($result as $user_object)
30             {
31                 echo "case '$i':\n".
32                     "document.form.txi.value = \"".$user_object->name."\";\n".
33                     "document.form.tx.value = \"".$user_object->description."\";\n".
34                     "break;\n";
35             }
36         }
37     $result = gettesttext(1);
38     foreach($result as $user_object)
39     {
40         echo "default:\n";

```

```

41         echo "document.form.txi.value = \"".$user_object->name."\";\n";
42         echo "document.form.tx.value = \"".$user_object->description."\";\n";
43     }
44
45     echo "};";
46
47     echo "};";
48
49 echo "</script>";
50 } ?>
51 </head>
52 <body bgcolor="#70BDED"
53 <?php if(($page != "i") && ($notstart != "1")){ ?> onload="settext('0')" <?php } ?>
54 <?php if($logind == "1"){ ?> onload="document.frmLogin.txtUsername.focus();" <?php } ?>
55 >
56 <table id="content" width="835" border="0" cellpadding="0" cellspacing="0">
57 <tr>
58 <td colspan="3" valign="bottom" align="center"
59     style="width: 835px; height: 74px;
60     background-image: url('http://sikkerhed.tre-ben.dk/images/Slice_3.jpg');">
61 <div id="menu">
62 <?php
63     if($validuser == false)
64     {
65         echo "<ul><li>".
66             "<a href='\"http://sikkerhed.tre-ben.dk/opretbruger.php\">Opret bruger</a></li>".
67             "<li id='\"menur\">".
68             "<a href='\"http://sikkerhed.tre-ben.dk/logind.php\">Log ind</a></li></ul>";
69
70         if($restricted != "")
71         {
72             echo "<script type='\"text/javascript\"'> window.location="
73                 "\"http://sikkerhed.tre-ben.dk/ingenadgang.php\"</script>";
74         }
75     }
76 }
77 else
78 {
79     echo "<ul><li><a href='\"http://sikkerhed.tre-ben.dk/start.php\">Start</a></li>".
80         "<li id='\"menur\">".
81         "<a href='\"http://sikkerhed.tre-ben.dk/logud.php\">Log ud</a></li></ul>";
82 }
83 ?>
84 </div>
85 </td>
86 </tr>
87 <tr>
88 <td align="left" valign="top"
89     style="width: 19px; height: auto;
90     background: url(http://sikkerhed.tre-ben.dk/images/Slice_5pixel.jpg) repeat-y;"></td>
91 <td align="center" valign="middle"
92     style="width: 799px; height:440px; background-color: white; padding:0px;">
93

```

### A.5.3 footer.php

```

1 </td>
2 <td style="width: 17px; height: auto;
3     background: url(http://sikkerhed.tre-ben.dk/images/Slice_7pixel.jpg) repeat-y;"></td>
4 </tr>
5 <tr>
6 <td colspan="3" align="left" valign="top">
7 </td>

```

```
9 </tr>
10 </table>
11 </body>
12 </html>
```

### A.5.4 index.php

```
1 <?php $page = "i"; include("header.php"); ?>
2
3 <div style="width: 80%; text-align: left; padding: 13px;">
4
5 <p style="width: 600px;">
6 Denne side er en del af et eksamensprojekt i informatik på Danmarks Tekniske Universitet, og
7 indeholder spørgeskemaer om sikkerhedsbevidsthed.<br />
8 <br />
9 Vil du vide lidt mere om computersikkerhed, så test dig selv via. spørgeskemaerne på denne side.
10 Det er helt gratis og mens du lærer af spørgeskemaerne, lagres testene til statistisk brug.
11 <br /><br />
12 Tryk på start for at komme igang og oprette en ny bruger.
13 <br /><br /><br /><br />
14
15 </p>
16 <table width="100%" cellpadding="0" cellspacing="0">
17 <tr>
18 <td colspan="2" align="center">
19 <form name="form" action="opretbruger.php" method="post">
20 <input type="submit" value="Start" style="width: 90px; height: 30px;" />
21 </form>
22 <br /><br /><br /><br /><br />
23 </td>
24 </tr>
25 <tr>
26 <td></td>
27 <td>
28 <p style="width: 600px;">
29 Denne side er lavet som del af et kandidatspeciale <br />
30 på Danmarks tekniske Universitet.<br /><br />
31 Theo Andersen, <a href="mailto:sikkerhed@tre-ben.dk">sikkerhed@tre-ben.dk</a>
32 </p>
33 </td>
34 </tr>
35 </table>
36 </div>
37
38 <?php include("footer.php"); ?>
```

### A.5.5 classes.php

```
1 <?php
2
3
4 function connect()
5 {
6     mysql_connect("localhost", "tre_ben_dk", "kablooie") or die(mysql_error());
7     mysql_select_db("tre_ben_dk") or die(mysql_error());
8 }
9
10 function countgettesttext()
11 {
12     connect();
13
14     $result = mysql_query("SELECT count(*) FROM sik_test;") or die(mysql_error());
```

```

15 $row = mysql_fetch_row($result);
16 if ( $row )
17 { return $row[0]; }
18 else
19 { return false; }
20 mysql_close();
21 }
22
23 function getusername($id)
24 {
25     connect();
26
27     $result = mysql_query("SELECT username FROM sik_person WHERE id = ".$id."); or die(mysql_error());
28     $row = mysql_fetch_row($result);
29     if ( $row )
30     { return $row[0]; }
31     else
32     { return false; }
33     mysql_close();
34 }
35
36 function gettestfinished($tid)
37 {
38     connect();
39     $pid = $_SESSION['user'];
40     $result = mysql_query("SELECT data32 FROM sik_testdata where person_id = ".$pid.
41         " AND test_id = ".$tid."); or die(mysql_error());
42     $row = mysql_fetch_row($result);
43     if ( $row[0] == "OK" )
44     { return true; }
45     else
46     { return false; }
47     mysql_close();
48 }
49
50 function gettesttext($id)
51 {
52     connect();
53     $query = "SELECT * FROM sik_test WHERE id = ".$id.";";
54     $result = mysql_query($query) or die("Forespørgelsen lykkedes ikke : ". mysql_error());
55
56     while ($row = mysql_fetch_object($result))
57     {
58         $array[] = $row;
59     }
60     if($array)
61     { return $array; }
62     else
63     { return null; }
64
65     mysql_free_result($result);
66     mysql_close($link);
67 }
68
69 function adduser($username, $password, $name, $gender, $age,
70     $skill)
71 {
72     connect();
73
74     $result = mysql_query("INSERT INTO sik_person (id, username, password, name, gender, age, ".
75         "computerskill)".
76         "VALUES ('', '".$username."', '".$password."', '".$name."', ".
77         "'".$gender."', '".$age."', '".$skill.'');") or die(mysql_error());
78
79     if ( $result )
80     { return true; }

```

```
80     else
81     { return false; }
82     mysql_close();
83 }
84
85 function userauth($username, $password)
86 {
87     connect();
88     $result = mysql_query("SELECT * FROM sik_person WHERE username = '". $username.'" AND password = '".
89         $password."'"); or die(mysql_error());
90     $row = mysql_fetch_array( $result );
91     if ( $row )
92     { return true; }
93     else
94     { return false; }
95     mysql_close();
96 }
97
98 // Bruger Login
99 function loginuser($username, $password)
100 {
101     if(userauth($username, $password))
102     {
103         $_SESSION['user'] = user_id($username);
104         return true;
105     }
106     else
107     {
108         return false;
109     }
110 }
111
112
113 function user_id($username)
114 {
115     connect();
116     $result = mysql_query("SELECT id FROM sik_person WHERE username = '".
117         $username."'"); or die(mysql_error());
118     $row = mysql_fetch_row($result);
119     if ( $row )
120     { return $row[0]; }
121     else
122     { return false; }
123     mysql_close();
124 }
125
126 function userexist($username)
127 {
128     connect();
129     $result = mysql_query("SELECT * FROM sik_person WHERE username = '".
130         $username."'"); or die(mysql_error());
131     $row = mysql_fetch_array( $result );
132     if ( $row )
133     { return true; }
134     else
135     { return false; }
136     mysql_close();
137 }
138
139 // Kontroller om en bruger er logget ind
140 function check()
141 {
142     return $_SESSION['user'];
143 }
144
```

```
145
146 // Bruger Logout
147 function logout()
148 {
149     if($_SESSION['user'])
150     {
151         $_SESSION = array();
152         session_destroy();
153         return true;
154     }
155 }
156
157
158
159 ?>
```

## A.5.6 script.js

```
1
2
3 function validatelogin()
4 {
5     if((document.frmLogin.txtUsername.value.length == 0) ||
6         (document.frmLogin.txtPassword.value.length == 0))
7     {
8         alert('Et af tekstfelterne er tomme');
9         return false;
10    }
11    else
12    {
13        return true;
14    }
15 }
16
17 function validatetestslut()
18 {
19     if((document.form.radTest[0].checked == false) &&
20         (document.form.radTest[1].checked == false))
21     {
22         alert('Enten \'ja\' eller \'nej\' skal vælges for at kunne fortsætte!');
23         return false;
24     }
25     else
26     {
27         return true;
28     }
29 }
30
31
32 function IsNumeric(sText)
33 {
34     var ValidChars = "0123456789";
35     var Char;
36
37     for (i = 0; i < sText.length; i++)
38     {
39         Char = sText.charAt(i);
40         if (ValidChars.indexOf(Char) == -1)
41         {
42             return false;
43         }
44     }
45     return true;
46 }
```

```
47
48 function validateopretbruger()
49 {
50     if((document.frmLogin.txtUsername.value.length > 0) &&
51         (document.frmLogin.txtPassword.value.length > 0) &&
52         (document.frmLogin.txtPassword2.value.length > 0) &&
53         (document.frmLogin.txtName.value.length > 0) &&
54         (document.frmLogin.txtAge.value.length > 0) &&
55         ((document.frmLogin.radGender[0].checked == true) ||
56         (document.frmLogin.radGender[1].checked == true)) &&
57         ((document.frmLogin.radComputerskill[0].checked == true) ||
58         (document.frmLogin.radComputerskill[1].checked == true) ||
59         (document.frmLogin.radComputerskill[2].checked == true)) )
60     {
61         // Test af gentaget kodeord er ens
62         if(document.frmLogin.txtPassword.value != document.frmLogin.txtPassword2.value)
63             { alert("De to kodeord er ikke ens!"); return false; }
64
65         // Test Alder er tal og mellem 0 og 100
66         if((!isNaN(parseFloat(document.frmLogin.txtAge.value)) ||
67             (document.frmLogin.txtAge.value < 1) ||
68             (document.frmLogin.txtAge.value > 99)))
69             { alert("Alder skal være specificeret ved et tal mellem 0 og 100"); return false; }
70
71         return true;
72     }
73     else
74     {
75         alert('Et fornødent tekstfelt var ikke udfyldt!');
76         return false;
77     }
78 }
79
80 // Phishing Web
81 var gstrEmailTo = "";
82 var gstrEmailSubject = "";
83 var gstrEmailBody = "";
84 var myVar = 0;
85
86 function appendEmailBody(strBody)
87 {
88     var temp_str = strBody;
89     while(temp_str.indexOf("|") > -1)
90     {
91         temp_str = temp_str.substr(0,temp_str.indexOf("|")) + "\r" +
92             temp_str.substr(temp_str.indexOf("|") + 1)
93     }
94     gstrEmailBody += temp_str + "\r";
95 }
96
97 function sendMail()
98 {
99     //alert(gstrEmailBody);
100
101     document.emailForm.elements["Results"].value = "\r" + gstrEmailBody;
102     document.emailForm.submit();
103 }
104
105 function padMail(strAddress, strSubject, strBody)
106 {
107     gstrEmailTo = strAddress;
108     gstrEmailSubject = strSubject;
109     var temp_str = strBody;
110     while(temp_str.indexOf("|") > -1)
111     {
```

```
112     temp_str = temp_str.substr(0,temp_str.indexOf("|")) + "\r" +
113     temp_str.substr(temp_str.indexOf("|") + 1)
114 }
115 gstrEmailBody = temp_str + "\r";
116 }
117
118 function writeDocument(s){document.write(s);}
119 // End -->
120
121 function vurderstoreogsmaa(form)
122 {
123     var reges =/[a-z]/; //[^A-Za-z0-9]
124     var regeb =/[A-Z]/; //[^A-Za-z0-9]
125     results= reges.test(form);
126     resultb= regeb.test(form);
127     if((results == true) && (resultb == true))
128     {
129         return true;
130     }
131     else
132     {
133         return false;
134     }
135 }
136
137 function vurderpecial(form)
138 {
139     // se om en af tegnene er ikke A-Z eller a-z
140     var reges =/[^A-Za-z]/; //[^A-Za-z0-9]
141     result= reges.test(form);
142     return result;
143 }
144
145
146
147 function vurderkodeord()
148 {
149     document.form.bigsmall.value = "";
150     document.form.specialnumber.value = "";
151     document.form.points.value = "";
152
153     var kodeord = document.form.pass.value;
154     var point = 0;
155
156     if(kodeord.length > 0)
157     {
158         // Længde af kodeordet..
159         var temp = ( kodeord.length * 15 ) - 50
160         if( temp > 0 )
161         {
162             if( temp < 70 )
163             { point = temp; }
164             else
165             { point = 70; }
166         }
167
168         // Er der både små og store bogstaver?..
169
170         if(vurderstoreogsmaa(kodeord))
171         {
172             point = point + 15;
173             document.form.bigsmall.value = "true";
174         }
175         else
176         {
```



```
177     document.form.bigsmall.value = "false";
178 }
179
180 if(vurderspecial(kodeord))
181 {
182     point = point + 15;
183     document.form.specialnumber.value = "true";
184 }
185 else
186 {
187     document.form.specialnumber.value = "false";
188 }
189
190 if(point < 0) point = 0;
191 if(point > 100) point = 100;
192
193 document.form.points.value = point;
194 document.form.length.value = kodeord.length;
195
196 //alert(point +
197 //     " points, "+
198 //     vurderespecial(kodeord) +
199 //     " specialtal, " +
200 //     vurderstoreogsmå(kodeord)+" storesmå.");
201
202     return true;
203 }
204 else
205 {
206     alert("Du skal skrive et kodeord i feltet");
207 }
208
209 }
```

### A.5.7 style.css

```
1 body {
2     margin:50px 0px; padding:0px;
3     text-align:center;
4     font-family: Verdana;
5     font-size: 11px;
6 }
7
8 #content {
9     margin:0px auto;
10    text-align:left;
11 }
12
13 #menu {
14     margin: 0px; padding: 0px;
15     width: 770px; height:15px;
16     border-bottom: 1px solid #cccccc;
17 }
18
19 #menu ul {
20     list-style: none;
21     margin: 0px; padding: 0px;
22     border: 1px;
23 }
24
25 #menu ul li {
26     float: left;
27     margin-left: 0px;
28     padding-left: 0px; padding-bottom: 2px;
```

```
29 }
30 #menu #menur {
31   float:right;
32   padding-bottom: 2px;
33 }
34
35
36 #testlist {
37   padding-left: 20px; margin-left: 4px;
38   border-left: 1px solid black;
39
40 }
41 #testlist li {
42   padding-bottom: 4px; margin-left: 4px;
43 }
44
45 a {
46   text-decoration: none;
47   color: black;
48 }
49 a:hover {
50   text-decoration: underline;
51   color: black;
52 }
53 a:visited {
54   color: black;
55 }
56
57 #txi {
58   font-family: Verdana;
59   font-size:11px;
60   font-weight: bold;
61   border: 1px solid white;
62   text-align: left;
63 }
64
65 #tx {
66   border: 1px solid white;
67   font-size: 11px;
68   font-family: verdana;
69   overflow: hidden;
70   text-align: left;
71 }
72
73 .textfield {
74   margin-left: 1px;
75 }
76 .button {
77   margin: 0px; padding: 0px;
78   outline-style: none;
79 }
80
81 table td h1 {
82   font-family: Verdana;
83   font-size: 15pt;
84   color: #003466;
85 }
86
87 table td h3 {
88   font-family: Verdana;
89   font-weight: normal;
90   padding-top: 15px; padding-left: 10px;
91 }
92
93 .svar {
```

```

94 background-color: white;
95     }
96
97 b {
98     font-size: 12px;
99     }

```

## A.5.8 start.php

```

1 <?php $restricted = true; $start = true; include("header.php"); require_once("classes.php");?>
2
3 <div style="width: 80%; text-align: left; padding: 13px;">
4 Velkommen, <?php echo getusername($_SESSION['user']); ?><br />
5
6 <p style="width: 650px;" onmouseover="settext('0')">
7 Fra denne side kan du vælge og prøve forskellige animerede spørgeskemaer, designet til at
8 teste din viden om forskellige aspekter af computersikkerhed. I slutningen af næsten alle
9 spørgeskemaerne vil emnet blive forklaret som en gennemgang af spørgsmålene og dine svar sådan
10 at du lærer hvordan du kan sikre dig på din computer.
11 </p><br /><br />
12 <table cellpadding="0" cellspacing="0" >
13 <tr>
14     <td width="60%" valign="top" onmouseover="settext('0')">
15         <form name="form" action="">
16             <input id="txi" type="text" value="" readonly ="readonly" /><br />
17             <textarea id="tx" rows="10" cols="55"  readonly ="readonly"></textarea>
18         </form>
19     </td>
20     <td width="40%" valign="top" align="left" style="padding-left:85px;" >
21         <b style="font-size: 11px;">Test dig selv:</b><br />
22         <ul id="testlist">
23             <?php
24                 for($i=2; $i <= countgettesttext(); $i++)
25                 {
26                     $result = gettesttext($i);
27                     foreach($result as $user_object)
28                     {
29                         echo "<li><a class=\"info\" href=\"\".$user_object->url.\"\">\n";
30                         echo "onmouseover=\"settext('\".($i).\"')\" >\n".
31                             $user_object->name."</a>";
32                         if(gettestfinished(($i)))
33                         {
34                             echo " <img src=\"images/check.jpg\" alt=\"\" ".
35                                 "title=\"Du har allerede kørt denne test.\" />\n";
36                         }
37                         echo "</li>";
38                     }
39                 }
40             ?>
41
42
43
44         </ul>
45     </td>
46 </tr>
47 </table>
48 </div>
49
50 <?php include("footer.php"); ?>

```

## A.5.9 logind.php

```

1 <?php $page = "i"; $logind = "1"; include("header.php"); ?>
2
3 <div style="width: 80%; text-align: left; padding: 13px;">
4
5
6 <form name="frmLogin" action="logindh.php" method="post" onsubmit="return validatelogin();">
7 <table border="0" align="center">
8 <tr>
9 <td></td>
10 <td colspan="2"><b>Log ind</b></td>
11 </tr>
12 <tr>
13 <td align="right">Login:</td>
14 <td colspan="2"><input class="text" name="txtUsername" type="text" size="33"
15 <td colspan="2" style="padding-left: 100px;" value="" maxlength="34" /></td>
16 </tr>
17 <tr>
18 <td align="right">Password:</td>
19 <td colspan="2"><input class="text" name="txtPassword" type="password" size="33"
20 <td colspan="2" style="padding-left: 100px;" value="" maxlength="345" /></td>
21 </tr>
22 <tr>
23 <td></td>
24 <td><input type="submit" class="button" value="Log ind" style="width:112px;"/></td>
25 <td><input type="reset" class="button" value="Ryd" style="width:111px;"/></td>
26 </tr>
27 </table>
28 </form>
29 <br /><br /><br />
30 </div>
31 <?php include("footer.php"); ?>

```

## A.5.10 logindh.php

```

1 <?php $page = "i"; include("header.php"); ?>
2 <div style="width: 50%; text-align: left; padding: 13px;">
3
4 <?php
5 require_once 'classes.php';
6
7 if($_POST['txtUsername'])
8 {
9     if(loginuser($_POST['txtUsername'], MD5($_POST['txtPassword'])))
10     {
11         echo "<b>Velkommen, ".check()."</h2>";
12         echo "Du bliver nu omdirigeret til hovedsiden";
13     }
14     <script type="text/javascript"> window.location="start.php"</script><?php
15     }
16     else
17     {
18         echo "<b>Brugernavn eller kodeord forkert, <a href=\"logind.php\">tilbage!</a></b>";
19     }
20 }
21 ?>
22
23 <br /><br /><br /><br /><br /><br />
24 </div>
25 <?php include("footer.php"); ?>

```

### A.5.11 logud.php

```

1 <?php include("header.php"); ?>
2
3 <div style="width: 80%; text-align: left; padding: 13px;">
4
5 <?php
6     require_once 'classes.php';
7     logout();
8 ?>
9 <script type="text/javascript"> window.location="http://sikkerhed.tre-ben.dk"</script>
10
11 <?php include("footer.php"); ?>

```

### A.5.12 opretbruger.php

```

1 <?php $page = "i"; $logind = "1"; include("header.php"); ?>
2
3 <div style="width: 80%; text-align: left; padding: 13px;">
4
5     <form name="frmLogin" action="opretbrugerh.php" method="post"
6         onsubmit="return validateopretbruger();">
7     <table border="0" align="center">
8     <tr>
9         <td></td>
10        <td colspan="2"><b>Opret bruger</b></td>
11    </tr>
12    <tr>
13        <td align="right">Brugernavn:</td>
14        <td colspan="2">
15            <input class="text" name="txtUsername" type="text" size="33" maxlength="34" />
16        </td>
17    </tr>
18    <tr>
19        <td align="right">Kodeord:</td>
20        <td colspan="2">
21            <input class="text" name="txtPassword" type="password" size="33" maxlength="345" />
22        </td>
23    </tr>
24    <tr>
25        <td align="right">Gentag kodeord:</td>
26        <td colspan="2">
27            <input class="text" name="txtPassword2" type="password" size="33" maxlength="345" />
28        </td>
29    </tr>
30    <tr style="height:10px;"><td></td></tr>
31    <tr>
32        <td align="right">Fuldt navn:</td>
33        <td colspan="2">
34            <input class="text" name="txtName" type="text" size="33" maxlength="70" />
35        </td>
36    </tr>
37    <tr>
38        <td align="right">Alder:</td>
39        <td colspan="2">
40            <input class="text" name="txtAge" type="text" size="33" maxlength="2" />
41        </td>
42    </tr>
43    <tr style="height:3px;"><td></td></tr>
44    <tr style="margin-bottom:4px;">
45        <td align="right">Køn:</td>
46        <td colspan="2">
47            <label><input name="radGender" type="radio" value="M" />Mand</label>
48            <label><input name="radGender" type="radio"

```

```

49         style="margin-left: 4px;" value="F" />Kvinde</label>
50     </td>
51 </tr>
52 <tr style="height:3px;"><td></td></tr>
53 <tr>
54     <td align="right">Computerfærdighed:</td>
55     <td colspan="2">
56         <label><input name="radComputerskill" type="radio" value="0" />Uerfaren</label>
57         <label><input name="radComputerskill" type="radio" value="1" />Erfaren</label>
58         <label><input name="radComputerskill" type="radio" value="2" />Ekspert</label>
59     </td>
60 </tr>
61 <tr style="height:10px;"><td></td></tr>
62 <tr>
63     <td></td>
64     <td><input type="submit" class="button" value="Send" style="width:112px;" /></td>
65 </tr>
66 </table>
67 </form>
68 </div>
69 <br /><br /><br />
70 </div>
71 <?php include("footer.php"); ?>

```

### A.5.13 opretbrugerh.php

```

1 <?php $page = "i"; include("header.php"); ?>
2 <div style="width: 50%; text-align: left; padding: 13px;">
3
4 <?php
5     require_once 'classes.php';
6
7     if(userexist($_POST['txtUsername']))
8     {
9         echo "<h2>Brugernavnet: '".stripslashes($_POST['txtUsername'])."' er allerede i brug.</h2>";
10        echo "<a href=\"javascript:history.go(-1)\">>".
11            "Klik her for at komme tilbage og vælge et andet.</a>";
12    }
13    else
14    {
15        $result = adduser($_POST['txtUsername'], MD5($_POST['txtPassword'], $_POST['txtName'],
16            $_POST['radGender'], $_POST['txtAge'], $_POST['radComputerskill']));
17        if($result)
18        {
19            $id = user_id($_POST['txtUsername']);
20            echo "<form name=\"form\" action=\"loginh.php\" method=\"post\">";
21            echo "<h2>Tak, ".stripslashes($_POST['txtUsername'])." blev oprettet!</h2>";
22            echo "<a href=\"javascript:document.form.submit();\">Videre!</a>";
23
24            echo "<input name=\"txtUsername\" type=\"hidden\" value=\"".
25                stripslashes($_POST['txtUsername'])."\">";
26            echo "<input name=\"txtPassword\" type=\"hidden\" value=\"".$_POST['txtPassword']."\">";
27            echo "</form>";
28        }
29        else
30        {
31            echo "<h2>Fejl! Det er sket en fejl.";
32            echo "<h2>Kontakt <a href=\"mailto:theo@paradis.dk\">Webmasteren</a>".
33                " om denne fejl og prøv lidt senere igen!";
34        }
35    }
36 ?>
37
38 <br /><br /><br /><br /><br /><br />

```

```
39 </div>
40 <?php include("footer.php"); ?>
```

### A.5.14 ingenadgang.php

```
1 <?php $page = "i"; include("header.php"); ?>
2
3 <div style="width: 50%; text-align: center; padding: 13px;">
4   <b>Ingen adgang!</b><br /><br /><br />
5
6   Du er ikke logget ind på denne side. <br /><br />
7   <a href="http://sikkerhed.tre-ben.dk/">Klik her for at gå tilbage</a>
8 </div>
9 <br /><br /><br /><br /><br /><br /><br /><br />
10 <?php include("footer.php"); ?>
```

### A.5.15 testslut.php

```
1 <?php $restricted = true; $page = "i"; include("header.php"); ?>
2
3 <?php
4   // Skal komme fra en test
5
6   if(($_GET['id'] == "") && ($_POST['id'] == ""))
7   {
8     echo "<script type=\"text/javascript\"> window.location="
9         "\"http://sikkerhed.tre-ben.dk/start.php\"</script>";
10  }
11
12  if($_POST['id'])
13  {
14
15    mysql_connect("localhost", "tre_ben_dk", "kablooie") or die(mysql_error());
16    mysql_select_db("tre_ben_dk") or die(mysql_error());
17    $result = mysql_query("UPDATE sik_testdata SET data31 = '".$_POST['radTest']
18                          "' WHERE id = '".$_POST['id']."'"); or die(mysql_error());
19    mysql_close();
20
21
22    echo "<script type=\"text/javascript\"> window.location="
23        "\"http://sikkerhed.tre-ben.dk/start.php\"</script>";
24
25  }
26 ?>
27
28 <div style="width: 80%; text-align: center; padding: 13px;">
29   <form name="form" action="testslut.php" method="post" onsubmit="return validatetestslut()">
30     <b>Fik du noget ud af denne test?</b><br /><br />
31     <label><input name="radTest" type="radio" value="JA" />Ja</label>
32     <label><input name="radTest" type="radio" value="NEJ" style="margin-left: 70px;" />Nej</label>
33     <br /><br />
34     <input name="id" type="hidden" value="<?php if($_GET['id'] != "") { echo $_GET['id']; }?>" />
35     <input type="submit" value="Videre" />
36   </form>
37 </div>
38 <br /><br /><br /><br />
39 <?php include("footer.php"); ?>
```

## A.5.16 Phishing internet.php

```

1 <?php $restricted = true; $notstart = "1"; include("../header.php"); ?>
2 <br />
3 <form name="emailForm" action="inethandler.php" method="POST">
4 <input name="Results" type="hidden">
5 </form>
6 <div style="padding: 0px; width:790px; border: 1px solid #4286b0;">
7 <script type="text/javascript">
8 <!--
9 writeDocument(' <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000" ' +
10 "codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0"'+
11 " width="790" height="545" ID="Captivate1">' +
12 ' <param name="movie" value="phishingweb.swf">' +
13 ' <param name="quality" value="high">' +
14 ' <param name="menu" value="false">' +
15 ' <param name="loop" value="0">' +
16 ' <embed src="phishingweb.swf" width="790" height="545" loop="0" quality="high" ' +
17 "pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash" ' +
18 ' menu="false"></embed>' +
19 '</object>');
20 //-->
21 </script>
22 <noscript>
23 <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000"'+
24 ' codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0"'+
25 ' width="790" height="545" ID="Captivate1">
26 <param name="movie" value="phishingweb.swf">
27 <param name="quality" value="high">
28 <param name="menu" value="false">
29 <param name="loop" value="0">
30 <embed src="phishingweb.swf" width="790" height="545" loop="0" quality="high" ' +
31 ' pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash" ' +
32 ' menu="false"></embed>
33 </object>
34 </noscript>
35 </div>
36 <?php include("../footer.php"); ?>

```

## A.5.17 Phishing inethandler.php

```

1 <?php $restricted = true; $notstart = "1"; include("../header.php");require_once '../classes.php';?>
2
3 <div style="width: 790px; margin:0px; padding:0px;" >
4 <?php
5     function striptoststrings( $result )
6     {
7         // strip the start info
8
9         $result = strstr(stripslashes($result), "\",\"Student Response\", \"Result\", \"
10         \"Weight\", \"Latency\"");
11         $num = strpos($result, "\n");
12         $result = substr($result, $num+1);
13
14         // now $result is only the five lines of results
15         // strip these five lines into an array
16
17         for($i = 0; $i < 4; $i++)
18         {
19             $num = strpos($result, "\n");
20             $arr[$i] = substr($result, 0, $num);
21             $result = substr($result, $num+1);
22         }
23         $arr[4] = $result;

```



```

24     return $arr;
25 }
26
27 function stripastring( $string, $print )
28 {
29     // is in 10 parts seperated by "\", we need parts 3 and 7
30     $string;
31     $num = strpos($string, "\\"); // 1
32     $string = substr($string, $num+1);
33     $num = strpos($string, "\\"); // 2
34     $string = substr($string, $num+1);
35     $num = strpos($string, "\\"); // 3 (we need this one)
36     $arr[0] = substr($string, 2, $num-2);
37     $string = substr($string, $num+1);
38     $num = strpos($string, "\\"); // 4
39     $string = substr($string, $num+1);
40     $num = strpos($string, "\\"); // 5
41     $string = substr($string, $num+1);
42     $num = strpos($string, "\\"); // 6
43     $string = substr($string, $num+1);
44     $num = strpos($string, "\\"); // 7 (last one we need)
45     $arr[1] = substr($string, 2, $num-2);
46     $string = substr($string, $num+1);
47
48     if($print == 1)
49     {
50         if($arr[1] != 1)
51             echo $arr[0].".".$arr[1]."<br />";
52         else
53             echo $arr[0]."<br />";
54     }
55     if($arr[1] != 1)
56         return $arr[0].".".$arr[1];
57     else
58         return $arr[0];
59 }
60
61 }
62
63 $result = nl2br($_POST['Results']).": til";
64 //echo $result;
65 //echo "<br />Her: <br />";
66
67 $a = striptostings( $result );
68
69 stripastring( $a[0], 0 );
70 stripastring( $a[1], 0 );
71 stripastring( $a[2], 0 );
72 stripastring( $a[3], 0 );
73 stripastring( $a[4], 0 );
74
75 // Læg alt op i databasen
76 if(gettestfinished(2) == false)
77 {
78     mysql_connect("localhost", "tre_ben_dk", "kablooiie") or die(mysql_error());
79     mysql_select_db("tre_ben_dk") or die(mysql_error());
80     $result = mysql_query("INSERT INTO sik_testdata (id, person_id, test_id, data1, data2, data3, "+
81 "data4, data5, data32) VALUES (".
82         "'', ".
83         ",'" .$_SESSION['user'] ."' " .
84         "','2' ".
85         ",'" .stripastring( $a[0], 0 ) ."' " .
86         ",'" .stripastring( $a[1], 0 ) ."' " .
87         ",'" .stripastring( $a[2], 0 ) ."' " .
88         ",'" .stripastring( $a[3], 0 ) ."' " .

```

```

89         ", '".stripastring( $a[4], 0 )."' ".
90         ", 'OK' ".
91         ");") or die(mysql_error());
92
93     $id = mysql_insert_id();
94
95     mysql_close();
96 }
97 ?>
98 <table id="content" width="790" border="0" cellpadding="0" cellspacing="0" style="margin:0px;" >
99 <tr>
100 <td align="center"><h1>Svar fra test</h1></td>
101 </tr>
102 <tr>
103 <td align="right"><a href=" ../testslut.php?id=<?php echo $id; ?>">Tilbage til startsiden</a></td>
104 </tr>
105 <tr>
106 <td>
107 <!-- Start Svarblok ----->
108 <table class="svar" width="790" border="0" cellpadding="0"
109     cellspacing="0" style="border: 1px solid #4286b0;">
110 <tr style="height: 49px; background-color: #70bde4;">
111 <td colspan="2"><h3>Spørgsmål 1 ud af 5 - Danske netbank</h3></td>
112 </tr>
113 <tr>
114 <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
115 <div style="font-weight: bold; font-color: green;">
116 <?php
117     $r = substr(stripastring( $a[0], 0 ), 2, 1);
118     $r2 = substr(stripastring( $a[0], 0 ), 3);
119     if( ($r == "N") &&
120         (substr_count($r2, "A") > 0) &&
121         (substr_count($r2, "B") == 0) &&
122         (substr_count($r2, "C") == 0) &&
123         (substr_count($r2, "E") == 0))
124     { echo "Korrekt!"; }
125     else
126     { echo "Forkert!"; }
127
128     ?>
129 </div>
130 <br />
131 <?php
132     echo "De svarede: ";
133     if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
134     if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
135     if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
136 ?>
137 <div style="font-size: 8pt; padding-left: 10px;">
138 <?php
139     if(substr_count($r2, "A") > 0) {
140         echo "- <font style=\"color:green\">Adressen var forkert</font><br />"; }
141     if(substr_count($r2, "B") > 0) {
142         echo "- <font style=\"text-decoration: underline; color:red\">".
143             "Der manglede 'https' i adressen</font><br />"; }
144     if(substr_count($r2, "C") > 0) {
145         echo "- <font style=\"text-decoration: underline; color:red\">".
146             "Der manglede en lås</font><br />"; }
147     if(substr_count($r2, "D") > 0) {
148         echo "- <font style=\"color:green\">Siden virkede forkert</font><br />"; }
149     if(substr_count($r2, "E") > 0) {
150         echo "- <font style=\"text-decoration: underline; color:red\">Ved ikke</font><br />"; }
151     ?>
152 </div>
153 </td>

```



```

219     echo "- <font style=\"color:green;\">Siden virkede forkert</font><br />"; }
220     if(substr_count($r2, "E") > 0) {
221         echo "- <font style=\"text-decoration: underline; color:red;\">Ved ikke</font><br />"; }
222     }?>
223 </div>
224 </td>
225 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
226     
227 </td>
228 </tr>
229 <tr>
230 <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px;
231 padding-bottom: 10px; font-size: 13px;">
232 <br />
233 Denne sides adresse er rigtig nok men låsen mangler.<br /><br />
234 Den lukkede lås sammen med 'https' til forskel for det normale 'http' istarten
235 af adressen, viser at websiden har certifikat og kan sende data sikkert mellem
236 dig og siden. <br />
237 Da låsen ikke er tilstede frarådes det at stole på denne side og indtaste nogen
238 form for brugerinformation på den.<br />
239 <br />
240 Bemærk: På nogle browsere vises låseikonet ikke oppe vedsiden af adressen,
241 men nede på baren under websiden.
242 </td>
243 </tr>
244 </table>
245 <!-- Slut Svarblok ----->
246 </td>
247 </tr>
248 <tr style="padding-top: 50px;">
249 <td align="right"><a href=" ../testslut.php?id=<?php echo $id; ?>">Tilbage til startside</a></td>
250 </tr>
251 <tr>
252 <td>
253 <!-- Start Svarblok ----->
254 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0"
255 style="border: 1px solid #4286b0;">
256 <tr style="height: 49px; background-color: #70bded;">
257 <td colspan="2"><h3>Spørgsmål 3 ud af 5 - Paypal</h3></td>
258 </tr>
259 <tr>
260 <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
261 <div style="font-weight: bold; font-color: green;">
262 <?php
263 $r = substr(stripastring( $a[2], 0 ), 2, 1);
264 $r2 = substr(stripastring( $a[2], 0 ), 3);
265 if( ($r == "Y"))
266 { echo "Korrekt!"; }
267 else
268 { echo "Forkert!"; }
269
270 }?>
271 </div>
272 <br />
273 <?php
274 echo "De svarede: ";
275 if($r == "Y") { echo "<font style=\"color:green\">Ja</font><br />"; }
276 if($r == "N") { echo "<font style=\"color:red\">Nej</font><br />"; }
277 if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
278 }?>
279 <div style="font-size: 8pt; padding-left: 10px;">
280 <?php
281 if(substr_count($r2, "A") > 0) {
282 echo "- <font style=\"text-decoration: underline; color:red;\">Adressen var forkert</font><br />"; }
283 if(substr_count($r2, "B") > 0) {

```

```

284 echo "- <font style=\"text-decoration: underline; color:red;\">".
285 "Der manglede 'https' i adressen</font><br />"; }
286     if(substr_count($r2, "C") > 0) {
287 echo "- <font style=\"text-decoration: underline; color:red;\">Der manglede en lås</font><br />"; }
288     if(substr_count($r2, "D") > 0) {
289 echo "- <font style=\"text-decoration: underline; color:red;\">Siden virkede forkert</font><br />"; }
290     if(substr_count($r2, "E") > 0) {
291 echo "- <font style=\"text-decoration: underline; color:red;\">Ved ikke</font><br />"; }
292     ?>
293 </div>
294 </td>
295 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
296 
297 </td>
298 </tr>
299 <tr>
300 <td align="left" colspan="2" style="padding-left: 15px; padding-right: 100px;
301 padding-bottom: 10px; font-size: 13px;">
302 <br />
303 Denne side er ikke phished.<br /><br />
304 Siden har 'https' og en lås sådan at vi ved der kan sendes data
305 sikkert mellem os og siden.<br />
306 Derudover kan vi se at adressen passer.<br />
307 Da disse tre ting er opfyldt er der ingen grund til ikke at stole på siden.
308 </td>
309 </tr>
310 </table>
311 <!-- Slut Svarblok ----->
312 </td>
313 </tr>
314 <tr style="padding-top: 50px;">
315 <td align="right"><a href=" ../testslut.php?id=<?php echo $id; ?>">Tilbage til startside</a></td>
316 </tr>
317 <tr>
318 <td>
319 <!-- Start Svarblok ----->
320 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0"
321 style="border: 1px solid #4286b0;">
322 <tr style="height: 49px; background-color: #70bde4;">
323 <td colspan="2"><h3>Spørgsmål 4 ud af 5 - Apple store</h3></td>
324 </tr>
325 <tr>
326 <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
327 <div style="font-weight: bold; font-color: green;">
328 <?php
329 $r = substr(stripastring( $a[3], 0 ), 2, 1);
330 $r2 = substr(stripastring( $a[3], 0 ), 3);
331 if( ($r == "N") &&
332 (substr_count($r2, "A") == 0) &&
333 ((substr_count($r2, "B") > 0) || (substr_count($r2, "C") > 0)) &&
334 (substr_count($r2, "E") == 0))
335 { echo "Korrekt!"; }
336 else
337 { echo "Forkert!"; }
338 }
339 ?>
340 </div>
341 <br />
342 <?php
343 echo "De svarede: ";
344 if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
345 if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
346 if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
347 }
348 ?>

```

```

349 <div style="font-size: 8pt; padding-left: 10px;">
350 <?php
351     if(substr_count($r2, "A") > 0) {
352 echo "- <font style=\"text-decoration: underline; color:red;\">Adressen var forkert</font><br />"; }
353     if(substr_count($r2, "B") > 0) {
354         echo "- <font style=\"color:green;\">Der manglede 'https' i adressen</font><br />"; }
355     if(substr_count($r2, "C") > 0) {
356         echo "- <font style=\"color:green;\">Der manglede en lås</font><br />"; }
357     if(substr_count($r2, "D") > 0) {
358         echo "- <font style=\"color:green;\">Siden virkede forkert</font><br />"; }
359     if(substr_count($r2, "E") > 0) {
360         echo "- <font style=\"text-decoration: underline; color:red;\">Ved ikke</font><br />"; }
361     ?>
362 </div>
363 </td>
364 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
365 
366 </td>
367 </tr>
368 <tr>
369 <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px;
370 padding-bottom: 10px; font-size: 13px;">
371 <br />
372 Denne sides adresse er rigtig nok, men låsen mangler sammen med 'https'.<br /><br />
373 Låse eller andre ikoner på selve websiden, holder ingen garantier for at
374 siden virkelig er sikker.<br />
375 Man bør kun stole på browserens egne indikatorer.
376 <br />
377 Da låsen og 'https' ikke er tilstede, frarådes det at stole på
378 denne side og indtaste nogen form for brugerinformation på den.<br />
379 <br />
380 </td>
381 </tr>
382 </table>
383 <!-- Slut Svarblok ----->
384 </td>
385 </tr>
386 <tr style="padding-top: 50px;">
387 <td align="right"><a href="..">Tilbage til startside</a></td>
388 </tr>
389 <tr>
390 <td>
391 <!-- Start Svarblok ----->
392 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0"
393     style="border: 1px solid #4286b0;">
394 <tr style="height: 49px; background-color: #70bde4;">
395 <td colspan="2"><h3>Spørgsmål 5 ud af 5 - Jyske bank</h3></td>
396 </tr>
397 <tr>
398 <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
399 <div style="font-weight: bold; font-color: green;">
400 <?php
401     $r = substr(stripastring( $a[4], 0 ), 2, 1);
402     $r2 = substr(stripastring( $a[4], 0 ), 3);
403     if( ($r == "N") &&
404         (substr_count($r2, "A") > 0) &&
405         (substr_count($r2, "B") == 0) &&
406         (substr_count($r2, "C") == 0) &&
407         (substr_count($r2, "E") == 0))
408     { echo "Korrekt!"; }
409     else
410     { echo "Forkert!"; }
411
412     ?>
413 </div>

```

```

414 <br />
415 <?php
416     echo "De svarede: ";
417     if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
418     if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
419     if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
420     ?>
421     <div style="font-size: 8pt; padding-left: 10px;">
422         <?php
423             if(substr_count($r2, "A") > 0) {
424                 echo "- <font style=\"color:green\">Adressen var forkert</font><br />"; }
425             if(substr_count($r2, "B") > 0) {
426 echo "- <font style=\"text-decoration: underline; color:red\">".
427 "Der manglede 'https' i adressen</font><br />"; }
428             if(substr_count($r2, "C") > 0) {
429 echo "- <font style=\"text-decoration: underline; color:red\">Der manglede en lås</font><br />"; }
430             if(substr_count($r2, "D") > 0) {
431                 echo "- <font style=\"color:green\">Siden virkede forkert</font><br />"; }
432             if(substr_count($r2, "E") > 0) {
433                 echo "- <font style=\"text-decoration: underline; color:red\">Ved ikke</font><br />"; }
434             ?>
435         </div>
436     </td>
437     <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
438         
439     </td>
440 </tr>
441 <tr>
442     <td align="left" colspan="2" style="padding-left: 15px; padding-right: 100px;
443         padding-bottom: 10px; font-size: 13px;">
444         <br />
445         Adressen på denne side er forkert.
446         Bemærk at det sidste 'k' er udskiftet til et 'c'.<br /><br />
447         Så små ændringer som dette kan gøre at siden man ser,
448         rent faktisk er en helt anden. Derfor bør man
449         altid kigge efter når man følger et link (især fra en email)
450         om den peger det rigtige sted hen.<br />
451         <br />
452         Da vi ikke kan være sikre på at denne side virkelig er jyske banks
453         kan det ikke tilrådes at stole på den eller indtaste nogen
454         form for følsomme oplysninger på den.
455     </td>
456 </tr>
457 </table>
458 <!-- Slut Svarblok ----->
459 </td>
460 </tr>
461 </table>
462 </div>
463 <?php include("../footer.php"); ?>

```

### A.5.18 Phishing email.php

```

1 <?php $restricted = true; $notstart = "1"; include("../header.php"); ?>
2 <br />
3
4 <form name="emailForm" action="mailhandler.php" method="POST">
5 <input name="Results" type="hidden">
6 </form>
7 </head>
8 <div style="padding: 0px; width: 790px; border: 1px solid #4286b0;">
9 <script type="text/javascript">
10 <!--
11 writeDocument(' <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000" ' +

```

```

12 'codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0" '+
13 'width="790" height="545" ID="Captivatel">'+
14   '<param name="movie" value="phishingmail.swf">'+
15   '<param name="quality" value="high">'+
16   '<param name="menu" value="false">'+
17   '<param name="loop" value="0">'+
18   '<embed src="phishingmail.swf" width="790" height="545" loop="0" quality="high" '+
19   '<pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash" '+
20   '<menu="false"></embed>'+
21 '</object>');
22 //-->
23 </script>
24 <noscript>
25   <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000" '+
26   'codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0" '+
27   'width="790" height="545" ID="Captivatel">
28     <param name="movie" value="phishingmail.swf">
29     <param name="quality" value="high">
30     <param name="menu" value="false">
31     <param name="loop" value="0">
32     <embed src="phishingmail.swf" width="790" height="545" loop="0" quality="high" '+
33     'pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash" '+
34     'menu="false"></embed>
35   </object>
36 </noscript>
37 </div>
38 <?php include("../footer.php"); ?>

```

## A.5.19 Phishing mailhandler.php

```

1 <?php $restricted = true; $notstart = "1"; include("../header.php");require_once '../classes.php';?>
2
3 <div style="width: 790px; margin:0px; padding:0px;" >
4 <?php
5   function striptostrings( $result )
6   {
7     // strip the start info
8     $result = strstr(stripslashes($result), "\",\"Student Response\", \"Result\", \"Weight\"".
9     "\",\"Latency\"");
10    $num = strpos($result, "\n");
11    $result = substr($result, $num+1);
12
13    // now $result is only the five lines of results
14    // strip these five lines into an array
15
16    for($i = 0; $i < 4; $i++)
17    {
18      $num = strpos($result, "\n");
19      $arr[$i] = substr($result, 0, $num);
20      $result = substr($result, $num+1);
21    }
22    $arr[4] = $result;
23
24    return $arr;
25  }
26
27 function stripastring( $string, $print )
28 {
29   // is in 10 parts separated by "," we need parts 3 and 7
30
31   $num = strpos($string, "\",\""); // 1
32   $string = substr($string, $num+1);
33   $num = strpos($string, "\",\""); // 2
34   $string = substr($string, $num+1);

```



```

35     $num = strpos($string, "\",\""); // 3 (we need this one)
36     $arr[0] = substr($string, 2, $num-2);
37     $string = substr($string, $num+1);
38     $num = strpos($string, "\",\""); // 4
39     $string = substr($string, $num+1);
40     $num = strpos($string, "\",\""); // 5
41     $string = substr($string, $num+1);
42     $num = strpos($string, "\",\""); // 6
43     $string = substr($string, $num+1);
44     $num = strpos($string, "\",\""); // 7 (last one we need)
45     $arr[1] = substr($string, 2, $num-2);
46     $string = substr($string, $num+1);
47
48     if($print == 1)
49     {
50     if($arr[1] != 1)
51         echo $arr[0].";".$arr[1]."<br />";
52     else
53         echo $arr[0]."<br />";
54     }
55     if($arr[1] != 1)
56         return $arr[0].";".$arr[1];
57     else
58         return $arr[0];
59
60 }
61
62 $result = nl2br($_POST['Results']);
63 //echo $result;
64 //echo "<br />Her: <br />";
65
66 $a = striptostings( $result );
67 stripastring( $a[0], 0 );
68 stripastring( $a[1], 0 );
69 stripastring( $a[2], 0 );
70 stripastring( $a[3], 0 );
71 stripastring( $a[4], 0 );
72
73 if(gettestfinished(3) == false)
74 {
75     mysql_connect("localhost", "tre_ben_dk", "kabloolie") or die(mysql_error());
76     mysql_select_db("tre_ben_dk") or die(mysql_error());
77     $result = mysql_query("INSERT INTO sik_testdata (id, person_id, test_id ,data1, data2, data3, ".
78     "data4, data5, data32) VALUES (".
79     "'' ' ".
80     "',' ".$_SESSION['user'].' ' ".
81     "','3' ".
82     "','.stripastring( $a[0], 0 ).'' ' ".
83     "','.stripastring( $a[1], 0 ).'' ' ".
84     "','.stripastring( $a[2], 0 ).'' ' ".
85     "','.stripastring( $a[3], 0 ).'' ' ".
86     "','.stripastring( $a[4], 0 ).'' ' ".
87     "','OK' ".
88     ");") or die(mysql_error());
89
90     $id = mysql_insert_id();
91     mysql_close();
92 }
93 ?>
94 <table id="content" width="790" border="0" cellpadding="0" cellspacing="0">
95 <tr>
96 <td align="center"><h1>Svar fra test</h1></td>
97 </tr>
98 <tr>
99 <td align="right"><a href=" ../testslut.php?id=<?php echo $id; ?>">Tilbage til startsiden</a></td>

```

```
100 </tr>
101 <tr>
102 <td>
103 <!-- Start Svarblok ----->
104 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0"
105 style="border: 1px solid #4286b0;">
106 <tr style="height: 49px; background-color: #70bded;">
107 <td colspan="2"><h3>Spørgsmål 1 ud af 5 - My Travel</h3></td>
108 </tr>
109 <tr>
110 <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
111 <div style="font-weight: bold; font-color: green;">
112 <?php
113 $r = substr(stripastring( $a[0], 0 ), 2, 1);
114 $r2 = substr(stripastring( $a[0], 0 ), 3);
115 if( ($r == "N") &&
116 (substr_count($r2, "A") == 0) &&
117 (substr_count($r2, "B") > 0) &&
118 (substr_count($r2, "E") == 0))
119 { echo "Korrekt!"; }
120 else
121 { echo "Forkert!"; }
122
123 >
124 </div>
125 <br />
126 <?php
127 echo "De svarede: ";
128 if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
129 if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
130 if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
131 >
132 <div style="font-size: 8pt; padding-left: 10px;">
133 <?php
134 if(substr_count($r2, "A") > 0) {
135 echo "- <font style=\"text-decoration: underline; color:red;\">.
136 Forkert afsenderemail</font><br />"; }
137 if(substr_count($r2, "B") > 0) {
138 echo "- <font style=\"color:green;\">Et link \"passer\" ikke ind</font><br />"; }
139 if(substr_count($r2, "C") > 0) {
140 echo "- <font style=\"color:green;\">Der blev bedt om &nbsp;&nbsp;&nbsp;.
141 upassende oplysninger</font><br />"; }
142 if(substr_count($r2, "D") > 0) {
143 echo "- <font style=\"color:green;\">Emailen virkede forkert</font><br />"; }
144 if(substr_count($r2, "E") > 0) {
145 echo "- <font style=\"text-decoration: underline; color:red;\">Ved ikke</font><br />"; }
146 >
147 </div>
148 </td>
149 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
150 
152 </td>
153 </tr>
154 <tr>
155 <td align="left" colspan="2" style="padding-left: 15px; padding-right: 100px;
156 padding-bottom: 10px; font-size: 13px;">
157 <br />
158 Linket i denne email går til: http://www.ourtravel.com<br /><br />
159
160 Idet at linket ikke går til mytravel men en anden udekendt side,
161 kan denne mail ikke stoles på og det
162 frarådes at følge linket.<br />
163 Desuden skal det bemærkes at der bliver bedt om cpr-nummer,
164 hvilket ikke er passende. Hvis et cpr-nummer kommer
```

```

165     i de forkert hænder kan man risikere identitetssvindel,
166     hvor ondsindede personer forsøger at få adgang til
167     beskyttede entiteter med ens cpr-nummer.
168
169     </td>
170 </tr>
171 </table>
172 <!-- Slut Svarblok ----->
173 </td>
174 </tr>
175 <tr style="padding-top: 50px;">
176   <td align="right"><a href=" ../testslut.php?id=<?php echo $id; ?>">Tilbage til startside</a></td>
177 </tr>
178 <tr>
179 <td>
180 <!-- Start Svarblok ----->
181   <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0"
182     style="border: 1px solid #4286b0;">
183     <tr style="height: 49px; background-color: #70bde2;">
184       <td colspan="2"><h3>Spørgsmål 2 ud af 5 - Paypal</h3></td>
185     </tr>
186     <tr>
187       <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
188         <div style="font-weight: bold; font-color: green;">
189           <?php
190             $r = substr(stripastring( $a[1], 0 ), 2, 1);
191             $r2 = substr(stripastring( $a[1], 0 ), 3);
192             if( ($r == "N") &&
193               (substr_count($r2, "B") > 0) &&
194               (substr_count($r2, "E") == 0))
195               { echo "Korrekt!"; }
196             else
197               { echo "Forkert!"; }
198
199           ?>
200         </div>
201       <br />
202       <?php
203         echo "De svarede: ";
204         if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
205         if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
206         if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
207       ?>
208       <div style="font-size: 8pt; padding-left: 10px;">
209         <?php
210           if(substr_count($r2, "A") > 0) {
211             echo "- <font style=\"color:green\">Forkert afsenderemail</font><br />"; }
212           if(substr_count($r2, "B") > 0) {
213             echo "- <font style=\"color:green\">Et link \"passer\" ikke ind</font><br />"; }
214           if(substr_count($r2, "C") > 0) {
215             echo "- <font style=\"text-decoration: underline; color:red\">".
216               "Der blev bedt om &nbsp;&nbsp;upassende oplysninger</font><br />"; }
217           if(substr_count($r2, "D") > 0) {
218             echo "- <font style=\"color:green\">Emailen virkede forkert</font><br />"; }
219           if(substr_count($r2, "E") > 0) {
220             echo "- <font style=\"text-decoration: underline; color:red\">Ved ikke</font><br />"; }
221           ?>
222         </div>
223       </td>
224       <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
225         
227       </td>
228     </tr>
229 </tr>

```



```

295     </div>
296 </td>
297 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
298     
300 </td>
301 </tr>
302 <tr>
303     <td align="left" colspan="2" style="padding-left: 15px; padding-right: 100px;
304         padding-bottom: 10px; font-size: 13px;">
305     <br />
306     Linket i denne email går til: http://tastselv\_skat.DK/SKAT.aspx?oID=349844<br /><br />
307
308     Denne mail ser helt rigtig ud, men hvis man kigger efter på linket
309     kan man se at punktummet imellem tast selv og skat
310     er blevet byttet ud med et '_'. Dette gør at linket potentielt slet
311     ikke ender hos SKAT.<br />
312     Hvis man er usikker på om et link virkelig ender det rigtige sted,
313     bør man hvis muligt indtaste adressen i hånden
314     med en adresse man er sikker på og så klikke sig frem til det rigtige sted. <br />
315     I dette tilfælde er den sikre metode at gå ind på www.skat.dk og
316     derfra klikke sig vej hen til tast selv skat. På
317     denne måde er man sikker på at man ikke kommer ind på en phished hjemmeside.<br />
318     Det kan ikke anbefales at stole på denne email.
319 </td>
320 </tr>
321 </table>
322 <!-- Slut Svarblok ----->
323 </td>
324 </tr>
325 <tr style="padding-top: 50px;">
326 <td align="right"><a href="mailto:../testslut.php?id=?php echo $id; ?">Tilbage til startside</a></td>
327 </tr>
328 <tr>
329 <td>
330 <!-- Start Svarblok ----->
331 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0"
332     style="border: 1px solid #4286b0;">
333 <tr style="height: 49px; background-color: #70bde4;">
334 <td colspan="2"><h3>Spørgsmål 4 ud af 5 - Apple store</h3></td>
335 </tr>
336 <tr>
337 <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
338 <div style="font-weight: bold; font-color: green;">
339 <?php
340     $r = substr(stripstring( $a[3], 0 ), 2, 1);
341     $r2 = substr(stripstring( $a[3], 0 ), 3);
342     if($r == "Y")
343     { echo "Korrekt!"; }
344     else
345     { echo "Forkert!"; }
346
347     ?>
348 </div>
349 <br />
350 <?php
351     echo "De svarede: ";
352     if($r == "Y") { echo "<font style=\"color:green\">Ja</font><br />"; }
353     if($r == "N") { echo "<font style=\"color:red\">Nej</font><br />"; }
354     if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
355     ?>
356 <div style="font-size: 8pt; padding-left: 10px;">
357 <?php
358     if(substr_count($r2, "A") > 0) {
359         echo "- <font style=\"text-decoration: underline; color:red;\">".

```

```

360     "Forkert afsenderemail</font><br />"; }
361   if(substr_count($r2, "B") > 0) {
362     echo "- <font style=\text-decoration: underline; color:red;\>".
363     "Et link \passer\" ikke ind</font><br />"; }
364   if(substr_count($r2, "C") > 0) {
365     echo "- <font style=\text-decoration: underline; color:red;\>".
366     "Der blev bedt om &nbsp;&nbsp;&nbsp;upassende oplysninger</font><br />"; }
367   if(substr_count($r2, "D") > 0) {
368     echo "- <font style=\text-decoration: underline; color:red;\>".
369     "Emailen virkede forkert</font><br />"; }
370   if(substr_count($r2, "E") > 0) {
371     echo "- <font style=\text-decoration: underline; color:red;\>Ved ikke</font><br />"; }
372   ?>
373 </div>
374 </td>
375 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
376   
378 </td>
379 </tr>
380 <tr>
381   <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px;
382   padding-bottom: 10px; font-size: 13px;">
383     <br />
384     Denne mail beder ikke om nogle informationer, og de link der er tilstede går til
385     rent informerende sider på www.aok.dk<br />
386     Pga. dette er der ingen grund til ikke at stole på emailen.
387     <br />
388   </td>
389 </tr>
390 </table>
391 <!-- Slut Svarblok ----->
392 </td>
393 </tr>
394 <tr style="padding-top: 50px;">
395   <td align="right"><a href=" ../testslut.php?id=<?php echo $id; ?>">Tilbage til startsidan</a></td>
396 </tr>
397 <tr>
398 <td>
399 <!-- Start Svarblok ----->
400   <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0"
401   style="border: 1px solid #4286b0;">
402   <tr style="height: 49px; background-color: #70bded;">
403     <td colspan="2"><h3>Spørgsmål 5 ud af 5 - Falck</h3></td>
404   </tr>
405   <tr>
406     <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
407       <div style="font-weight: bold; font-color: green;">
408         <?php
409           $r = substr(stripastring( $a[4], 0 ), 2, 1);
410           $r2 = substr(stripastring( $a[4], 0 ), 3);
411           if( ($r == "N") &&
412             (substr_count($r2, "A") == 0) &&
413             (substr_count($r2, "B") > 0) &&
414             (substr_count($r2, "E") == 0))
415             { echo "Korrekt!"; }
416           else
417             { echo "Forkert!"; }
418         ?>
419       </div>
420     <br />
421   <?php
422     echo "De svarede: ";
423     if($r == "Y") { echo "<font style=\color:red\>Ja</font><br />"; }
424

```

```

425     if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
426     if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
427     ?>
428     <div style="font-size: 8pt; padding-left: 10px;">
429     <?php
430         if(substr_count($r2, "A") > 0) {
431             echo "- <font style=\"text-decoration: underline; color:red;\>".
432                 "Forkert afsenderemail</font><br />"; }
433         if(substr_count($r2, "B") > 0) {
434             echo "- <font style=\"color:green;\>Et link \"passer\" ikke ind</font><br />"; }
435         if(substr_count($r2, "C") > 0) {
436             echo "- <font style=\"text-decoration: underline; color:red;\>".
437                 "Der blev bedt om &nbsp;&nbsp;&nbsp;opassende oplysninger</font><br />"; }
438         if(substr_count($r2, "D") > 0) {
439             echo "- <font style=\"text-decoration: underline; color:red;\>".
440                 "Emailen virkede forkert</font><br />"; }
441         if(substr_count($r2, "E") > 0) {
442             echo "- <font style=\"text-decoration: underline; color:red;\>Ved ikke</font><br />"; }
443         ?>
444     </div>
445 </td>
446 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
447     
449 </td>
450 </tr>
451 <tr>
452     <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px; ".
453     "padding-bottom: 10px; font-size: 13px;">
454     <br />
455     Linket i denne email går til: http://www.givmigdinadgangskode.dk<br /><br />
456
457     Denne mail ser rigtig ud, og de første links går virkelig til falcks hjemmeside.
458     Men det sidste link der leder
459     til ændring af ens brugernavn og adgangskode er tydeligvis falsk.<br />
460     Det kan derfor ikke anbefales at stolle på denne email.
461 </td>
462 </tr>
463 </table>
464 <!-- Slut Svarblok ----->
465 </td>
466 </tr>
467 </table>
468 </div>
469 <?php include("../footer.php"); ?>

```

## A.5.20 Trådløst wireless.php

```

1 <?php $restricted = true; $notstart = "1"; include("../header.php"); ?>
2 <br />
3
4 <form name="emailForm" action="wirelessh.php" method="POST">
5 <input name="Results" type="hidden">
6 </form>
7 </head>
8
9 <div style="padding: 0px; width:660px; border: 1px solid #4286b0;">
10 <script type="text/javascript">
11 <!--
12 writeDocument(' <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000" ' +
13 'codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0" ' +
14 'width="660" height="509" ID="Captivatel">' +
15 ' <param name="movie" value="wireless_skin.swf">' +
16 ' <param name="quality" value="high">' +

```

```

17 '<param name="menu" value="false">'+
18 '<param name="loop" value="0">'+
19 '<embed src="wireless_skin.swf" width="660" height="509" loop="0" quality="high" '+
20 'pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash" '+
21 ' menu="false"></embed>'+
22 '</object>');
23 //-->
24 </script>
25 <noscript>
26 <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000" '+
27 'codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0" '+
28 ' width="660" height="509" ID="Captivate1">
29 <param name="movie" value="wireless_skin.swf">
30 <param name="quality" value="high">
31 <param name="menu" value="false">
32 <param name="loop" value="0">
33 <embed src="wireless_skin.swf" width="660" height="509" loop="0" quality="high" '+
34 'pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash" '+
35 ' menu="false"></embed>
36 </object>
37 </noscript>
38 </center>
39 </div>
40 <?php include("../footer.php"); ?>

```

## A.5.21 Trådløst wirelessh.php

```

1 <?php $restricted = true; $notstart = "1"; include("../header.php");require_once '../classes.php';?>
2
3 <div style="width: 790px; margin:0px; padding:0px;" >
4 <?php
5
6 function striptostrings( $result )
7 {
8     // strip the start info
9
10    $result = strstr(stripslashes($result), "\",\"Student Response\", \"Result\", \"",
11    "\"Weight\", \"Latency\"");
12    $num = strpos($result, "\n");
13    $result = substr($result, $num+1);
14
15    // now $result is only the five lines of results
16    // strip these five lines into an array
17
18    for($i = 0; $i < 4; $i++)
19    {
20        $num = strpos($result, "\n");
21        $arr[$i] = substr($result, 0, $num);
22        $result = substr($result, $num+1);
23    }
24    $arr[4] = $result;
25
26    return $arr;
27 }
28
29 function stripastring( $string, $print )
30 {
31     // is in 10 parts seperated by "," we need parts 3 and 7
32     $string;
33     $num = strpos($string, "\",\""); // 1
34     $string = substr($string, $num+1);
35     $num = strpos($string, "\",\""); // 2
36     $string = substr($string, $num+1);
37     $num = strpos($string, "\",\""); // 3 (we need this one)

```



```

38 $arr[0] = substr($string, 2, $num-2);
39 $string = substr($string, $num+1);
40 $num = strpos($string, "\",\""); // 4
41 $string = substr($string, $num+1);
42 $num = strpos($string, "\",\""); // 5
43 $string = substr($string, $num+1);
44 $num = strpos($string, "\",\""); // 6
45 $string = substr($string, $num+1);
46 $num = strpos($string, "\",\""); // 7 (last one we need)
47 $arr[1] = substr($string, 2, $num-2);
48 $string = substr($string, $num+1);
49
50 if($print == 1)
51 {
52 if($arr[1] != 1)
53 echo $arr[0].";".$arr[1]."<br />";
54 else
55 echo $arr[0]."<br />";
56 }
57 if($arr[1] != 1)
58 return $arr[0].";".$arr[1];
59 else
60 return $arr[0];
61 }
62 }
63
64 $result = nl2br($_POST['Results']).": til";
65 //echo $result;
66 //echo "<br />Her: <br />";
67
68 $a = striptostings( $result );
69
70 $r = stripastring( $a[0], 0 );
71
72
73 // Læg alt op i databasen
74 if(gettestfinished(4) == false)
75 {
76 mysql_connect("localhost", "tre_ben_dk", "kabloolie") or die(mysql_error());
77 mysql_select_db("tre_ben_dk") or die(mysql_error());
78 $result = mysql_query("INSERT INTO sik_testdata (id, person_id, test_id ,data1, data32) VALUES (".
79 "'' ".
80 ",'".$_SESSION['user']."' ".
81 ",'4' ".
82 ",'".$stripastring( $a[0], 0 )."' ".
83 ",'OK' ".
84 ");") or die(mysql_error());
85
86 $id = mysql_insert_id();
87
88 mysql_close();
89 }
90 ?>
91
92 <table id="content" width="660" border="0" cellpadding="0" cellspacing="0" style="margin:0px;" >
93 <tr>
94 <td align="center"><h1>Svar fra test</h1></td>
95 </tr>
96 <tr>
97 <td align="right"></td>
98 </tr>
99 <tr>
100 <td>
101 <!-- Start Svarblok ----->
102 <table class="svar" width="660" border="0" cellpadding="0" cellspacing="0"

```

```

103     style="border: 1px solid #4286b0;">
104 <tr style="height: 49px; background-color: #70bde4;">
105     <td colspan="2"><h3>Wireless spørgsmål - Vælg et sikkert trådløst netværk</h3></td>
106 </tr>
107 <tr>
108     <td style="padding-left: 15px; padding-top: 10px;">
109     <div style="font-weight: bold; font-color: green;">
110         <?php
111             if(($r == '1') || ($r == '2'))
112             { echo "Korrekt!"; }
113             else
114             { echo "Forkert!"; }
115         <?php
116             ?>
117         </div>
118     <br />
119     <?php
120         echo "De svarede: ";
121     if($r == '1') {
122         echo "<font style='color:green'>Wireless (med sikkerhedsaktivering)</font><br />"; }
123     if($r == '2') {
124         echo "<font style='color:green'>Ninjanet (med sikkerhedsaktivering - WPA)</font><br />"; }
125     if($r == '3') {
126         echo "<font style='color:red'>TOTSK (uden sikkerhedsaktivering)</font><br />"; }
127     if($r == '4') {
128         echo "<font style='color:red'>Belkinbro (uden sikkerhedsaktivering)</font><br />"; }
129     ?>
130     </td>
131 </tr>
132 <tr>
133 <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px;
134 padding-bottom: 10px; font-size: 13px;">
135 <br />
136 Trådløse netværk uden en form for sikkerhed som kryptering aktiveret, sender alt information
137 helt ubeskyttet over netværket. Derved kan ondsindede personer udnytte og stjæle eventuelt
138 følsomme oplysninger der bliver sendt over netværket.<br />
139 <br />
140 Trådløse netværk med sikkerhedsaktivering som kryptering, koder alle beskeder der bliver
141 sendt sådan at dette ikke kan ske. Desuden vil krypterede hjemmenetværk fraholde andre
142 ikke-inviterede brugere fra at hugge båndbredden.
143 </td>
144 </tr>
145 </table>
146 <!-- Slut Svarblok ----->
147 </td>
148 </tr>
149 </table>
150 </div>
151 <?php include("../footer.php"); ?>

```

## A.6 Iteration 2 - Website for sikkerhedsbevidsthed

### A.6.1 Beskrivelse af ajax-modulet til statistiksiden

Følgende afnit fra javascriptet *statscript.js* (bilag A.6.14) viser AJAX delen af statistik modulet:

```

function ajaxManager()
{

```

```

var args = ajaxManager.arguments;
switch (args[0])
{
case "load_page":
    if (document.getElementById)
    {
        var x = (window.ActiveXObject) ?
            new ActiveXObject("Microsoft.XMLHTTP") :
            new XMLHttpRequest();
    }
    if (x)
    {
        x.onreadystatechange = function()
        {
            if (x.readyState == 4 && x.status == 200)
            {
                e1 = document.getElementById(args[2]);
                e1.innerHTML = x.responseText;
                e2 = document.getElementById('loadfield');
                e2.innerHTML = "<div style=\"color: black;\">Done</div>";
            }
        }

        document.form.nocache.value = Date(); // ingen cache
        x.open("GET", args[1], true);
        x.send(null);
    }
    break;
case "start_up":
    var lget = "?age_from="+document.form.age_from.value+
        "&age_to="+document.form.age_to.value+
        "&gender_male="+document.form.gender_male.checked+
        "&gender_female="+document.form.gender_female.checked+
        "&skill_none="+document.form.skill_none.checked+
        "&skill_experienced="+document.form.skill_experienced.checked+
        "&skill_expert="+document.form.skill_expert.checked+
        "&dato="+Date();

    ajaxManager('load_page', 'statdata.php'+lget, 'statdata');
    break;
}
}

```

Funktionen *ajaxManager()* kaldes med argumentet 'start\_up', hvorved en variabel *lget* bliver initialiseret til de nuværende parametre for form-elementerne. Denne variabel bliver sammensat således at den kan puttes bagefter adressen på den fil der skal hentes ind, sådan at denne formes som et html 'GET' kald med parametre (*filnavn?var1=data1&var2=data2*). Dette sørger for at *statdata.php* får tilsendt afgrænsningerne til statistikkerne. Ved brug af et ActiveX object og funktionen *XMLHttpRequest()*, kalder AJAX så asynkront *statdata.php*, venter til at den har modtaget koden derfra og putter dem ind i et division-tag med id værende 'statdata'.

Idet AJAX kører asynkront i baggrunden har brugeren ikke umiddelbart nogen mulighed for at se hvor langt browseren er nået med hentningen af siden. Derfor har jeg tilføjet et felt øverst i højre hjørne af statistikken, som når færdig med opdateringen viser 'Done', og mens der opdateres vises i rød skrift som 'Opdaterer'. Dette er lavet som endnu et division tag, hvis indhold ændres i

javascriptet når afgrænsingerne ændres.

Under implementeringen forekom det at internetcachen til tider blev benyttet ved opdateringen, når det var med samme afgrænsninger som tidligere benyttet. Dette forsagede så at en tidligere version af statistikken blev hentet ind hvilket gjorde øjebliksbilledet af statistikken ukonsistent. I *statdata.php*, ændrede jeg derfor headeren til at bede om altid at undlade at blive gemt i cachen, men da jeg gentagne gange så at dette ikke virkede, tilføjede jeg et ekstra parameter til AJAX-kaldet indholdene den nuværende dato og tid, således at siden aldrig vil blive identificeret som en side liggende i internetcachen, da cachen aldrig ville matche den nuværende dato og tid.

## A.6.2 Test kodeord password.php

```

1 <?php $restricted = true; $notstart = "1"; include("../header.php"); ?>
2 <div style="padding: 0px; width:500px; text-align: left; ">
3   <b>Hvor sikkert er dit kodeord?</b><br />
4   <p>
5     Hvor sikkert et kodeord er, er bestemt af hvor nemt det er at gætte.<br />
6     <br />
7     Test dig selv og skriv dit kodeord i feltet nedenfor og tryk på 'test' for at få en
8     vurdering af hvor sikkert dit kodeord er, samt med foreslag til hvordan du kan forbedre.<br />
9     <br />
10    Dit kodeord vil ikke blive gemt, og kun hvor sikkert det blev vurderes registreres til statistik.
11  </p>
12 <br /><br /><br />
13  <form name="form" action="passwordh.php" method="post" onsubmit="return vurderkodeord();">
14    <b>Test dit kodeord: </b><div style="display: inline; padding-left: 100px;">&nbsp;</div>
15    <input name="pass" type="password" size="20" maxvalue="30" />
16    <input type="submit" value="Test" style="width:75px;" />
17    <input name="points" type="hidden" value="" />
18    <input name="length" type="hidden" value="" />
19    <input name="bigsmall" type="hidden" value="" />
20    <input name="specialnumber" type="hidden" value="" />
21  </form>
22  <br /><br /><br /><br /><br /><br />
23
24 </div>
25 <?php include("../footer.php"); ?>

```

## A.6.3 Test kodeord passwordh.php

```

1 <?php $restricted = true; $notstart = "1"; include("../header.php");require_once '../classes.php';?>
2
3 <?php
4
5   // Læg alt op i databasen
6   if(gettestfinished(5) == false)
7   {
8     mysql_connect("localhost", "tre_ben_dk", "kablooie") or die(mysql_error());
9     mysql_select_db("tre_ben_dk") or die(mysql_error());
10    $result = mysql_query("INSERT INTO sik_testdata (id, person_id, test_id ,data1, ".
11                        " data2, data3, data4, data32) VALUES (".
12                        "'', ".

```

```

13         ",'" . $_SESSION['user'] . "' " .
14         ",'" .
15         ",'" . $_POST['points'] . "' " .
16         ",'" . $_POST['length'] . "' " .
17         ",'" . $_POST['bigsmall'] . "' " .
18         ",'" . $_POST['specialnumber'] . "' " .
19         ",'" . 'OK' " .
20         "');" or die(mysql_error());
21
22     $sid = mysql_insert_id();
23
24     mysql_close();
25 }
26
27 ?>
28 <div style="padding: 0px; width:500px; text-align: left; ">
29 <b>Vurdering af kodeord</b>
30 <a style="color: #6c6c6c;"
31 href=" ../testslut.php?id=<?php echo $id; ?>">(tilbage til startside)</a><br />
32 <br /><br />
33
34 <table width="340" cellpadding="0" cellspacing="0">
35 <tr><td></td><td style="padding-bottom: 2px;">Dit kodeord blev vurderet som: <?php
36 if($_POST['points'] <= 40) { echo "Dårligt sikret"; }
37 if($_POST['points'] > 70) { echo "Godt sikret"; }
38 if((($_POST['points'] <= 70) && ($_POST['points'] > 40)) { echo "Mellem sikret"; }
39 //echo " -".$_POST['points'];
40 ?></td><td></td>
41 <tr><td width="20"></td>
42 <td style=" width:300px; border: 1px solid black;" align="left">
43 <div style="width: <?php
44 echo ($_POST['points'] * 3) . "px; ";
45
46 if($_POST['points'] <= 40) { echo "background-color: #ff0000;"; }
47 if($_POST['points'] > 70) { echo "background-color: #00ff00;"; }
48 if((($_POST['points'] <= 70) && ($_POST['points'] > 40)){echo "background-color: #ffff00;"; }
49 ?>>&nbsp;  </div>
50 </td>
51 <td width="20"></td>
52 </tr>
53 <tr>
54 <td align="left" colspan="3">
55 <div style="display:inline;padding: 0px; margin: 0px; padding-right: 12px">&nbsp;  </div>0
56 <div style="display:inline;padding: 0px; margin: 0px;padding-right: 132px; ">&nbsp;  </div>50
57 <div style="display:inline;padding: 0px; margin: 0px;padding-right: 124px; ">&nbsp;  </div>100
58 </td>
59 </table>
60 <br /><br />
61 Et sikkert kodeord skal:<br /><br />
62 <?php
63 if($_POST['length'] < 8)
64 { echo "<img src=\" ../images/cross.jpg\" />"; }
65 else
66 { echo "<img src=\" ../images/check.jpg\" />"; }
67 echo " - være mindst otte felter langt.<br />\n";
68
69 if($_POST['bigsmall'] == "false")
70 { echo "<img src=\" ../images/cross.jpg\" />"; }
71 else
72 { echo "<img src=\" ../images/check.jpg\" />"; }
73 echo " - indeholde både små og store bogstaver<br />\n";
74
75 if($_POST['specialnumber'] == "false")
76 { echo "<img src=\" ../images/cross.jpg\" />"; }
77 else

```

```

78 { echo "<img src=\"../images/check.jpg\" />"; }
79 echo " - indeholde specielle tegn eller tal<br />\n";
80 ?>
81 <br />
82 Kodeordet må ikke indeholde elementer af det brugernavn det bliver benyttet sammen med.
83 Hvis det gør er det meget nemt at gætte. Kodeord der er dannet af et rigtigt ord er
84 nemmere at gætte. Så dan ikke kodeord af rigtige ord.<br />
85 <br />
86 Forsøg at overholde så mange af reglerne her så dit kodeord bliver vurderet grøn
87 (godt sikret). Men sørg stadigvæk for at kodeordet ikke bliver så kompliceret at du ikke
88 kan huske det.
89 <br /><br /><br /><br /><br /><br /><br /><br />
90 <br />
91
92 </div>
93 <?php include("../footer.php"); ?>
94
95
96
97
98
99

```

## A.6.4 classes.php

```

1 <?php
2
3
4 function connect()
5 {
6     mysql_connect("localhost", "tre_ben_dk", "kablooie") or die(mysql_error());
7     mysql_select_db("tre_ben_dk") or die(mysql_error());
8 }
9
10 function statperson()
11 {
12     connect();
13
14     $result = mysql_query("SELECT count(*) FROM sik_person;") or die(mysql_error());
15     $row = mysql_fetch_row($result);
16     if ( $row )
17     { return $row[0]; }
18     else
19     { return false; }
20
21 }
22
23 function statpersonw($where)
24 {
25     connect();
26
27     $result = mysql_query("SELECT count(*) FROM sik_person ".$where." ;") or die(mysql_error());
28     $row = mysql_fetch_row($result);
29     if ( $row )
30     { return $row[0]; }
31     else
32     { return false; }
33
34 }
35
36 function stattestdata()
37 {
38     connect();
39

```

```
40 $result = mysql_query("SELECT count(*) FROM sik_testdata where data32 = 'OK';") or die(mysql_error());
41 $row = mysql_fetch_row($result);
42 if ( $row )
43 { return $row[0]; }
44 else
45 { return false; }
46
47 }
48
49 function stattestdataw($where)
50 {
51     connect();
52
53     $result = mysql_query("SELECT count(*) FROM sik_testdata where person_id = ANY ".
54         "(select id from sik_person ".$where.");") or die(mysql_error());
55     $row = mysql_fetch_row($result);
56     if ( $row )
57     { return $row[0]; }
58     else
59     { return false; }
60
61 }
62
63 function stattest()
64 {
65     connect();
66
67     $result = mysql_query("SELECT count(*) FROM sik_test;") or die(mysql_error());
68     $row = mysql_fetch_row($result);
69     if ( $row )
70     { return ($row[0]-1); }
71     else
72     { return false; }
73
74 }
75
76 function stattestdatat($where, $test)
77 {
78     connect();
79     if($where != "")
80     {
81         $result = mysql_query("SELECT count(*) FROM sik_testdata where person_id = ANY ".
82             "(select id from sik_person ".$where.") AND test_id = ".$test.
83             ";;") or die(mysql_error());
84     }
85     else
86     {
87         $result = mysql_query("SELECT count(*) FROM sik_testdata WHERE test_id = "
88             ".$test.;") or die(mysql_error());
89     }
90     //echo $result;
91     $row = mysql_fetch_row($result);
92     if ( $row )
93     { return $row[0]; }
94     else
95     { return false; }
96
97 }
98
99 function stattest4($where, $data)
100 {
101     connect();
102     switch($data)
103     {
104         case 1:
```

```

105     $result = mysql_query("SELECT count(*) FROM sik_testdata ".
106         "where person_id = ANY ".
107         "(select id from sik_person ".$where.") AND test_id = 4 ".
108         "AND (datal = 1 OR datal = 2);")
109         or die(mysql_error());
110     break;
111
112 case 2:
113     $result = mysql_query("SELECT count(*) FROM sik_testdata ".
114         "where person_id = ANY ".
115         "(select id from sik_person ".$where.") AND test_id = 4 ".
116         "AND (datal = 3 OR datal = 4 OR datal = 0);")
117         or die(mysql_error());
118     break;
119 case 3:
120     $result = mysql_query("SELECT count(*) FROM sik_testdata ".
121         "where person_id = ANY ".
122         "(select id from sik_person ".$where.") AND test_id = 4".
123         " AND data31 = 'JA';")
124         or die(mysql_error());
125     break;
126 }
127
128
129 $row = mysql_fetch_row($result);
130 if ( $row )
131 { return $row[0]; }
132 else
133 { return false; }
134
135 }
136
137 function stattest5($where, $data)
138 {
139     connect();
140     switch($data)
141     {
142     case 1:
143         $result = mysql_query("SELECT avg(datal) FROM sik_testdata ".
144             "where person_id = ANY ".
145             "(select id from sik_person ".$where.") AND test_id = 5;")
146             or die(mysql_error());
147         break;
148
149     case 2:
150         $result = mysql_query("SELECT avg(data2) FROM sik_testdata ".
151             "where person_id = ANY ".
152             "(select id from sik_person ".$where.") AND test_id = 5;")
153             or die(mysql_error());
154         break;
155
156     case 3:
157         $result = mysql_query("SELECT count(*) FROM sik_testdata ".
158             "where person_id = ANY ".
159             "(select id from sik_person ".$where.") AND test_id = 5".
160             " AND data2 > 7;")
161             or die(mysql_error());
162         break;
163     case 4:
164         $result = mysql_query("SELECT count(*) FROM sik_testdata ".
165             "where person_id = ANY ".
166             "(select id from sik_person ".$where.") AND test_id = 5".
167             " AND data31 = 'JA';")
168             or die(mysql_error());
169         break;

```



```
170     }
171
172
173     $row = mysql_fetch_row($result);
174     if ( $row )
175     { return $row[0]; }
176     else
177     { return false; }
178
179 }
180
181 function stattet2($where, $data, $row)
182 {
183     connect();
184     $line = "";
185     switch($row)
186     {
187         case 1:
188             $line = "AND data".$data." = 'Q".$data."Y'";
189             break;
190         case 2:
191             $line = "AND data".$data." like 'Q".$data."N%'";
192             break;
193         case 3:
194             $line = "AND data".$data." = 'Q".$data."D'";
195             break;
196     }
197
198     switch($data)
199     {
200         case 1:
201             if($row == 4)
202             {
203                 $line = "AND data".$data." like 'Q".$data."N%".
204                     "AND data".$data." like '%A%' ".
205                     "AND data".$data." not like '%B%' ".
206                     "AND data".$data." not like '%C%' ".
207                     "AND data".$data." not like '%E%' ";
208             }
209
210             $result = mysql_query("SELECT count(*) FROM sik_testdata ".
211                 "where person_id = ANY ".
212                 "(select id from sik_person ".$where.") AND test_id = 2 ".$line.
213                 ";")
214                 or die(mysql_error());
215             break;
216
217         case 2:
218             if($row == 4)
219             {
220                 $line = "AND data".$data." like 'Q".$data."N%".
221                     "AND data".$data." like '%C%' ".
222                     "AND data".$data." not like '%A%' ".
223                     "AND data".$data." not like '%B%' ".
224                     "AND data".$data." not like '%E%' ";
225             }
226
227             $result = mysql_query("SELECT count(*) FROM sik_testdata ".
228                 "where person_id = ANY ".
229                 "(select id from sik_person ".$where.") AND test_id = 2 ".$line.
230                 ";")
231                 or die(mysql_error());
232             break;
233
234         case 3:
```

```

235     if($row == 4)
236     {
237         $line = "AND data".$data." = 'Q'".$data."Y' ";
238     }
239
240     $result = mysql_query("SELECT count(*) FROM sik_testdata ".
241         "where person_id = ANY ".
242         "(select id from sik_person ".$where.") AND test_id = 2 ".$line.
243         " ");
244         or die(mysql_error());
245     break;
246 case 4:
247     if($row == 4)
248     {
249         $line = "AND data".$data." like 'Q'".$data."N' ".
250         "AND (data".$data." like '%B%' ".
251         "OR data".$data." like '%C%' ) ".
252         "AND data".$data." not like '%A%' ".
253         "AND data".$data." not like '%E%' ";
254     }
255
256     $result = mysql_query("SELECT count(*) FROM sik_testdata ".
257         "where person_id = ANY ".
258         "(select id from sik_person ".$where.") AND test_id = 2 ".$line.
259         " ");
260         or die(mysql_error());
261     break;
262 case 5:
263     if($row == 4)
264     {
265         $line = "AND data".$data." like 'Q'".$data."N' ".
266         "AND data".$data." like '%A%' ".
267         "AND data".$data." not like '%B%' ".
268         "AND data".$data." not like '%C%' ".
269         "AND data".$data." not like '%E%' ";
270     }
271
272     $result = mysql_query("SELECT count(*) FROM sik_testdata ".
273         "where person_id = ANY ".
274         "(select id from sik_person ".$where.") AND test_id = 2 ".$line.
275         " ");
276         or die(mysql_error());
277     break;
278 case 'JA':
279     $result = mysql_query("SELECT count(*) FROM sik_testdata ".
280         "where person_id = ANY ".
281         "(select id from sik_person ".$where.") AND test_id = 2 ".
282         " AND data31 = 'JA'");
283         or die(mysql_error());
284     break;
285 }
286
287
288 $row = mysql_fetch_row($result);
289 if ( $row )
290 { return $row[0]; }
291 else
292 { return false; }
293
294 }
295
296 function stattest2intro($where, $data, $row)
297 {
298     connect();
299

```

```
300 switch($data)
301 {
302     case 1:
303
304         $result = mysql_query("SELECT count(*) FROM sik_testdata ".
305                               "where person_id = ANY ".
306                               "(select id from sik_person ".$where.") AND test_id = 2 AND ".
307                               "data7 = '". $row.'" ".
308                               ";")
309                               or die(mysql_error());
310         break;
311
312     case 2:
313
314         $result = mysql_query("SELECT count(*) FROM sik_testdata ".
315                               "where person_id = ANY ".
316                               "(select id from sik_person ".$where.") AND test_id = 2 AND ".
317                               "data8 = '". $row.'" ".
318                               ";")
319                               or die(mysql_error());
320         break;
321     }
322
323
324     $row = mysql_fetch_row($result);
325     if ( $row )
326     { return $row[0]; }
327     else
328     { return false; }
329
330 }
331
332 function getphishinginetscore($pid)
333 {
334     $score = 0;
335     for($i = 1; $i < 6; $i ++)
336     {
337         //echo "slide(".$i.)";
338         if(phishinginetscore($i, $pid) != 0)
339         {
340             $score = $score + 1;
341         }
342     }
343     //echo "(".$score.")\n";
344     return $score;
345 }
346
347 function phishinginetscore($slide, $pid)
348 {
349     switch($slide)
350     {
351         case 1:
352
353             $line = "AND data1 like 'Q1N%' ".
354                   "AND data1 like '%A%' ".
355                   "AND data1 not like '%B%' ".
356                   "AND data1 not like '%C%' ".
357                   "AND data1 not like '%E%' ";
358
359
360             break;
361
362         case 2:
363
364             $line = "AND data2 like 'Q2N%' ".

```

```

365         "AND data2 like '%C%' ".
366         "AND data2 not like '%A%' ".
367         "AND data2 not like '%B%' ".
368         "AND data2 not like '%E%' ";
369
370
371     break;
372
373     case 3:
374
375         $line = "AND data3 = 'Q3Y'";
376
377         break;
378     case 4:
379
380         $line = "AND data4 like 'Q4N%' ".
381         "AND (data4 like '%B%' ".
382         "OR data4 like '%C%') ".
383         "AND data4 not like '%A%' ".
384         "AND data4 not like '%E%' ";
385
386         break;
387     case 5:
388
389         $line = "AND data5 like 'Q5N%' ".
390         "AND data5 like '%A%' ".
391         "AND data5 not like '%B%' ".
392         "AND data5 not like '%C%' ".
393         "AND data5 not like '%E%' ";
394
395         break;
396
397 }
398 $result = mysql_query("SELECT count(*) FROM sik_testdata ".
399         "where person_id = ".$pid." AND test_id = 2 ".$line.
400         " ");
401         or die(mysql_error());
402 $row = mysql_fetch_row($result);
403 if ( $row )
404
405 { return $row[0]; }
406 else
407 { return false; }
408
409 }
410
411 function getsumphishinginetscore($where)
412 {
413     connect();
414
415     $result = mysql_query("SELECT sum(data30) FROM sik_testdata ".
416         "where person_id = ANY ".
417         "(select id from sik_person ".$where.") AND test_id = 2".
418         " AND data32 = 'OK'");
419         or die(mysql_error());
420 $row = mysql_fetch_row($result);
421 if ( $row )
422 { return $row[0]; }
423 else
424 { return false; }
425 mysql_close();
426 }
427
428 function stattest3($where, $data, $row)
429 {

```

```

430 connect();
431
432 switch($row)
433 {
434     case 1:
435         $line = "AND data".$data." = 'Q".$data."Y'";
436         break;
437     case 2:
438         $line = "AND data".$data." like 'Q".$data."N%";
439         break;
440     case 3:
441         $line = "AND data".$data." = 'Q".$data."D'";
442         break;
443 }
444
445 switch($data)
446 {
447     case 1:
448         if($row == 4)
449         {
450             $line = "AND data1 like 'Q1N%'.
451                 "AND (data1 like '%B%' ".
452                 "OR data1 like '%C%' ) ".
453                 "AND data1 not like '%A%' ".
454                 "AND data1 not like '%E%' ";
455         }
456
457         $result = mysql_query("SELECT count(*) FROM sik_testdata ".
458                             "where person_id = ANY ".
459                             "(select id from sik_person ".$where.") AND test_id = 3 ".$line.
460                             ";")
461                             or die(mysql_error());
462
463         break;
464
465     case 2:
466         if($row == 4)
467         {
468             $line = "AND data2 like 'Q2N%'.
469                 "AND (data2 like '%B%' ".
470                 "OR data2 like '%A%' ) ".
471                 "AND data2 not like '%E%' ";
472         }
473
474         $result = mysql_query("SELECT count(*) FROM sik_testdata ".
475                             "where person_id = ANY ".
476                             "(select id from sik_person ".$where.") AND test_id = 3 ".$line.
477                             ";")
478                             or die(mysql_error());
479
480         break;
481
482     case 3:
483         if($row == 4)
484         {
485             $line = "AND data3 like 'Q3N%'.
486                 "AND data3 like '%B%' ".
487                 "AND data3 not like '%C%' ".
488                 "AND data3 not like '%A%' ".
489                 "AND data3 not like '%E%' ";
490         }
491
492         $result = mysql_query("SELECT count(*) FROM sik_testdata ".
493                             "where person_id = ANY ".
494                             "(select id from sik_person ".$where.") AND test_id = 3 ".$line.
495                             ";")

```

```

495         or die(mysql_error());
496     break;
497 case 4:
498     if($row == 4)
499     {
500         $line = "AND data4 = 'Q4Y' ";
501     }
502
503     $result = mysql_query("SELECT count(*) FROM sik_testdata ".
504         "where person_id = ANY ".
505         "(select id from sik_person ".$where.") AND test_id = 3 ".$line.
506         ";")
507         or die(mysql_error());
508     break;
509 case 5:
510     if($row == 4)
511     {
512         $line = "AND data5 like 'Q5N' ".
513             "AND data5 like '%B%' ".
514             "AND data5 not like '%A%' ".
515             "AND data5 not like '%C%' ".
516             "AND data5 not like '%E%' ";
517     }
518
519     $result = mysql_query("SELECT count(*) FROM sik_testdata ".
520         "where person_id = ANY ".
521         "(select id from sik_person ".$where.") AND test_id = 3 ".$line.
522         ";")
523         or die(mysql_error());
524     break;
525 case 'JA':
526     $result = mysql_query("SELECT count(*) FROM sik_testdata ".
527         "where person_id = ANY ".
528         "(select id from sik_person ".$where.") AND test_id = 3".
529         " AND data31 = 'JA'");
530         or die(mysql_error());
531     break;
532 }
533
534
535 $row = mysql_fetch_row($result);
536 if ( $row )
537 { return $row[0]; }
538 else
539 { return false; }
540
541 }
542
543 function getphishingemailscore($pid)
544 {
545     $score = 0;
546     for($i = 1; $i < 6; $i ++)
547     {
548         //echo "slide(".$i.)";
549         if(phishingemailscore($i, $pid) != 0)
550         {
551             $score = $score + 1;
552         }
553     }
554     //echo "(".$score.)\n";
555     return $score;
556 }
557
558 function phishingemailscore($slide, $pid)
559 {

```

```
560 switch($slide)
561 {
562     case 1:
563         $line = "AND data1 like 'Q1N%' ".
564             "AND (data1 like '%B%' ".
565             "OR data1 like '%C%') ".
566             "AND data1 not like '%A%' ".
567             "AND data1 not like '%E%' ";
568
569         break;
570
571     case 2:
572         $line = "AND data2 like 'Q2N%' ".
573             "AND (data2 like '%B%' ".
574             "or data2 like '%A%') ".
575             "AND data2 not like '%E%' ";
576
577         break;
578
579     case 3:
580         $line = "AND data3 like 'Q3N%' ".
581             "AND data3 like '%B%' ".
582             "AND data3 not like '%C%' ".
583             "AND data3 not like '%A%' ".
584             "AND data3 not like '%E%' ";
585
586         break;
587
588     case 4:
589         $line = "AND data4 = 'Q4Y' ";
590
591         break;
592
593     case 5:
594         $line = "AND data5 like 'Q5N%' ".
595             "AND data5 like '%B%' ".
596             "AND data5 not like '%A%' ".
597             "AND data5 not like '%C%' ".
598             "AND data5 not like '%E%' ";
599
600         break;
601 }
602 $result = mysql_query("SELECT count(*) FROM sik_testdata ".
603     "where person_id = ".$pid." AND test_id = 3 ".$line.
604     " ");
605 or die(mysql_error());
606 $row = mysql_fetch_row($result);
607 if ( $row )
608 { return $row[0]; }
609 else
610 { return false; }
611 }
612
613 function getsumphishingemailscore($where)
614 {
```

```

625     connect();
626     $result = 0;
627     $line = "AND data2 like 'Q2N%'".
628           "AND (data2 like '%B%' ".
629           "or data2 like '%A%' ) ".
630           "AND data2 not like '%E%' ";
631
632     $result = mysql_query("SELECT count(*) FROM sik_testdata ".
633                          "where person_id = ANY ".
634                          "(select id from sik_person ".$where.") AND test_id = 3".
635                          " AND data32 = 'OK' ".$line)
636           or die(mysql_error());
637
638     $row = mysql_fetch_row($result);
639     if ( $row )
640     { $result = $row[0]; }
641     else
642     { $result = 0; }
643     mysql_close();
644
645     connect();
646
647     $result = mysql_query("SELECT sum(data30) FROM sik_testdata ".
648                          "where person_id = ANY ".
649                          "(select id from sik_person ".$where.") AND test_id = 3".
650                          " AND data32 = 'OK'");
651           or die(mysql_error());
652
653     $row = mysql_fetch_row($result);
654     if ( $row )
655     { return ($row[0] + $result); }
656     else
657     { return false; }
658     mysql_close();
659 }
660
661 function countgettesttext()
662 {
663     connect();
664
665     $result = mysql_query("SELECT count(*) FROM sik_test;") or die(mysql_error());
666     $row = mysql_fetch_row($result);
667     if ( $row )
668     { return $row[0]; }
669     else
670     { return false; }
671     mysql_close();
672 }
673
674 function getusername($id)
675 {
676     connect();
677
678     $result = mysql_query("SELECT username FROM sik_person WHERE id = ".$id.";") or die(mysql_error());
679     $row = mysql_fetch_row($result);
680     if ( $row )
681     { return $row[0]; }
682     else
683     { return false; }
684     mysql_close();
685 }
686
687 function gettestfinished($tid)
688 {
689     connect();
690     $pid = $_SESSION['user'];

```



```
690 $result = mysql_query("SELECT data32 FROM sik_testdata where person_id = ".$pid.
691 " AND test_id = ".$tid.";") or die(mysql_error());
692 $row = mysql_fetch_row($result);
693 if ( $row[0] == "OK" )
694 { return true; }
695 else
696 { return false; }
697 mysql_close();
698 }
699
700 function gettesttext($id)
701 {
702 connect();
703 $query = "SELECT * FROM sik_test WHERE id = ".$id.";";
704 $result = mysql_query($query) or die("Forespørgelsen lykkedes ikke : " . mysql_error());
705
706 while ($row = mysql_fetch_object($result))
707 {
708 $array[] = $row;
709 }
710 if($array)
711 { return $array; }
712 else
713 { return null; }
714
715 mysql_free_result($result);
716 mysql_close($link);
717 }
718
719 function adduser($username, $password, $name, $gender, $age,
720 $skill, $bversion)
721 {
722 connect();
723
724 $result = mysql_query("INSERT INTO sik_person (id, username, password, name, gender, age, ".
725 "computerskill, browserversion)".
726 "VALUES ('', '$username.', '$password.', '$name.', ".
727 "'".$gender."', '$age.', '$skill.', '$bversion.'");
728 or die(mysql_error());
729
730 if ( $result )
731 { return true; }
732 else
733 { return false; }
734 mysql_close();
735 }
736
737 function userauth($username, $password)
738 {
739 connect();
740 $result = mysql_query("SELECT * FROM sik_person WHERE username = '$username.'" AND password = '$".
741 $password.'""); or die(mysql_error());
742 $row = mysql_fetch_array( $result );
743 if ( $row )
744 { return true; }
745 else
746 { return false; }
747 mysql_close();
748 }
749 // Bruger Login
750 function loginuser($username, $password)
751 {
752 if(userauth($username, $password))
753 {
754 $_SESSION['user'] = user_id($username);
```

```
755     return true;
756 }
757 else
758 {
759     return false;
760 }
761 }
762
763
764 function user_id($username)
765 {
766     connect();
767     $result = mysql_query("SELECT id FROM sik_person WHERE username = '".
768         $username."'"); or die(mysql_error());
769     $row = mysql_fetch_row($result);
770     if ( $row )
771     { return $row[0]; }
772     else
773     { return false; }
774     mysql_close();
775 }
776
777 function userexist($username)
778 {
779     connect();
780     $result = mysql_query("SELECT * FROM sik_person WHERE username = '".
781         $username."'"); or die(mysql_error());
782     $row = mysql_fetch_array( $result );
783     if ( $row )
784     { return true; }
785     else
786     { return false; }
787     mysql_close();
788 }
789
790 // Kontroller om en bruger er logget ind
791 function check()
792 {
793     return $_SESSION['user'];
794 }
795
796
797 // Bruger Logout
798 function logout()
799 {
800     if($_SESSION['user'])
801     {
802         $_SESSION = array();
803         session_destroy();
804         return true;
805     }
806 }
807
808
809
810 ?>
```

### A.6.5 script.js

```
1
2
3 function validatelogin()
4 {
5     if((document.frmLogin.txtUsername.value.length == 0) ||
```

```
6         (document.frmLogin.txtPassword.value.length == 0))
7     {
8         alert('Et af tekstfelterne er tomme');
9         return false;
10    }
11    else
12    {
13        return true;
14    }
15 }
16
17 function validatetestslut()
18 {
19     if((document.form.radTest[0].checked == false) &&
20        (document.form.radTest[1].checked == false))
21     {
22         alert('Enten \'ja\' eller \'nej\' skal vælges for at kunne fortsætte!');
23         return false;
24     }
25     else
26     {
27         return true;
28     }
29 }
30
31
32 function IsNumeric(sText)
33 {
34     var ValidChars = "0123456789";
35     var Char;
36
37     for (i = 0; i < sText.length; i++)
38     {
39         Char = sText.charAt(i);
40         if (ValidChars.indexOf(Char) == -1)
41         {
42             return false;
43         }
44     }
45     return true;
46 }
47
48     var BrowserDetect = {
49 init: function () {
50     this.browser = this.searchString(this.dataBrowser) || "An unknown browser";
51     this.version = this.searchVersion(navigator.userAgent)
52         || this.searchVersion(navigator.appVersion)
53         || "an unknown version";
54     this.OS = this.searchString(this.dataOS) || "an unknown OS";
55 },
56 searchString: function (data) {
57     for (var i=0;i<data.length;i++) {
58         var dataString = data[i].string;
59         var dataProp = data[i].prop;
60         this.versionSearchString = data[i].versionSearch || data[i].identity;
61         if (dataString) {
62             if (dataString.indexOf(data[i].subString) != -1)
63                 return data[i].identity;
64         }
65         else if (dataProp)
66             return data[i].identity;
67     }
68 },
69 searchVersion: function (dataString) {
70     var index = dataString.indexOf(this.versionSearchString);
```

```
71     if (index == -1) return;
72     return parseFloat(dataString.substring(index+this.versionSearchString.length+1));
73 },
74 dataBrowser: [
75     { string: navigator.userAgent,
76       subString: "OmniWeb",
77       versionSearch: "OmniWeb/",
78       identity: "OmniWeb"
79     },
80     {
81       string: navigator.vendor,
82       subString: "Apple",
83       identity: "Safari"
84     },
85     {
86       prop: window.opera,
87       identity: "Opera"
88     },
89     {
90       string: navigator.vendor,
91       subString: "iCab",
92       identity: "iCab"
93     },
94     {
95       string: navigator.vendor,
96       subString: "KDE",
97       identity: "Konqueror"
98     },
99     {
100      string: navigator.userAgent,
101      subString: "Firefox",
102      identity: "Firefox"
103     },
104     {
105      string: navigator.vendor,
106      subString: "Camino",
107      identity: "Camino"
108     },
109     { // for newer Netscapes (6+)
110      string: navigator.userAgent,
111      subString: "Netscape",
112      identity: "Netscape"
113     },
114     {
115      string: navigator.userAgent,
116      subString: "MSIE",
117      identity: "Explorer",
118      versionSearch: "MSIE"
119     },
120     {
121      string: navigator.userAgent,
122      subString: "Gecko",
123      identity: "Mozilla",
124      versionSearch: "rv"
125     },
126     { // for older Netscapes (4-)
127      string: navigator.userAgent,
128      subString: "Mozilla",
129      identity: "Netscape",
130      versionSearch: "Mozilla"
131     }
132 ],
133 dataOS : [
134     {
135       string: navigator.platform,
```

```
136     subString: "Win",
137     identity: "Windows"
138 },
139 {
140     string: navigator.platform,
141     subString: "Mac",
142     identity: "Mac"
143 },
144 {
145     string: navigator.platform,
146     subString: "Linux",
147     identity: "Linux"
148 }
149 ]
150
151 };
152 BrowserDetect.init();
153
154
155
156 function validateopretbruger()
157 {
158     if((document.frmLogin.txtUsername.value.length > 0) &&
159         (document.frmLogin.txtPassword.value.length > 0) &&
160         (document.frmLogin.txtPassword2.value.length > 0) &&
161         (document.frmLogin.txtName.value.length > 0) &&
162         (document.frmLogin.txtAge.value.length > 0) &&
163         ((document.frmLogin.radGender[0].checked == true) ||
164          (document.frmLogin.radGender[1].checked == true)) &&
165         ((document.frmLogin.radComputerskill[0].checked == true) ||
166          (document.frmLogin.radComputerskill[1].checked == true) ||
167          (document.frmLogin.radComputerskill[2].checked == true)) )
168     {
169         // Test af gentaget kodeord er ens
170         if(document.frmLogin.txtPassword.value != document.frmLogin.txtPassword2.value)
171         { alert("De to kodeord er ikke ens!"); return false; }
172
173         // Test Alder er tal og mellem 0 og 100
174         if(!IsNumeric(document.frmLogin.txtAge.value) ||
175            (document.frmLogin.txtAge.value < 1) ||
176            (document.frmLogin.txtAge.value > 99))
177         { alert("Alder skal være specificeret ved et tal mellem 0 og 100"); return false; }
178
179
180         document.frmLogin.bversion.value = BrowserDetect.OS+ " - " +
181         BrowserDetect.browser+ " "+BrowserDetect.version;
182
183         return true;
184     }
185     else
186     {
187         alert('Et fornødent tekstfelt var ikke udfyldt!');
188         return false;
189     }
190 }
191
192 // Phishing Web
193 var gstrEmailTo = "";
194 var gstrEmailSubject = "";
195 var gstrEmailBody = "";
196 var myVar = 0;
197
198 function appendEmailBody(strBody)
199 {
200     var temp_str = strBody;
```

```
201 while(temp_str.indexOf("|") > -1)
202 {
203     temp_str = temp_str.substr(0,temp_str.indexOf("|")) + "\r" +
204     temp_str.substr(temp_str.indexOf("|") + 1)
205 }
206 gstrEmailBody += temp_str + "\r";
207 }
208
209 function sendMail()
210 {
211
212
213     //document.emailForm.elements["Results"].value = "\r" + gstrEmailBody;
214     document.emailForm.Results.value = "\r" + gstrEmailBody;
215
216     // alert(document.emailForm.Results.value);
217     document.emailForm.submit();
218 }
219
220 function padMail(strAddress, strSubject, strBody)
221 {
222     gstrEmailTo = strAddress;
223     gstrEmailSubject = strSubject;
224     var temp_str = strBody;
225     while(temp_str.indexOf("|") > -1)
226     {
227         temp_str = temp_str.substr(0,temp_str.indexOf("|")) + "\r" +
228         temp_str.substr(temp_str.indexOf("|") + 1)
229     }
230     gstrEmailBody = temp_str + "\r";
231 }
232
233 function writeDocument(s){document.write(s);}
234 // End -->
235
236 function vurderstoreogsmaa(form)
237 {
238     var reges =/[a-z]/; //[^A-Za-z0-9]
239     var regeb =/[A-Z]/; //[^A-Za-z0-9]
240     results= reges.test(form);
241     resultb= regeb.test(form);
242     if((results == true) && (resultb == true))
243     {
244         return true;
245     }
246     else
247     {
248         return false;
249     }
250 }
251
252 function vurderspecialchars(form)
253 {
254     // se om en af tegnene er ikke A-Z eller a-z
255     var reges =/[^A-Za-z]/; //[^A-Za-z0-9]
256     result= reges.test(form);
257     return result;
258 }
259
260
261
262 function vurderkodeord()
263 {
264     document.form.bigsmall.value = "";
265     document.form.specialnumber.value = "";
```

```
266 document.form.points.value = "";
267
268 var kodeord = document.form.pass.value;
269 var point = 0;
270
271 if(kodeord.length > 0)
272 {
273     // Længde af kodeordet..
274     var temp = ( kodeord.length * 15 ) - 50
275     if( temp > 0 )
276     {
277         if( temp < 70 )
278         { point = temp; }
279         else
280         { point = 70; }
281     }
282
283     // Er der både små og store bogstaver?..
284
285     if(vurderstoreogsmå(kodeord))
286     {
287         point = point + 15;
288         document.form.bigsmall.value = "true";
289     }
290     else
291     {
292         document.form.bigsmall.value = "false";
293     }
294
295     if(vurderspecial(kodeord))
296     {
297         point = point + 15;
298         document.form.specialnumber.value = "true";
299     }
300     else
301     {
302         document.form.specialnumber.value = "false";
303     }
304
305     if(point < 0) point = 0;
306     if(point > 100) point = 100;
307
308     document.form.points.value = point;
309     document.form.length.value = kodeord.length;
310
311     //alert(point +
312     //     " points, "+
313     //     vurderpecial(kodeord) +
314     //     " specialtal, " +
315     //     vurderstoreogsmå(kodeord)+" storesmå.");
316
317     return true;
318 }
319 else
320 {
321     alert("Du skal skrive et kodeord i feltet");
322 }
323
324 }
325
326 function validateinetintro()
327 {
328     if(((document.formp.radPhishing[0].checked == true) ||
329         (document.formp.radPhishing[1].checked == true) ||
330         (document.formp.radPhishing[2].checked == true) ||
```

```
331     (document.formp.radPhishing[3].checked == true) ||
332     (document.formp.radPhishing[4].checked == true)) &&
333     ((document.formp.radPopupsafe[0].checked == true) ||
334     (document.formp.radPopupsafe[1].checked == true) ||
335     (document.formp.radPopupsafe[2].checked == true) ))
336     {
337
338         return true;
339     }
340     else
341     {
342         alert("Du skal vælge et punkt i hvert spørgsmål.");
343         return false;
344     }
345
346 }
347
348
```

## A.6.6 style.css

```
1 body {
2     margin:50px 0px; padding:0px;
3     text-align:center;
4     font-family: Verdana;
5     font-size: 11px;
6 }
7
8 #content {
9     margin:0px auto;
10    text-align:left;
11 }
12
13 #menu {
14     margin: 0px; padding: 0px;
15     width: 770px; height:15px;
16     border-bottom: 1px solid #cccccc;
17 }
18
19 #menu ul {
20     list-style: none;
21     margin: 0px; padding: 0px;
22     border: 1px;
23 }
24
25 #menu ul li {
26     float: left;
27     margin-left: 0px;
28     padding-left: 0px; padding-bottom: 2px;
29 }
30 #menu #menur {
31     float:right;
32     padding-bottom: 2px;
33 }
34
35
36 #testlist {
37     padding-left: 20px; margin-left: 4px;
38     border-left: 1px solid black;
39 }
40
41 #testlist li {
42     padding-bottom: 4px; margin-left: 4px;
43 }
```



```
44
45 a {
46   text-decoration: none;
47   color: black;
48 }
49 a:hover {
50   text-decoration: underline;
51   color: black;
52 }
53 a:visited {
54   color: black;
55 }
56
57 #txi {
58   font-family: Verdana;
59   font-size: 11px;
60   font-weight: bold;
61   border: 1px solid white;
62   text-align: left;
63 }
64
65 #tx {
66   border: 1px solid white;
67   font-size: 11px;
68   font-family: verdana;
69   overflow: hidden;
70   text-align: left;
71 }
72
73 .textfield {
74   margin-left: 1px;
75 }
76 .button {
77   margin: 0px; padding: 0px;
78   outline-style: none;
79 }
80
81 table td h1 {
82   font-family: Verdana;
83   font-size: 15pt;
84   color: #003466;
85 }
86
87 table td h2 {
88   font-family: Verdana;
89   font-size: 10pt; font-weight: bold;
90   padding-bottom: 0px; margin-bottom: 0px;
91 }
92
93 table td h3 {
94   font-family: Verdana;
95   font-weight: normal;
96   padding-top: 15px; padding-left: 10px;
97 }
98
99 .svar {
100   background-color: white;
101 }
102
103 b {
104   font-size: 12px;
105 }
106
107 a.info {
108   position: relative; /*this is the key*/
```

```

109     z-index:24; padding-bottom: 0px;
110     line-height: 15px;
111     color:#000;
112     text-decoration: none;
113     border-bottom: 1px dashed #b3c3d4;
114     }
115
116     a.info:hover{z-index:25; }
117
118     a.info span{display: none}
119
120     a.info:hover span{ /*the span will display just on :hover state*/
121         line-height: 18px;
122         display:block; padding-left: 4px; padding-right: 4px;
123         position:absolute; width: 15em;
124         left: 5em; top: 1.5em;
125         border:1px solid;
126         background-color: #cbcbcb;
127         text-align: center
128     }

```

## A.6.7 start.php

```

1 <?php $restricted = true; $start = true; include("header.php"); require_once("classes.php");?>
2
3 <div style="width: 80%; text-align: left; padding: 13px;">
4 Velkommen, <?php echo getusername($_SESSION['user']); ?><br />
5
6 <p style="width: 650px;" onmouseover="settext('0')">
7 Fra denne side kan du vælge og prøve forskellige animerede spørgeskemaer, designet til at
8 teste din viden om forskellige aspekter af computersikkerhed. I slutningen af næsten alle
9 spørgeskemaerne vil emnet blive forklaret som en gennemgang af spørgsmålene og dine svar sådan
10 at du lærer hvordan du kan sikre dig på din computer.
11 </p><br /><br />
12 <table cellpadding="0" cellspacing="0" >
13 <tr>
14     <td width="60%" valign="top" onmouseover="settext('0')">
15         <form name="form" action="">
16             <input id="txi" type="text" value="" readonly = "readonly" /><br />
17             <textarea id="tx" rows="10" cols="55" readonly = "readonly"></textarea>
18         </form>
19     </td>
20     <td width="40%" valign="top" align="left" style="padding-left:85px;" >
21         <b style="font-size: 11px;">Test dig selv:</b><br />
22         <ul id="testlist">
23             <?php
24                 for($i=2; $i <= countgettesttext(); $i++)
25                 {
26                     $result = gettesttext($i);
27                     foreach($result as $user_object)
28                     {
29                         echo "<li><a class=\"info\" href=\"\".$user_object->url.\"\">\n";
30                         echo "onmouseover=\"settext('\".(\".$i).\"')\" >\n";
31                         $user_object->name."</a>";
32                         if(gettestfinished(($i)))
33                         {
34                             echo " <img src=\"images/check.jpg\" alt=\"\" ".
35                                 "title=\"Du har allerede kørt denne test.\" />\n";
36                         }
37                         echo "</li>";
38                     }
39                 }
40             ?>
41

```

```

42
43
44     </ul>
45 </td>
46 </tr>
47 </table>
48 </div>
49
50 <?php include("footer.php"); ?>

```

### A.6.8 opretbruger.php

```

1 <?php $page = "i"; $logind = "1"; include("header.php"); ?>
2
3 <div style="width: 80%; text-align: left; padding: 13px;">
4
5 <form name="frmLogin" action="opretbrugerh.php" method="post"
6     onsubmit="return validateopretbruger();">
7 <table border="0" align="center">
8 <tr>
9 <td></td>
10 <td colspan="2"><b>Opret bruger</b></td>
11 </tr>
12 <tr>
13 <td align="right">Brugernavn:</td>
14 <td colspan="2">
15 <input class="text" name="txtUsername" type="text" size="33" maxlength="34" />
16 </td>
17 </tr>
18 <tr>
19 <td align="right">Kodeord:</td>
20 <td colspan="2">
21 <input class="text" name="txtPassword" type="password" size="33" maxlength="345" />
22 </td>
23 </tr>
24 <tr>
25 <td align="right">Gentag kodeord:</td>
26 <td colspan="2">
27 <input class="text" name="txtPassword2" type="password" size="33" maxlength="345" />
28 </td>
29 </tr>
30 <tr style="height:10px;"><td></td></tr>
31 <tr>
32 <td align="right">Fuldt navn:</td>
33 <td colspan="2">
34 <input class="text" name="txtName" type="text" size="33" maxlength="70" />
35 </td>
36 </tr>
37 <tr>
38 <td align="right">Alder:</td>
39 <td colspan="2">
40 <input class="text" name="txtAge" type="text" size="33" maxlength="2" />
41 </td>
42 </tr>
43 <tr style="height:3px;"><td></td></tr>
44 <tr style="margin-bottom:4px;">
45 <td align="right">Køn:</td>
46 <td colspan="2">
47 <label><input name="radGender" type="radio" value="M" />Mand</label>
48 <label><input name="radGender" type="radio"
49     style="margin-left: 4px;" value="F" />Kvinde</label>
50 </td>
51 </tr>
52 <tr style="height:3px;"><td></td></tr>

```

```

53 <tr>
54 <td align="right">Computerfærdighed:</td>
55 <td colspan="2">
56 <label><input name="radComputerskill" type="radio" value="0" />
57 <a class="info" href="#">Uerfaren<span>
58 Benytter ikke en computer ofte.
59 </span></a></label>
60 <label><input name="radComputerskill" type="radio" value="1" />
61 <a class="info" href="#">Erfaren<span>
62 Benytter en computer regelmæssigt og surfer på internettet.
63 </span></a></label>
64 <label><input name="radComputerskill" type="radio" value="2" />
65 <a class="info" href="#">Ekspert<span>
66 Superbruger. Muligvis programmeringserfaring eller andet.
67 </span></a></label>
68 </td>
69 </tr>
70 <tr style="height:10px;"><td></td></tr>
71 <tr>
72 <td></td>
73 <td><input type="submit" class="button" value="Send" style="width:112px;" /></td>
74 </tr>
75 </table>
76 <input type="hidden" name="bversion" value="" />
77 </form>
78 <br /><br /><br />
79 </div>
80 <?php include("footer.php"); ?>

```

## A.6.9 Phishing internet.php

```

1 <?php $restricted = true; $notstart = "1"; include("../header.php"); ?>
2 <br />
3 <form name="emailForm" action="inethandler.php" method="POST">
4 <input name="Results" type="hidden">
5 <input name="phishing" type="hidden" value="<?php echo $_POST['radPhishing']; ?>">
6 <input name="popup" type="hidden" value="<?php echo $_POST['radPopupsafe']; ?>">
7 </form>
8 <div style="padding: 0px; width:790px; border: 1px solid #4286b0;">
9 <script type="text/javascript">
10 <!--
11 writeDocument(' <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000" ' +
12 ' codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,0,0" ' +
13 ' width="790" height="545" ID="Captivatel">'+
14 ' <param name="movie" value="phishingweb.swf">'+
15 ' <param name="quality" value="high">'+
16 ' <param name="menu" value="false">'+
17 ' <param name="loop" value="0">'+
18 ' <embed src="phishingweb.swf" width="790" height="545" loop="0" quality="high" ' +
19 ' pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash" ' +
20 ' menu="false"></embed>'+
21 ' </object>');
22 //-->
23 </script>
24 <noscript>
25 <object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000" ' +
26 ' codebase="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#'+
27 ' version=7,0,0,0" width="790" height="545" ID="Captivatel">
28 <param name="movie" value="phishingweb.swf">
29 <param name="quality" value="high">
30 <param name="menu" value="false">
31 <param name="loop" value="0">
32 <embed src="phishingweb.swf" width="790" height="545" loop="0" quality="high" ' +

```

```

33     'pluginspage="http://www.macromedia.com/go/getflashplayer" '+
34     'type="application/x-shockwave-flash" menu="false"></embed>
35 </object>
36 </noscript>
37 </div>
38 <?php include("../footer.php"); ?>

```

### A.6.10 Phishing inethandler.php

```

1 <?php $restricted = true; $notstart = "1"; include("../header.php");require_once '../classes.php';?>
2
3 <div style="width: 790px; margin:0px; padding:0px;" >
4 <?php
5     function striptostings( $result )
6     {
7         // strip the start info
8
9         //$result = strstr(stripslashes($result), "\",\"Student Response\", \"Result\", \"Weight\", \"Latency\"");
10        \"/>

```

```

56 $string = substr($string, $num+1);
57 $num = strpos($string, "\",\""); // 7 (last one we need)
58 $arr[1] = substr($string, 2, $num-2);
59 $string = substr($string, $num+1);
60
61 if($print == 1)
62 {
63 if($arr[1] != 1)
64 echo $arr[0].".".$arr[1]."<br />";
65 else
66 echo $arr[0]."<br />";
67 }
68 if($arr[1] != 1)
69 return $arr[0].".".$arr[1];
70 else
71 return $arr[0];
72
73 }
74
75 $result = nl2br($_POST['Results']);
76 //echo $result;
77 //echo "<br />Her: <br />";
78
79 $a = striptostings( $result );
80
81 stripastring( $a[0], 0 );
82 stripastring( $a[1], 0 );
83 stripastring( $a[2], 0 );
84 stripastring( $a[3], 0 );
85 stripastring( $a[4], 0 );
86
87 //echo "user:". $_SESSION['user'];
88 // Læg alt op i databasen
89 if(gettestfinished(2) == false)
90 {
91
92 //echo "lagt op";
93 mysql_connect("localhost", "tre_ben_dk", "kablooie") or die(mysql_error());
94 mysql_select_db("tre_ben_dk") or die(mysql_error());
95 $result = mysql_query("INSERT INTO sik_testdata (id, person_id, test_id ,data1, data2, ".
96 "data3, data4, data5, data29, data32) VALUES (".
97 "'' ".
98 ",'"$_SESSION['user']"."' ".
99 "','2' ".
100 ",'"stripastring( $a[0], 0 )."' ".
101 ",'"stripastring( $a[1], 0 )."' ".
102 ",'"stripastring( $a[2], 0 )."' ".
103 ",'"stripastring( $a[3], 0 )."' ".
104 ",'"stripastring( $a[4], 0 )."' ".
105 ",'"$_POST['Results']"."' ".
106 "','OK' ".
107 ");") or die(mysql_error());
108
109 $id = mysql_insert_id();
110
111 mysql_close();
112 //echo "id: ".$id;
113 mysql_connect("localhost", "tre_ben_dk", "kablooie") or die(mysql_error());
114 mysql_select_db("tre_ben_dk") or die(mysql_error());
115 $result = mysql_query("UPDATE sik_testdata SET ".
116 "data7 = '".$_POST['phishing']"."', ".
117 "data8 = '".$_POST['popup']"."', ".
118 "data30 = '".getphishinginetscore($_SESSION['user'])."' ".
119 "WHERE id = ".$id.";") or die(mysql_error());
120 mysql_close();

```

```

121     }
122 }
123
124
125 ?>
126 <table id="content" width="790" border="0" cellpadding="0" cellspacing="0" style="margin:0px;" >
127 <tr>
128 <td align="center"><h1>Svar fra test</h1></td>
129 </tr>
130 <tr>
131 <td align="right"><a href="../testslut.php?id=<?php echo $id; ?>">Tilbage til startside</a></td>
132 </tr>
133 <tr>
134 <td>
135 <?php echo "<!--".stripastring( $a[0], 0 ).
136         "-".stripastring( $a[1], 0 ).
137         "-".stripastring( $a[2], 0 ).
138         "-".stripastring( $a[3], 0 ).
139         "-".stripastring( $a[4], 0 )."-- >\n"; ?>
140 <!-- Start Svarblok ----->
141 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0"
142       style="border: 1px solid #4286b0;">
143 <tr style="height: 49px; background-color: #70bded;">
144 <td colspan="2"><h3>Spørgsmål 1 ud af 5 - Danske netbank</h3></td>
145 </tr>
146 <tr>
147 <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
148 <div style="font-weight: bold; font-color: green;">
149 <?php
150 $r = substr(stripastring( $a[0], 0 ), 2, 1);
151 $r2 = substr(stripastring( $a[0], 0 ), 3);
152 if( ($r == "N") &&
153     (substr_count($r2, "A") > 0) &&
154     (substr_count($r2, "B") == 0) &&
155     (substr_count($r2, "C") == 0) &&
156     (substr_count($r2, "E") == 0) )
157 { echo "Korrekt!"; }
158 else
159 { echo "Forkert!"; }
160
161 ?>
162 </div>
163 <br />
164 <?php
165 echo "De svarede: ";
166 if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
167 if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
168 if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
169 ?>
170 <div style="font-size: 8pt; padding-left: 10px;">
171 <?php
172 if(substr_count($r2, "A") > 0) { echo "- <font style=\"color:green\">Adressen var forkert".
173   " </font><br />"; }
174 if(substr_count($r2, "B") > 0) { echo "- <font style=\"text-decoration: underline; ".
175   "color:red\">Der manglede 'https' i adressen</font><br />"; }
176 if(substr_count($r2, "C") > 0) { echo "- <font style=\"text-decoration: underline; ".
177   "color:red\">Der manglede en lås</font><br />"; }
178 if(substr_count($r2, "D") > 0) { echo "- <font style=\"color:green\">Siden virkede ".
179   "forkert</font><br />"; }
180 if(substr_count($r2, "E") > 0) { echo "- <font style=\"text-decoration: underline; ".
181   "color:red\">Ved ikke</font><br />"; }
182 ?>
183 </div>
184 </td>
185 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">

```

```

186     
187     </td>
188 </tr>
189 <tr>
190     <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px;
191         padding-bottom: 10px; font-size: 13px;">
192         <br />
193         Adressen på denne side viser en ip-adresse istedet for Danske NetBanks egen adresse. <br />
194         <br />
195         Pga. dette kan man ikke være sikker på om den viste side virkelig er Danske NetBank<br />
196         og det frarådes derfor at stole på denne side og indtaste nogen form for brugerinformation
197         på den.
198     </td>
199 </tr>
200 </table>
201 <!-- Slut Svarblok ----->
202 </td>
203 </tr>
204 <tr style="padding-top: 50px;">
205     <td align="right"><a href=" ../testslut.php?id=<?php echo $id; ?>">Tilbage til startside</a></td>
206 </tr>
207 <tr>
208 <td>
209 <!-- Start Svarblok ----->
210     <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0"
211         style="border: 1px solid #4286b0;">
212     <tr style="height: 49px; background-color: #70bded;">
213         <td colspan="2"><h3>Spørgsmål 2 ud af 5 - Tastselv Skat</h3></td>
214     </tr>
215     <tr>
216         <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
217         <div style="font-weight: bold; font-color: green;">
218             <?php
219                 $r = substr(stripastring( $a[1], 0 ), 2, 1);
220                 $r2 = substr(stripastring( $a[1], 0 ), 3);
221                 if( ($r == "N") &&
222                     (substr_count($r2, "A") == 0) &&
223                     (substr_count($r2, "B") == 0) &&
224                     (substr_count($r2, "C") > 0) &&
225                     (substr_count($r2, "E") == 0))
226                 { echo "Korrekt!"; }
227                 else
228                 { echo "Forkert!"; }
229             >
230         </div>
231     <br />
232     <?php
233         echo "De svarede: ";
234         if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
235         if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
236         if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
237     >
238     <div style="font-size: 8pt; padding-left: 10px;">
239     <?php
240         if(substr_count($r2, "A") > 0) { echo "- <font style=\"text-decoration: underline; ".
241             "color:red;\>Adressen var forkert</font><br />"; }
242         if(substr_count($r2, "B") > 0) { echo "- <font style=\"text-decoration: underline; ".
243             "color:red;\>Der manglede 'https' i adressen</font><br />"; }
244         if(substr_count($r2, "C") > 0) { echo "- <font style=\"color:green;\>Der manglede en ".
245             "lås</font><br />"; }
246         if(substr_count($r2, "D") > 0) { echo "- <font style=\"color:green;\>Siden virkede ".
247             "forkert</font><br />"; }
248         if(substr_count($r2, "E") > 0) { echo "- <font style=\"text-decoration: underline; ".
249             "color:red;\>Ved ikke</font><br />"; }
250     >

```



```

251     ?>
252     </div>
253 </td>
254 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
255     
256 </td>
257 </tr>
258 <tr>
259     <td align="left" colspan="2" style="padding-left: 15px; padding-right: 100px;
260     padding-bottom: 10px; font-size: 13px;">
261     <br />
262     Denne sides adresse er rigtig nok men låsen mangler.<br /><br />
263     Den lukkede lås sammen med 'https' til forskel for det normale 'http' istarten
264     af adressen, viser at websiden har certifikat og kan sende data sikkert mellem dig og
265     siden. <br />
266     Da låsen ikke er tilstede frarådes det at stole på denne side og indtaste nogen form for
267     brugerinformation på den.<br />
268     <br />
269     Bemærk: På nogle browsere vises låseikonet ikke oppe vedsiden af adressen, men nede på
270     baren under websiden.
271 </td>
272 </tr>
273 </table>
274 <!-- Slut Svarblok ----->
275 </td>
276 </tr>
277 <tr style="padding-top: 50px;">
278     <td align="right"><a href=" ../testslut.php?id=<?php echo $id; ?>">Tilbage til startside</a></td>
279 </tr>
280 <tr>
281 <td>
282 <!-- Start Svarblok ----->
283 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0"
284     style="border: 1px solid #4286b0;">
285     <tr style="height: 49px; background-color: #70bde4;">
286         <td colspan="2"><h3>Spørgsmål 3 ud af 5 - Paypal</h3></td>
287     </tr>
288     <tr>
289         <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
290             <div style="font-weight: bold; font-color: green;">
291                 <?php
292                 $r = substr(stripastring( $a[2], 0 ), 2, 1);
293                 $r2 = substr(stripastring( $a[2], 0 ), 3);
294                 if( ($r == "Y"))
295                 { echo "Korrekt!"; }
296                 else
297                 { echo "Forkert!"; }
298
299                 ?>
300             </div>
301             <br />
302             <?php
303             echo "De svarede: ";
304             if($r == "Y") { echo "<font style=\"color:green\">Ja</font><br />"; }
305             if($r == "N") { echo "<font style=\"color:red\">Nej</font><br />"; }
306             if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
307             ?>
308             <div style="font-size: 8pt; padding-left: 10px;">
309                 <?php
310                 if(substr_count($r2, "A") > 0) { echo "- <font style=\"text-decoration: underline; ".
311                 "color:red\">Adressen var forkert</font><br />"; }
312                 if(substr_count($r2, "B") > 0) { echo "- <font style=\"text-decoration: underline; ".
313                 "color:red\">Der manglede 'https' i adressen</font><br />"; }
314                 if(substr_count($r2, "C") > 0) { echo "- <font style=\"text-decoration: underline; ".
315                 "color:red\">Der manglede en lås</font><br />"; }

```

```

316     if(substr_count($r2, "D") > 0) { echo "- <font style=\"text-decoration: underline; ".
317         "color:red;\">Siden virkede forkert</font><br />"; }
318     if(substr_count($r2, "E") > 0) { echo "- <font style=\"text-decoration: underline; ".
319         "color:red;\">Ved ikke</font><br />"; }
320     ?>
321 </div>
322 </td>
323 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
324 
325 </td>
326 </tr>
327 <tr>
328 <td align="left" colspan="2" style="padding-left: 15px; padding-right: 100px;
329     padding-bottom: 10px; font-size: 13px;">
330 <br />
331 Denne side er ikke phished.<br /><br />
332 Siden har 'https' og en lås sådan at vi ved der kan sendes data sikkert mellem os og siden. <br />
333 Derudover kan vi se at adressen passer.<br />
334 Da disse tre ting er opfyldt er der ingen grund til ikke at stole på siden.
335 </td>
336 </tr>
337 </table>
338 <!-- Slut Svarblok ----->
339 </td>
340 </tr>
341 <tr style="padding-top: 50px;">
342 <td align="right"><a href=" ../testslut.php?id=<?php echo $id; ?>">Tilbage til startside</a></td>
343 </tr>
344 <tr>
345 <td>
346 <!-- Start Svarblok ----->
347 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0"
348     style="border: 1px solid #4286b0;">
349 <tr style="height: 49px; background-color: #70bded;">
350 <td colspan="2"><h3>Spørgsmål 4 ud af 5 - Apple store</h3></td>
351 </tr>
352 <tr>
353 <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
354 <div style="font-weight: bold; font-color: green;">
355 <?php
356     $r = substr(stripastring( $a[3], 0 ), 2, 1);
357     $r2 = substr(stripastring( $a[3], 0 ), 3);
358     if( ($r == "N") &&
359         (substr_count($r2, "A") == 0) &&
360         ((substr_count($r2, "B") > 0) || (substr_count($r2, "C") > 0)) &&
361         (substr_count($r2, "E") == 0))
362     { echo "Korrekt!"; }
363     else
364     { echo "Forkert!"; }
365
366     ?>
367 </div>
368 <br />
369 <?php
370     echo "De svarede: ";
371     if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
372     if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }
373     if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
374     ?>
375 <div style="font-size: 8pt; padding-left: 10px;">
376 <?php
377     if(substr_count($r2, "A") > 0) { echo "- <font style=\"text-decoration: underline; ".
378         "color:red;\">Adressen var forkert</font><br />"; }
379     if(substr_count($r2, "B") > 0) { echo "- <font style=\"color:green;\">Der manglede ".

```

```

381     "'https' i adressen</font><br />"; }
382     if(substr_count($r2, "C") > 0) { echo "- <font style=\"color:green;\">Der manglede en ".
383         "lås</font><br />"; }
384     if(substr_count($r2, "D") > 0) { echo "- <font style=\"color:green;\">Siden virkede ".
385         "forkert</font><br />"; }
386     if(substr_count($r2, "E") > 0) { echo "- <font style=\"text-decoration: underline; ".
387         "color:red;\">Ved ikke</font><br />"; }
388     ?>
389 </div>
390 </td>
391 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
392 
393 </td>
394 </tr>
395 <tr>
396 <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px;
397     padding-bottom: 10px; font-size: 13px;">
398 <br />
399 Denne sides adresse er rigtig nok, men låsen mangler sammen med 'https'.<br /><br />
400 Låse eller andre ikoner på selve websiden, holder ingen garantier for at siden
401 virkelig er sikker.<br />
402 Man bør kun stole på browserens egne indikatorer.
403 <br />
404 Da låsen og 'https' ikke er tilstede, frarådes det at stole på denne side og indtaste
405 nogen form for brugerinformation på den.<br />
406 <br />
407 </td>
408 </tr>
409 </table>
410 <!-- Slut Svarblok ----->
411 </td>
412 </tr>
413 <tr style="padding-top: 50px;">
414 <td align="right"><a href=" ../testslut.php?id=<?php echo $id; ?>">Tilbage til startsiden</a></td>
415 </tr>
416 <tr>
417 <td>
418 <!-- Start Svarblok ----->
419 <table class="svar" width="790" border="0" cellpadding="0" cellspacing="0"
420     style="border: 1px solid #4286b0;">
421 <tr style="height: 49px; background-color: #70bde2;">
422 <td colspan="2"><h3>Spørgsmål 5 ud af 5 - Jyske bank</h3></td>
423 </tr>
424 <tr>
425 <td style="width: 280px; padding-left: 15px; padding-top: 10px;">
426 <div style="font-weight: bold; font-color: green;">
427 <?php
428     $r = substr(stripastring( $a[4], 0 ), 2, 1);
429     $r2 = substr(stripastring( $a[4], 0 ), 3);
430     if( ($r == "N") &&
431         (substr_count($r2, "A") > 0) &&
432         (substr_count($r2, "B") == 0) &&
433         (substr_count($r2, "C") == 0) &&
434         (substr_count($r2, "E") == 0))
435     { echo "Korrekt!"; }
436     else
437     { echo "Forkert!"; }
438
439     ?>
440 </div>
441 <br />
442 <?php
443     echo "De svarede: ";
444     if($r == "Y") { echo "<font style=\"color:red\">Ja</font><br />"; }
445     if($r == "N") { echo "<font style=\"color:green\">Nej</font><br />"; }

```

```

446     if($r == "D") { echo "<font style=\"color:red\">Ved Ikke</font><br />"; }
447     ?>
448     <div style="font-size: 8pt; padding-left: 10px;">
449     <?php
450     if(substr_count($r2, "A") > 0) { echo "- <font style=\"color:green;\">Adressen var ".
451     "forkert</font><br />"; }
452     if(substr_count($r2, "B") > 0) { echo "- <font style=\"text-decoration: underline;".
453     " color:red;\">Der manglede 'https' i adressen</font><br />"; }
454     if(substr_count($r2, "C") > 0) { echo "- <font style=\"text-decoration: underline; ".
455     "color:red;\">Der manglede en lås</font><br />"; }
456     if(substr_count($r2, "D") > 0) { echo "- <font style=\"color:green;\">Siden virkede ".
457     "forkert</font><br />"; }
458     if(substr_count($r2, "E") > 0) { echo "- <font style=\"text-decoration: underline; ".
459     "color:red;\">Ved ikke</font><br />"; }
460     ?>
461     </div>
462 </td>
463 <td align="right" valign="top" style="padding-top: 10px; padding-right: 10px;">
464 
465 </td>
466 </tr>
467 <tr>
468 <td align="left" colspan="2" style="padding-left: 15px; padding-right:100px;
469 padding-bottom: 10px; font-size: 13px;">
470 <br />
471 Adressen på denne side er forkert. Bemærk at det sidste 'k' er udskiftet til et
472 'c'.<br /><br />
473 Så små ændringer som dette kan gøre at siden man ser, rent faktisk er en helt anden.
474 Derfor bør man
475 altid kigge efter når man følger et link (især fra en email) om den peger det rigtige
476 sted hen.<br />
477 <br />
478 Da vi ikke kan være sikre på at denne side virkelig er jyske banks kan det ikke tilrådes
479 at stole på den eller indtaste nogen
480 form for følsomme oplysninger på den.
481 </td>
482 </tr>
483 </table>
484 <!-- Slut Svarblok ----->
485 </td>
486 </tr>
487 </table>
488 </div>
489 <?php include("../footer.php"); ?>

```

## A.6.11 Phishing internetintro.php

```

1 <?php $restricted = true; $notstart = "1"; include("../header.php"); ?>
2 <br />
3 <form name="formp" action="internet.php" method="POST" onsubmit="return validateinetintro();">
4
5 <div style="padding: 0px; width:400px; text-align: left;">
6 <h1>Phishing</h1>
7 <table border="0" cellpadding="2" cellspacing="2">
8 <tr><td colspan="2"><h2>Hvad er phishing på internettet?</h2></td></tr>
9 <tr>
10 <td valign="top"><input name="radPhishing" type="radio" value="1" /></td>
11 <td>Internetsider der forsøger at give dig virus, eller andre ubehageligheder.</td>
12 </tr>
13 <tr>
14 <td valign="top"><input name="radPhishing" type="radio" value="2" /></td>
15 <td>
16 Internetsider der giver sig ud for at være en andre sider for at lokke personlige
17 oplysninger, så som kodeord eller kreditkort oplysninger, ud af dig.

```

```

18     </td>
19 </tr>
20 <tr>
21 <td valign="top"><input name="radPhishing" type="radio" value="3" /></td>
22 <td>
23     Internetsider der giver sig ud for at være en andre sider for at overtage din computer og
24     benytte den til at udføre kriminelle handlinger.
25 </td>
26 </tr>
27 <tr>
28 <td valign="top"><input name="radPhishing" type="radio" value="4" /></td>
29 <td>
30     En metode hvorved firmaer benyttet internettet til at prøve at 'fiske' nye kunder til sig.
31 </td>
32 </tr>
33 <tr>
34 <td valign="top"><input name="radPhishing" type="radio" value="D" /></td>
35 <td>Ved ikke</td>
36 </tr>
37 <tr><td>&nbsp;</td></tr>
38 <tr>
39 <td colspan="2">
40 <h2>Det er sikkert at indtaste personlige oplysninger
41     eller bankoplysninger i et pop-up-vindue</h2>
42 </td>
43 </tr>
44 <tr>
45 <td valign="top"><input name="radPopupsafe" type="radio" value="Y" /></td>
46 <td>Rigtigt</td>
47 </tr>
48 <tr>
49 <td valign="top"><input name="radPopupsafe" type="radio" value="N" /></td>
50 <td>Forkert</td>
51 </tr>
52 <tr>
53 <td valign="top"><input name="radPopupsafe" type="radio" value="D" /></td>
54 <td>Ved ikke</td>
55 </tr>
56 <tr><td>&nbsp;</td></tr>
57 <tr>
58 <td colspan="2"><input type="submit" value="Videre" /></td>
59
60 </tr>
61
62 </table>
63 </div>
64 </form>
65 <?php include("../footer.php"); ?>

```

### A.6.12 Statistik statistik.php

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3 <html>
4 <head>
5 <title>Sikkerhed - Statistik</title>
6 <meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
7 <script src="http://sikkerhed.tre-ben.dk/statistik/statscript.js" type="text/javascript"></script>
8 <script type="text/javascript">
9
10 </script>
11
12 <style type="text/css">
13 body, body table tr td {

```

```
14 font-family: Verdana;
15 font-size: 9pt;
16
17 }
18
19 h1 {
20 font-family: Verdana;
21 font-size: 12pt;
22 }
23 h2 {
24 font-family: Verdana;
25 font-size: 10pt;
26 display: inline;
27 margin-left: 4px;
28 }
29
30 a.info {
31 position:relative; /*this is the key*/
32 z-index:24; padding-bottom: 0px;
33 line-height: 15px;
34 color:#000;
35 text-decoration: none;
36 border-bottom: 1px dashed #b3c3d4;
37 }
38
39 a.info:hover{z-index:25; }
40
41 a.info span{display: none}
42
43 a.info:hover span{ /*the span will display just on :hover state*/
44 line-height: 18px;
45 display:block; padding-left: 4px; padding-right: 4px;
46 position:absolute; width: 15em;
47 left: 5em; top: 1.5em;
48 border:1px solid;
49 background-color: #cbcbcb;
50 text-align: center
51 }
52 </style>
53 </head>
54 <body onload="ajaxManager('start_up');">
55
56 <h1>Statistik for sikkerhed.tre-ben.dk</h1>
57
58 <!-- KRITERIER -->
59
60 <form name="form" action="#" method="post">
61 <table width="800" border="0" cellpadding="1" cellspacing="1" style="background-color: #dce5ee;">
62 <tr style="background-color: #bdcbda">
63 <td width="180"><h2>Kriterier</h2></td><td colspan="2" width="500"><br /></td>
64 <td rowspan="6" width="100" align="center"><div id="loadfield">Done</div></td>
65 </tr>
66 <tr>
67 <td align="right" style="padding-right: 5px">Alder:</td>
68 <td colspan="2" style="padding-left: 5px">
69 <input name="age_from" type="text" maxlength="2" value="0" onkeyup="return validate(this);"
70 style="margin-left: 4px; width: 22px; background-color: #fcfcfd; border: 1px solid #bdcbda" />
71 til
72 <input name="age_to" type="text" maxlength="2" value="99" onkeyup="return validate(this);"
73 style="margin-left: 4px; width: 22px; background-color: #fcfcfd; border: 1px solid #bdcbda" />
74 <div id="agefault" style="display: inline; color: #ff0000;"></div>
75 </td>
76 </tr>
77 <tr style="height:3px;"><td></td></tr>
78 <tr style="margin-bottom:4px;">
```

```

79     <td align="right" style="padding-right: 5px">Køn:</td>
80     <td colspan="2" style="padding-left: 5px">
81         <label><input name="gender_male" type="checkbox" checked="checked"
82             onmouseup="return validate(this);" />Mand</label>
83         <label><input name="gender_female" type="checkbox" checked="checked"
84             onmouseup="return validate(this);" />Kvinde</label>
85     </td>
86 </tr>
87 <tr style="height:3px;"><td><input name="nocache" type="hidden" /></td></tr>
88 <tr>
89     <td align="right" style="padding-right: 5px">Computerfærdighed:</td>
90     <td colspan="2" style="padding-left: 5px">
91         <label><input name="skill_none" type="checkbox" checked="checked"
92             onmouseup="return validate(this);" />
93         <a class="info" href="#">Uerfaren<span>
94             Benytter ikke en computer regelmæssigt.
95         </span></a></label>
96         <label><input name="skill_experienced" type="checkbox" checked="checked"
97             onmouseup="return validate(this);" /><a class="info" href="#">Erfaren<span>
98             Benytter en computer regelmæssigt og surfer på internettet.
99         </span></a></label>
100        <label><input name="skill_expert" type="checkbox" checked="checked"
101            onmouseup="return validate(this);" />
102        <a class="info" href="#">Ekspert<span>
103            Superbruger. Programmeringserfaring.
104        </span></a></label>
105    </td>
106
107
108 </tr>
109 </table>
110 </form>
111 <br />
112 <!-- STATISTIK - VHA. AJAX -->
113 <div id="statdata" />
114 </body>
115 </html>

```

### A.6.13 Statistik statdata.php

```

1 <?php
2 // slå caching fra så denne bliver opdateret hver gang
3 header( "Expires: Mon, 20 Dec 1998 01:00:00 GMT" );
4 header( "Last-Modified: " . gmdate("D, d M Y H:i:s") . " GMT" );
5 header( "Cache-Control: no-cache, must-revalidate" );
6 header( "Content-Type: text/html; charset=ISO-8859-1" );
7
8 header( "Pragma: no-cache" );
9
10 require_once '../classes.php';
11
12
13 // Lav where-clausen der skal bruges til databasen
14 $where = "WHERE age >= ".$_GET[age_from]. " AND ".
15     "age <= ".$_GET[age_to]. " AND ".
16
17     "( gender = ";
18     if($_GET['gender_male'] == "true")
19     { $where = $where."M' OR "; } else { $where = $where."' ' OR "; }
20
21     $where = $where."gender = ";
22     if($_GET['gender_female'] == "true")
23     { $where = $where."F'"; } else { $where = $where."' '"; }
24

```

```

25     $where = $where." AND (computerskill = ";
26     if($_GET[skill_none] == "true")
27     {$where = $where."0 OR "; } else {$where = $where."10 OR "; }
28
29
30     $where = $where."computerskill = ";
31     if($_GET[skill_experienced] == "true")
32     {$where = $where."1 OR "; } else {$where = $where."10 OR "; }
33
34     $where = $where."computerskill = ";
35     if($_GET[skill_expert] == "true")
36     {$where = $where."2 "; } else {$where = $where."10"; }
37
38     ?>
39     <br />
40     <table width="800" border="0" cellpadding="1" cellspacing="1">
41     <tr>
42     <td width="176" align="right" style="padding-right: 10px">Valgt statistik:</td>
43     <td style="padding: 0px; margin-left: 4px; padding-left: 5px;">
44     <?php
45     echo statpersonw($where). " / ". statperson()
46     ." personer i ".statstestdataw($where). " / ".statstestdata()
47     ." testrækker fra ". statstest(). " spørgeskemaer";
48     ?>
49     </td>
50     <td align="right">
51     <?php echo date('j/n-Y H:i:s'); ?>
52     </td>
53     </tr>
54     </table>
55     <hr style="width: 800px; padding: 0px; margin: 0px;
56     text-align: left; float: left; display:block;" /><br />
57
58     <table width="800" border="0" cellpadding="1" cellspacing="1" style="background-color: #dce5ee;">
59     <tr style="background-color: #dbcbda">
60     <td width="180"><h2>Adgangskoder</h2></td>
61     <td colspan="2" width="600" align="right">
62     <?php $total = statstestdatat($where, 5); echo $total." / ".statstestdatat("", 5); ?>
63     </td>
64     </tr>
65     <tr>
66     <td align="right" style="padding-right: 5px">
67     <a class="info" href="#">Point (avg):<span>0-35 : Dårligt sikret<br />35-71 : Mellem sikret
68     <br />71-100 : Godt sikret
69     </span></a>
70
71     </td>
72     <td style="padding-left: 5px">
73     <?php
74     if($total != 0)
75     { echo substr(statstest5($where, 1),0, 4); } else { echo " - "; }
76     ?>
77     </td></td>
78     </tr>
79     <tr style="height:3px;"><td></td></tr>
80     <tr>
81     <td align="right" style="padding-right: 5px">
82     <a class="info" href="#">Længde (avg):<span>Gennemsnits længde af indtastede kodeord
83     </span></a>
84
85     </td>
86     <td style="padding-left: 5px">
87     <?php
88     if($total != 0)
89     { echo substr(statstest5($where, 2),0,4). " tegn"; } else { echo " - tegn "; }
90     ?>

```



```

90     </td>
91     <td></td>
92 </tr>
93 <tr style="height:3px;"><td></td></tr>
94 <tr>
95     <td align="right" style="padding-right: 5px">
96         <a class="info" href="#">Længde > 7:<span>Procentdel af indtastede kodeord der er mindst 8
97             tegn lange.
98
99                                     </span></a>
100     </td>
101     <td style="padding-left: 5px">
102         <?php
103             if($total != 0)
104                 { echo substr(((statstest5($where, 3) / $total) * 100),0,4)." %"; } else { echo "- %"; }
105         </td>
106     <td align="right">Testen hjælp
107     <?php
108         if($total != 0)
109             { echo substr(((statstest5($where, 4) / $total) * 100),0,4)." % "; } else { echo "- %"; }
110         </td>
111 </tr>
112 </table>
113 <br />
114 <table border="1" style="background-color: #dce5ee;">
115 <tr style="background-color: #bdcbda">
116     <td width="180"><h2>Phishing - Web</h2></td>
117     <td colspan="2" width="600" align="right">
118         <?php $total = statstestdatat($where, 2); echo $total." / ".statstestdatat("", 2); ?>
119     </td>
120 </tr>
121 <tr>
122     <td align="right" style="padding-right: 5px">
123         <a class="info" href="#">Hvad er phishing?<span>Hvad er phishing? <br />Tekstspørgsmål
124             </span></a>
125     </td>
126     <td style="padding-left: 5px">
127         <table border="0" width="450">
128             <tr>
129                 <?php
130                 if($total != 0)
131                 {
132                     echo "<td width=\"100\">";
133                     echo "<a class=\"info\" href=\"#\>A: ".
134                         substr(((statstest2intro($where, 1, 1) / $total) * 100), 0, 3)." %, <span>".
135                         "Internetsider der forsøger at give dig virus,".
136                         "eller andre ubehageligheder.</span></a></td>";
137
138                     echo "<td width=\"100\">";
139                     echo "<a class=\"info\" href=\"#\>B: ".
140                         substr(((statstest2intro($where, 1, 2) / $total) * 100), 0, 3)." %, <span>".
141                         "(Korrekt)<br />Internetsider der giver sig ud for at være en andre sider for ".
142                         "at lokke personlige ".
143                         "oplysninger, så som kodeord eller kreditkort oplysninger, ud af dig.</span></a></td>";
144
145                     echo "<td width=\"100\">";
146                     echo "<a class=\"info\" href=\"#\>C: ".
147                         substr(((statstest2intro($where, 1, 3) / $total) * 100), 0, 3)." %, <span>".
148                         "Internetsider der giver sig ud for at være en andre sider for at overtage din computer ".
149                         "og benytte den til at udføre kriminelle handlinger.</span></a></td>";
150
151                     echo "<td width=\"100\">";
152                     echo "<a class=\"info\" href=\"#\>D: ".

```

```

155     substr(((statte2intro($where, 1, 4) / $total) * 100), 0, 3)." %, <span>".
156     "En metode hvorved firmaer benyttet internettet til at prøve at 'fiske' nye kunder til".
157     " sig.</span></a></td>";
158
159     echo "<td width=\"100\">";
160     echo "<a class=\"info\" href=\"#\">E: ".
161     substr(((statte2intro($where, 1, 'D') / $total) * 100), 0, 3)." %, <span>".
162     "Ved ikke</span></a></td>";
163
164
165
166     } else { echo "<td width=\"100\">A: - %,</td><td width=\"100\">B: - %,</td>".
167             "<td width=\"100\">C: - %,</td><td width=\"100\">D: - %,</td> ".
168             "<td width=\"100\">E: - %,</td>"; }
169
170     ?>
171     </tr>
172     </table>
173 </td></td>
174 </tr>
175 <tr><td style="height:3px;"</td></tr>
176 <tr>
177 <td align="right" style="padding-right: 5px">
178 <a class="info" href="#">Pop-up vindue, sikkert?<span>Det er sikkert at indtaste personlige
179 oplysninger eller bankoplysninger i et pop-up-vindue
180 </span></a>
181 </td>
182 <td style="padding-left: 5px">
183 <table border="0" width="270">
184 <tr>
185
186 <?php
187     if($total != 0)
188     {
189         echo "<td width=\"100\">";
190         echo "<a class=\"info\" href=\"#\">A: ".
191         substr(((statte2intro($where, 2, 'Y') / $total) * 100), 0, 3)." %, <span>".
192         "Rigtigt</span></a></td>";
193
194         echo "<td width=\"100\">";
195         echo "<a class=\"info\" href=\"#\">B: ".
196         substr(((statte2intro($where, 2, 'N') / $total) * 100), 0, 3)." %, <span>".
197         "(Korrekt)<br />Forkert</span></a></td>";
198
199         echo "<td width=\"100\">";
200         echo "<a class=\"info\" href=\"#\">C: ".
201         substr(((statte2intro($where, 2, 'D') / $total) * 100), 0, 3)." %, <span>".
202         "Ved ikke</span></a></td>";
203
204
205
206
207     } else { echo "<td width=\"100\">A: - %,</td><td width=\"100\">B: - %,</td>".
208             "<td width=\"100\">C: - %"; }
209
210     ?>
211     </tr>
212     </table>
213 </td></td>
214 </tr>
215 <tr><td style="height:15px;"></td></tr>
216 <tr>
217 <td align="right" style="padding-right: 5px">
218 <a class="info" href="#">1) Danskenetbank:<span>Forkert adresse (IP istedet for url)
219 </span></a>

```

```

220 </td>
221 <td style="padding-left: 5px">
222 <table border="0">
223 <tr>
224
225 <?php
226     if($total != 0)
227     {
228         echo "<td width=\"95\">Ja: ".substr(((statte2($where, 1, 1) / $total) * 100), 0, 3).
229             " %, </td>";
230         echo "<td width=\"95\">Nej: ".substr(((statte2($where, 1, 2) / $total) * 100), 0, 3).
231             " %, </td>";
232         echo "<td width=\"150\">Ved ikke: ".substr(((statte2($where, 1, 3) / $total) *100),0,3).
233             " %, </td>";
234
235         echo "<td width=\"120\">";
236         echo "Korrekt: ".substr(((statte2($where, 1, 4) / $total) * 100), 0, 3)." % </td>";
237
238     } else { echo "<td width=\"95\">Ja - %, </td><td width=\"95\"> Nej - %, </td>".
239               "<td width=\"150\"> Ved ikke - %</td><td width=\"120\">Korrekt: - %</td> "; }
240     ?>
241 </tr>
242 </table>
243 </td>
244 <td></td>
245 </tr>
246 <tr style="height:3px;"><td></td></tr>
247 <tr>
248 <td align="right" style="padding-right: 5px">
249 <a class="info" href="#">2) Tastselv skat:<span>Ingen lås</span></a>
250 </td>
251 <td style="padding-left: 5px">
252 <table border="0">
253 <tr>
254
255 <?php
256     if($total != 0)
257     {
258         echo "<td width=\"95\">Ja: ".substr(((statte2($where, 2, 1) / $total) * 100), 0, 3).
259             " %, </td>";
260         echo "<td width=\"95\">Nej: ".substr(((statte2($where, 2, 2) / $total) * 100), 0, 3).
261             " %, </td>";
262         echo "<td width=\"150\">Ved ikke: ".substr(((statte2($where, 2, 3) / $total)*100),0,3).
263             " %, </td>";
264
265         echo "<td width=\"120\">";
266         echo "Korrekt: ".substr(((statte2($where, 2, 4) / $total) * 100), 0, 3)." % </td>";
267
268     } else { echo "<td width=\"95\">Ja - %, </td><td width=\"95\"> Nej - %, </td>".
269               "<td width=\"150\"> Ved ikke - %</td><td width=\"120\">Korrekt: - %</td> "; }
270     ?>
271
272 </tr>
273 </table>
274 </td>
275 <td align="right"></td>
276 </tr>
277 <tr style="height:3px;"><td></td></tr>
278 <tr>
279 <td align="right" style="padding-right: 5px">
280 <a class="info" href="#">3) Paypal:<span>Ikke phished</span></a></td>
281 <td style="padding-left: 5px">
282 <table border="0">
283 <tr>
284

```

```

285 <?php
286     if($total != 0)
287     {
288         echo "<td width=\"95\">Ja: ".substr(((statstest2($where, 3, 1) / $total) * 100), 0, 3).
289             " %,</td>";
290         echo "<td width=\"95\">Nej: ".substr(((statstest2($where, 3, 2) / $total) * 100), 0, 3).
291             " %,</td>";
292         echo "<td width=\"150\">Ved ikke: ".substr(((statstest2($where, 3, 3) / $total)*100), 0, 3).
293             " %,</td>";
294
295         echo "<td width=\"120\">";
296         echo "Korrekt: ".substr(((statstest2($where, 3, 4) / $total) * 100), 0, 3)." % </td>";
297
298     } else { echo "<td width=\"95\">Ja - %,</td><td width=\"95\"> Nej - %,</td>".
299             "<td width=\"150\"> Ved ikke - %</td><td width=\"120\">Korrekt: - %</td> "; }
300     ?>
301
302 </tr>
303 </table>
304 </td>
305 <td align="right">
306 </td>
307 </tr>
308 <tr style="height:3px;"><td></td></tr>
309 <tr>
310 <td align="right" style="padding-right: 5px">
311 <a class="info" href="#">4) Apple store:<span>Hverken https eller lås</span></a>
312 </td>
313 <td style="padding-left: 5px">
314 <table border="0">
315 <tr>
316
317 <?php
318     if($total != 0)
319     {
320         echo "<td width=\"95\">Ja: ".substr(((statstest2($where, 4, 1) / $total) * 100), 0, 3).
321             " %,</td>";
322         echo "<td width=\"95\">Nej: ".substr(((statstest2($where, 4, 2) / $total) * 100), 0, 3).
323             " %,</td>";
324         echo "<td width=\"150\">Ved ikke: ".substr(((statstest2($where, 4, 3) / $total)*100), 0, 3).
325             " %,</td>";
326
327         echo "<td width=\"120\">";
328         echo "Korrekt: ".substr(((statstest2($where, 4, 4) / $total) * 100), 0, 3)." % </td>";
329
330     } else { echo "<td width=\"95\">Ja - %,</td><td width=\"95\"> Nej - %,</td>".
331             "<td width=\"150\"> Ved ikke - %</td><td width=\"120\">Korrekt: - %</td> "; }
332     ?>
333
334 </tr>
335 </table>
336 </td>
337 <td align="right"></td>
338 </tr>
339 <tr style="height:3px;"><td></td></tr>
340 <tr>
341 <td align="right" style="padding-right: 5px">
342 <a class="info" href="#">5) Jyske bank:<span>Forkert adresse (jyskebank)</span></a>
343 </td>
344 <td style="padding-left: 5px">
345 <table border="0">
346 <tr>
347
348 <?php
349     if($total != 0)

```

```

350     {
351         echo "<td width=\"95\">Ja: ".substr((statstest2($where, 5, 1) / $total) * 100), 0, 3).
352             " %,</td>";
353         echo "<td width=\"95\">Nej: ".substr((statstest2($where, 5, 2) / $total) * 100), 0, 3).
354             " %,</td>";
355         echo "<td width=\"150\">Ved ikke: ".substr((statstest2($where, 5, 3) / $total) * 100), 0, 3).
356             " %,</td>";
357
358         echo "<td width=\"120\">";
359         echo "Korrekt: ".substr((statstest2($where, 5, 4) / $total) * 100), 0, 3)." % </td>";
360
361     } else { echo "<td width=\"95\">Ja - %,</td><td width=\"95\"> Nej - %,</td>".
362         "<td width=\"150\"> Ved ikke - %</td><td width=\"120\">Korrekt: - %</td> "; }
363     ?>
364
365     </tr>
366 </table>
367 </td>
368 <td align="right"></td>
369 </tr>
370 <tr style="height:3px;"><td></td></tr>
371 <tr>
372 <td align="right" style="padding-right: 5px">
373 <a class="info" href="#">Samlet (avg):<span>Antal korrekte svar i gennemsnit.</span></a></td>
374 <td style="padding-left: 5px">
375 <table border="0">
376 <tr>
377
378 <?php
379     if($total != 0)
380     {
381         echo "<td width=\"95\"></td>";
382         echo "<td width=\"95\"></td>";
383         echo "<td width=\"150\"></td>";
384
385         echo "<td width=\"110\">";
386         echo "Korrekt: ".(substr((getsumphishinginetscore($where) / $total), 0, 3))." / 5 </td>";
387
388     } else { echo "<td width=\"95\"></td><td width=\"95\"></td>".
389         "<td width=\"150\"></td><td width=\"110\">Korrekt: - %</td> "; }
390     ?>
391
392     </tr>
393 </table>
394 </td>
395 <td align="right">Testen hjalp <?php
396     if($total != 0)
397     { echo substr((statstest2($where, "JA", 0) / $total) * 100),0,2)." %"; } else{ echo "- %"; }
398     ?>
399     </td>
400 </tr>
401 </table>
402 <br />
403 <?php /<!-- PHISHING EMAIL STATISTIK ----->?>
404 <table width="800" border="0" cellpadding="1" cellspacing="1" style="background-color: #dce5ee;">
405 <tr style="background-color: #bdcbda">
406 <td width="180"><h2>Phishing - Email</h2></td>
407 <td colspan="2" width="600" align="right">
408 <?php $total = statstestdatat($where, 3); echo $total." / ".statstestdatat("", 3); ?>
409 </td>
410 </tr>
411 <tr>
412 <td align="right" style="padding-right: 5px">
413 <a class="info" href="#">1) Mytravel:<span>Forkert link, samt bedt om upassende oplysninger.
414 </span></a></td>

```

```

415 <td style="padding-left: 5px">
416 <table border="0">
417 <tr>
418
419 <?php
420     if($total != 0)
421     {
422         echo "<td width=\"95\">Ja: ".substr(((statte3($where, 1, 1) / $total) * 100), 0, 3).
423             " %, </td>";
424         echo "<td width=\"95\">Nej: ".substr(((statte3($where, 1, 2) / $total) * 100), 0, 3).
425             " %, </td>";
426         echo "<td width=\"150\">Ved ikke: ".substr(((statte3($where, 1, 3)/$total)* 100), 0, 3).
427             " %, </td>";
428
429         echo "<td width=\"120\">";
430         echo "Korrekt: ".substr(((statte3($where, 1, 4) / $total) * 100), 0, 3)." % </td>";
431
432     } else { echo "<td width=\"95\">Ja - %, </td><td width=\"95\"> Nej - %, </td>".
433         "<td width=\"150\"> Ved ikke - %</td><td width=\"120\">Korrekt: - %</td> "; }
434     ?>
435 </tr>
436 </table>
437 </td>
438 <td></td>
439 </tr>
440 <tr style="height:3px;"><td></td></tr>
441 <tr>
442 <td align="right" style="padding-right: 5px">
443     <a class="info" href="#">2) Paypal:<span>Åbenlyst forkert link
444     </span></a></td>
445 <td style="padding-left: 5px">
446 <table border="0">
447 <tr>
448
449 <?php
450     if($total != 0)
451     {
452         echo "<td width=\"95\">Ja: ".substr(((statte3($where, 2, 1) / $total) * 100), 0, 3).
453             " %, </td>";
454         echo "<td width=\"95\">Nej: ".substr(((statte3($where, 2, 2) / $total) * 100), 0, 3).
455             " %, </td>";
456         echo "<td width=\"150\">Ved ikke: ".substr(((statte3($where, 2, 3) / $total)*100), 0, 3).
457             " %, </td>";
458
459         echo "<td width=\"120\">";
460         echo "Korrekt: ".substr(((statte3($where, 2, 4) / $total) * 100), 0, 3)." % </td>";
461
462     } else { echo "<td width=\"95\">Ja - %, </td><td width=\"95\"> Nej - %, </td>".
463         "<td width=\"150\"> Ved ikke - %</td><td width=\"120\">Korrekt: - %</td> "; }
464     ?>
465
466 </tr>
467 </table>
468 </td>
469 <td align="right"></td>
470 </tr>
471 <tr style="height:3px;"><td></td></tr>
472 <tr>
473 <td align="right" style="padding-right: 5px">
474     <a class="info" href="#">3) Tast selv skat:<span>Forkert link (svært at se, 'tast_selv')
475     </span></a></td>
476 <td style="padding-left: 5px">
477 <table border="0">
478 <tr>
479

```

```

480 <?php
481     if($total != 0)
482     {
483         echo "<td width=\"95\">Ja: ".substr(((statte3($where, 3, 1) / $total) * 100), 0, 3).
484             " %,</td>";
485         echo "<td width=\"95\">Nej: ".substr(((statte3($where, 3, 2) / $total) * 100), 0, 3).
486             " %,</td>";
487         echo "<td width=\"150\">Ved ikke: ".substr(((statte3($where, 3, 3) / $total) * 100), 0, 3).
488             " %,</td>";
489
490         echo "<td width=\"120\">";
491         echo "Korrekt: ".substr(((statte3($where, 3, 4) / $total) * 100), 0, 3)." % </td>";
492
493     } else { echo "<td width=\"95\">Ja - %,</td><td width=\"95\"> Nej - %,</td>".
494         "<td width=\"150\"> Ved ikke - %</td><td width=\"120\">Korrekt: - %</td> "; }
495     ?>
496
497     </tr>
498 </table>
499 </td>
500 <td align="right">
501 </td>
502 </tr>
503 <tr style="height:3px;"><td></td></tr>
504 <tr>
505 <td align="right" style="padding-right: 5px">
506 <a class="info" href="#">4) AOK:<span>Ikke phisnet.
507 </span></a></td>
508 <td style="padding-left: 5px">
509 <table border="0">
510 <tr>
511 <?php
512     if($total != 0)
513     {
514         echo "<td width=\"95\">Ja: ".substr(((statte3($where, 4, 1) / $total) * 100), 0, 3).
515             " %,</td>";
516         echo "<td width=\"95\">Nej: ".substr(((statte3($where, 4, 2) / $total) * 100), 0, 3).
517             " %,</td>";
518         echo "<td width=\"150\">Ved ikke: ".substr(((statte3($where, 4, 3) / $total) * 100), 0, 3).
519             " %,</td>";
520
521         echo "<td width=\"120\">";
522         echo "Korrekt: ".substr(((statte3($where, 4, 4) / $total) * 100), 0, 3)." % </td>";
523
524     } else { echo "<td width=\"95\">Ja - %,</td><td width=\"95\"> Nej - %,</td>".
525         "<td width=\"150\"> Ved ikke - %</td><td width=\"120\">Korrekt: - %</td> "; }
526     ?>
527
528     </tr>
529 </table>
530 </td>
531 <td align="right"></td>
532 </tr>
533 <tr style="height:3px;"><td></td></tr>
534 <tr>
535 <td align="right" style="padding-right: 5px">
536 <a class="info" href="#">5) Falck:<span>Forkert link
537 </span></a></td>
538 <td style="padding-left: 5px">
539 <table border="0">
540 <tr>
541 <?php
542     if($total != 0)

```

```

545     {
546         echo "<td width=\"95\">Ja: ".substr(((statte3($where, 5, 1) / $total) * 100), 0, 3).
547             " %, </td>";
548         echo "<td width=\"95\">Nej: ".substr(((statte3($where, 5, 2) / $total) * 100), 0, 3).
549             " %, </td>";
550         echo "<td width=\"150\">Ved ikke: ".substr(((statte3($where, 5, 3) / $total) * 100), 0, 3).
551             " %, </td>";
552
553         echo "<td width=\"120\">";
554         echo "Korrekt: ".substr(((statte3($where, 5, 4) / $total) * 100), 0, 3)." % </td>";
555
556     } else { echo "<td width=\"95\">Ja - %, </td><td width=\"95\"> Nej - %, </td>".
557         "<td width=\"150\"> Ved ikke - %</td><td width=\"120\">Korrekt: - %</td> "; }
558     ?>
559     </tr>
560 </table>
561 </td>
562 <td align="right"></td>
563 </tr>
564 <tr style="height:3px;"><td></td></tr>
565 <tr>
566 <td align="right" style="padding-right: 5px"><a class="info" href="#">Samlet (avg):<span>
567     Antal korrekte svar i gennemsnit.</span></a></td>
568 <td style="padding-left: 5px">
569     <table border="0">
570     <tr>
571     <td align="right">
572     <?php
573         if($total != 0)
574         {
575             echo "<td width=\"95\"></td>";
576             echo "<td width=\"95\"></td>";
577             echo "<td width=\"150\"></td>";
578
579             echo "<td width=\"120\">";
580             echo "Korrekt: ".substr(((getsumphishingemailscore($where) / $total), 0, 3)." / 5 </td>";
581
582         } else { echo "<td width=\"95\"></td><td width=\"95\"></td>".
583             "<td width=\"150\"></td><td width=\"120\">Korrekt: - %</td> "; }
584     ?>
585     </tr>
586 </table>
587 </td>
588 <td align="right">Testen hjalp
589 <?php
590     if($total != 0)
591     { echo substr(((statte2($where, "JA", 0) / $total) * 100), 0, 3)." % ";}else{ echo "- %"; }
592     ?>
593 </td>
594 </tr>
595 </table>
596 <br />
597 <?php // START WIRELESS STATISTIK //-----?>
598 <table width="800" border="0" cellpadding="1" cellspacing="1" style="background-color: #dce5ee;">
599 <tr style="background-color: #bdcbda">
600 <td width="180"><h2>Trådløst internet</h2></td>
601 <td colspan="2" width="600" align="right">
602 <?php $total = statte2datat($where, 4); echo $total." / ".statte2datat("", 4); ?>
603 </td>
604 </tr>
605 <tr>
606 <td align="right" style="padding-right: 5px">Med sikkerhedsaktivering:</td>
607 <td style="padding-left: 5px">

```



```

610     <?php
611         if($total != 0)
612             { echo substr(((statstest4($where, 1) / $total) * 100),0,3)." %"; } else { echo "- % "; }
613     >?
614     </td>
615 </td></td>
616 </tr>
617 <tr style="height:3px;"><td></td></tr>
618 <tr>
619     <td align="right" style="padding-right: 5px">Uden sikkerhedsaktivering:</td>
620     <td style="padding-left: 5px">
621         <?php
622             if($total != 0)
623                 { echo substr(((statstest4($where, 2) / $total) * 100),0,3)." %"; } else { echo "- % "; }
624         >?
625     </td>
626     <td align="right">Testen hjælp
627     <?php
628         if($total != 0)
629             { echo substr(((statstest4($where, 3) / $total) * 100),0,3)." % "; } else { echo "- %"; }
630     >?
631     </td>
632 </tr>
633 </table>
634 <form><input type="hidden" value="<?php echo date('j/n-Y H:i:s'); ?>" /></form>

```

### A.6.14 Statistik statscript.js

```

1 function ajaxManager()
2 {
3     var args = ajaxManager.arguments;
4     switch (args[0])
5     {
6     case "load_page":
7         if (document.getElementById)
8         {
9             var x = (window.ActiveXObject) ? new ActiveXObject("Microsoft.XMLHTTP") : new XMLHttpRequest();
10        }
11        if (x)
12        {
13            x.onreadystatechange = function()
14            {
15                if (x.readyState == 4 && x.status == 200)
16                {
17                    e1 = document.getElementById(args[2]);
18                    e1.innerHTML = x.responseText;
19                    e2 = document.getElementById('loadfield');
20                    e2.innerHTML = "<div style=\"color: black;\">Done</div>";
21                }
22            }
23        }
24        document.form.nocache.value = Date(); // ingen cache
25        x.open("GET", args[1], true);
26        x.send(null);
27    }
28    break;
29    case "start_up":
30        var lget = "?age_from="+document.form.age_from.value+
31            "&age_to="+document.form.age_to.value+
32            "&gender_male="+document.form.gender_male.checked+
33            "&gender_female="+document.form.gender_female.checked+
34            "&skill_none="+document.form.skill_none.checked+
35            "&skill_experienced="+document.form.skill_experienced.checked+
36            "&skill_expert="+document.form.skill_expert.checked+

```

```
37         "&dato="+Date();
38
39     ajaxManager('load_page', 'statdata.php'+lget, 'statdata');
40     break;
41 }
42 }
43
44 function IsNumeric(sText)
45 {
46     var ValidChars = "0123456789";
47     var Char;
48
49     for (i = 0; i < sText.length; i++)
50     {
51         Char = sText.charAt(i);
52         if (ValidChars.indexOf(Char) == -1)
53         {
54             return false;
55         }
56     }
57     return true;
58 }
59
60 function validate(elem)
61 {
62     e2 = document.getElementById('loadfield');
63     e2.innerHTML = "<div style=\"color: red;\">Updating</div>";
64     if((IsNumeric(document.form.age_from.value) == false) ||
65        (IsNumeric(document.form.age_to.value) == false))
66     {
67         alert("Begge 'Alder' -felter skal være tal og mellem 0 og 99");
68         document.form.age_from.value = "0";
69         document.form.age_to.value = "99";
70         return false
71     }
72
73     if(parseInt(document.form.age_from.value) > parseInt(document.form.age_to.value))
74     {
75         e1 = document.getElementById('agefault');
76         e1.innerHTML = "(det første felt skal være mindre end det andet)";
77         return false
78     }
79     else
80     {
81         e1 = document.getElementById('agefault');
82         e1.innerHTML = "";
83     }
84     // hvis det er en checkbox der er blevet ændret må vi huske at sende ændret info...
85     var lget;
86     if(elem.name != "age_from" && elem.name != "age_to")
87     {
88         elem.click();
89         lget = "?age_from="+document.form.age_from.value+
90             "&age_to="+document.form.age_to.value+
91             "&gender_male="+document.form.gender_male.checked+
92             "&gender_female="+document.form.gender_female.checked+
93             "&skill_none="+document.form.skill_none.checked+
94             "&skill_experienced="+document.form.skill_experienced.checked+
95             "&skill_expert="+document.form.skill_expert.checked;
96         elem.click();
97     }
98     else
99     {
100         lget = "?age_from="+document.form.age_from.value+
101             "&age_to="+document.form.age_to.value+
```

```
102         "&gender_male="+document.form.gender_male.checked+
103         "&gender_female="+document.form.gender_female.checked+
104         "&skill_none="+document.form.skill_none.checked+
105         "&skill_experienced="+document.form.skill_experienced.checked+
106         "&skill_expert="+document.form.skill_expert.checked;
107     }
108     ajaxManager('load_page', 'statdata.php'+lget, 'statdata');
109     return false;
110 }
```



# Litteratur

---

- [1] Adobe. Captivate 2.  
<http://www.adobe.com/products/captivate>.
- [2] CERT. Gode råd om it-sikkerhed.  
<https://www.cert.dk/vejled/10raad.shtml>.
- [3] Thomas M. Connolly. *Database solutions*. Addison-Wesley, third edition, 2002.
- [4] ENISA. How to raise information security awareness. 2006.
- [5] Anti-Phishing Working Group. Phishing.  
<http://www.antiphishing.org/>.
- [6] Musikhuset Aage Jensen. Brugerforum.  
[http://www.aage.dk/default.asp?page=/nytforum/forum\\_posts.asp?TID=67192](http://www.aage.dk/default.asp?page=/nytforum/forum_posts.asp?TID=67192).
- [7] Craig Larman. *Applying UML and Patterns*. Prentice Hall, second edition, 2002.
- [8] Microsoft. Phishing, basisviden.  
<http://www.microsoft.com/danmark/athome/security/quiz/phishingbasics1.aspx>.
- [9] The European Network and Information Security Agency. Enisa homepage.  
<http://www.enisa.europa.eu/>.
- [10] Charles P. Pfleeger and Shari Lawrence Pfleeger. *Security in Computing*. Prentice Hall, third edition, 2003.
- [11] W3schools. Asynchronous javascript and xml.  
<http://www.w3schools.com/ajax/default.asp>.

- [12] W3schools. W3schools homepage.  
<http://www.enisa.europa.eu/>.
- [13] Wikipedia. Adware.  
<http://en.wikipedia.org/wiki/Adware>.
- [14] Wikipedia. Spyware.  
<http://en.wikipedia.org/wiki/Spyware>.