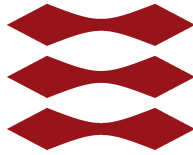


DTU



FIT – An Online Inspection Support Tool

Rita Petrolyte
s090657

Kongens Lyngby 2011
IMM-M.Sc.-2011-40

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
www.imm.dtu.dk

IMM-M.Sc.: ISSN 2011-40

Summary

Formal inspection is one of the techniques which help to find defects and mistakes in any kind of development documents. This technique can be applied at any stage of the development process and improve it.

In all development documents creation stages appear mistakes which lead to not acceptable results and quality. Formal inspection helps to find mistakes in the earlier stage, save time and budget. There are several ways to make a formal inspection and sometimes it is quite difficult to choose one.

Formal inspection can be done based on paper forms or using online tools. Both of the ways are useful if it is done properly. Standard formal inspection paper forms are complicated, requires a lot of time to fill, have limited editing. Using paper based formal inspection forms to organize and coordinate all inspection process becomes challenging. Online based inspection requires less time and gives the same result. The main problem appears when chosen online based inspection tool is not suitable for selected artifact. Mostly online inspection tools are created to inspect software.

To make the formal inspection process straightforward, less time consuming, online inspection support system FIT (Formal Inspection Tool) will be created. An online inspection support system will allow making a formal inspection for development documents like contracts, project plans, requirements documents, specifications, designs and code. Using FIT it will be possible to create reviews, plan inspections, share artifacts, submit individual inspection results, collect preparations, lead meetings, determine reworks, sign of reworks, compile inspectors' comments, log defects, forward results, inspect artifacts, submit results and close reviews online. All these mentioned functions are the main formal inspection process steps which are mandatory to make a formal inspection. Allowing making these functions online formal inspection will become more straightforward, more effective and efficient process.

FIT will change the way of making the formal inspection process but will have the same result - improvement of the development documents.

Preface

This thesis was prepared at Informatics Mathematical Modelling, the Technical University of Denmark in partial fulfilment of the requirements for acquiring the MSc degree of Science in Telecommunications.

This thesis deals with formal inspection. The main focus is to design and create formal inspection tool based on web technology which will change the way of making formal inspections.

30 ECTS credits worth project was started on the 1 of February and finished on the 1 of July. Supervisor of the project is Prof. Dr. Harald Störrle.

Lyngby, July 2011

Rita Petrolyte

Acknowledgements

I thank my supervisor Prof. Dr. Harald Störrle for a great supervising of the project, constructive critics and valuable advices. Hence, I thank my international family for support and the best colleagues of the MSc students' office for discussions and calm environment to work.

Table of Contents

Summary.....	3
Preface.....	4
Acknowledgements	5
List of figures	8
List of tables.....	10
List of abbreviations	11
1. Motivation	12
1.1. Experience	12
1.2. Existing tools.....	17
1.3. Goals	19
2. Fagan inspection.....	22
3. FIT design.....	26
3.1. FIT use case diagram.....	26
3.2. Inspection process using FIT.....	28
3.3. Life cycle of review process using FIT	30
3.4. Database structure	30
3.5. Technology	34
3.6. FIT layout and sketches	36
3.7. Logical GUI design steps of using FIT tool	46
4. Implementation.....	52
4.1. Adapting technologies.....	52
4.2. FIT tool code files and database	54
4.3. Implemented FIT tool	57
5. Operation.....	69
5.1. Installation, configure and run FIT.....	69

5.2.	Error messages and alerts	69
5.3.	Loading time of FIT	70
5.4.	Log files	71
5.5.	Adding or changing FIT	73
6.	Conclusions.....	74
	Bibliography.....	76

List of figures

Figure 1 Evaluation of formal inspection processes in utility and complexity scale	16
Figure 2 The formal inspection process	23
Figure 3 Use case diagram of FIT	26
Figure 4 The formal inspection process using FIT	29
Figure 5 Life cycle of review process using FIT	32
Figure 6 Database structure	33
Figure 7 FIT layout	37
Figure 8 General layout of the FIT changing review after login	38
Figure 9 FIT review section	39
Figure 10 FIT team section	40
Figure 11 FIT artifact section	40
Figure 12 FIT remarks section.....	41
Figure 13 FIT selecting edit.....	42
Figure 14 FIT selecting delete (1).....	42
Figure 15 FIT selecting delete (2).....	43
Figure 16 FIT import .csv file (1)	44
Figure 17 FIT import .csv file (2)	44
Figure 18 FIT import .csv file (3)	45
Figure 19 FIT import .csv file (4)	45
Figure 20 Edit function in FIT.....	48
Figure 21 Add function in FIT.....	48
Figure 22 Alert messages of FIT.....	49
Figure 23 Logical GUI design.....	50
Figure 24 FIT main page (login)	57
Figure 25 Administrator review tab	58
Figure 26 Administrator add moderator tab	58
Figure 27 Administrator user tab	59
Figure 28 Moderator review tab	59
Figure 29 Moderator team tab.....	60
Figure 30 Moderator add team tab.....	60
Figure 31 Moderator artifact tab.....	61

Figure 32 Moderator remarks tab	61
Figure 33 Moderator add remark function	62
Figure 34 Moderator edit remark function	62
Figure 35 Moderator delete remark function	63
Figure 36 changing the review function and inspector review tab	63
Figure 37 Inspector team tab	64
Figure 38 Inspector artifact tab	64
Figure 39 Inspector remarks tab	65
Figure 40 Inspector remarks tab uploading .csv file.....	66
Figure 41 Uploaded .csv file in remarks table	66
Figure 42 Author review tab.....	67
Figure 43 Author team tab	67
Figure 44 Author artifact tab	68
Figure 45 Author remarks tab	68
Figure 46 Admin page load time	70
Figure 47 Moderator page load time	70
Figure 48 Author page load time.....	70
Figure 49 Inspector page load time.....	70
Figure 50 Scribe page load time	71

List of tables

Table 1 Formal inspection roles, responsibilities and list of forms to fill	12
Table 2 Positive feedback of requirements engineering course students	14
Table 3 Negative feedback for inspection as a technique of requirements engineering course students.....	14
Table 4 Negative feedback for inspection as a process of requirements engineering course students.....	15
Table 5 Paper based and online inspection tools.....	17
Table 6 Comparison of the paper based and online inspection tools.....	19
Table 7 Inspection team responsibilities supported by FIT.....	19
Table 8 Goals and sub goals of the formal inspection tool	20
Table 9 Group review with individual preparation, formal inspection with individual preparation and individual peer desk-check	25
Table 10 Main functions of the FIT and implementation possibility.....	35
Table 11 Features allowed for specific role.....	47
Table 12 FIT tool code files	54
Table 13 FIT database user table.....	55
Table 14 FIT database review table	55
Table 15 FIT database comment table	55
Table 16 FIT database file table.....	56
Table 17 FIT data base role table.....	56
Table 18 FIT data base review_user_xref table.....	56
Table 19 Example of .csv file for uploading comments.....	65

List of abbreviations

AISA	-	Asynchronous Inspector of Software Artifacts
CAIS	-	Collaborative Asynchronous Software Inspection
CSI	-	Collaborative Software Inspection
CSS	-	Cascading Style Sheets
CSV	-	Comma Separated Values
FI	-	Formal Inspection
FIT	-	Formal Inspection Tool
GPL	-	General Public License
GUI	-	Grafical User Interface
HTML	-	Hyper Text Markup Language
ICICLE	-	Intelligent Code Inspection in a C Language Environment
IPA	-	Inspection Process Assistant
JSON	-	Java Script Object Notation
MIT	-	Massachusetts Institute of Technology
MySQL	-	My Structured Query Language
PHP	-	Personal Home Page
SQL	-	Structured Query Language
UML	-	Unified Modeling Language
WiP	-	Web Inspection Prototype
XAMPP	-	X Apache MySQL PHP Perl

1. Motivation

During the Requirements engineering course there were a possibility to organize two formal inspections. Inspections were based on Fagan inspection method. This chapter describes the inspection process highlighting advantages and disadvantage. Requirements engineering course students' detailed feedback is presented in the end of the chapter.

1.1. Experience

The first formal inspection in requirements engineering course was organized and based on provided guidelines of formal inspection. Inspection was based on Fagan method filling paper forms. First of all, students had to pick up the roles of formal inspection team and follow the main steps of formal inspection process: preparation, inspection meeting and follow-up rework. Before starting the formal inspection, students were introduced to the process. Formal inspection team members had different roles, responsibilities and tasks, forms to fill (Table 1).

Table 1 Formal inspection roles, responsibilities and list of forms to fill

Role	Responsibilities	Forms to fill
Moderator	<ul style="list-style-type: none"> Plan inspection; Collect preparation; Lead meeting; Determine rework; Follow up on rework; Sign off process steps. 	<ul style="list-style-type: none"> Inspection Process Summary; Inspection Preparation Summary; Additional Rework Assignments.
Scribe	<ul style="list-style-type: none"> Compile and consolidate inspectors comments; Log defects during inspection meeting; Support moderator if it is assigned by moderator; 	<ul style="list-style-type: none"> Inspection Preparation Summary; Additional Rework Assignments.
Author	<ul style="list-style-type: none"> Prepare and make available inspection artifact; Present inspection artifact in the meeting; Answer questions to inspectors; Forward inspection results to co-authors. 	
Inspector	<ul style="list-style-type: none"> Inspect artifact as meeting preparation ; Submit results to moderator; Explain and elaborate comments during inspection. 	<ul style="list-style-type: none"> Individual Inspection Preparation; Additional Comments.

Moderator was responsible to fill Inspection Process Summary, Inspection Preparation Summary and Additional Rework Assignments. Inspectors were responsible to fill Individual Inspection Preparation and Additional Comments forms. Moderators were responsible to organize inspection, collect preparation, lead meeting, determine rework, follow up on rework and sign off process steps. Author provided an artifact for the inspection, inspectors had to inspect the given artifact and before the deadline submit the result for

moderators. When individual inspection was submitted, the filled forms had to be reviewed before the inspection meeting by moderator. Individual inspection comments had to be compiled before the meeting. Scanned filled paper forms during the formal inspection were distributed by e-mails or arranging meetings to give filled forms directly to moderator. During the formal inspection meeting moderator led the discussion and took notes. Inspectors were following their individual inspection comments using their filled paper forms. In the end of the discussion all documents were collected by moderator and submitted to author for the rework process.

The second formal inspection had modifications. The main difference between the first and the second inspections was individual inspection preparation sheet. It was changed to digital in order to make it easier to write the comments for inspectors and afterwards to collect, read and compile them for moderator. In one excel sheet it was asked to note the issue number, inspector assessment (major, minor), type (criteria list entry), artifact (number), location (line number), additional remarks (yes or no) and inspector number. Additional remarks, comments were presented in other excel sheet. Information was easy to share. It was more comfortable to collect and compile the comments. The idea of making it in digital way was reasonable but as it was not obligatory some of the inspection team members chose the previous, paper based way. All in all, moderators get mixed forms, some of them were digital and some of them were paper based. In conclusion it can be assumed that to digitize the forms was the right way but it should be strictly pointed to fill paper or digital forms by moderator. Mixed way of forms can become more complicated.

Formal inspection process feedback was collected from the requirements engineering course participants in order to get more opinion about the first and second inspections. They were asked to indicate their role in formal inspection and answer few questions. From 44 participants 15 of them gave feedback. Participants were asked to answer these questions:

- What was your role during Formal Inspection?
- What you liked and disliked about Formal Inspection?
- Was it hard or easy, and why?

Results of the feedback are given in Table 2, Table 3 and Table 4. Feedback is grouped in positive and negative, considering the inspection process role (moderator, author and inspector). Hence, negative feedback is grouped into negative as inspection technique or negative as process. There was no reason to separate positive feedback because all comments and remarks are dedicated for inspection as a technique.

Table 2 Positive feedback of requirements engineering course students

Role	Positive feedback
Moderator	<ul style="list-style-type: none"> • Useful outcomes; • Good experience in formalities, helps to adopt later when you proceed on the business carrier; • Allows to set the tone of the review process; • Real improvements to the inspected material.
Author	<ul style="list-style-type: none"> • Good feedback; • Easy to prepare for the review; • Effective, several problems were found in short time; • Significant improvement if the artifacts.
Inspector	<ul style="list-style-type: none"> • Individual preparation, discussion, useful knowledge; • The communication part during the inspection; • The strict form; • Easy, helped the author to improve the work; • Good inspection group gives great value to the project; • Interesting and constructive process; • All participants benefit from the meeting; • Active group, all members could interact with each other; • Good number of participants, easy to control for moderator, everybody can express his/her opinion; • Easy procedure. • Did not take a lot of time comparing with the positive feedback you get.

Almost all participants agreed that formal inspection is useful and helpful tool in order to improve the artifact. They got experience and were satisfied with the outcome of the formal inspection. Few quotations are presented: *"They really read the artifact carefully and they delivered a great feedback that I'm sure the author could use for improvements."*, *"I found the whole concept and process very interesting and very constructive, as it was the first time that I was doing something like that. I believe that all the participants could benefit from these meetings, because all of us were active and could interact with each other."*

Table 3 Negative feedback for inspection as a technique of requirements engineering course students

Role	Negative feedback
Moderator	<ul style="list-style-type: none"> • Deadlines (emphasize on the deadlines); • Focus on minor defects (inspectors should form the same case study to improve it); • Format and layout too ambiguous, non-understandable; • Subjective; • Hard to make decisions during the review process; • Too much attention for details.
Author	<ul style="list-style-type: none"> • Hard to explain for inspectors because they are not related to the case study; • Felt judged; • Hard to explain later for group members what was discussed in the inspection meeting; • The most difficult role in formal inspection, a lot of responsibility.
Inspector	<ul style="list-style-type: none"> • Problems in counting minor, major defects; • Found defects mostly minor defects; • Too long discussions on minor defects;

- Lack of moderators last word;
- Too much formalism and rules;
- It is not feasible technique to be applied broadly in forms;
- If it could be less formal it would be more attractive and more effective.

Table 4 Negative feedback for inspection as a process of requirements engineering course students

Role	Negative feedback
Moderator	<ul style="list-style-type: none"> • Difficulties to insert and track inspectors comments in to the sheet (new comments were added during the inspection); • Time consuming; • No clear idea how to fill inspection preparation summary, more explanation needed; • Rework stage is not clear; • Slow process (lack of knowledge how to perform); • No clear idea what paper should be handled; • Too much time focusing on formalities (prefer more information why it is important to keep standard layouts); • Too short preparation time; • Author did not know how to prepare inspection material correctly; • Inspectors did not know what they need to do; • A lot of stress for moderator; • Inspectors expect moderator to say what to do; • Not clear inspection roles; • No repository for inspection documents; • No tool that can help to find syntax, semantics and logical errors; • Challenging to coordinate all process; • Filling in the paper work was confusing; • Artifact were not ready for review; • Hard to coordinate the inspection group, hard to find the time slot for the meeting; • Stressful to make notes during the meeting and listen to the discussion; • Prefer software then paper work; • Process is not clear; • Slow and rigid procedure; • Everything is done in writing. There is little space for modifications of the outcomes.
Author	<ul style="list-style-type: none"> • Hard to agree about the meeting time, needed external webpage schedule system; • Hand writing; • Tiring to read everything loud.
Inspector	<ul style="list-style-type: none"> • Repeating errors; • Waste of time writing down all comments, review sheet header had to be filled again; • Hand writing (prefer digital format); • Moderator needs scribe, takes too much time of making notes; • Moderators are not organized and prepared, it slows down the process; • Hard to understand provided template tables.

Majority of the comments were negative which are dedicated to the way of making the formal inspection. It is seen that participants were not satisfied in filling paper forms. Inspection could be easier process for all inspection team if it will be in digital form. The main arguments for making it digital are less time consuming, more editing possibilities, easier to coordinate the process. As evidence few quotations are

presented: *“First of all it would be nice if all the nodes would be in digital format so everything that the inspectors notice is analysed together with the author and the moderator.”*, *“Some kind of repository for the inspection documents should be created instead of sending mails. Predefined e-documents are good instead of pictures and scans.”*, *“It was a bit challenging to orchestrate the entire thing, such as the meeting time and getting everybody to send their documents on time.”*, *“On the moderator role, I thought that one could easily make it easier by having some software, to help you note down corrections, and gather the comments from the inspectors. At times it was quite stressful to make notes, and listen to the discussion about the artifact, and guide the discussion in the right direction.”*, *“Time consuming review (waiting for the moderator to take notes) what would be great/were missing would be: an easier way to combine the suggestions and comments to a specific fault/line, before the review for the moderator, an easier way to select the “correct” suggestions and add comments for the moderator under the review.”* In the end students were asked to evaluate formal inspection in utility and complexity scale (Figure 1). They were asked to put a dot in the graph individually (not seeing each other opinion).

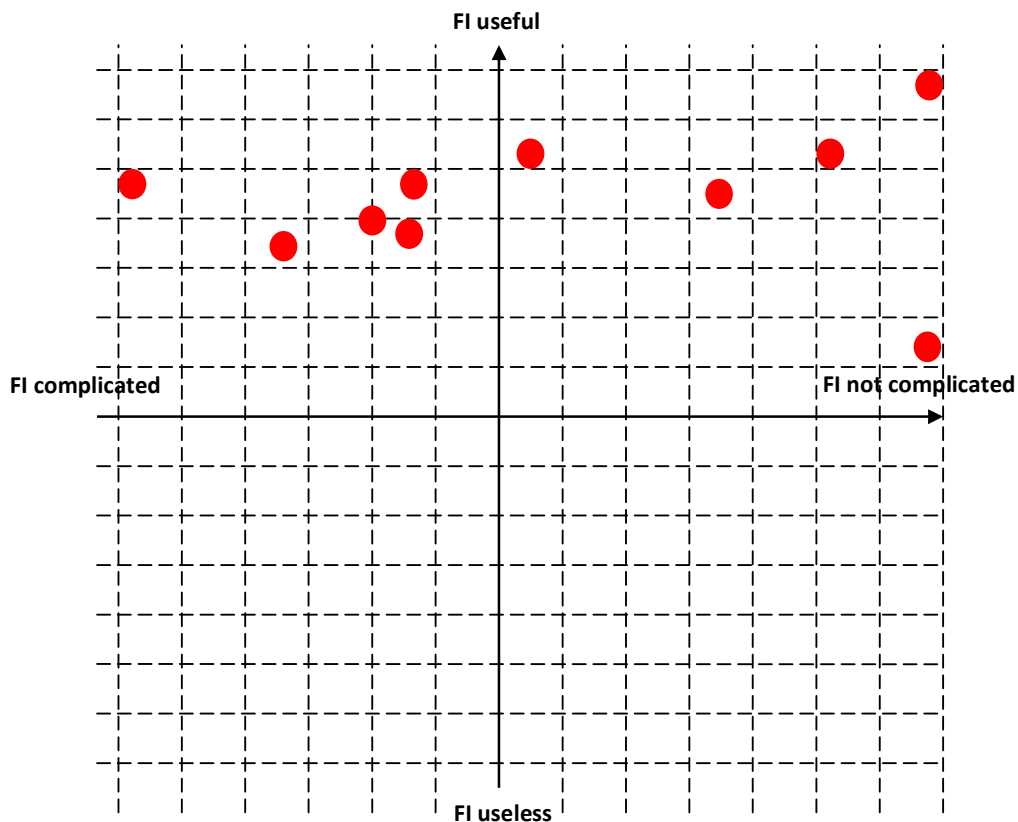


Figure 1 Evaluation of formal inspection processes in utility and complexity scale

It was expected that they will consider formal inspection as useful and quite complicated process. Analysing the results it is seen that they agreed that it is useful, but the complexity level is different for all of them. Taking an average and comparing results it is obvious that it should be somewhere in the middle, between complicated and not complicated side. From the given feedback it can be assumed that complexity level was evaluated in two different ways. For some of the students it was challenging to understand and use formal inspection as a technique and for others the way of the process was not acceptable and too complicated.

Some of the students of requirement engineering course are using formal inspection and found it useful in order to improve their projects, reports and assignments. All in all, to make the formal inspection straightforward and less time consuming there is necessity to make some changes which will lead to the increasement of the formal inspection usage.

1.2. Existing tools

There are two types of existing computer supported inspection tools: paper based inspection tools which provide data from paper based inspections and online based inspection tools which provide online inspection of artifact. Paper based tools were created based on moving the inspection process online. Few paper based tools Compas, Quality Group 400, Inspection Process Assistant (IPA) descriptions are presented in Table 5. (Macdonald 1998)

Online inspection tools: Intelligent Code Inspection in a C Language Environment (ICICLE), Collaborative Software Inspection (CSI), Collaborative Asynchronous Software Inspection (CAIS), Asynchronous Inspector of Software Artifacts (AISA), Web Inspection Prototype (WiP) and etc. All tools are more complex than paper based tools. Detailed descriptions are presented in the Table 5. (Macdonald 1998)

Table 5 Paper based and online inspection tools

Tools	Description
Compas	Compas is development process support tool, document management system. Compas allows the inspection to be scheduled, allows participants to be name, to set time and place of the inspection. Electronic notification can be sending for participants. Can generate a set of inspection forms.
Quality Group 400	Quality Group 400 is used to support software inspection. Comments during the reviews are supported by simple text files. There is a possibility for moderator to collect comments into single file, after that paper report can be generated. There is possibility to store data from multiple reviews.
Inspection Process Assistant	The main use is to allow defects to be entered on-line. Inspection process consists of planning (checking artifacts and organizing the inspection team), meeting, and verification. Inspection Process Assistant allows storing information.
ICICLE	Designed to support the inspection of C and C++ code. Very specific tool and is not suitable to inspect other types of artifact. ICICLE supports individual preparation

	and inspection meeting. Meeting support means that meeting has to be held with all inspectors in the same room using ICICLE. All meeting participants have access to all inspection documents as well as to their own made comments. During the discussion, accepted comments are sending as output of the meeting. In the end of the meeting the list of accepted comments is generated.
CSI	Designed to support inspection of all software development products. Inspection process using CSI starts from giving the document for inspection by the author and during the preparation stage inspectors creates the list of the mistakes and faults founded in the provided artifact. The main responsibility to deal with the founded defects belongs to the author as well, before the inspection meeting author is responsible to correlate the list of the defects. CSI allows seeing the inspected document using the browser and make notes on specifically selected line. Specific comment for particular line can be made. This function is supported by hyperlinks. Annotations can be made individually by all inspectors, and all collected results can be seen by author. Author has possibility to categorize defects into accepted or rejected. Sorting function is possible as well. During the inspection meeting all defects can be seen on the screen for all inspection team. The responsibility to guide the meeting belongs to author. All detected defects are discussed and after agreeing moved to action list. More annotations can be added during the inspection meeting. CSI allows audio conferencing. Data collection of team members' info, time of meeting and number of defects is possible.
AISA	Designed to allow asynchronous inspection of graphical documents. This tool supports defect collection, defect correlation and asynchronous meeting support. Inspected document is a clickable image map, where annotation can be done by clicking on the specific part of the graphical document. Individual annotation can be done and after that all information is collected. The producer has possibility to correlate the list of the defects by arranging them and removing the duplicates, change the order and prepare material for inspection meeting. Using voting system defects are accepted or rejected. Output of the meeting is a summary of all defects and suggestions how to improve it. AISA does not allow data collection.
WiP	Designed to supports only text documents. Inspection process starts of providing the artifact for inspection assigning inspectors and defining the roles. During preparation stage all inspection team members have access to the artifact, annotations can be done line by line. During the public inspection WiP combines all defects and makes a single list of defects. More annotations can be added. There is no meeting support in WiP, all phases are asynchronous. WiP collects inspection time and number of defects.

Comparing Compas, Quality Group 400 and Inspection Process Assistant (IPA) tools the main criteria's are document handling, individual preparation, meeting support and data collection. Document handling stands for allowing browsing documents online, annotation of documents and all information available during formal inspection. Individual preparation means allowing providing inspector with checklists and other documentation like guidelines during the inspection process. Meeting support allows organizing meeting date, time and place. Data collection stands for providing data about inspection process, taking in account time which was spend during the individual preparation, meeting and overall process. Data is collected automatically, there is no additional work to get it and in this way moderator can concentrate on other parts of formal inspection.

An online inspection tools are more complex than paper based tools, those tools can be classified as well as paper based ones, comparing suitability to any kind of artifact, support more than one review for one user, document handling, individual preparation, meeting support and data collection criteria's. Comparison of paper based and online inspection tools is given in Table 6 (Macdonald 1998).

Table 6 Comparison of the paper based and online inspection tools

Feature / Tool	Compas	Quality Group 400	IPA	ICICLE	CSI	AISA	WiP
Suitability for any kind of artifact	X	-	X	-	-	-	X
Support more than one review for one user	-	-	-	-	-	-	-
Document handling	X	X	-	X	X	X	-
Individual preparation	X	X	X	X	X	X	X
Meeting support	-	-	X	X	X	-	-
Data collection	-	X	X	-	X	-	X

Mentioned tools are designed to support the inspection of specific artifacts, for codes, for software development products or for graphical documents. Just some of them are more or less suitable for any kind of artifact. None of them supports more than one review for one user.

1.3. Goals

Considering on the feedback of the requirements engineering courses students and the analysis of the existing inspection tool the list of the goals for FIT tool was made. The main goals is to digitize formal inspection, allowing to plan inspection, collect preparation, lead meeting, determine rework, sign of process steps, compile and consolidate inspectors comments, log defects during the inspection meeting, prepare and make available inspection artifacts, forward inspection results to co-authors, insect artifact as meeting preparation, submit results to moderator. Table 7 provides all responsibilities of the inspection team members (Storrle 2010). Responsibilities marked in bold will be supported by the FIT tool. Apart from moving formal inspection online FIT will allow to support all stages of inspection process, will support any type of artifact. Inspection will be based online and there will be no necessity to install the program on personal computer. FIT can be used everywhere and at any time if the user has internet connection.

Table 7 Inspection team responsibilities supported by FIT

Role	Responsibilities supported by FIT
Moderator	<ul style="list-style-type: none"> • Plan inspection; • Collect preparation; • Lead meeting; • Determine rework; • Follow up on rework; • Sign off process steps.
Scribe	<ul style="list-style-type: none"> • Compile and consolidate inspectors comments;

	<ul style="list-style-type: none"> • Log defects during inspection meeting; • Support moderator if it is assigned by moderator;
Author	<ul style="list-style-type: none"> • Prepare and make available inspection artifact; • Present inspection artifact in the meeting; • Answer questions to inspectors; • Forward inspection results to co-authors.
Inspector	<ul style="list-style-type: none"> • Inspect artifact as meeting preparation; • Submit results to moderator; • Explain and elaborate comments during inspection.

FIT tool will cover the main tasks of the formal inspection which can be transferred to online platform. Tasks which are not supported by the tool because it does not require specific way of improving or does not have it. Follow up of the rework consist of signing of the process steps and does not requires to be as an additional feature in the creating tool as well as support moderator if it is assigned by moderator. Present inspection artifact in the meeting is just the process of the reading, as the artifacts will be available online it can be read as well. To answer the questions to inspectors is the task of the author which is done during the meeting it is a part of the discussion and it cannot be shifted on the FIT tool as well as explain and elaborate comments during inspection. Table 8 gives all goals and sub goals which will be achieved creating FIT tools.

Table 8 Goals and sub goals of the formal inspection tool

No.	Goals	Sub goals
1.	Organize inspection	<ul style="list-style-type: none"> • Create review; • Close review; • Upload guidelines; • Create review team; • Enter review information; • Distribute information; • Confirm uploaded artifact; • Reject uploaded artifact.
2.	Collect preparation	<ul style="list-style-type: none"> • Consolidate individual inspection result.
3.	Lead meeting	<ul style="list-style-type: none"> • View comments during the inspection meeting; • Add additional remarks; • Edit existing comments.
4.	Determine rework	<ul style="list-style-type: none"> • Inform author about rework (dates).
5.	Sign off process steps	<ul style="list-style-type: none"> • Confirm process step; • Reject process step; • Return process step to earlier stage.
6.	Compile and consolidate inspectors comments	<ul style="list-style-type: none"> • Delete remark; • Change status of remark; • Sort by line number, artifact name, author of the comment name, type of the comment and date when comment was submitted.
7.	Takes notes during inspection meeting	<ul style="list-style-type: none"> • Add additional comments or remarks during the inspection meeting.

8.	Prepare and make available inspection artifact	<ul style="list-style-type: none">• Upload artifact;• Upload guidelines.
9.	Forward inspection results to co-authors	<ul style="list-style-type: none">• Download rework.
10.	Inspect artifact as meeting preparation	<ul style="list-style-type: none">• Define remark location in artifact;• Define type of remark;• Enter the remark;• Define the status of remark;• Edit remark;• Delete remark.
11.	Submit results to moderator	<ul style="list-style-type: none">• Upload individual inspection results.

All mentioned goals and sub goals supported by the FIT tool will make formal inspection process easier and more straightforward.

2. Fagan inspection

The inspection technique was developed by Michael E. Fagan at IBM Kingston NY Laboratories. Fagan inspection is a very early upstream development and maintenance process which aims at both quality improvement and work process improvement (Gilb 1993) (Strauss 1993).

Fagan inspection is a structured procedure to find faults and mistakes in artifacts. It is a simple, powerful and much cost efficient technique for quality assurance. Fagan inspection is a group review method used to evaluate output of the given process. There is lists of fields were Fagan inspection method can be used (Wikipedia 2011):

- Requirements specification;
- Software and information systems architecture;
- Programming;
- Software testing.

Fagan inspection process consists of six operations: planning, overview, preparation, inspection meeting, rework and follow-up. Operations consists number of tasks which have to be strictly followed in order to reach useful results of inspection. Operations includes list of tasks:

- Planning: preparation of materials, arranging of participants, arranging of meeting place and time;
- Overview: assignment of roles;
- Preparation: individual inspection, preparing material for the meeting, noting all possible defects and questions;
- Inspection meeting: finding the defects during the discussion;
- Rework: defects resolving;
- Follow-up: verifying defects resolving stage.

To start formal inspection it is necessary to form inspection team which consists of moderator, author/designer/coder, reader and reviewers. Team members have their own tasks and responsibilities:

- Moderator organizes the inspection process and guides the discussion during the inspection meeting;
- Author/Designer/Coder is a person presenting the artifact and reading it to the inspection team during the meeting;
- Reader is responsible to paraphrase the document;

- Reviewers'/Inspectors main responsibility is to assess the artifact and take notes in advance. In the table there is show all inspection team responsibilities grouped by the inspection role.

The UML activity diagram (Figure 2) provides an overview of the formal inspection process. The inspection process is divided into three phases: preparation, inspection meeting and follow-up work (Storrle 2010).

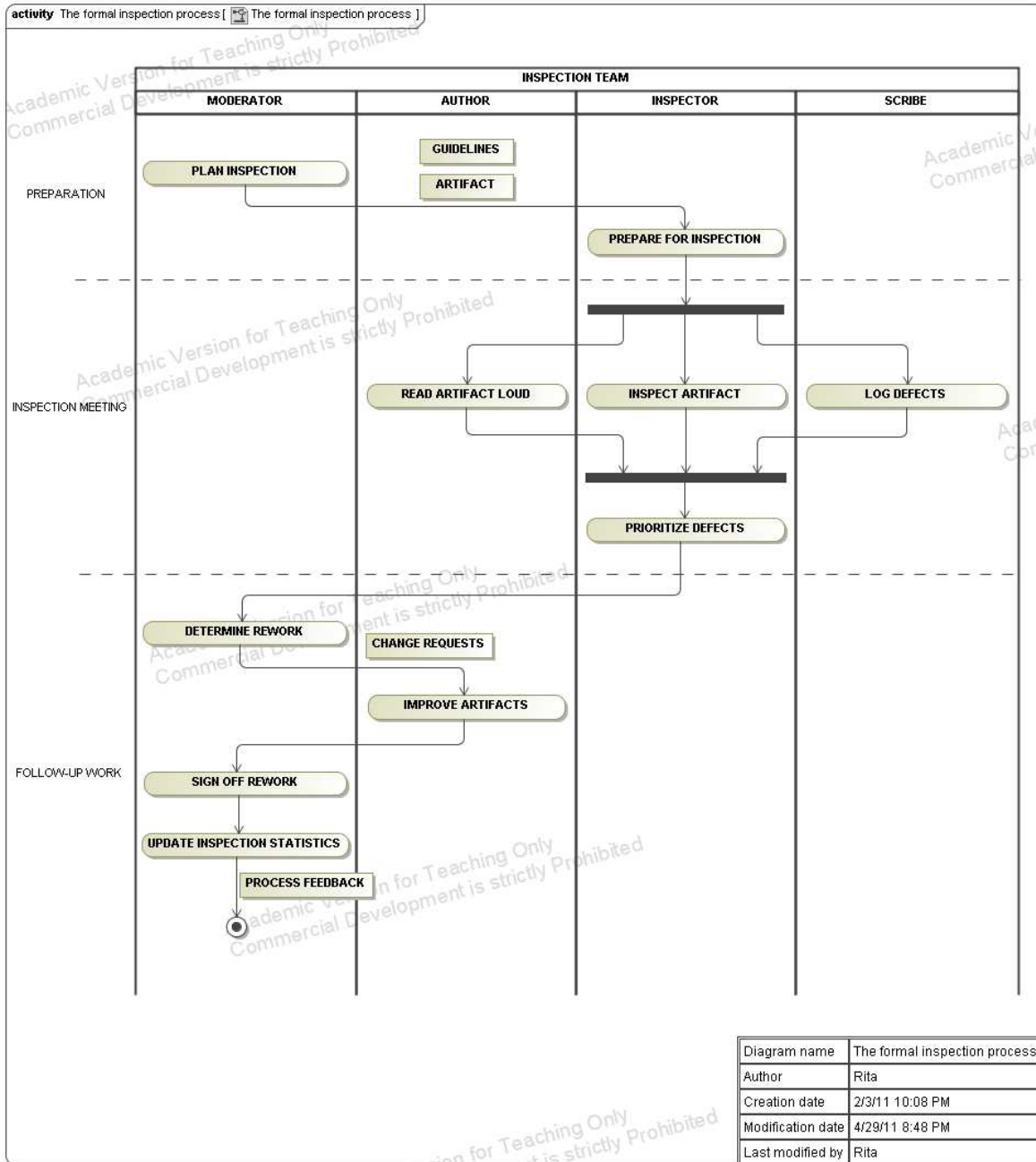


Figure 2 The formal inspection process

Preparation phase consists of planning inspection and preparing for inspection. To plan the inspection is moderator's responsibility. In this stage moderator is responsible for picking up the applicable guidelines for the artifact, sets deadlines and inform inspection team about the decisions. Moreover, everything has to be documented. When artifact and guideless are available inspectors are responsible to inspect the artifact (note errors, questions, comments). Prepared documentation has to be submitted to moderator. The last step of preparation phase is to collect all individual inspectors' comments and prepare for the meeting.

During the meeting author reads artifact, inspectors inspect artifacts and scribe log defects. In the end of the meeting defects are prioritized. After the meeting follow-up work phase starts. The first step is done by moderator. Moderator is responsible to determine rework. Then improvement of artifact is done by author. Improvement of artifact has to be signed by moderator. Finally inspection statistics are updated and inspection is over.

Using inspection process number of mistakes and faults can be found, correcting those mistakes the quality of the inspected artifact increases. Fagan inspection is the first inspection technique which were started to use. Later on there appear number on inspection techniques which had the same base as Fagan inspection with minor number of changes.

Moreover, there is a list of existing process to find faults in artifacts with its advantages and disadvantages. The most popular and often used are walkthrough and peer review techniques (Wieger 2011):

- Walkthrough is a group activity in which artifact author guides the discussion of the review. Walkthroughs are less rigorous than formal inspections, they are less successful at detecting faults than formal inspection. Walkthroughs includes all important aspects but not necessary details. Walkthrough is a review in the form of report.
- A peer review is a group activity in which author plays passive role. Process and content of the peer review may be different depending on the purpose of the profession and the purpose of the review. Artifact is send for experts of the field, later on the evaluation is returned highlighting and noting weaknesses, problems and providing suggestions for improvement.

Walkthrough and review techniques are less formal than formal inspection and less effective at identifying defects. Walkthroughs and reviews are peer group discussion activities without much focus on defect identification and correction. There is not so much focus on quality improvement compared with formal inspection.

All mentioned methods starting from Fagan inspection (formal inspection), walkthrough and peer reviews can be called reviewing techniques with their advantages and disadvantages. Group review with individual preparation, formal inspection with individual preparation and individual peer desk-check is presented in the Table 9 (Wieger 2011):

Table 9 Group review with individual preparation, formal inspection with individual preparation and individual peer desk-check

Advantages	Disadvantages
Group review with individual inspection preparation	
<ul style="list-style-type: none"> • Multiple participants can find more defects; • Inspection meeting can lead to finding more defects. 	<ul style="list-style-type: none"> • Most defects are found redundantly and during the preparation stage; • Cost of review is high because of multiple reviewers.
Formal inspection with individual preparation	
<ul style="list-style-type: none"> • More coverage of artifact; • Find the most defects this way. 	<ul style="list-style-type: none"> • Slower procedure compared with group review; • Necessity to prepare, knowledge or guiding the process, different tasks for different roles; • Accurate preparation before inspection meeting; Cost is even higher than group review because of multiple inspectors and slow procedure.
Individual peer desk-check	
<ul style="list-style-type: none"> • Only one reviewer, cheaper; • Works well if reviewer is experienced; • More comfortable for artifact author because of no need to discuss and argue. 	<ul style="list-style-type: none"> • Single person plays all roles, can be difficult to manage; • Author is not there to answer questions and participate in the discussion; • Necessity to have follow-up session to inform author about detected faults; No group synergy.

All mentioned processes have their own pluses and minuses. The main challenge comes when there is a need to find the right way to inspect the artifact.

3. FIT design

The main idea of creating the formal inspection tool is to support all stages of inspection process, support any type of document, allow individual preparation, collect data, and include supporting documentation and guidelines.

3.1. FIT use case diagram

Formal inspection tool will be created considering the formal inspection process. The process and sequence will be the same. Difference will appear in the way of noting and collecting information. The responsibilities and tasks of formal inspection team members will be supported by the tool (Figure 3).

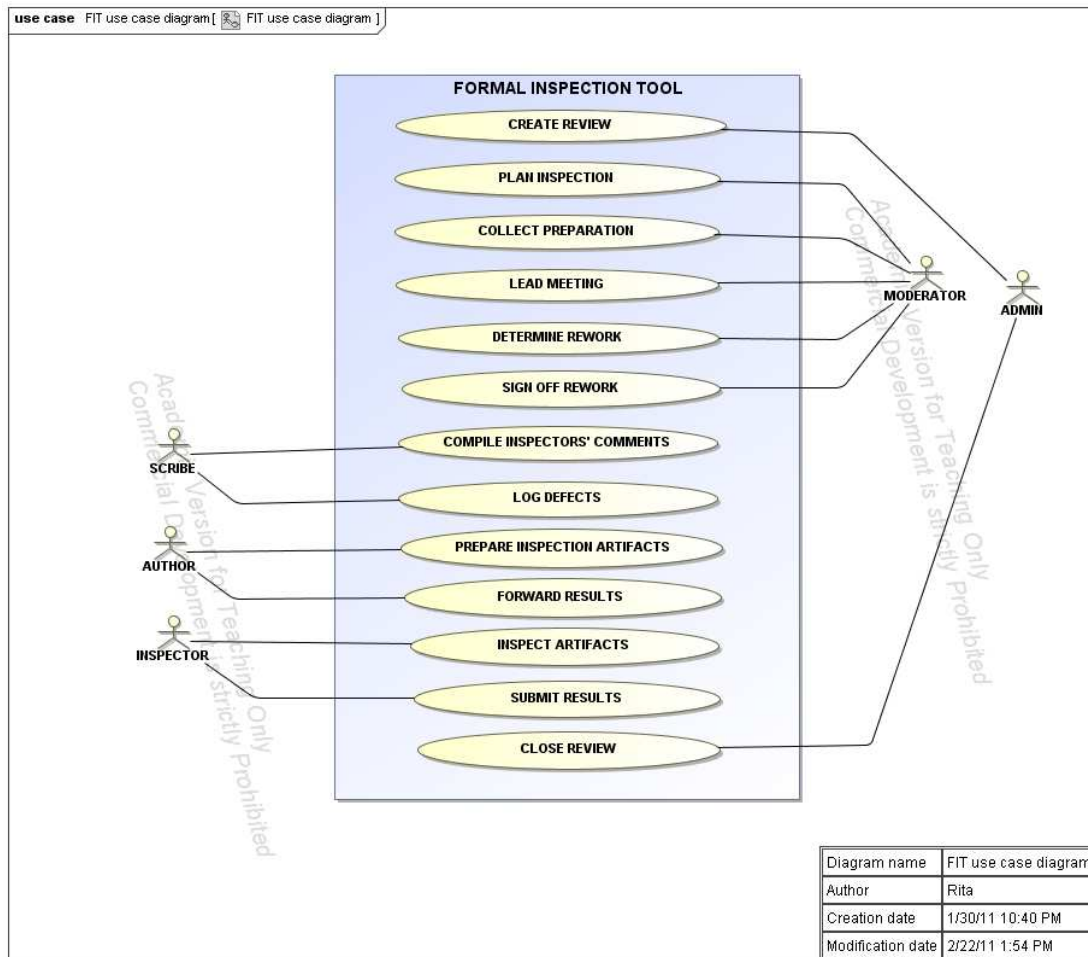


Figure 3 Use case diagram of FIT

It is important to mention that new actor appears in the formal inspection process, it is admin. Admin will have two main functions, create review and close review. Actors are divided into two groups: moderator and admin who are responsible of planning and creating the inspections; scribe, author and inspector following the inspection process and fulfilling the set tasks. Description of the main formal inspection functions:

- Create review allows for admin to create a review, with a possibility to specify start, end dates of inspection;
- Plan inspection allows for moderator to create an inspection process summary (project, time table, participants);
- Collect preparation allows for moderator to review individual inspection preparation results before inspection meeting;
- Lead meeting allows for moderator to enter and edit collected and compiled individual inspection results during the inspection meeting;
- Determine rework allows for moderator to resolve all defects which were found by inspectors and provide material for author for rework stage;
- Sign of rework step allows for moderator to accept or reject improved artifact results submitted by author;
- Compile inspectors' comments allows for scribe to compile and consolidate inspectors' individual inspection preparation results. If there is a high number of comment moderator can assign scribe to compile and consolidate the individual inspection results before the inspection meeting;
- Log defects allows for scribe and moderator to take notes during inspection meeting, edit comments;
- Prepare inspection artifacts allows for author to upload and delete artifact, as well as to upload and delete guidelines for moderator;
- Forward results allows for author to forward inspection results and improved artifact to co-authors;
- Inspect artifact allows for inspectors to enter individual inspection preparation results;
- Submit results allows for inspectors to submit individual inspection preparation results;
- Close review allows for admin closing review at any stage of formal inspection process.

FIT will allow making all main functions of formal inspection online. The only difference will be the way of making the formal inspection process but not sequence.

3.2. Inspection process using FIT

In the section 2 there were described formal inspection process based on paper forms. During all phases of this process it is necessary to fill the formal inspection forms. Moderators have to fill the inspection process summary, the inspection preparation summary and additional rework assignments. Inspectors are responsible for filling individual inspection preparation, and additional comments forms.

Inspection team will need to follow the main process steps mentioned in section 2 in order to use FIT. Difference will appear when entering, distributing data, and organizing process. Activity diagram using FIT gives an overview over the formal inspection process using online based formal inspection tool (Figure 4). Process starts from creating the review and finishes when administrator closes the review. All processes between those two stages are done by moderator, author, scribe and inspectors. There is one more additional actor – administrator. Administrator's main responsibilities are to create the review and close it. More or less the main functions of other actors stay the same. However, functions will not be made on paper forms but in digital format.

When review is created the moderator can start planning inspection. Afterwards the preparation for inspection meeting is following. During the preparation all of the actors have their responsibilities. Artifact has to be uploaded by author. Guidelines have to be uploaded by moderator. When artifact and guidelines are available the inspectors have to inspect artifacts and upload individual inspection results. Moderator is responsible for assuring that all inspectors contribute to individual preparation process. When individual preparation is confirmed by moderator, scribe compiles comments.

When preparation is done, inspection meeting starts. Author reads artifact, while moderator leads the meeting at the same time editing the compiled comments. Inspectors inspect the artifact and participate in the discussion. Comments are visible for all inspection team members. Scribe helps moderator if it is necessary to log defects. While the defects are logged all inspection team can see the changes and at the same time discuss about them. Later moderator prioritizes the defects and determines rework.

When author improves the artifact it is necessary that moderator signs off the rework. Next author can send the rework to co-authors. When inspection is updated the review can be closed by admin. Moreover it is important to mention that admin can close review at any stage.

Moderator does not have power to close the review. But after the indicated deadlines the moderator can collect preparation and note if some of the inspectors did not contribute in the individual inspection.

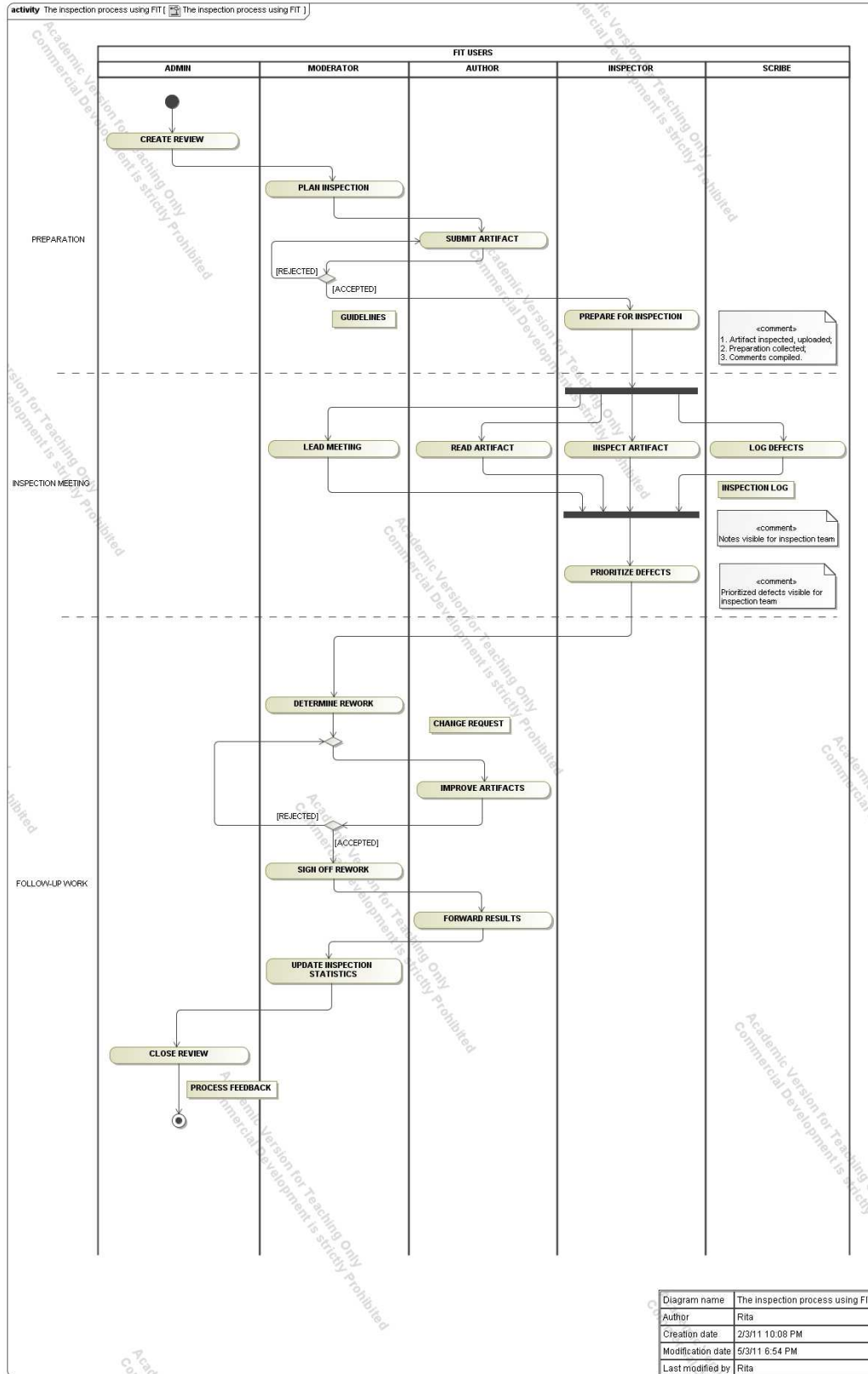


Figure 4 The formal inspection process using FIT

Comparing activity diagrams in Figure 2 and Figure 4, the activity diagram for formal inspection process using FIT is different. New processes in preparation stage and follow up rework appear. These processes are to create review and close review. Moderator has the possibility to reject the artifact if it does not pass the criterias. In inspection meeting stage moderator gets the possibility to lead meeting using FIT. Moreover, moderator can confirm or reject the improved artifact. If improved artifact was rejected the process goes back to determine rework stage. If rework is signed, author can forward results to co-author.

3.3. Life cycle of review process using FIT

The state machine diagram of life cycle of review process (Figure 5) describes the behaviour of the system and shows all states in the sequence. States of the life-cycle of review are: review created, inspection planed, individual inspection results submitted, preparation collected and approved, results compiled, defects prioritized, rework determined, artifact improved, rework signed, results forwarded, review closed. It is important to note that review can be closed at any state. User case diagram, activity diagram and state machine diagram presents the review of the process.

3.4. Database structure

Database structure (

Figure 6) presents a detailed data model of the database which will be used for FIT. Database will consist of user, review, comment, role and artifact information. It is important to note that one user will have the possibility to participate in more than one review having different roles. All reviews will have assigned artifacts (inspected document, guidelines). Comments made by inspectors will be collected and will have identified severity (major or minor).

The database structure will be used in FIT creating process. In order to create review it will be necessary to indicate title, group name, course, start date, preparation due and inspection meeting date. To add user to the data base it will be necessary to enter user's first name, last name, student ID, e-mail address, username, password and, in the end, to indicate the role.

Comments are created by inspectors. When comment is created it will hold comment state, page number, line number, content of the comment, severity and author who made a comments information.

Data types used in the database are: integers, strings (varchar), dates and enumeration. Integer allows numbers between -32,768 and 32,767. Varchar allows variable length strings, including letters, numbers and special characters, and can store up to 255 characters. Date allows date format, in this case it was

chosen YYYY-MM-DD. Enumeration allows to enter a list of possible values which have to be chosen (w3schools 2011).

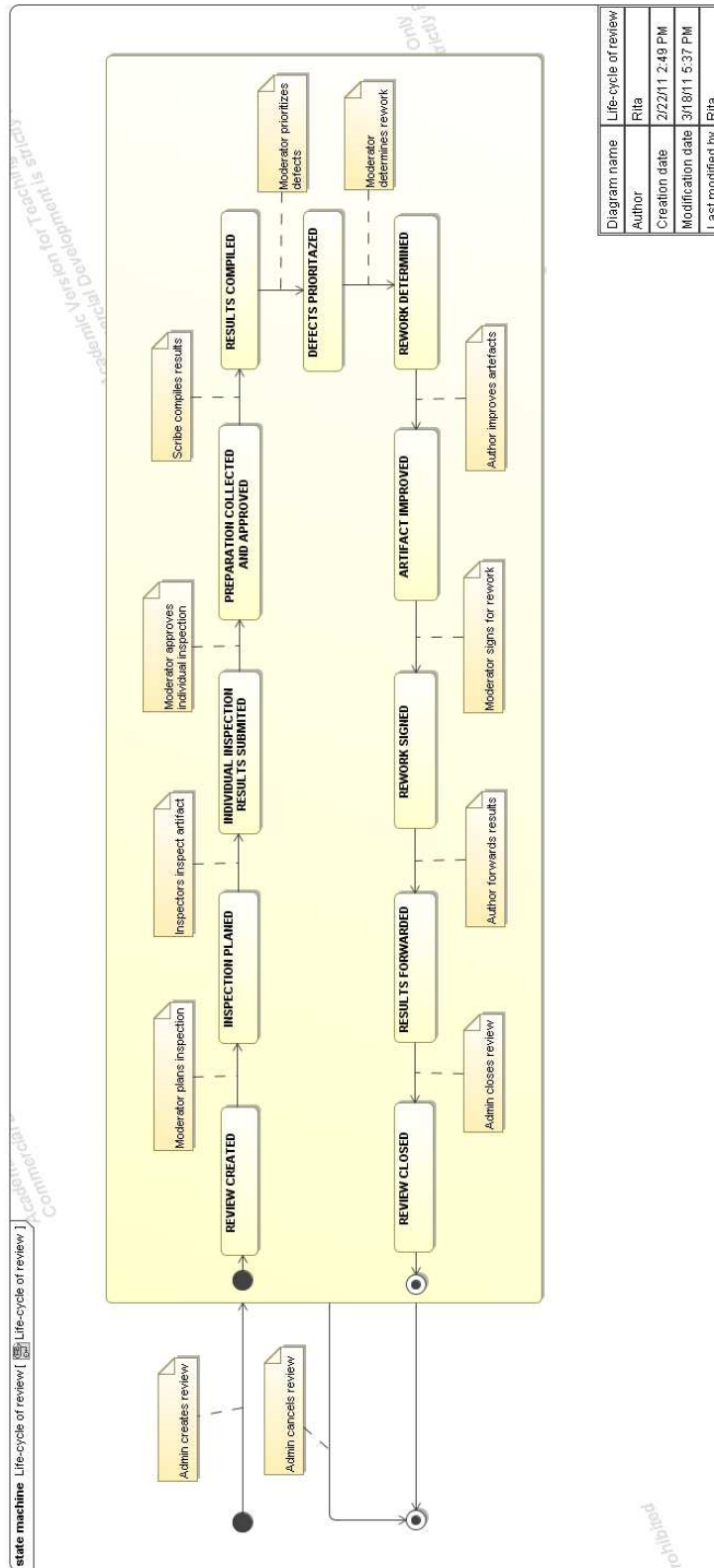


Figure 5 Life cycle of review process using FIT

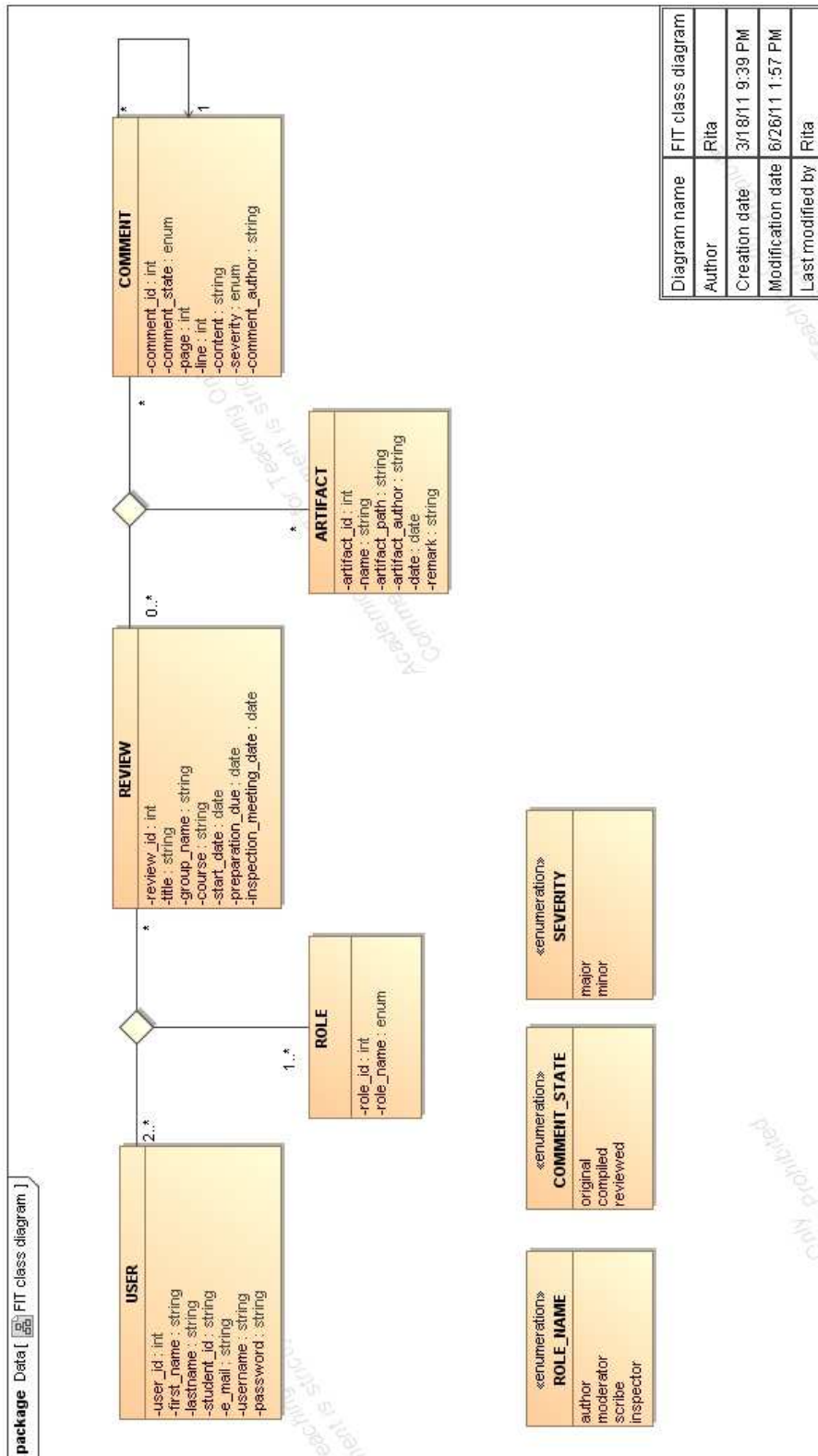


Figure 6 Database structure

3.5. Technology

To realize an idea of creating an online inspection tool few technologies were chosen: HTML, PHP, MySQL, CSS and JavaScript. HTML was chosen because it is the predominant markup language for creating the web pages. It works perfectly with PHP, MySQL and JavaScript. PHP was chosen as a simple and powerful scripting language designed for web development and MySQL a relational database management system as solid and reliable database server. The main advantages of PHP and MySQL are (Quigley 2006):

- Open source projects – there is no license fee associated with using PHP and MySQL;
- Build-in functions – the official PHP web site provides documentation explaining how to use all of the functions which are currently available;
- Run on many platforms, including Linux, Windows, Mac OS X, Solaris and etc.;
- Developer community – easy to find the solution, to solve the problem.

In order to design FIT tool CSS (Cascading Style Sheets) will be used. CSS is a style sheet language. CSS will help to separate document content from document presentation, including the main design elements: layout, colours and fonts. The separation will improve content accessibility, provide more flexibility and will allow more control possibilities in changing some specific presentation characteristics. Hence, it will be easier to use the same formatting for multiple pages. The complexity will be reduced. The main advantages using this technology solution are (Wikipedia 2011):

- Saves time – style details are specified in one page;
- Pages will load faster because of less code;
- Easy maintenance – if there is a need to change the style of the specific element it is done in one page;
- CSS has wide array of attributes.

To make FIT more user friendly and to add more useful features, JavaScript will be used as well. JavaScript is a prototype based, object orientated scripting language. It is implemented as part of a web browser with purpose to provide more user friendly interfaces and dynamic websites. The main advantages of using JavaScript are (Ezine 2011):

- JavaScript is fast because functions runs immediately;
- Get along with other languages – can be used with scripts written with HTML, PHP and other languages;

- No need to download a plug-in (in comparison with Flash and Java) – all browsers supports JavaScript;
- Possibility to create advanced user interfaces;
- Developer community.

All mentioned advantages of HTML, PHP, MySQL, CSS and JavaScript (jQuery grid plug) gives a possibility to create a reliable, straightforward an online inspection support tool and achieve all raised goals and sub goals.

Detailed description of the main functions which will be supported by the FIT tool is given in Table 10. Functions are arranged by the FIT tool pages and in the end there is a separate common section which describes the functions applied in all of them.

Table 10 Main functions of the FIT and implementation possibility

	Function	Description	Implementation
Front	Login	User logs in to the specific review, with the assigned role.	HTML/PHP/MySQL
	Select review	User selects the specific review to switch to if she/he is assigned to more than one.	HTML/PHP/MySQL
	View info	Get the main information about FIT.	HTML
Admin	Create review	Administrator creates review entering the main information (project type, project group, project course, start date).	PHP/MySQL/jQgrid
	Close review	Administrator can close review at any stage.	PHP/MySQL
	Create review team	Administrator assigns moderators for created reviews by entering moderators' first name, last name, student id, e-mail and role.	HTML/PHP/MySQL
Moderator	Remove review	Administrator removes passed reviews.	PHP/MySQL/jQgrid
	Upload guidelines	Moderator uploads guidelines.	HTML/PHP/MySQL
	Create review team	Moderator assigns inspection team (author, inspectors and scribe) by entering first name, last name, student id, e-mail and assigning role.	PHP/MySQL/jQgrid
	Enter review info	Moderator enters preparation due and meeting dates for individual inspection.	PHP/MySQL/jQgrid
	Change status of comment	Moderator changes status of comment: proposed, rejected and accepted.	PHP/MySQL/jQgrid
	Edit comment	Moderator can edit comments entered by inspectors.	PHP/MySQL/jQgrid
	Delete comment	Moderator can delete comment form the comments list.	PHP/MySQL/jQgrid
Author	Upload artifact	Author uploads artifact.	HTML/PHP/MySQL
	Upload rework	Author uploads rework after inspection process.	HTML/PHP/MySQL
	Review comments	Author can review entered comments before the inspection meeting in order to prepare for discussion.	PHP/MySQL/jQgrid
Inspector	Submit individual inspection	Inspector submits inspected artifact comments entering page, line numbers, comment and	PHP/MySQL/jQgrid

		comment type.	
	Import comments	Import comments as excel sheet.	PHP/MySQL/jQgrid
	Define type of comment	Inspector defines the type of the comment (major, minor or remark).	PHP/MySQL/jQgrid
Scribe	Change status of comment	Scribe changes status of comment: original, compiled or reviewed if it is assigned by Moderator.	PHP/MySQL/jQgrid
	Delete comment	Scribe can delete comment form the comments list if it is assigned by Moderator.	PHP/MySQL/jQgrid
Common	Log out	All users can logout from the logged page and returns to the main page.	HTML/PHP/MySQL
	View comments	All team members can view individual inspection comments.	PHP/MySQL/jQgrid
	Tab	Main tasks and responsibilities separated in tabs, in order to save loading time.	JavaScript
	Table sorter	Table sorter allows sorting information in the table.	jQgrid
	Alert of removing	Alert: are you sure you want to remove?	jQgrid
	Scroll	Scroll frame inside the page.	jQgrid
	View review info	View review info (project type, project group, project course, start date, preparation due, meeting date).	PHP/MySQL/jQgrid
	Download artifact and guidelines	Download specific artifact or guidelines.	HTML/PHP/MySQL

Design decisions and the first sketches of the FIT tool are presented in the next section, presenting logical GUI design as well.

3.6. FIT layout and sketches

Before starting the development and implementation processes sketching, as fast and cheap type of prototyping, was chosen. Sketching will allow to have the first impression how FIT should look like. At this stage it is quite hard to predict how users will interact with the tool, will it be easy or it will be challenging. It will help in finding the problems and defects in the tool. Usually it is easier to find defects when it is seen visually. The layout and the main functionalities were presented in the paper form depending on the main goals.

FIT tool layout is divided into three sections (Figure 7):

1. Tool section, which consists of FIT and DTU logo, login, user information fields;
2. Information section, which consists of inspection stages fields (preparation, inspection, rework);
3. Tasks section, which consists of review, team, artifact and remark fields.

Before the login, tool section of the FIT is the same for all users no matter to which review they belong or to which role they are assigned. In the tool section appears additional line which identifies the user role,

name, surname, review and role after the user logins. Login function is orientated in the top right corner not to confuse user. It is a standard position, widely used in the websites where it is required to login (e-mails, e-banking, social media websites). Sequence of text and tasks was chosen from left to write in order to keep the reading track.

Information section presents the main stages of the formal inspection (preparation stage, inspection stage and rework stage) including the time lines. Information section appears for all users except administrator. Administrator does not need to follow the sequence of the formal inspection, administrator is responsible for creating and closing the reviews.

Task section consists of review, team, artifact and remarks fields. These fields are seen for all users except administrator. The main difference in the task fields appears in the features which are allowed for one or another role. For example, moderator can add new member to the team and edit existing team members' information while author, inspectors and scribe has possibility just to review the information.

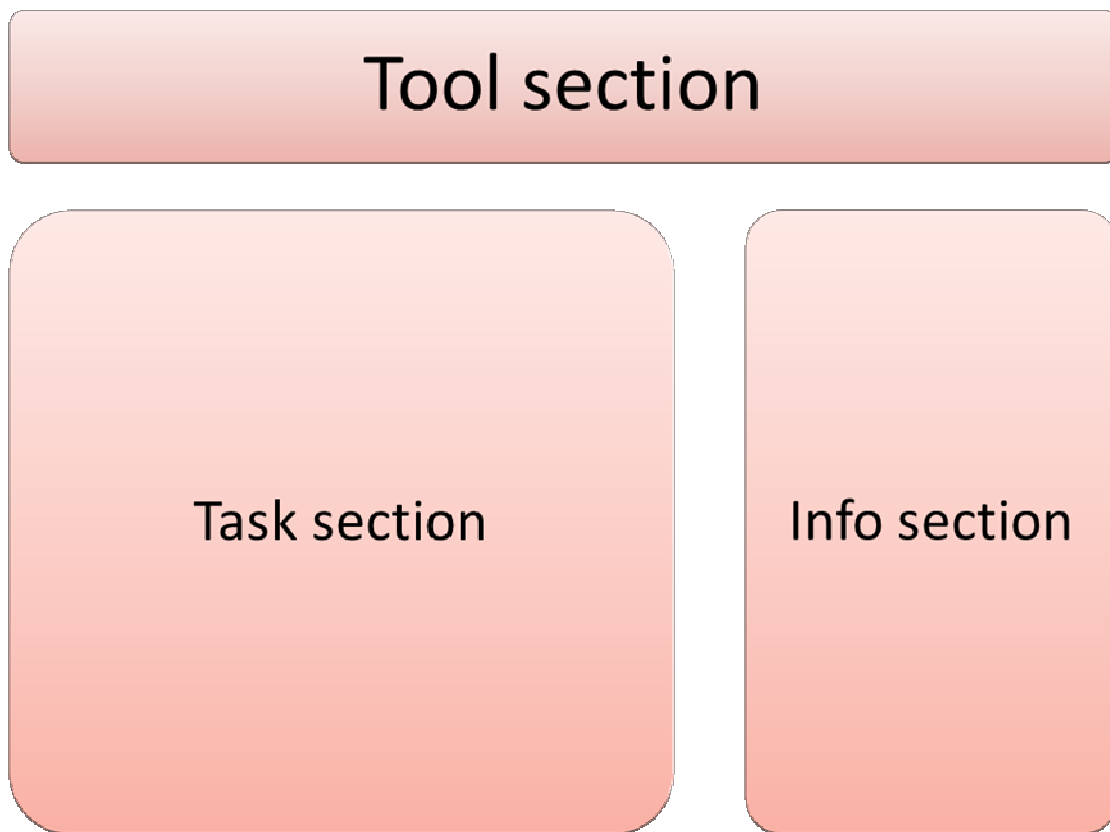


Figure 7 FIT layout

Few sketches which presents more detailed layout of the FIT tool are given in Figure 8 – Figure 19. Role is not defined in the sketches because differences appear in allowing editing, deleting, adding or importing data. Functions edit, delete, add or import data activation depends on the role. After log in all team members sees the same layout of the page and navigation in it kept the same as well in order not to confuse the user if user has different roles in different reviews. If user wants to switch to other review s/he needs to identify it in the right top corner (Figure 8). There is no necessity to log out and login again in order to go through the assigned reviews.

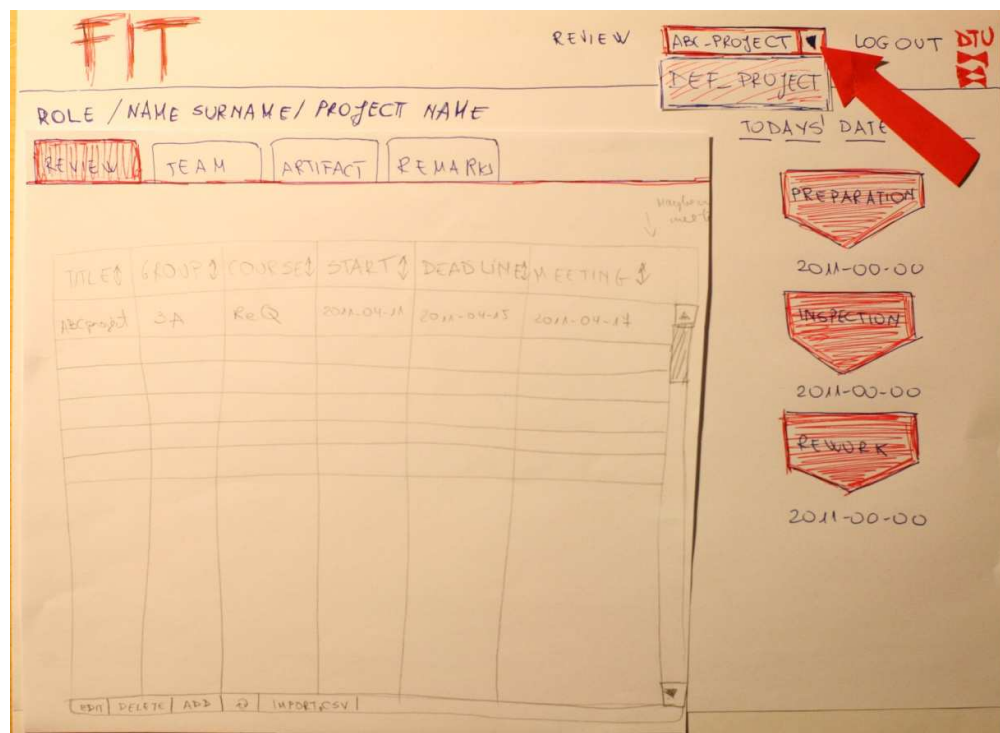


Figure 8 General layout of the FIT changing review after login

When the review field is chosen table with the information about the assigned review appears (Figure 9). Title of the project, group name, course name, inspection process start date, deadline for individual inspection submission and meeting date are presented allowing editing, deleting, adding and importing data depending of the role. The same review table will be used for administrator just presenting all existing reviews. Table has scroll part in order to present information in one page. Moreover sorting feature is possible, like sorting by date which will allow to tracking table information if it will be necessary. This feature is important, especially for administrator because of the high number of reviews.

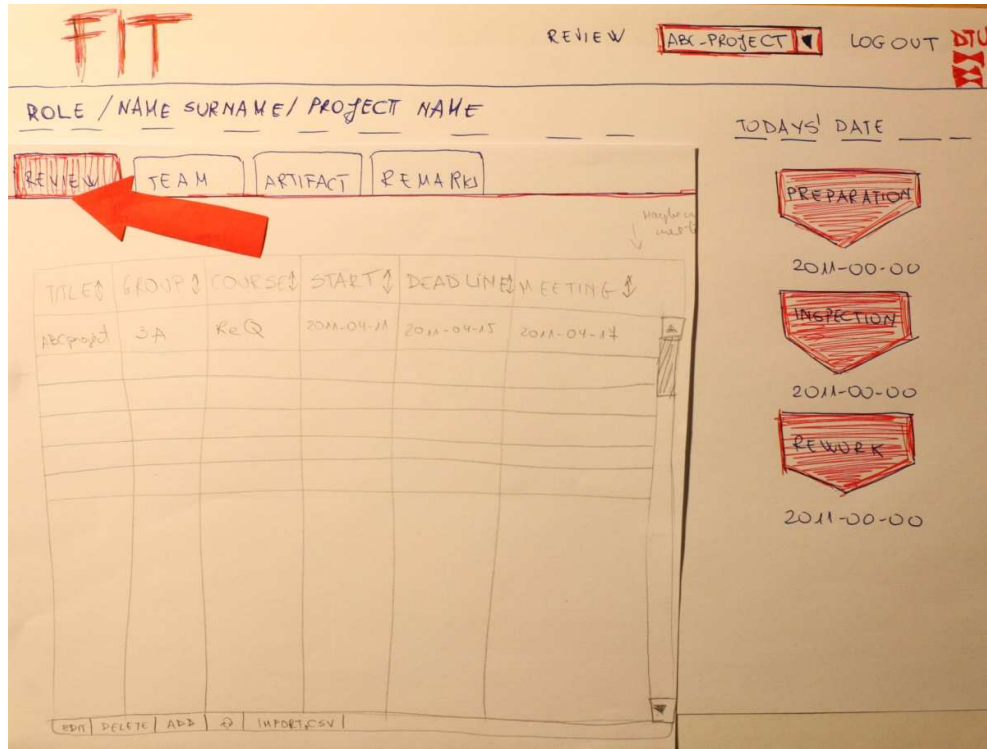


Figure 9 FIT review section

When the team field is chosen table of team members who belongs to the same review appears (Figure 10). Information of users like the names, surnames, student ids, e-mails, reviews and roles will be presented. Personal information will be allowed to edit in case if it was entered incorrect by moderator. The same layout table will be used for administrator adding username and password fields, presenting all existing users and allowing editing, deleting and adding data. Scrolling and sorting features are implemented in the table as well, in order ease tracking information.

When the artifact field is chosen table of artifacts appears, presenting name and file of artifact, date when it was uploaded, authors name and remarks in order to inform other team members about updates (Figure 11). Artifact table will present all artifacts, including documents for inspection, guidelines as well as rework documentation.

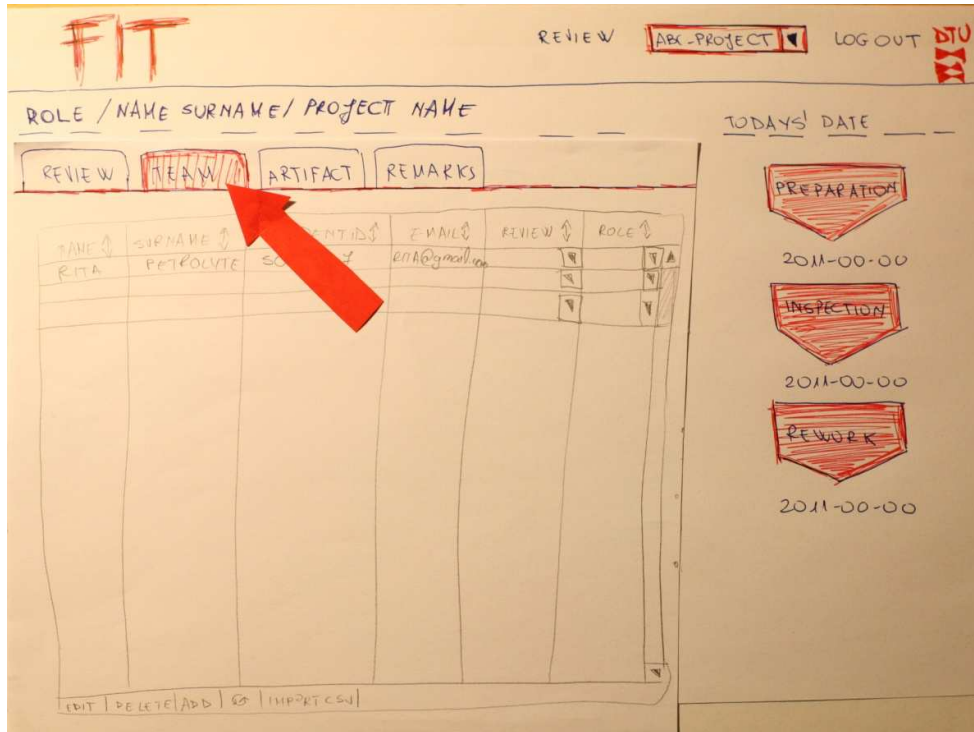


Figure 10 FIT team section

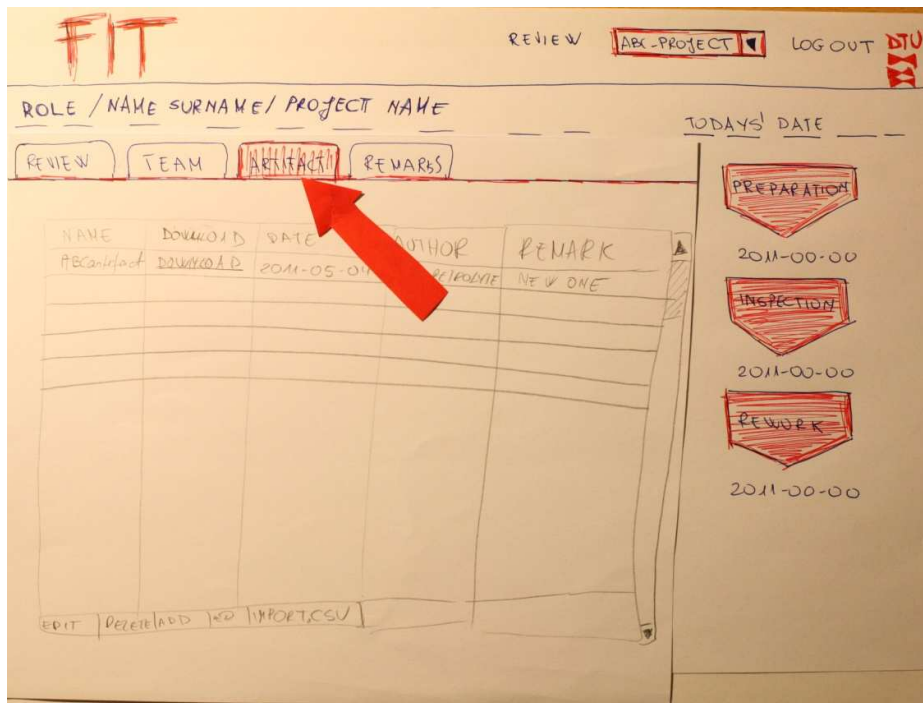


Figure 11 FIT artifact section

When the remark field is chosen table with all merged individual inspection results is presented, identifying the page number and line where mistake was found, comment context, type of the comments (major, minor), authors name and status of the comment (Figure 12). It is important to mention that sorting feature is necessary in the remarks table in order to ease compilation and consolidation of the comments. In this case it becomes very easy to find all mistakes founded in specific page and specific row, notice and remove the duplicates.

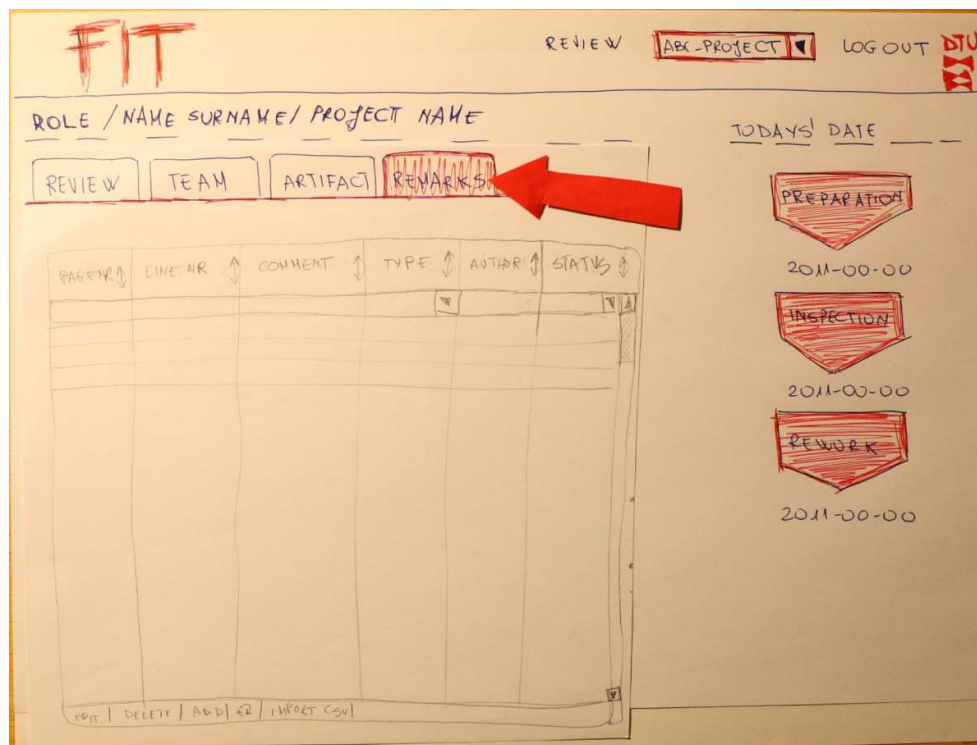


Figure 12 FIT remarks section

Status of the comment can be changed from original (original inspector comment before compiling), compiled (when comment is compiled before the formal inspection meeting) and reviewed (when comment is reviewed during formal inspection meeting and accepted by all inspection team (these reviewed comments will be presented in the rework stage)). Features as editing, deleting and adding are allowed as well. Importing data is one of the most important features. Comments can be made in .csv file and later on uploaded to the remarks table.

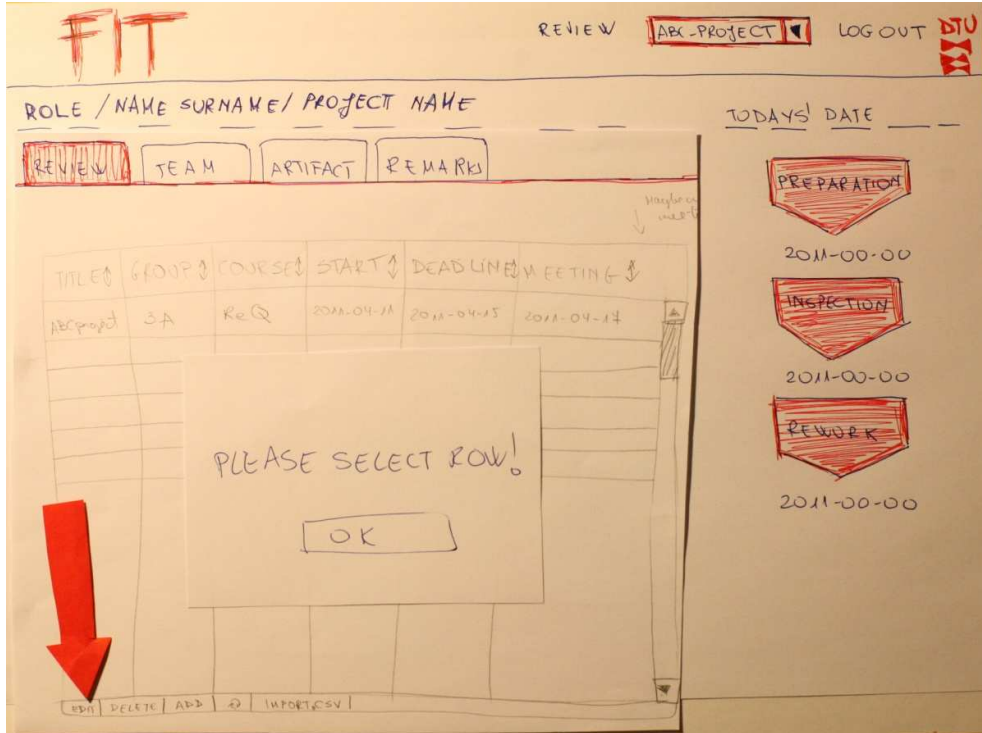


Figure 13 FIT selecting edit

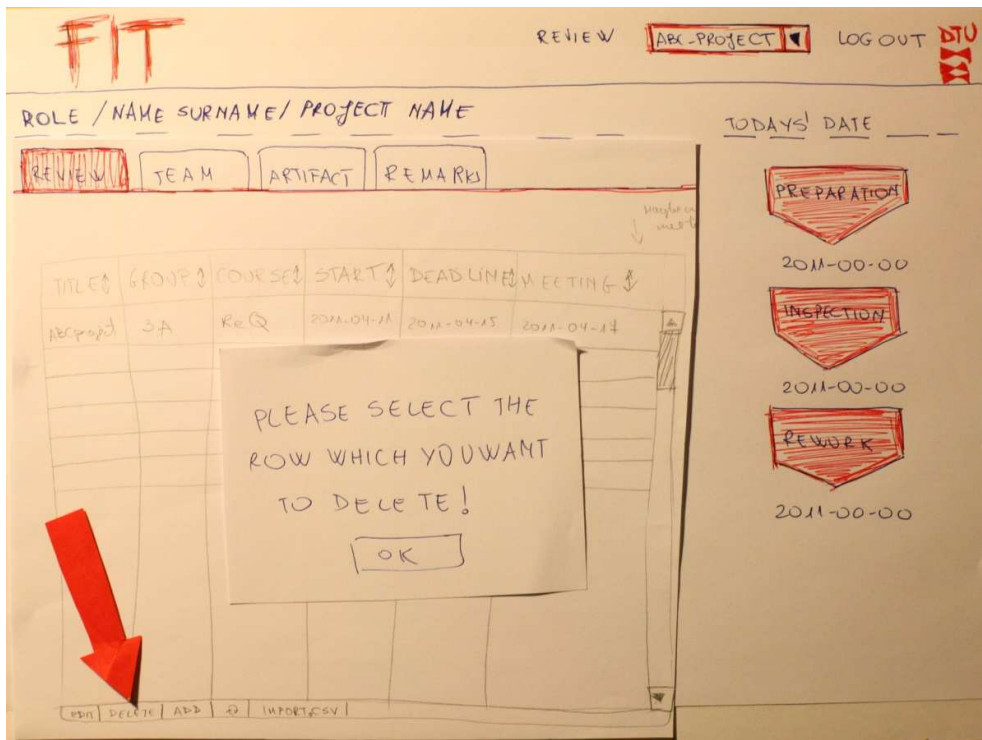


Figure 14 FIT selecting delete (1)

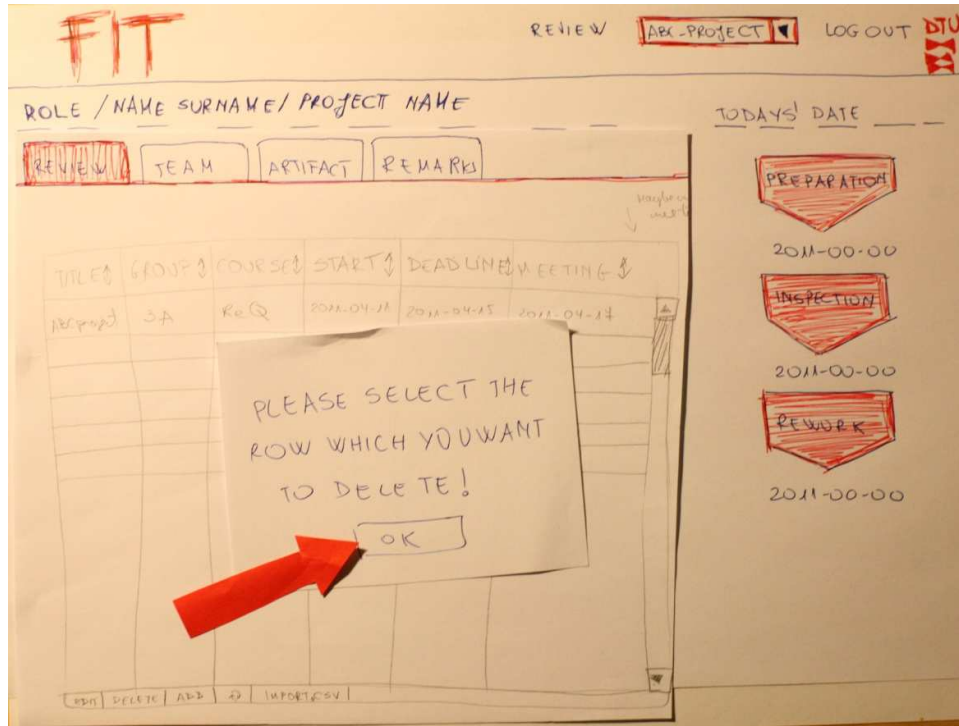


Figure 15 FIT selecting delete (2)

In order to edit or delete information from the table, the row has to be selected. If the button edit or delete are clicked before the row is selected the alert appears: "Please select row!". When the row is selected then edit and delete features can be activated. In order to avoid mistakes in deleting the additional alert ("Are you sure you want to delete this row?") appears requiring to confirm the deletion (Figure 13 – Figure 15).

The importation procedure of .csv file is presented in Figure 16 – Figure 19. When the import button is clicked the alert of informing about the file format which is allowed to import appears as well as fields to browse and upload .csv file. It is important to note that .csv files' data have to be separated with semicolon (;). Otherwise it will not be read and uploaded. After uploading the alert of successful upload will appear.

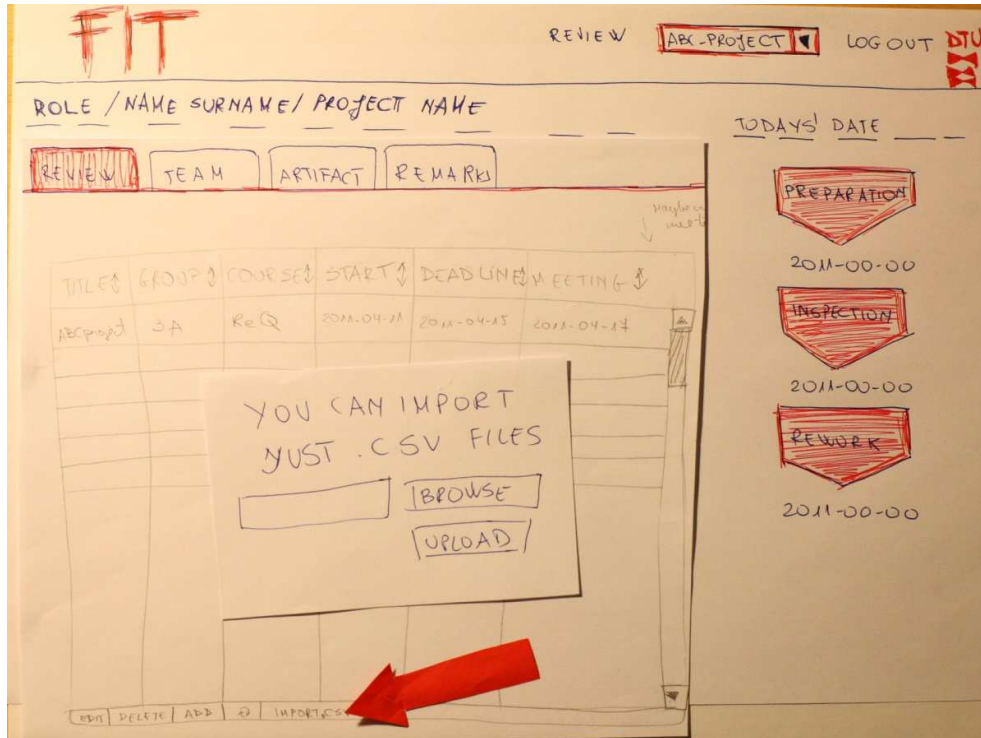


Figure 16 FIT import .csv file (1)

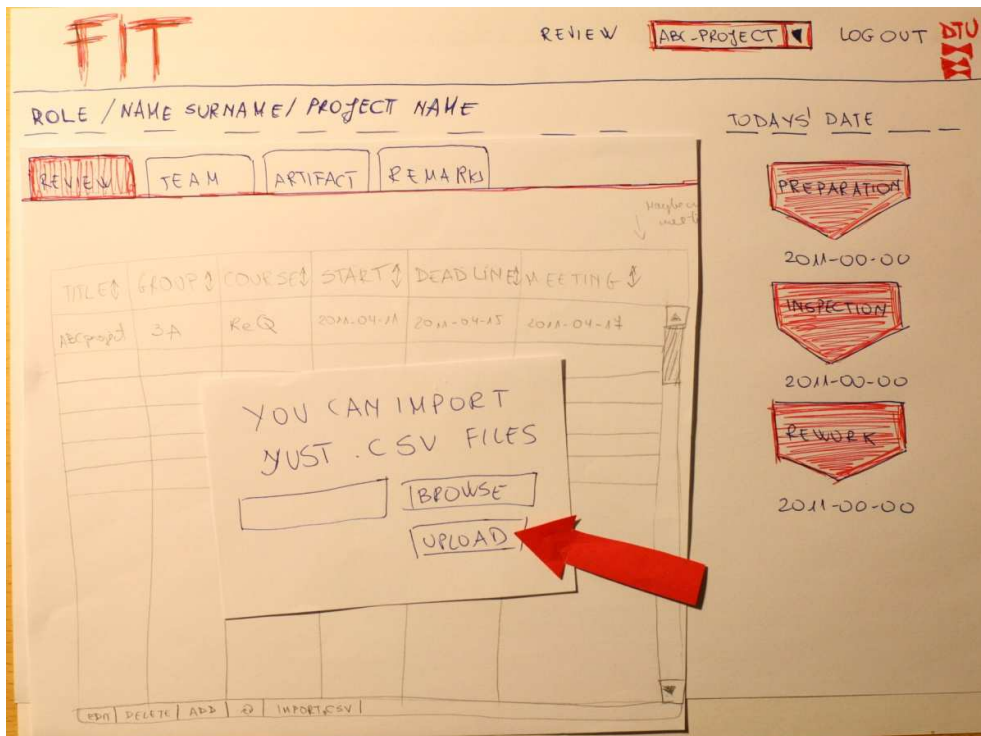


Figure 17 FIT import .csv file (2)

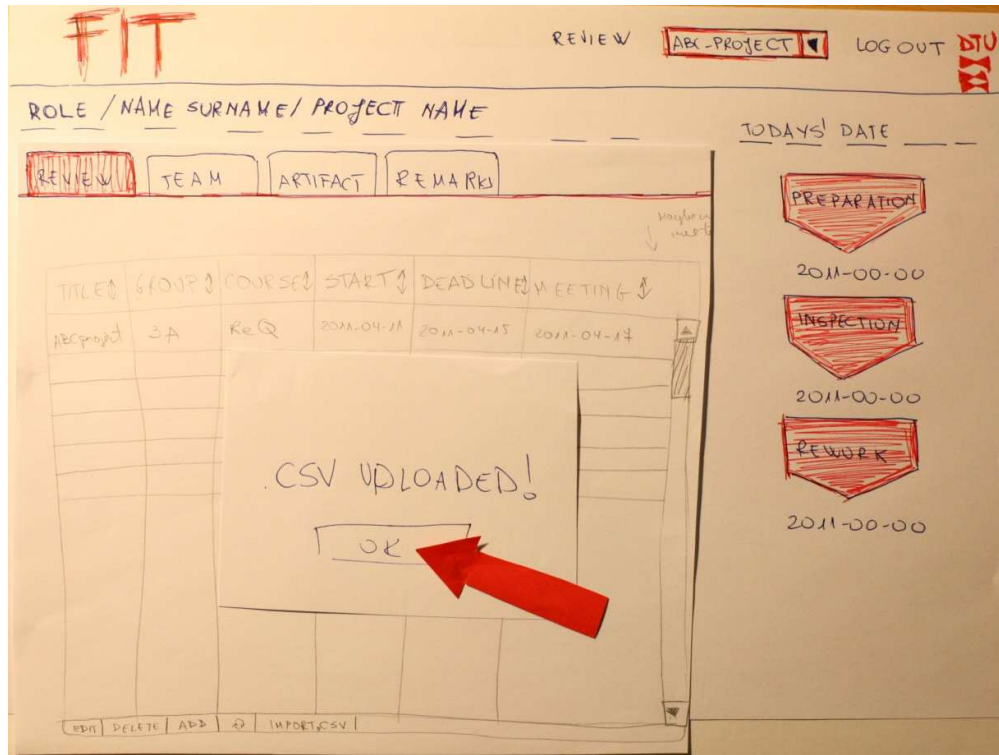


Figure 18 FIT import .csv file (3)

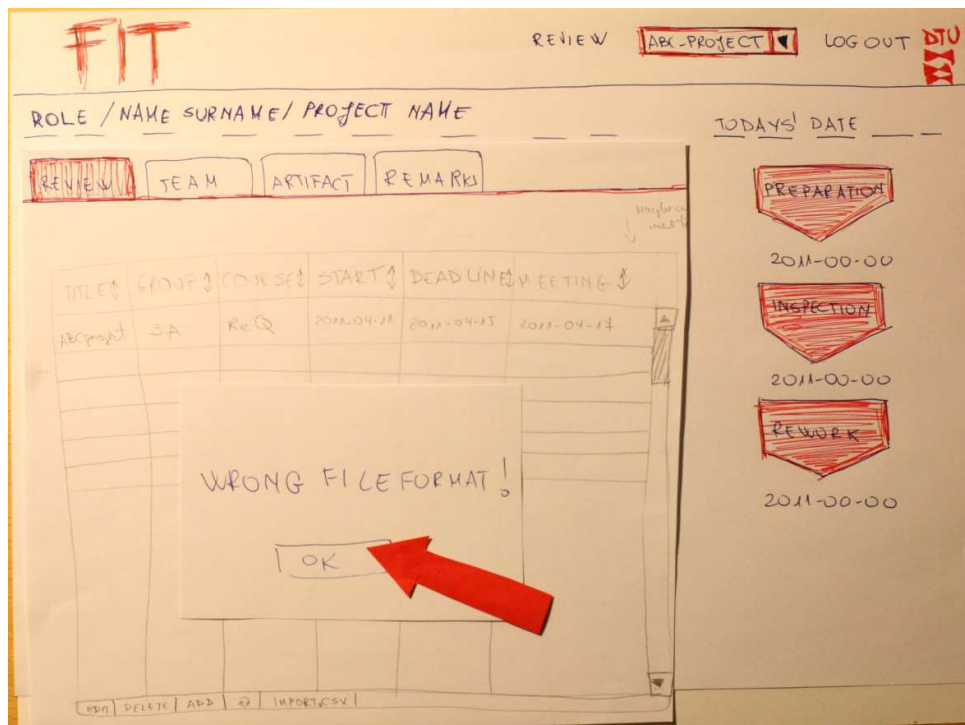


Figure 19 FIT import .csv file (4)

If the selected file was not .csv file the message about the wrong format appears and after pressing ok the action can be repeated again.

In order to save place and avoid unnecessary information in the creating tool, decision not to use descriptions about table fields was done. Pre readings about the formal inspection process and introduction about FIT are requirements to start using the tool.

3.7. Logical GUI design steps of using FIT tool

Sequence of the interaction with the FIT tool will be presented considering on the user role keeping the sequence of the formal inspection process.

Admin functions and tasks starts from creating review. Admin is responsible to create review entering the main information: title, group, course, start date, end date, meeting date and time. When review is created admin is responsible to assign moderator per each review. Admin can form all inspection team for specific review but usually this responsibility belongs for moderator. Entering first inspection team member with role moderator admin has to identify first name, last name, student id, e-mail address role and review.

When review is created and moderator assigned the other inspection team members have to be added to review. This responsibility belongs to moderator. After log in moderator page and adds all inspection team identifying first name, last name, student id, e-mail and role. When inspection team is formed moderator have possibility to set end date of individual inspection and set the date of the meeting in review tab. To make sure that all inspection team members are informed what are the main issues and considerations to inspect the artifact guidelines can be uploaded in artifact tab. Moderator can review uploaded artifacts and download them. Before inspection meeting moderator is responsible to compile individual inspection result. Individual inspection results of all inspectors (if they were submitted) can be reviewed and edited in remarks tab.

Author during inspection process is responsible to upload artifact using artifact tab, otherwise inspection process cannot start. When artifact is uploaded moderator informs all inspection team. During the preparation process author have possibility to track inspectors submitted comments in remarks tab. This feature gives possibility to prepare for inspection meeting thinking about possible answer for remarks. When author is prepared, inspection meeting discussions take less time. As all other inspection team members' author can see all review info in review tab. When inspection meeting is over author have to improve artifact and submit the rework using functions presented in artifact tab. Moderator can confirm or return to improve again if majority of the submitted comments were not taken in the consideration.

Inspector has possibility to download all review artifacts, see review info in artifact and review tabs. Individual inspection is done by selecting remarks field, entering page, line number of the artifact, comment, selecting type of the comment.

Scribe is responsible to help moderator compiling comments and preparing material for inspection meeting and this functions can be done using remarks tab.

Inspection team members' usage of functions is presented in Table 11, specifying what is allowed (x) and not allowed (-).

Table 11 Features allowed for specific role

Feature/Role	Moderator	Author	Inspector	Scribe
See review info	X	X	X	X
Edit info in review table	X	-	-	-
Change start date of review	X	-	-	-
Change date of individual submissions deadline	X	-	-	-
Change date if meeting date	X	-	-	-
See list of team members	X	X	X	X
Add team member	X	-	-	-
Edit team members info	X	-	-	-
Delete team member	X	-	-	-
Change the role of the team member	X	-	-	-
Sort list of team members	X	X	X	X
See list of artifacts	X	X	X	X
Upload artifact	X	X	-	-
Download artifact	X	X	X	X
Change remark of artifact	X	X	-	-
See list of remarks	X	X	X	X
Add remark	X	-	X	X
Edit remark	X	-	X	X
Delete remark	X	-	X	X
Change type of remark	X	-	-	X
Import remarks as .csv file	-	-	X	-
Sort list of remarks	X	X	X	X

Considering on the features which will be allowed for specific role the interface of the FIT tool tables will change. Basically the main change will be seen in the last row of the table where there are located add, edit and delete buttons. If user with the role which allows editing information of the table will see active lines while going through the table and by pressing on the edit button s/he will be allowed to change tables' context (Figure 20).

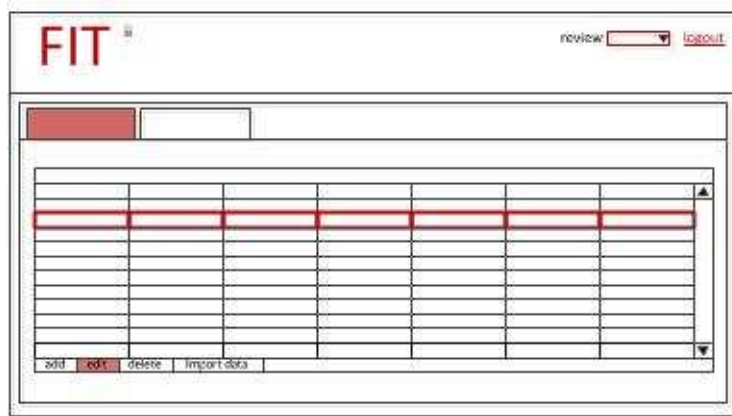


Figure 20 Edit function in FIT

If user wants to add additional line to the existing table s/he has to click on edit and after that additional line with empty, active fields appears in the end of the table. This feature is necessary because of noting new remarks and comments which appear during the inspection meeting (Figure 21).

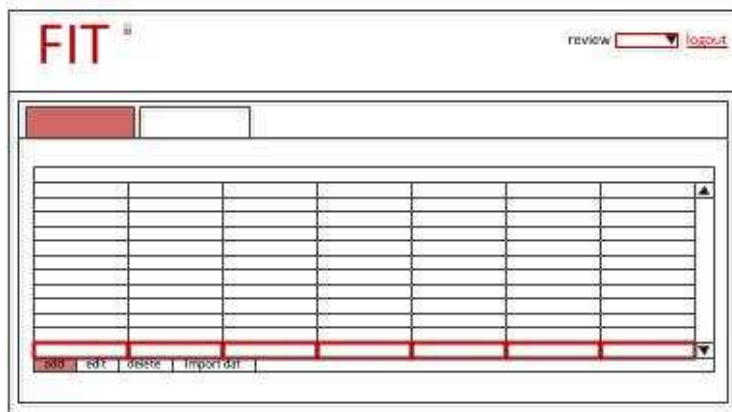


Figure 21 Add function in FIT

Identification about in which task tab user is will be presented in the changing the colours of the tabs. Active tab will be collared.

If user belongs to more than one review and more than one role s/he will have possibility to switch between them identifying it in the top right corner the review and role. It will be not necessary to logout

and login again. The same username and password will be used (Figure 20 and Figure 21). Preparation stage inspection stage and rework stage will present the time layout of the process in order to track deadlines.

Logical GUI design is presented in Figure 23, separating administrators and inspection team task. Administrator has two tabs which presents reviews and users. It is not reasonable to follow inspection process as administrator is responsible to create and close the reviews. Reviews and user information is presented in the table format to make it easier to review, add, edit or delete data. Alert messages used in FIT tool are given in Figure 22.

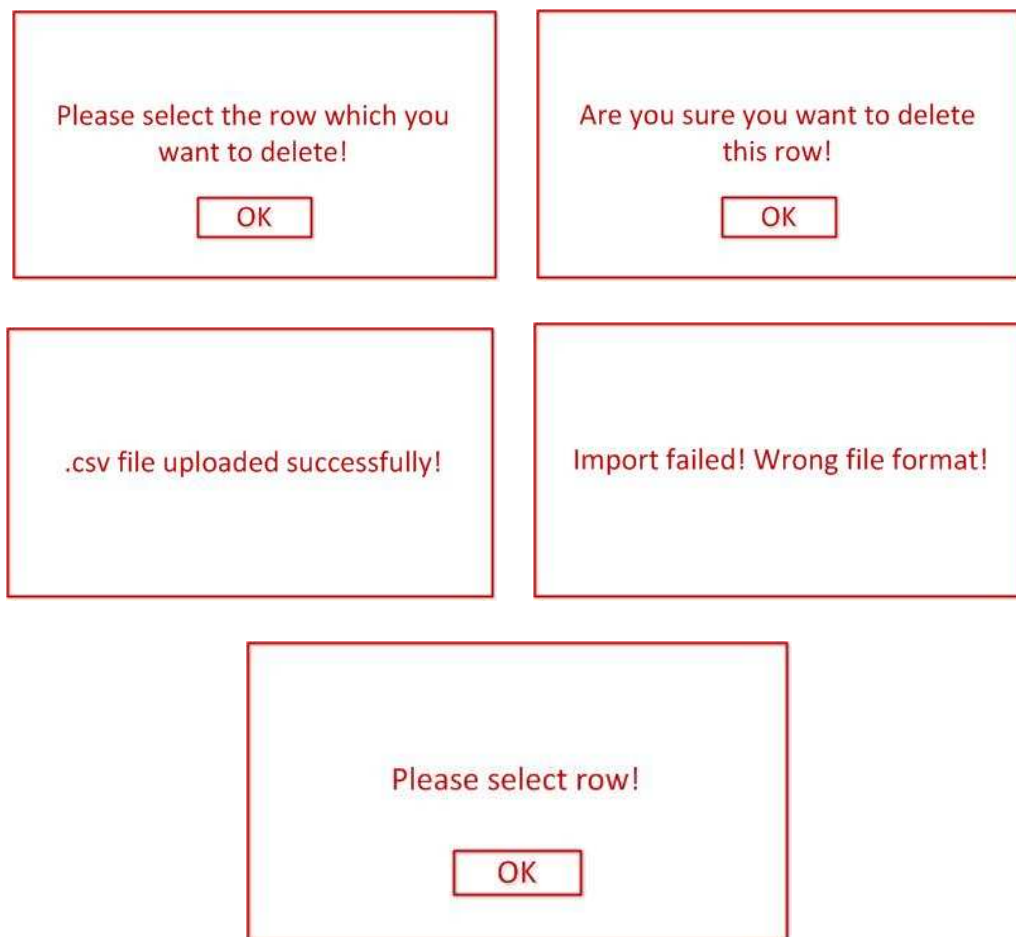


Figure 22 Alert messages of FIT

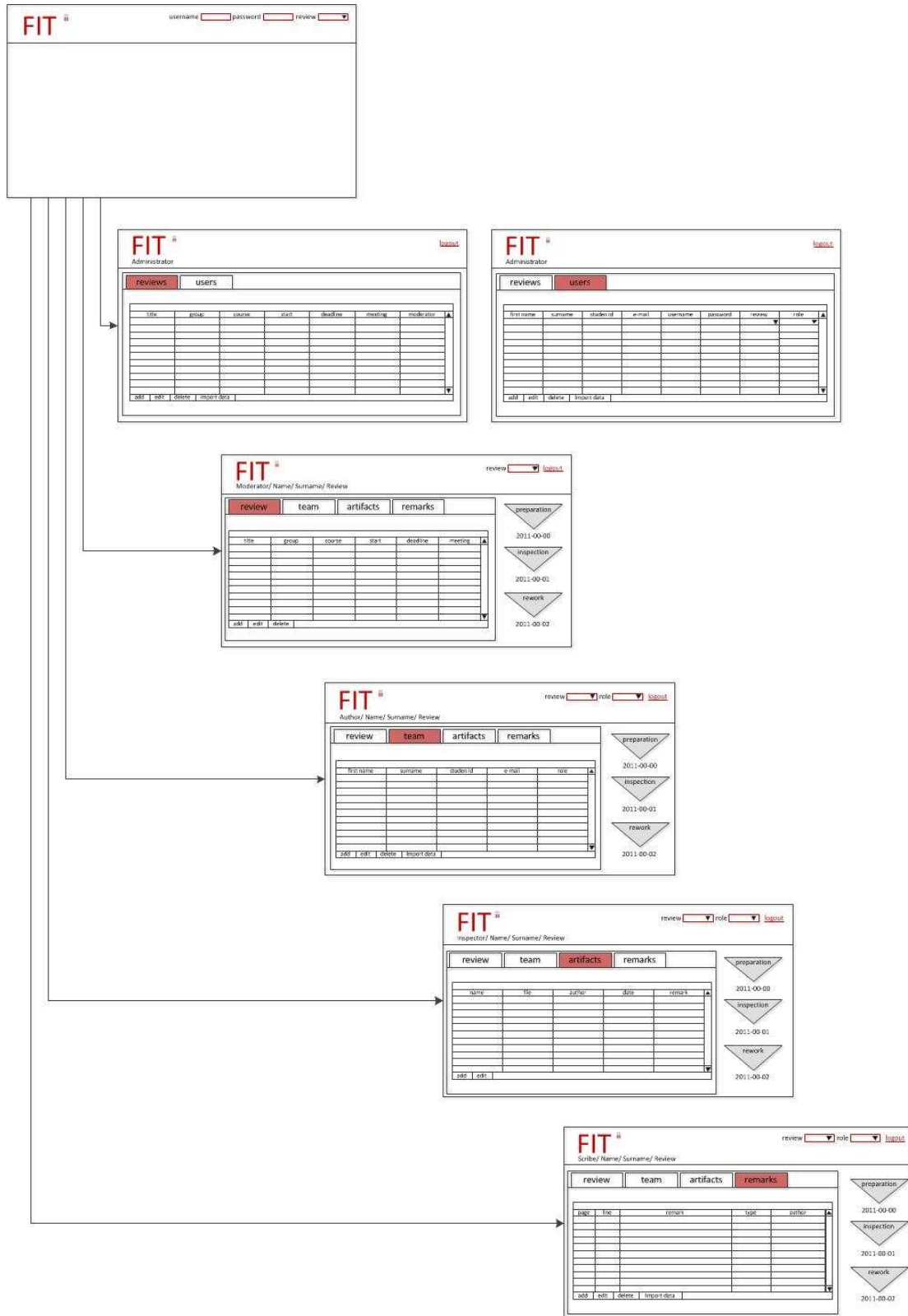


Figure 23 Logical GUI design

FIT tool GUI design is simple and easy to navigate but it is necessary to have knowledge about formal inspection process before starting to use FIT tool, what is the main tasks and responsibilities for each role and in which sequence it should be done.

Comparing FIT tool with other existing tools it has list of advantages. As some of the tools FIT allows document handling, individual preparation function, has a meeting support when all participants meets in the same room and can track changes of remarks table. Hence, FIT allows to inspect any kind of artifact, can be used for multiply reviews with the different roles (it is not necessary to run the tool again in order to switch the reviews or roles) and import data (remarks) what saves time and prevents of the one by one entering mistakes. FIT can be accessible where is internet connection. All information entered into FIT tool is visible for all team members what allows for them keep track of going process and follow time deadlines.

4. Implementation

Before implementation of FIT few PHP and MySQL books were studied (Atkinson 2004) (Beaulieu 2005) (Quigley 2006) (Taylor 2003). Implementation section will present technologies which were adapted and used in order to create FIT tool. There will be given detailed list of FIT tool files with description. Hence, data base structure will be presented.

4.1. Adapting technologies

As it was mentioned, to implement FIT tool HTML, PHP, MySQL, CSS and JavaScript were chosen. To start creating tool a XAMPP free and open source cross-platform web server solution stack packed was used as a testing environment. Used XAMPP 1.7.4 version consisted of (Apache Friends 2011):

- Apache 2.2.17;
- MySQL 5.5.8;
- PHP 5.3.5;
- phpMyAdmin 3.3.9;
- FileZilla FTP Server 0.9.37;
- Tomcat 7.0.3 (with mod_proxy_ajp as connector).

XAMPP was used as development tool to test FIT without access to the internet. XAMPP is very useful while creating dynamic websites as FIT using PHP and JavaScript.

CSS style was selected in order to separate FIT content form FIT presentation (layout, colours and fonts). There is a bunch of existing CSS templates provided in the internet. One of the trustful websites was chosen, freecsstemplates.org. The template Personified with fixed width and images matched the expected FIT layout. CSS license: *All of the free CSS templates on this website are released and licensed under the Creative Commons Attribution 3.0 Unported License. They are completely free (and royalty-free) to use for any personal or commercial purpose provided you credit me in the form of a link back to this website* (Free CSS Templates 2011).

The library of the jQuery was chosen in order to simplify the client side scripting of HTML. The jQuery is the most popular JavaScript library. It is free and open source. The jQuery is designed to make easier navigation in document, create animations, handle events and develop ajax applications. The jQuery is the library which helps to create powerful and dynamic website allowing (Wikipedia 2011):

- Feature detection;

- HTML request data retrieval;
- JSON data retrieval;
- Server push data retrieval;
- Drag and drop;
- Simple visual effects;
- Animations and advanced visual effects;
- Back button support and history management;
- Input forms widgets and validation;
- Grid;
- Rich text editor;
- HTML generation tools;
- GUI resizable panels and modal dialogs;
- GUI page layout;
- Mobile, tablet support;
- Developer tools and visual design;
- Internet Explorer, Mozilla Firefox, Safari, Opera and Chrome.

The jQuery license: *You are free to use a jQuery project in commercial projects as long as the copyright header is left intact.* Hence, there are a lot of exiting JavaScript languages like Ally, Ample, SDK, Dojo, Google web toolkit, Midori, YUI, ZK and etc. The biggest disadvantage of all of them is that those libraries do not have feature detection. Good feature detection is one of the fundamental indicators of a quality browser script. Some of mentioned do not have simple visual effects, animation, advanced visual effects features which narrows down to create attractive and user friendly interface. Hence, some of them do not include grid, rich text editor, developer tools, visual design or offline storage features (Wikipedia 2011).

In order to represent and manipulate the data in FIT tool jqGrid grid plug in was chosen. Chosen grid has many features which can make FIT more user friendly and allow making task faster. The jqGrid includes CSS based themes, paging possibility, resizable columns, sorting various types of data, auto data loading when scrolling, working with local data, integration with any server side solution, cross browser, support JSON and arrays as data sources, formatting, inline editing, cell editing, form editing, searching and filtering. Mentioned features will allow creating user friendly interface. The jqGrid license: *the jqGrid is released under GPL and MIT licenses. This license policy makes the software available to everyone for free and you can use it for commercial or open source projects, without any restriction* (jQuery Grid Plugin 2011).

4.2. FIT tool code files and database

FIT consists of number of PHP, CSS, jQuery and MySQL files. The names of the files and description are given in Table 12, explaining the main context and functionality of the file.

Table 12 FIT tool code files

Type	File name	Description
PHP	index	Generates the main (login) page, including general information about formal inspection.
	logging_process	Identifies user, to which reviews and roles user belongs.
	logout	Ends user connection and returns to the main page.
	connect_database	Connects to the database.
	admin	Presents the admin page with the main tasks and responsibilities.
	admin_insert_process	Inserts user information to database.
	server_admin_reviews	Updates database when changes were made in the editable review table.
	server_admin_users	Updates database when changes were made in the editable users table.
	author	Presents the author page with the main tasks and responsibilities.
	upload_artifact	Uploads artifact to uploaded folder and saves the path in the database.
	inspector	Presents the inspector page with the main tasks and responsibilities.
	moderator	Presents the moderator page with the main tasks and responsibilities.
	moderator_insert_process	Inserts added team member information to database.
	upload_guidelines	Uploads guidelines to uploaded folder and saves the path in the database.
	server	Updates data base when changes are made in remark section
	server_moderator	Inserts review information to database when editable table of review is changed.
	server_moderator_users	Updates database when changes were made in the editable users table.
	scribe	Presents the scribe page with the main tasks and responsibilities.
	CSS	default
SQL	requirements_engineering	Database.
Folders	images	CSS images.
	jquery	JavaScript libraries.
	uploaded	Files uploaded by inspection team members.

PHP and JavaScript were combined together in order to develop powerful and dynamic web applications. PHP is a server side technology and JavaScript a client side scripting language. PHP does not have a way to work client side (Web Cheat Sheet 2011). Using PHP files and JavaScript separately it adds complexity, slower the process.

FIT database was created, which consists of number of the tables handling specific information necessary to form FIT tool database. There were created six tables: user, review, comment, file, role and review_user_xref (Table 13 – Table 18).

Two reviews ABC project and DEF project were created identifying group name, course, start, preparation due and meeting dates to test FIT tool. When reviews were created for each of them four team members were assigned (moderator, inspector, author and scribe). To test the switching between reviews when user

belongs to more than one user Rita Petrolyte was assigned as second inspector to the DEF project team (in ABC project she is assigned as moderator) (Table 18).

Table 13 FIT database user table

user_id	first_name	last_name	student_id	e_mail	username	password
1	admin	admin	admin	admin@admin.dk	admin	admin
2	Rita	Petrolyte	s090657	s090657@student.dtu.dk	rita_petrolyte	12345
3	Robertas	Petrolis	s090658	s090658@student.dtu.dk	robertas_petrolis	54321
8	Raimonda	Lukosiunaite	s090659	s090659@student.dtu.dk	raimonda_lukosiunaite	abcde
9	Danute	Apulskyte	s090660	s090660@student.dtu.dk	danute_petroliene	edcba
10	Tomas	Tomasauskas	s094534	s094534@student.dtu.dk	tomas_tomasauskas	11223
11	Lina	Linkaite	s093598	s093598@student.dtu.dk	lina_linkaite	22334
12	Mantas	Monkunas	s092312	s092312@student.dtu.dk	mantas_monkunas	33445
31	Ruta	Rutkauskaite	s0907734	s0907734@student.dtu.dk	ruta_rutkauskaite	55667

Table 14 FIT database review table

review_id	title	group_name	course	start_date	preparation_due	inspection_meeting
3	ABC project	3A	Requirements engineering	2011-04-04	2011-04-11	2011-04-29
4	DEF project	3B	Requirements engineering	2011-04-03	2011-04-06	2011-04-08

Table 15 FIT database comment table

comment_id	review_id	page	line	content	type	author
1	3	2	1	These sentences have to be more specific.	MA	raimonda_lukosiunaite
2	3	1	3	The requirements have to highlight the actor (pers...	MA	raimonda_lukosiunaite
8	3	4	45	It is only one requirement with 4 different use ca...	MA	raimonda_lukosiunaite
24	3	1	78	"#" has to be in the glossory or has to be changed...	MI	raimonda_lukosiunaite
32	3	23	55	Spelling mistake.	MI	raimonda_lukosiunaite
36	3	54	3	"#" has to be in the glossory or has to be changed...	MA	raimonda_lukosiunaite
45	4	2	4	Incorect usage of "remark"	MI	mantas_monkunas
46	4	4	6	Spelling mistake.	MA	mantas_monkunas
47	4	7	34	Users are not specified.	MA	mantas_monkunas
48	4	6	19	Why it is important?	MA	mantas_monkunas
49	4	34	12	Incorrecct sequence of tasks.	MA	rita_petrolyte
50	4	16	34	Tables are mixed.	MA	rita_petrolyte

Table 16 FIT database file table

id	name	path	review_id	file_user_id	date	remark
60	ABC_artifact.doc	uploaded/ABC_artifact.doc	3	3	2011-06-25	Afrifact to inspect 13 pages
61	DEF_artifact.doc	uploaded/DEF_artifact.doc	4	11	2011-06-25	Afrifact to inspect 5 pages
62	ABC_guidelines.doc	uploaded/ABC_guidelines.doc	3	2	2011-06-25	Guidelines for the inspection
63	ABC_rework.doc	uploaded/ABC_rework.doc	3	3	2011-06-25	Rework. Last version.
64	DEF_guidelines.doc	uploaded/DEF_guidelines.doc	4	10	2011-06-25	Guidelines for the inspection
65	DEF_rework.doc	uploaded/DEF_rework.doc	4	11	2011-06-25	Rework. Last version.

Table 17 FIT data base role table

role_id	role_name
2	author
3	moderator
4	inspector
5	scribe

Table 18 FIT data base review_user_xref table

id	review_id	user_id	role_id
3	3	2	3
4	3	3	2
5	3	8	4
6	3	9	5
7	4	10	3
8	4	11	2
9	4	12	4
15	4	31	5
16	4	2	4

Role table was created in order easier to identify the role in the code. All information is connected via review_user_xref table, where it can be seen which user belongs to which review and what is the role assigned to him/her (review_id, user_id, role_id). SQL data base language allows managing data, as well as insert, query, update, delete, modify and access. All these features are necessary and well suited for FIT.

In order to implement a login function sessions were used. All registered users have their own, unique session id. In FIT case there were created three sessions called username session, review session and select review session. Username session to identify user and checking existence in the database comparing username and password matches. If entered user name and password matches with the database it is checked to which reviews user belongs. If user belongs to one review with one role the page of the assign role loads. If user belongs to more than one review with the specific role page of the first assigned review loads with the possibility to switch to another's. The list of the reviews appears and user has to select on which review s/he wants to work.

The page of admin, moderator, author, inspector or scribe loads depending on the review and role to which belongs user. Layout kept the same just allowing different functions. Task section is presented in the tab style: review, team, artifact and remark fields. This structure is kept in admin.php, moderator.php, inspector.php, author.php, scribe.php files.

4.3. Implemented FIT tool

The main page of FIT, administrators', moderators', inspectors' and authors' pages showing the navigation in tabs is given in Figure 24 – Figure 45. Database presented in the FIT tool code files and database chapter was used for testing the FIT.

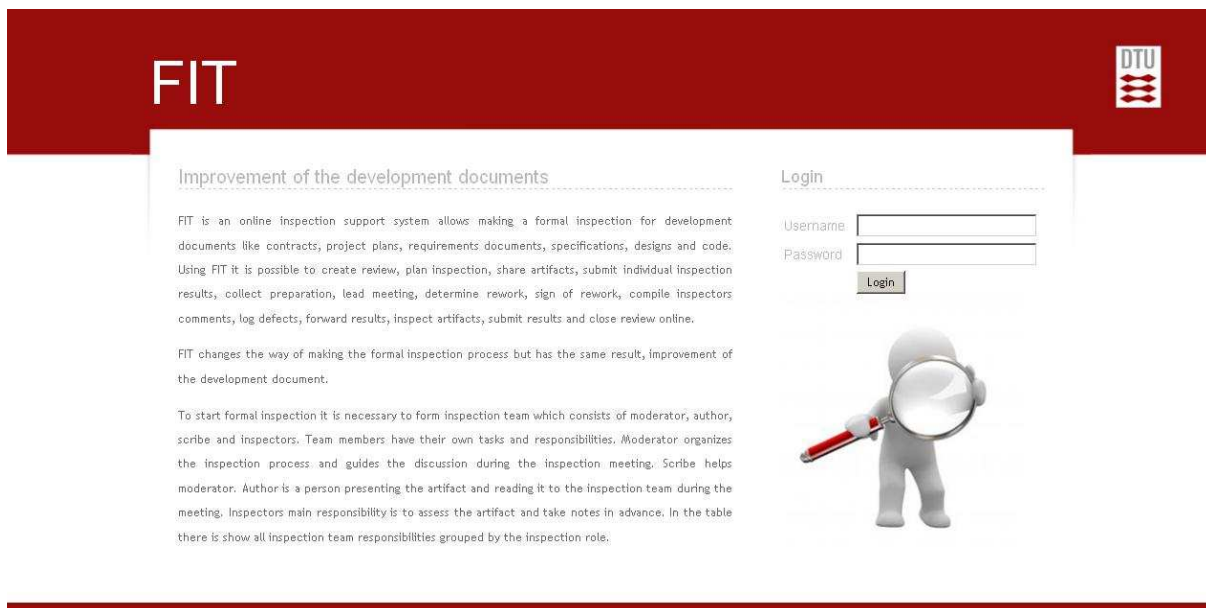


Figure 24 FIT main page (login)

Administrator

Today is 2011.06.26

1.Review 2.Add moderator 3.User

Basic Example

id	title	group	course	start	deadline	meeting
4	DEF project	3B	Requirement	2011-04-03	2011-04-06	2011-04-08
3	ABC project	3A	Requirement	2011-04-04	2011-04-18	2011-04-29

Page 0 of 1 View 1 - 2 of 2

Figure 25 Administrator review tab

Administrator

Today is 2011.06.26

1.Review 2.Add moderator 3.User

First name:

Last name:

Student ID:

E-mail:

User name:

Password:

Role:

Review:

Submit

Figure 26 Administrator add moderator tab

During the implementation additional tab add moderator was added to the administrators' page because it was not possible to implement the drop box (option feature) in the jqGrid grid plug. Administrator has to add users/moderators one by one, assigning the review and role (Figure 26).

Administrator

Today is 2011.06.26

1.Review 2.Add moderator 3.User

Basic Example

id	name	surname	student	e-mail	username	password
31	Ruta	Rutkauskaitė	s0907734	s0907734@stud	ruta_rutkausk	55667
12	Mantas	Monkunas	s092312	s092312@stud	mantas_monk	33445
11	Lina	Linkaitė	s093598	s093598@stud	lina_linkaitė	22334
10	Tomas	Tomasauska	s094534	s094534@stud	tomas_tomas	11223
9	Danute	Apulskytė	s090660	s090660@stud	danute_petro	edcba
8	Raimond	Lukosiunaite	s090659	s090659@stud	raimonda_luk	abcde

Page 0 of 1 10 View 1 - 9 of 9

Figure 27 Administrator user tab

Entered users' information can be modified in the administrators' user tab. Administrator has right to remove user from data base pressing delete icon (Figure 27). Moderators' page has additional inspection stages section which shows the main deadlines entered by her/him (Figure 28).

Moderator / Rita Petrolyte / ABC project

Today is 2011.06.26

1.Review 2.Team 3.Add team 4.Artifact 5.Remarks

Basic Example

id	title	group	course	start	deadline	meeting
3	ABC project	3A	Requirements	2011-04-04	2011-04-18	2011-04-29

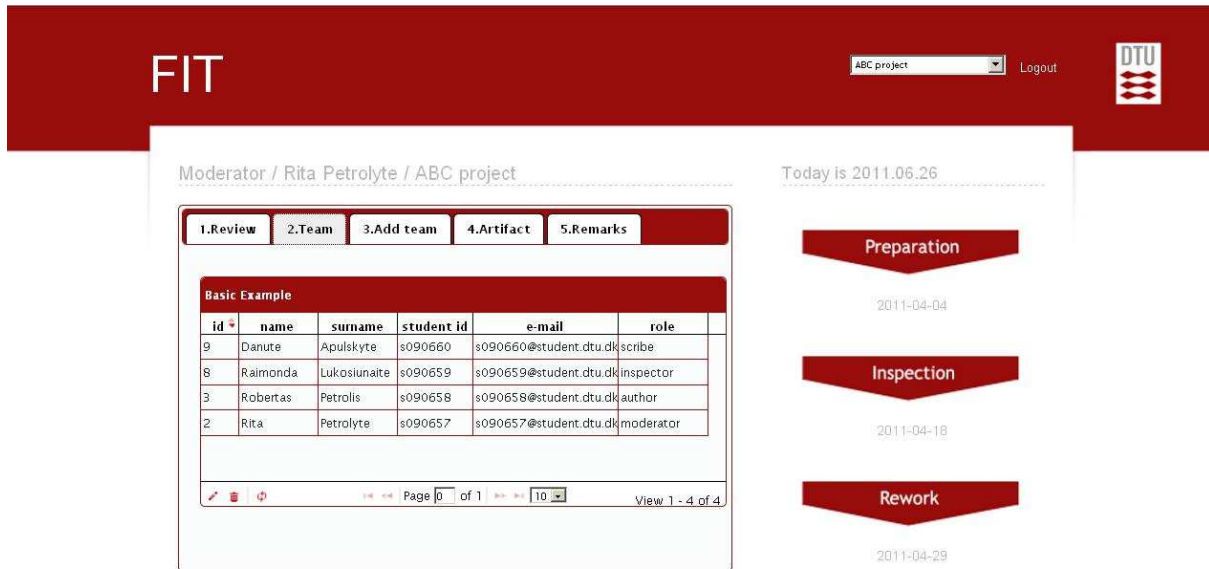
Page 0 of 1 10 View 1 - 1 of 1

Preparation
2011-04-04

Inspection
2011-04-18

Rework
2011-04-29

Figure 28 Moderator review tab



Moderator / Rita Petrolyte / ABC project

Today is 2011.06.26

1.Review 2.Team 3.Add team 4.Artifact 5.Remarks

Basic Example

id	name	surname	student id	e-mail	role
9	Danute	Apulskyte	s090660	s090660@student.dtu.dk	scribe
8	Raimonda	Lukosiunaite	s090659	s090659@student.dtu.dk	inspector
3	Robertas	Petrolis	s090658	s090658@student.dtu.dk	author
2	Rita	Petrolyte	s090657	s090657@student.dtu.dk	moderator

Page 10 of 1 | 10 | View 1 - 4 of 4

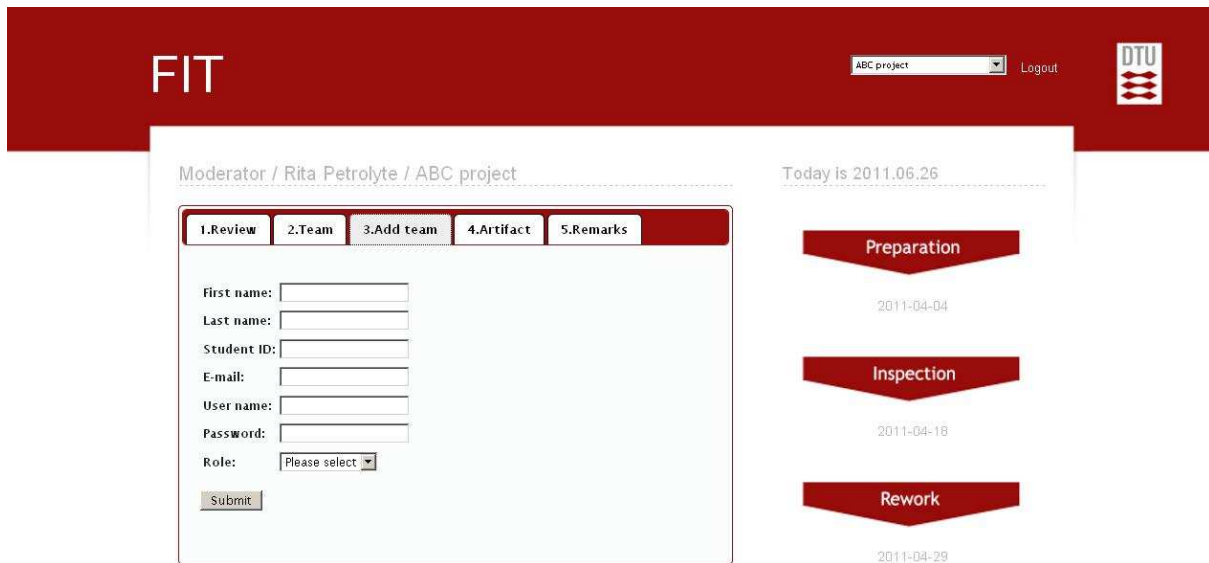
Preparation
2011-04-04

Inspection
2011-04-18

Rework
2011-04-29

Figure 29 Moderator team tab

Issue that it was not possible to implement the drop box (option feature) in the jqGrid grid plug appears in the moderators' role as well. The same solution to add additional tab add team was used (Figure 30).



Moderator / Rita Petrolyte / ABC project

Today is 2011.06.26

1.Review 2.Team 3.Add team 4.Artifact 5.Remarks

First name:

Last name:

Student ID:

E-mail:

User name:

Password:

Role:

Submit

Preparation
2011-04-04

Inspection
2011-04-18

Rework
2011-04-29

Figure 30 Moderator add team tab

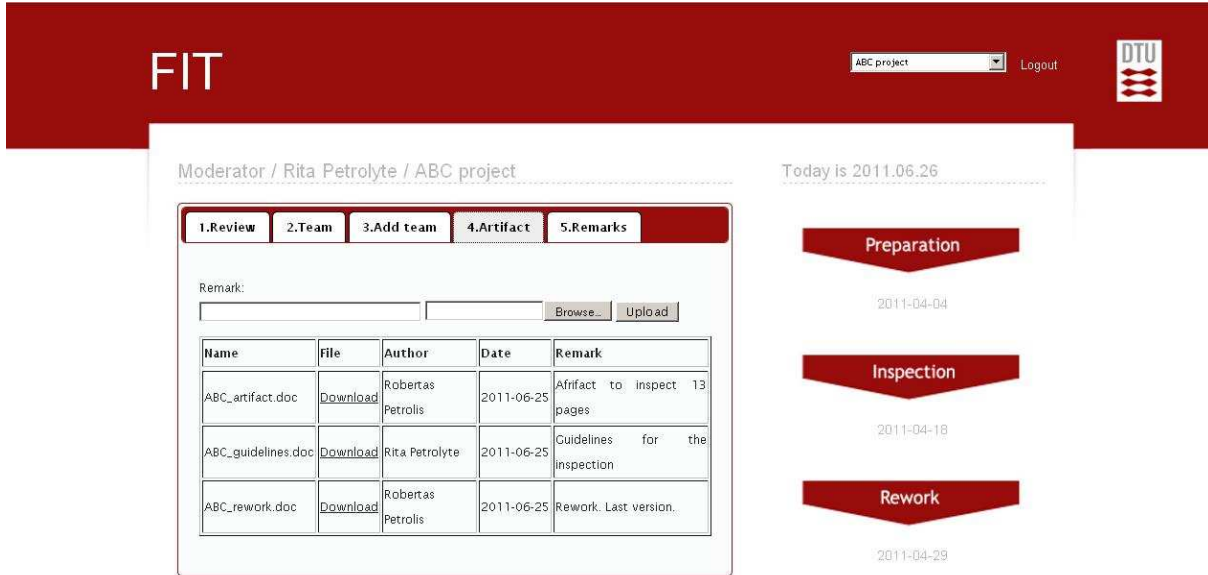


Figure 31 Moderator artifact tab

Comparing with the design described in the 3.6. FIT layout and sketches section. Changes in the artifact tab were made. Upload artifact function and already uploaded artifacts table are presented separately (Figure 31).

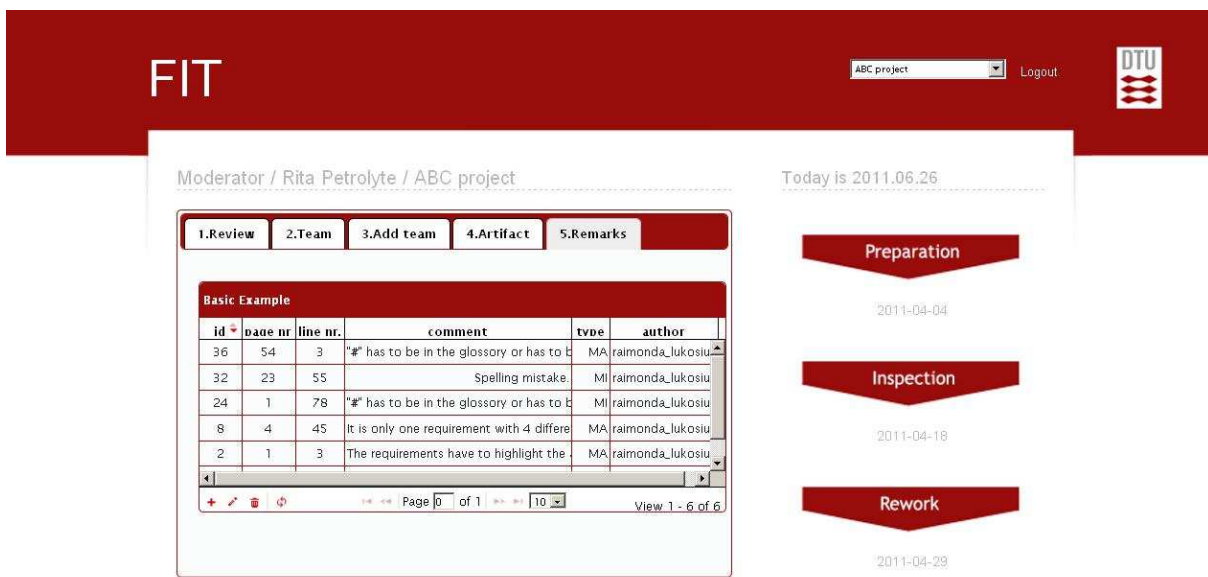


Figure 32 Moderator remarks tab

Moderator / Rita Petrolyte / ABC project

Today is 2011.06.26

1.Review 2.Team 3.Add team 4.Artifact 5.Remarks

Add Record

page nr.

line nr.

comment

type

author

	type	author
ssory or has t	MA	raimonda_lukos
lling mistake.	MI	raimonda_lukos
ssory or has t	MI	raimonda_lukos
nt with 4 diff	MA	raimonda_lukos
to highlight th	MA	raimonda_lukos
to be more sp	MA	raimonda_lukos

Page 0 of 1 View 1 - 6 of 6

Preparation 2011-04-04

Inspection 2011-04-18

Rework 2011-04-29

Figure 33 Moderator add remark function

After pressing to add the remark icon (+) table with fields which are required to add appears. Submit have to be pressed after entering all necessary information. If it was decided not to add remark it can be canceled by pressing cancel (Figure 33).

Moderator / Rita Petrolyte / ABC project

Today is 2011.06.26

1.Review 2.Team 3.Add team 4.Artifact 5.Remarks

Edit Record

page nr.

line nr.

comment

type

author

	type	author
ssory or has t	MA	raimonda_lukos
lling mistake.	MI	raimonda_lukos
ssory or has t	MI	raimonda_lukos
nt with 4 diff	MA	raimonda_lukos
to highlight th	MA	raimonda_lukos
to be more sp	MA	raimonda_lukos

Page 0 of 1 View 1 - 6 of 6

Preparation 2011-04-04

Inspection 2011-04-18

Rework 2011-04-29

Figure 34 Moderator edit remark function

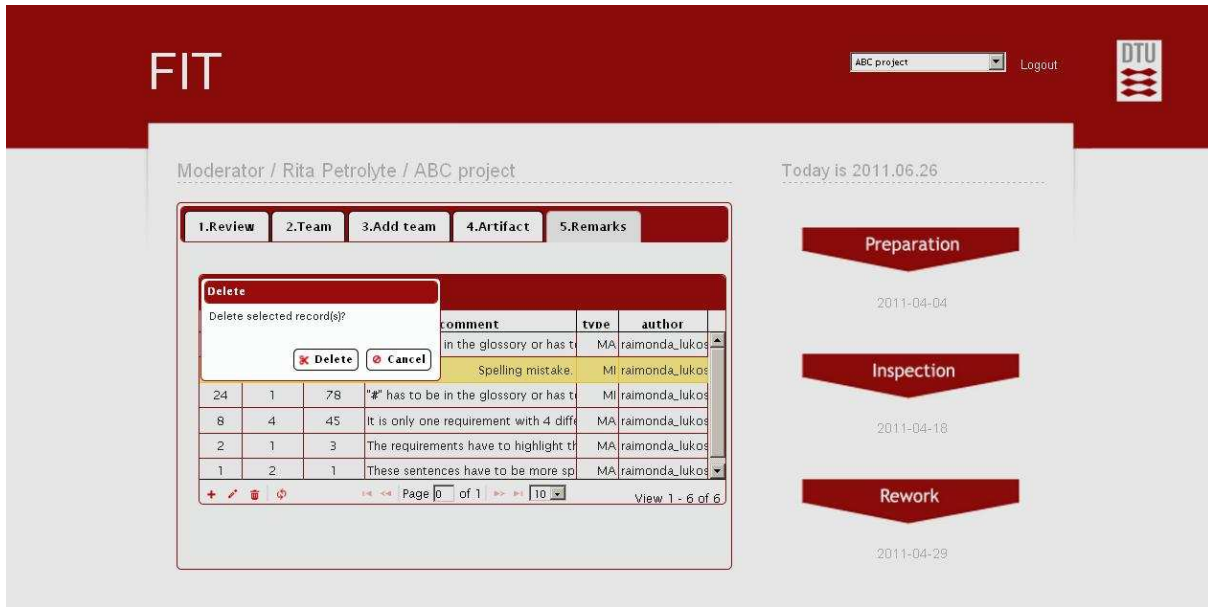


Figure 35 Moderator delete remark function

In order to edit or delete the remark it is necessary to select the row which to edit or delete. When row is selected pressing edit or delete icons functions can be activated. Confirmation about the deleting the row is required in case if it was pressed by accident (Figure 34 and Figure 35).

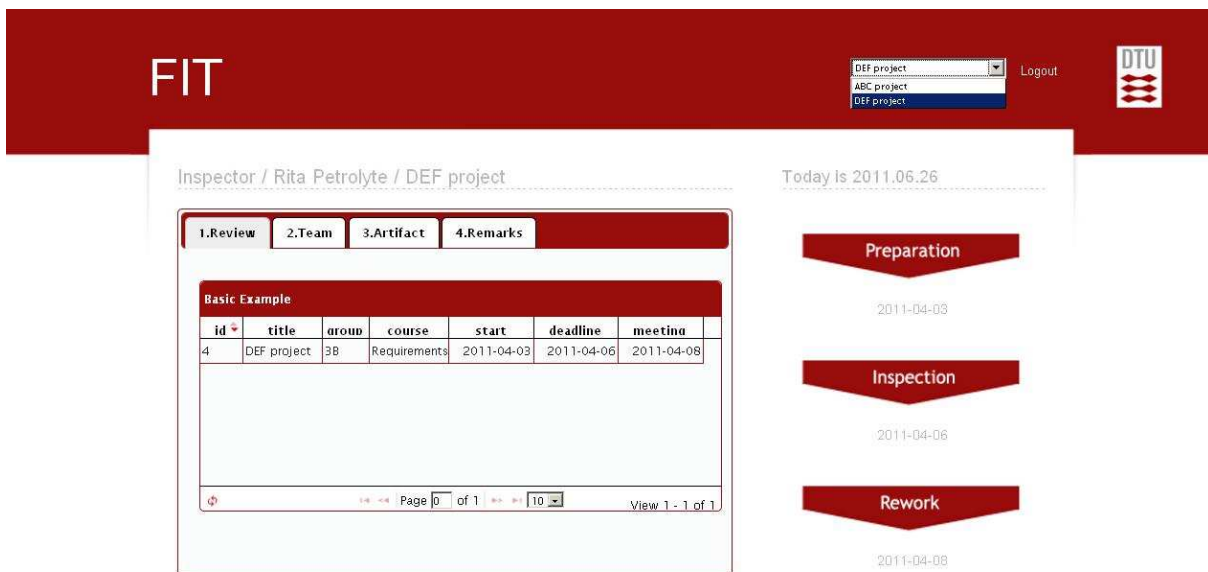


Figure 36 changing the review function and inspector review tab

Inspector / Rita Petrolyte / DEF project

Today is 2011.06.26

id	name	surname	student id	e-mail	role
31	Ruta	Rutkauskaitė	s0907734	s0907734@student.dtu.dk	scribe
12	Mantas	Monkunas	s092312	s092312@student.dtu.dk	inspector
11	Lina	Linkaite	s093598	s093598@student.dtu.dk	author
10	Tomas	Tomasauskas	s094534	s094534@student.dtu.dk	moderator
2	Rita	Petrolyte	s090657	s090657@student.dtu.dk	inspector

Page 0 of 1 | View 1 - 5 of 5

Preparation
2011-04-03

Inspection
2011-04-06

Rework
2011-04-08

Figure 37 Inspector team tab

Inspector / Rita Petrolyte / DEF project

Today is 2011.06.26

Name	File	Author	Date	Remark
DEF_artifact.doc	Download	Lina Linkaite	2011-06-25	Afrifact to inspect 5 pages
DEF_guidelines.doc	Download	Tomas Tomasauskas	2011-06-25	Guidelines for the inspection
DEF_rework.doc	Download	Lina Linkaite	2011-06-25	Rework. Last version.

Preparation
2011-04-03

Inspection
2011-04-06

Rework
2011-04-08

Figure 38 Inspector artifact tab

If user belongs to one review that review is loaded and select review field does not appear. If user is assigned more then to one review it is not necessary to log out in order to switch to another. It is enough to change the review which is wished to see (Figure 36). Then page switches to other roles page, in example it

is shown how from moderator role of one project (ABC project / Rita Petrolyte) is switched to inspector role to another project (DEF project / Rita Petrolyte). From the team members lists it can be seen that Rita Petrolyte is assigned as moderator in ABC project (Figure 29) and as inspector in DEF project (Figure 37).

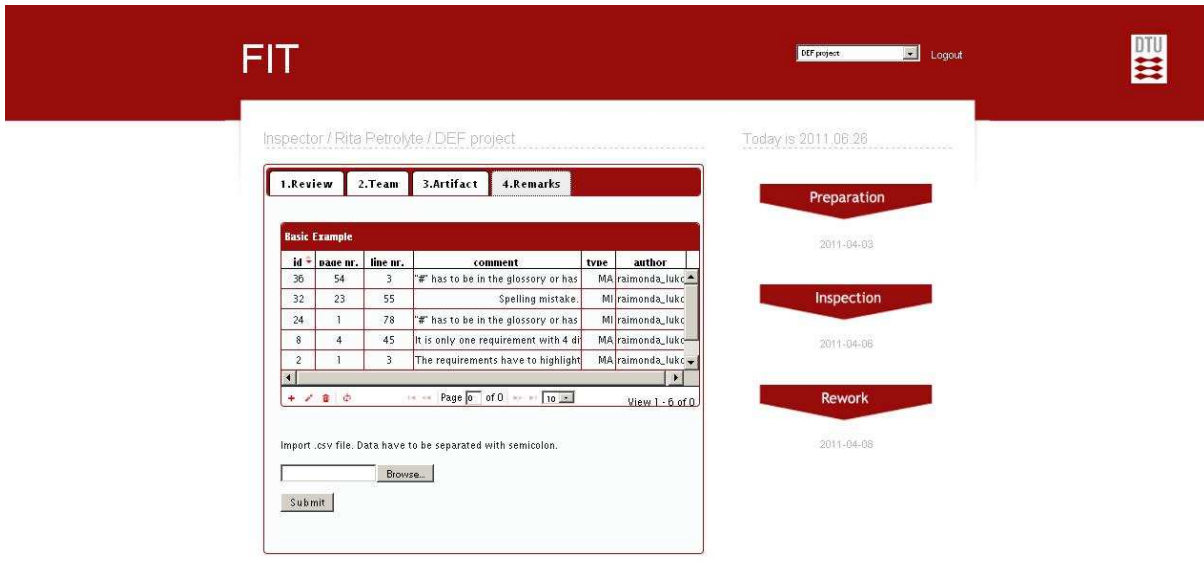


Figure 39 Inspector remarks tab

Inspectors' remarks tab is presented in Figure 39. Remarks can be added one by one if inspector has just few of them. In case if there are a lot of remarks or inspector does not have possibility to access the internet in order to submit remarks, s/he can create .csv file and later on upload it to the remarks table.

Table 19 Example of .csv file for uploading comments

22	2	Test.	MA	rita_petrolyte
33	3	Test.	MI	rita_petrolyte
44	4	Test.	MA	rita_petrolyte
55	5	Test.	MA	rita_petrolyte
66	6	Test.	MA	rita_petrolyte
77	7	Test.	MI	rita_petrolyte
88	8	Test.	MA	rita_petrolyte
99	9	Test.	MI	rita_petrolyte

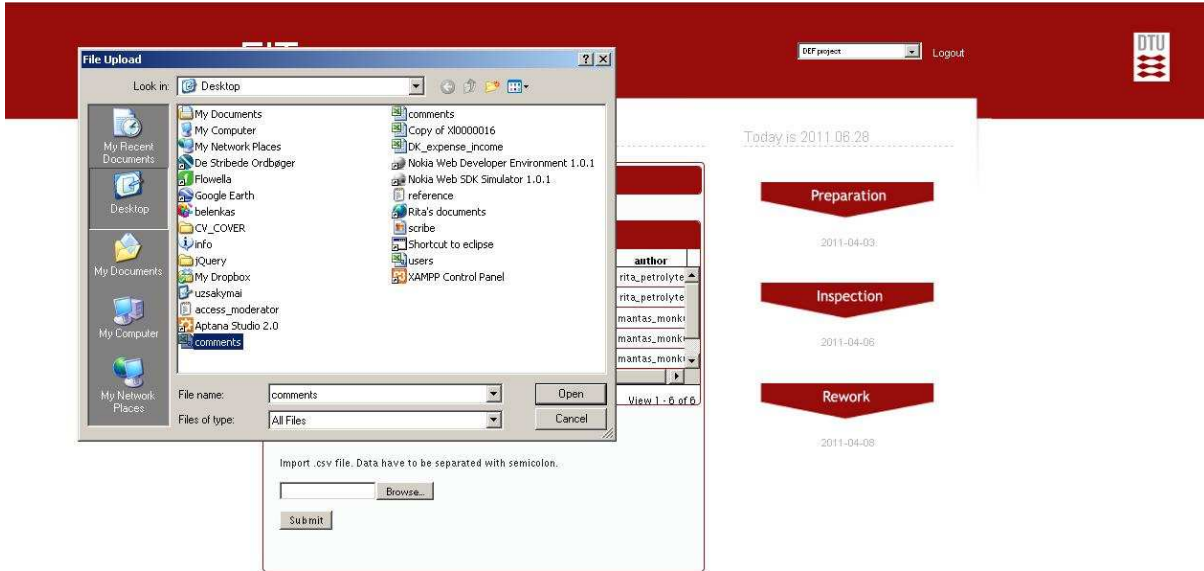


Figure 40 Inspector remarks tab uploading .csv file

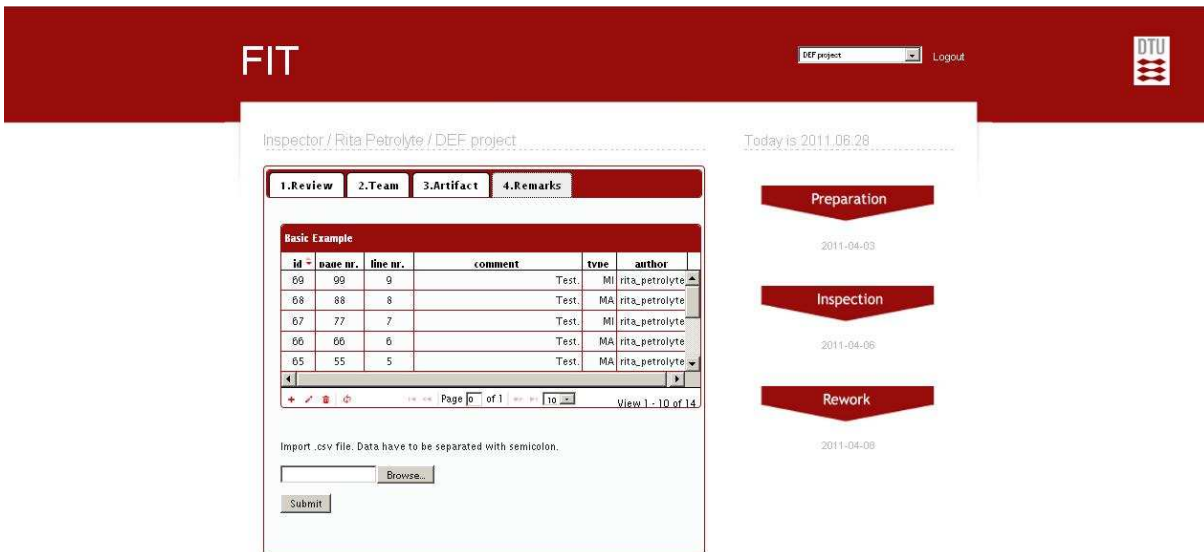


Figure 41 Uploaded .csv file in remarks table

Table 19 presents .csv files content. Figure 40 presents action how to upload .csv file and Figure 41 the result after uploading.

Author / Robertas Petrolis / ABC project

Today is 2011.06.26

1.Review 2.Team 3.Artifact 4.Remarks

Basic Example

id	title	group	course	start	deadline	meeting
3	ABC project	3A	Requirements	2011-04-04	2011-04-18	2011-04-29

Page 0 of 1 | 10 | View 1 - 1 of 1

Preparation
2011-04-04

Inspection
2011-04-18

Rework
2011-04-29

Figure 42 Author review tab

Author / Robertas Petrolis / ABC project

Today is 2011.06.26

1.Review 2.Team 3.Artifact 4.Remarks

Basic Example

id	name	surname	student id	e-mail	role
9	Danute	Apulskyte	s090660	s090660@student.dtu.dk	scribe
8	Raimonda	Lukosiunaite	s090659	s090659@student.dtu.dk	inspector
3	Robertas	Petrolis	s090658	s090658@student.dtu.dk	author
2	Rita	Petrolyte	s090657	s090657@student.dtu.dk	moderator

Page 0 of 1 | 10 | View 1 - 4 of 4

Preparation
2011-04-04

Inspection
2011-04-18

Rework
2011-04-29

Figure 43 Author team tab

Author / Robertas Petrolis / ABC project

Today is 2011.06.26

1.Review 2.Team 3.Artifact 4.Remarks

Remark:

Browse... Upload

Name	file	Author	Date	Remark
ABC_artifact.doc	Download	Robertas Petrolis	2011-06-25	Afrfact to inspect 13 pages
ABC_guidelines.doc	Download	Rita Petrolyte	2011-06-25	Guidelines for the inspection
ABC_rework.doc	Download	Robertas Petrolis	2011-06-25	Rework. Last version.

Preparation
2011-04-04

Inspection
2011-04-18

Rework
2011-04-29

Figure 44 Author artifact tab

Author / Robertas Petrolis / ABC project

Today is 2011.06.26

1.Review 2.Team 3.Artifact 4.Remarks

Basic Example

id	page Nr.	line Nr.	comment	type	author
36	54	3	"#" has to be in the glossory or has t	MA	raimonda_lukos
32	23	55	Spelling mistake.	MI	raimonda_lukos
24	1	78	"#" has to be in the glossory or has t	MI	raimonda_lukos
8	4	45	It is only one requirement with 4 diff	MA	raimonda_lukos
2	1	3	The requirements have to highlight th	MA	raimonda_lukos
1	2	1	These sentences have to be more sp	MA	raimonda_lukos

Page 0 of 1 | 10 | View 1 - 6 of 6

Preparation
2011-04-04

Inspection
2011-04-18

Rework
2011-04-29

Figure 45 Author remarks tab

Authors' tabs are presented in Figure 42 – Figure 45. Author can see review to which s/he belongs, see team members, artifacts' list with the function to upload and remarks.

5. Operation

This chapter will present how to install, configure and run FIT application, how to interpret error messages, log files and how to add or change the system.

5.1. Installation, configure and run FIT

To run the created FIT tool in the testing environment it necessary to have cross-platform web server described in the adapting technologies part (or similar one). If XAMPP is used following steps should be done in order to run FIT:

1. Create a folder in \xampp\htdocs with a web application name, in our case it is fit (\xampp\htdocs\fit);
2. Copy and paste all FIT related files in created fit folder;
3. FIT can be viewed through localhost. FIT tool runs with the index.php file. To see the main page of the FIT it is necessary to type in the browser address bar `http://localhost\fit\index.php` (as the name of the main page is named index.php it is not necessary to specify it because the server will automatically look for index.php and run it);
4. When the FIT run it can be used by entering username and password, which can be given by administrator and saved in database.

Hence, images and uploaded folders have to be created in side of the fit folder in order reach design images and save uploaded files. The requirements_engineering.sql database has to be imported to the MySQL.

5.2. Error messages and alerts

Error messages in the FIT tool alerts user about the problems that has occurred using FIT. For example, in the login page there are few error messages with inform user about incorrect login and requires trying entering the username and password once again. Login Error messages:

1. Wrong password or username! In case if entered username and password do not match.
2. Username or password missing! In case if username or password were not entered in the login field.

Hence warning messages were used as well in order to notify the used what condition it may cause. Basically warning messages were used in order to confirm modifications made in the data tables, especially when the data is going to be removed, deleted. Double confirmation of deleting is asked.

5.3. Loading time of FIT

In order to get loading time of the FIT firebug was used. Firebug is web development tool which allows inspecting HTML style and layout, analysing network usage and performance.

In Figure 46 – Figure 50 there is shown the request time made to the server after logging to the FIT tool and activating tabs. Request time is presented separately for all roles.



Figure 46 Admin page load time

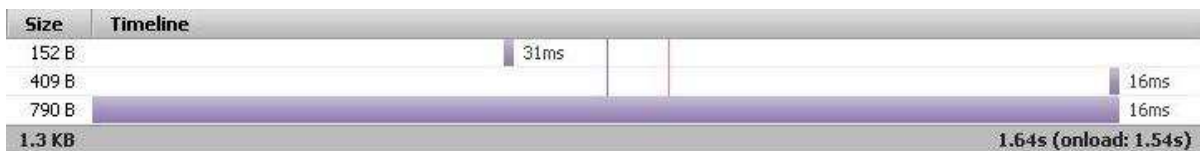


Figure 47 Moderator page load time

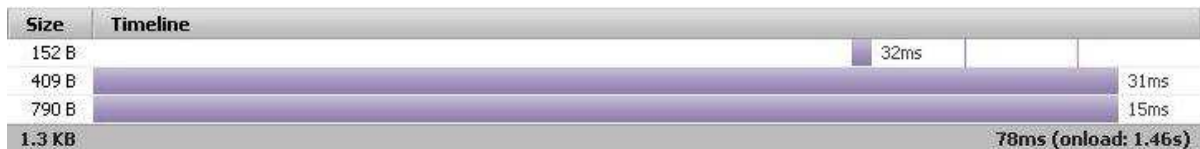


Figure 48 Author page load time

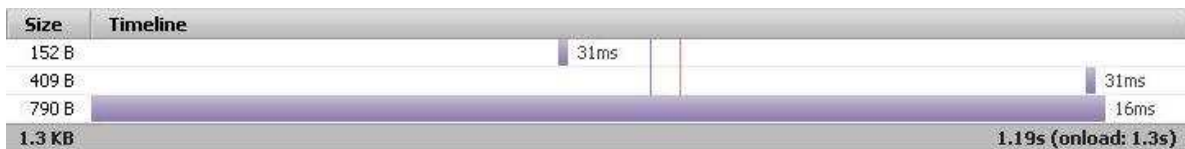


Figure 49 Inspector page load time

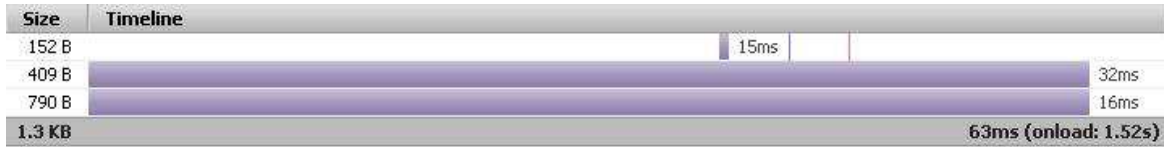


Figure 50 Scribe page load time

Moderator has the highest loading time. Moderator has more tasks and responsibilities than other inspection team members what leads to implementing more function to the moderator page and it increases the loading time. In order to load the page faster in the beginning decision to load information from the server after the click on the tab was made. Meaning that when user loads page the connection to the server is made once for the first tab and if user needs to get more information or make some tasks after clicking on the tasks fields' information is transferred from the server. All in all loading time does not reach more than 1,54s loading all information (3 requests).

5.4. Log files

Log files listing few requests made to the server by moderator is presented below. With log files it is possible to see how the user navigates through the FIT web site and see all requests and connections to the server when making one or another action. Decision to analyse the moderator log files was made because of the variety and high number of tasks what leads to the high number of connections to the server.

Loading main FIT page loads main page index.php, default.css, jquery.css and images needed to present the layout of the page:

- 127.0.0.1 -- [26/Jun/2011:16:42:43 +0200] "GET /fit/index.php HTTP/1.1" 200 3951 "http://localhost/fit/scribe.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
- 127.0.0.1 -- [26/Jun/2011:16:42:43 +0200] "GET /fit/default.css HTTP/1.1" 304 - "http://localhost/fit/index.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
- 127.0.0.1 -- [26/Jun/2011:16:42:43 +0200] "GET /fit/front.jpg HTTP/1.1" 304 - "http://localhost/fit/index.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
- 127.0.0.1 -- [26/Jun/2011:16:42:43 +0200] "GET /fit/images/dtu.jpg HTTP/1.1" 304 - "http://localhost/fit/index.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
- 127.0.0.1 -- [26/Jun/2011:16:42:43 +0200] "GET /fit/images/img01.jpg HTTP/1.1" 304 - "http://localhost/fit/default.css" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
- 127.0.0.1 -- [26/Jun/2011:16:42:43 +0200] "GET /fit/images/img02.jpg HTTP/1.1" 304 - "http://localhost/fit/default.css" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"

Login to moderator role starts from running the logging_process.php where it is checked username, password and identified number of reviews to which belongs user. In our case moderator page was loaded. As the moderators first tab is review it is loaded as well and it can be seen third line:

1. 127.0.0.1 -- [26/Jun/2011:16:43:20 +0200] "POST /fit/logging_process.php HTTP/1.1" 302 36 "http://localhost/fit/index.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
2. 127.0.0.1 -- [26/Jun/2011:16:43:20 +0200] "GET /fit/moderator.php HTTP/1.1" 200 11943 "http://localhost/fit/index.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
3. 127.0.0.1 -- [26/Jun/2011:16:43:21 +0200] "GET /fit/server_moderator.php?&_search=false&nd=1309099401490&rows=10&page=1&sidx=review_id&sord=desc HTTP/1.1" 200 152 "http://localhost/fit/moderator.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"

When moderator activates the team tab connection to the database is done, it is seen from the first line. Second connection appears when the new team member was added to the team, second line. An in the end when team member was removed from the review one more connection to the database appears, fourth line:

1. 127.0.0.1 -- [26/Jun/2011:16:44:03 +0200] "GET /fit/server_moderator_users.php?&_search=false&nd=1309099443405&rows=10&page=1&sidx=user_id&sord=desc HTTP/1.1" 200 409 "http://localhost/fit/moderator.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
2. 127.0.0.1 -- [26/Jun/2011:16:44:54 +0200] "POST /fit/moderator_insert_process.php HTTP/1.1" 302 75 "http://localhost/fit/moderator.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
3. 127.0.0.1 -- [26/Jun/2011:16:44:54 +0200] "GET /fit/moderator.php HTTP/1.1" 200 11943 "http://localhost/fit/moderator.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
4. 127.0.0.1 -- [26/Jun/2011:16:44:54 +0200] "GET /fit/server_moderator.php?&_search=false&nd=1309099494911&rows=10&page=1&sidx=review_id&sord=desc HTTP/1.1" 200 152 "http://localhost/fit/moderator.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"

Uploading guidelines is presented in the line seven, where connection to the upload_guidelines.php is done, first line. After uploading moderators page reloads, second line:

1. 127.0.0.1 -- [26/Jun/2011:16:46:08 +0200] "POST /fit/upload_guidelines.php HTTP/1.1" 302 25 "http://localhost/fit/moderator.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
2. 127.0.0.1 -- [26/Jun/2011:16:46:08 +0200] "GET /fit/moderator.php HTTP/1.1" 200 12089 "http://localhost/fit/moderator.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"

Adding comment and after words deleting it requires connection to the data base as well. Getting the list of existing comments presents first line, adding second line and deleting third line:

1. 127.0.0.1 -- [26/Jun/2011:16:46:28 +0200] "GET /fit/server.php?&_search=false&nd=1309099588613&rows=10&page=1&sidx=comment_id&sord=desc HTTP/1.1" 200 790 "http://localhost/fit/moderator.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"

2. 127.0.0.1 - - [26/Jun/2011:16:46:46 +0200] "POST /fit/server.php HTTP/1.1" 200 - "http://localhost/fit/moderator.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
3. 127.0.0.1 - - [26/Jun/2011:16:46:46 +0200] "GET /fit/server.php?&_search=false&nd=1309099606367&rows=10&page=1&sidx=comment_id&sord=desc HTTP/1.1" 200 833 "http://localhost/fit/moderator.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
4. 127.0.0.1 - - [26/Jun/2011:16:46:59 +0200] "POST /fit/server.php HTTP/1.1" 200 - "http://localhost/fit/moderator.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"
5. 127.0.0.1 - - [26/Jun/2011:16:46:59 +0200] "GET /fit/server.php?&_search=false&nd=1309099619738&rows=10&page=1&sidx=comment_id&sord=desc HTTP/1.1" 200 790 "http://localhost/fit/moderator.php" "Mozilla/5.0 (Windows; U; Windows NT 5.1; en-GB; rv:1.9.2.18) Gecko/20110614 Firefox/3.6.18"

Basically using log files it is easy to see the navigation between the files and record the sequence of the connecting to the server making one or another task:

5.5. Adding or changing FIT

System can be modified and changed editing information in created .php files. Files index.php, admin.php, author.php, inspector.php, moderator.php, scribe.php includes the main layout and description of the logged page depending on the role. Connecting process to the data base is saved separately in case if changes are required to make.. Hence, processes to insert or get information from the database are separated as well. Style of the FIT tool can be modified changing CSS files. All files of FIT code including database are given in CD which is attached in the end of the thesis.

6. Conclusions

Formal inspection is one of the most powerful tools to improve development processes, find mistakes, save time and budget. Finding the right technique for the inspection is not easy and always requires a lot of time. The main focus of the thesis was achieved. An online based formal inspection tool was created which allows inspecting any type of artifact at any time of development document at any stage.

Analysis of the requirements engineering course formal inspections showed that there is a need to improve formal inspections process in order to make it more attractive to the users. Feedback that formal inspection is time consuming and challenging process was taken into consideration creating FIT.

The main problems of the existing online inspection tools are that they do not support any kind of artifacts, do not have document handling, individual preparation or meeting. It has a very narrow area of usage what makes it difficult to adapt for any kind of artifact. The main issues which were taken into consideration is how to make formal inspection discipline and flexible at the same time and how to arrange all inspection team to meet. Web technology which allows working independently at any place and time solved this problem. Used technologies HTML, PHP, MySQL, CSS and JavaScript gave a possibility to create a reliable, straightforward online inspection support tool and achieve raised goals.

Creating formal inspection tool FIT based on web technologies the main purpose was to narrow down the formal inspection process in creating tool which will not confuse the user, will make it simple and powerful, easy to use and with useful outcomes. The main aim was not to change an idea of the formal inspection, as it is reliable way to find faults and artifacts in development documents, but to change the way making it. Challenge how to make it less time consuming, allowing more editing possibilities and coordinate all over the process accured. Steps of the formal inspection process were left the same trying to consider making the tasks and responsibilities less time consuming. Hand writing was changed to the digital way what allowed easier for all team members to understand text and to avoid miss understandings. All shared information which requires more than few lines were presented in the sortable tables with possibility to edit, add and delete data. Information sharing is faster and does not require organizing meetings with team members. Created tool gives a possibility to belong to more than one inspection team with the different role and keeps track of process using the same login. It becomes much easier to cover few reviews at the time and not to get lost among the papers, as well as it makes easy to coordinate the process. All information about the inspection process is seen by all inspection team members.

Before starting the implementation part languages as HTML, PHP, SQL, JavaScript were learned from the basics. Step by step identifying the requirements for the tool list of the functions needed were implemented and tested. It required a lot of time and patient to understand the back of the functions, how it works, how it can be implemented and how it can be improved.

I can see the bright future of this tool in Demarks Technological University. FIT can be used to improve development documents created by students at any course. Requirements engineering course students found formal inspection useful and they are using it to improve their projects, reports and assignments. Using FIT it will allow making it easier and faster. Increasement of the formal inspection usage is expected after presenting FIT to the students.

Bibliography

- Ahmed, F. *An asynchronous software inspection model*. n.d.
<http://mjcs.fsktm.um.edu.my/document.aspx?FileName=81.pdf> (accessed 04 12, 2011).
- Apache Friends*. 2011. <http://www.apachefriends.org/en/xampp.html> (accessed 04 25, 2011).
- Atkinson, Leon. *Core PHP programming*. Prentice hall, 2004.
- Beaulieu, Alan. *Learning SQL*. Orley, 2005.
- Ezine. 2011. <http://ezinearticles.com/?JavaScript-for-Web-Design---Advantages-and-Disadvantages&id=645013> (accessed 04 02, 2011).
- Free CSS Templates*. 2011. <http://www.freecsstemplates.org/license/> (accessed 05 02, 2011).
- Gilb, T. *Software inspection*. Addison-Wesley, 1993.
- JQuery Grid Plugin*. 2011. http://www.trirand.com/blog/?page_id=154 (accessed 05 08, 2011).
- Macdonald, F. "Computer Supported Software Inspections." 1998.
<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.55.7099&rank=1> (accessed 02 19, 2011).
- Michaux, Peter. n.d. <http://peter.michaux.ca/articles/feature-detection-state-of-the-art-browser-scripting> (accessed 05 16, 2011).
- Parnas, D. *Active design reviews: principles and practices*. n.d.
<http://delivery.acm.org/10.1145/320000/319599/p132-parnas.pdf?key1=319599&key2=8195334031&coll=DL&dl=ACM&ip=130.225.70.145&CFID=18870258&CFTOKEN=80176107> (accessed 05 04, 2011).
- Quigley, E. *PHP and MySQL by example*. Prentice Hall, 2006.
- Storrie, Harald. "Process Guideline QA: Formal Inspection ." 2010.
- Strauss, Susan H. *Software inspection process*. MacGraw-Hill, 1993.
- Taylor, Allen G. *SQL for dummies*. Wiley publishing, 2003.
- w3schools*. 2011. http://www.w3schools.com/SQL/sql_datatypes.asp (accessed 05 29, 2011).

Web Cheat Sheet. 2011. http://www.webcheatsheet.com/PHP/passing_javascript_variables_php.php
(accessed 06 15, 2011).

Whitehead, C. *Evaluating Web Page and Web Site Usability*. n.d.
<http://www.joelstienstra.com/usability/sources/Evaluating.pdf> (accessed 04 28, 2011).

Wieger, K.E. *Process Impact*. 2011. <http://www.processimpact.com/articles/inspects.html> (accessed 03 11,
2011).

Wikipedia. 2011. http://en.wikipedia.org/wiki/Fagan_inspection (accessed 03 16, 2011).

Wikipedia. 2011. http://en.wikipedia.org/wiki/Peer_review (accessed 02 25, 2011).

Wikipedia. 2011. http://en.wikipedia.org/wiki/software_review (accessed 03 05, 2011).

Wikipedia. 2011. <http://en.wikipedia.org/wiki/JavaScript> (accessed 04 15, 2011).

Wikipedia. 2011. <http://en.wikipedia.org/wiki/Walkthrough> (accessed 02 15, 2011).

Wikipedia. 2011. http://en.wikipedia.org/wiki/Cascading_Style_Sheets (accessed 05 10, 2011).

Wikipedia. 2011. http://en.wikipedia.org/wiki/Comparison_of_JavaScript_frameworks (accessed 05 03,
2011).