Quantified Self & Mobile Health Monitoring

Aleksi Elias Pertola

Kongens Lyngby 2013
IMM-MSc-2013-February

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-MSc-2013-February

Abstract

The rapid evolution of mobile phones from mere vessels for voice- and text-based communication to personal assistants with a treasure chest of advanced features has created a new paradigm in mobile applications. The computational power and ubiquity of such devices has generated plenty of interest from the healthcare sector, particularly via the recent parallel development of body worn sensors that can communicate with them wirelessly. There is a strong incentive, both from an economic and an altruistic perspective, to develop mobile applications that enable self-help, patient homecare, and remote disease management. However, very little research exists at present in the field of affecting health change behavior in people through such mobile applications.

This thesis proposes a body worn sensor-based mobile platform for the physiotherapeutic treatment of patients suffering from specific back disorders and back pain at home. In particular, this thesis focuses on developing a user interface characterized by persuasive, encouraging and motivational aspects that ensure adherence and compliance with respect to physiotherapeutic exercise regimes prescribed by a physiotherapist.

Preface

This thesis was written at the Department of Informatics and Mathematical Modeling (IMM), at the Technical University of Denmark (DTU) after approval from Michael Kai Petersen, Associate Professor in Cognitive Systems and Head of Studies for the Digital Media Engineering MSc program at DTU Compute, in order to achieve the MSc degree in engineering from Digital Media Engineering. The project was carried out in the period from the 10th of September, 2012 to the 15th of February, 2013.

Kgs. Lyngby, 15th of February-2013

Aleksi Elias Pertola

Acknowledgements

I'd like to thank the following people for their role, direct or indirect, in this thesis:

My supervisor, Michael Kai Petersen, for his encouragement and enthusiasm with regard to the idea for this thesis project, as well as his continued support in its implementation;

Petteri Pertola, Lars Schöning and Nicholas Bishop for their second opinion, help and advice in the technical areas of this project; and

My colleagues Mikkel Leth Olsen, Aslak Johansen, Vanessa Julia Carpenter and Morten Georg Jensen for providing the foundation, advice and some technical assistance for this project.

Contents

Chapter 1 – Introduction

1.1 Background

This MSc thesis is based on *Lev Vel* – a four-year meta-project comprising six innovation projects that are designed to address the societal challenges related to an aging population in Denmark [1]. These projects are developed and driven in close co-operation between the public and private sector, and emphasize the importance of incorporating end users – in this case, primarily the elderly – into every aspect of the project life cycle.

Specifically, *Lev Vel* addresses the concept of Preventative Self-Monitoring (PSM); it aims to provide primarily citizens aged 45 to 65 with better access to information regarding their own health, condition and ailments, and thus support them in managing their own lives and acting pro-actively to address health-related changes. The goal of the initiative is to allow the target group to maintain a high quality of life, and to preserve their functional capabilities for as long as possible; and to save time and cut expenditures both for the citizens and for the public healthcare system [2].

One of the six innovation projects in this initiative has been dubbed *BackTrack* – a mobile application based on the Android platform that attempts to apply the PSM concept to the specific case of persons suffering a variety of medical conditions affecting the spine and consequently inducing pain in the back, such as Spinal Disc Herniation[1] (SDH) and Spinal Stenosis[2] (SS). The project is an

[1] A rupture in the outer fibrous ring of an intervertebral disc, which allows the soft central portion of the disc to bulge out, thus compressing the spinal nerves and/or spinal cord and resulting in the release of inflammatory mediators which may cause severe pain.

ongoing joint public/private sector between two Danish hospitals (Gentofte and Helsingør), two private companies (Mobile Fitness A/S and Welfare Solutions ApS) and two research institutes (DELTA and CSI – *Center for Sundhedsinnovation*); and is divided into three distinct phases:

1. Formulation and analysis of requirements;
2. Development of a functional prototype; and
3. Feasibility testing through a pilot study.

Much of this thesis describes the creation of an independent sub-component of *BackTrack* via the three phases outlined above: a standalone mobile application and wearable sensor system presently titled *IdemoBelt*. The application was developed in conjunction with DELTA's involvement in the *Lev Vel* project through its IdemoLab department, which specializes in Product Innovation through Electronic Sketching; as well as Climate and the Environment through Energy Harvesting and Sensor Networks.

1.2 Motivation

1.2.1 Personal background

One of the key reasons for my undertaking this project as a thesis was my strong personal interest in developing technologies that support people with specific needs resulting from various disorders. This interest stems in part from my own experience with suffering such a condition – In early 2010 I was diagnosed with an abnormal lateral curvature of the spine (a "tennis shoulder"), which causes my right shoulder blade to protrude more than its left counterpart, pulling the muscles in the area between the spine and the blade outward and exerting a constant, and at times painful strain on the muscles in that region; in part from my work at IdemoLab DELTA with projects such as *Lev Vel* and *UNIK – Projekt KOL* [3]; and in part from my enrolment on the Personalized and Context-Aware Mobile Applications course at DTU, for which I submitted a project titled *Personalized mobile application for monitoring the heart rate and blood oxygen content of chronic obstructive pulmonary disease patients* [4] – a project, which focused on integrating data from an external fingertip sensor into a motivational

[2] An abnormal narrowing of the spinal canal which results in neurological disorders such as tingling, numbness or weakness radiating from the affected area toward the limbs.

mobile user interface so as to encourage COPD patients to persist with their physical exercise based rehabilitation programs and thus attain a higher quality of life. The project was relatively simple in its nature and scope, but through preliminary user testing and feedback it nonetheless provided me with the groundwork to delve deeper into the field and explore the aspects that play a role in persuading users to adopt such applications, and motivating them – through an encouraging, feedback-driven user interface – to continue using them.

1.2.2 Societal challenges

What motivates me to work in this field above all, however, is the knowledge that such technologies can and will be deployed in the near future – even now, in some cases – for the betterment of the lives of people with chronic, and often difficult medical conditions that require physical rehabilitation.

Nearly 35% of the adult population (persons aged 16+), or approximately 1.5 million people in Denmark are currently affected by a back disorder or back pain, which is directly relatable to the approximately two million days of absence in the domestic labor market annually; while statistical data suggests that such disorders and/or pain inflict an aggregate cost of 13 billion Danish Kroner on society each year. An estimated 34 to 45% of the cost stems from treatment, with the remaining 55 to 66% incurred from productivity losses [5].

At the same time, between 2001 and 2011 the percentage of elderly citizens in the Danish population grew by 142,000, which constitutes a 2% increase as a fraction of the total population. By 2050, this population segment is expected to stretch across 25% of the total population [6], and while back disorders and back pain are not exclusively symptomatic of an aging population, it is certainly reasonable to presume that such a drastic increase will have repercussions on the expenditures of the Danish public health sector in general – particularly so in the current economic climate.

As the effects of the U.S. subprime mortgage crisis and the subsequent global financial meltdown began to trickle down the European economies toward the end of 2008 and 2009, governments were pressed into introducing remedies whose intention was to cut down on public sector expenditures so as to prevent bankruptcy. In countries, such as Denmark, with exceptionally large public sectors that include universal national healthcare, such measures forced regional hospitals to slice their budgets with hundreds of millions of Danish Kroner [7] [8]. In particular, it was thought that less on-duty personnel [8] and ensuring that patients would be released after a shorter period of time in acute care [7], would help meet the stringent budget requirements prescribed by the Danish government.

Given the prevalence of back disorders and pain in the Danish adult population as well as the enormous costs incurred on society as a result, then, there seems to be a pressing demand for innovations with the capacity at least partially to remedy these issues. With fewer personnel available per patient at Danish hospitals, and with commercial entities struggling to remain profitable, it would appear that "outsourcing" rehabilitative treatment in particular to the patients' homes is mutually beneficial.

If patients can carry out therapeutic exercises at home instead of at a hospital or chiropractor, they save valuable time by virtue of not needing to book numerous consultations at a hospital or chiropractor – with the prerequisite, of course, that any system or application enabling them to do so has a motivational aspect that persuades them to persist with the prescribed exercises, and maintain personal discipline. In addition, home based physical activity programs tend to have higher adherence [9], and the patients involved have been shown to experience a higher quality of life than those enrolled on center-based programs [10]. At the same time, if patients are carrying out the therapeutic exercises at home, the physiotherapists and chiropractors can limit the number of consultations per patient to a bare minimum, thus reducing the hospital's resource costs and easing the pressure that stems from fewer personnel due to budget cuts, against an unchanged or growing number of patients.

This is the fundamental motivation for the *IdemoBelt* application and for the *Lev Vel* project in general. There is a clear incentive – both altruistic and economic – to research and design solutions that have the capacity to support people in maintaining a high quality of life despite suffering from a medical condition.

1.2.3 Smart applications for smart devices

Technological advances over the past decade have made common stock of devices such as smart phones and tablets – yet curiously, there appears to be a sizable divide between the computational power, advanced features and ubiquity of such devices; and the extent to which current applications tend to exploit them in a smart way. At the same time, recent developments in sensors, low-power integrated circuits, and wireless communications have paved the way for designing low-cost, miniature, lightweight and intelligent sensor nodes that can sense, process and communicate relevant information about a person's condition.

Self-quantification – the acquisition of data related to a person's daily life in terms of inputs, states and performance – is neither difficult, nor rare in the mobile application space; the amount of data available, likewise, is stupendous, both by virtue of the proliferation of various sensors that can monitor, measure and interpret aspects of their user, and because of the Internet. Yet an overwhelming amount of applications that subscribe to Quantified Self (QS) have not yet made the leap from simply collecting and presenting the information to the user, to integrating it in a way that assigns *meaning* to the data, prompts an *emotional response* from the user, and *encourages* the user to interact with it to his/her personal gain.

Given the sheer magnitude of data available and the increased ability to share it, there seems to be huge potential rigged in designing applications that monitor activities and encourage patients to do more of the right things for their health, and fewer of the wrong things.

1.3 Project scope & goals

This thesis, then, attempts both to propose a viable solution for a commercial product that can address the needs of the stakeholders in the *BackTrack* project; and to illustrate the concepts of QS and Mobile Health (mHealth) in an academic context, so as to unearth their potential in developing intelligent, beneficial and motivational mobile applications that integrate data from external sensors and/or the web for everyday use.

While the analysis, design and testing of the *IdemoBelt* application forms the backbone of this project, the thesis also involves an investigation and discussion of the application of QS and mHealth in related contexts, such as Diabetes, as well as the possibilities related to integrating web-based data into QS/mHealth applications.

1.4 Hardware & software used

The mobile application described in this thesis was written in Java for the Android mobile operating system (specifically version 3.0 or higher), using the Android Development Kit (ADK) for the Eclipse Integrated Development Environment (IDE). The properties used for graphical elements of the user interface are based on XML.

The sensor used for external measurements is an Arduino based printed circuit board (PCB) with an integrated accelerometer. It is connected to an RN-42 Bluetooth™ module. The embedded software needed for serial transmission via the Bluetooth™ module was written in Processing, an open source programming language and IDE based on Java, albeit with a simpler syntax and graphics programming model.

Initial sketches were tested with the Samsung Galaxy SII mobile smartphone; later the implementation was optimized for 10.1” tablets – and tested specifically using the 10.1” Samsung Galaxy Tab 2.

1.5 Thesis structure

Following this introduction, Chapter 2 provides an overview of related work in mHealth and QS, before in Chapter 3 I describe the initial analysis that lead to the formulation of formal requirements for the *IdemoBelt* application, as well as the requirements themselves.

These requirements provide the basis for the design and implementation of a prototype application, discussed in Chapter 4; including early sketches, the design framework used, and the technical details of the programs involved.

Chapter 5 outlines the testing component of the *BackTrack* pilot study, which was carried out in close co-operation with physiotherapists and SDH patients from the Gentofte and Helsingør hospitals. I describe the testing strategy and briefly discuss the results and lessons learned from hands-on user experience testing.

In Chapter 6 I attempt to evaluate the project objectively by reviewing its goals and assessing the extent to which such goals have been met, with reference to the test results; and discuss the future ambitions for the *IdemoBelt* application, as well as potential further work in the field. Chapter 7 then summarizes the project with an overview and conclusion.

Chapter 2 – Related Work

2.1 Non-back related mHealth

Given the strong incentive to develop viable QS mHealth solutions, there is of course a profuse amount of research taking place in the field at present. As a rule of thumb, proposed solutions can be pigeon-holed into one of two approaches: wireless sensor networks (WSN) in the home and elsewhere; and wearable personal/body area networks (WPAN/WBAN) which consist of body worn or implanted sensors that communicate wirelessly with a handheld device.

Since 2002, the Danish Center for Pervasive Healthcare (CfPH) has been involved in multiple projects within WSN based mHealth, including a safe home designed to track the daily activities of debilitated elderly [11], and using sensors and a camera phone to detect and verify falling [12]. More recently, CfPH has also investigated the deployment of Nintendo's Wii Fit gaming platform in a supervised physical training context as a means to motivate the elderly to exceed their own expectations for their physical and mental capabilities [13]. In July 2012, Cognizant Technology Alliance proposed a remote health monitoring system involving advanced sensors such as Microsoft Kinect – which can detect three-dimensional movement with extreme accuracy; accelerometers, microphones, cameras and GPS; thus eliminating the need for body worn hardware used for monitoring [14]. It echoes the gaming approach to increasing motivation and compliance proposed by R. Aarhus et. al [13].

There are examples of much larger scale deployments of the WSN approach to mHealth, too, with Louisville, Kentucky – which has a relatively high average prevalence of asthma – having deployed air monitoring nodes that can identify conditions that might trigger the disease to flare up, and notify residents suffering from the condition wirelessly when weather or pollution could be a problem [15].

Members of the Telemedicine Group at the University of Twente have focused on a generic WBAN architecture for remote monitoring and treatment [16, 17, 18]; researchers at the University of Technology in Sydney have investigated and applied an alternative, albeit similar solution based on local and personal mHealth services [19, 20, 21]. Both systems subscribe to the philosophy of developing a wearable and unobtrusive WBAN system comprising sensors and a handheld mobile device, which can be adapted to different clinical applications such as detection and management.

Researchers at the Technical University of Denmark (DTU) looked at the specific case of type 1 Diabetes – which is characterized by lackluster production of pancreatic insulin, leading to a high blood glucose concentration (hyperglycemia). D. Boiroux et. al [22] noted that the current practice of insulin administration relies on the patient to perform infrequent glucose measurements and administer insulin personally where necessary, and surmised an algorithm based on model predictive control (MPC), for use in a system such as [23] or [24], which utilize a body worn sensor and injection apparatus to automatically manage the amount of insulin needed to maintain a healthy blood glucose concentration in the patient.

Indeed, Diabetes is a recurring focal point in WBAN QS mHealth related research. Like back disorders and back pain, it is a chronic disease which does not prevent its sufferers from leading relatively normal lives, provided they administer the necessary insulin or glucose injections. Because glucose is an essential substrate for brain metabolism, it can be assumed to result in synchronized brain waves that can be detected as electroencephalogram (EEG) patterns [25]. As such, an alternative means to continuous glucose monitoring (CGM) is the use of electrodes (or, more recently, the Emotiv Epoc wireless EEG helmet) to detect changes in brain activity. This approach has been studied by, among others, R. Elsborg et. al [26], C.B. Juhl et. al [27], and L. S. Snogdal et. al [28]. However, at present there appears to be a lack of sophisticated and user-friendly mobile interfaces that could enable this type of CGM to be used in conjunction with handheld mobile devices [29]. As such there is a considerable incentive to study how such interfaces could be implemented.

Pulse oximetry seems to be another area of interest in QS mHealth, which usually involves the use of an infrared fingertip sensor [30] or electronic patch [31] [32] to monitor pulse and blood oxygenation via reflectance, transmitting this data to a mobile device, and displaying some meaningful visualization based on the information. One key area of application for mobile pulse oximetry is in the pulmonary rehabilitation of COPD patients for increased motivation, compliance and relapse [4].

2.2 Back-related mHealth

Most back disorders and back pain, though acute, and certainly not without societal repercussions; are not commonly perceived as serious medical conditions. This is perhaps why very little research exists within the field of QS and mHealth in terms of supporting patients suffering one or both of them.

The *BackTrack* application, which was mentioned briefly in Chapter 1, and to which the *IdemoBelt* application is an independent sub-component; is the first real effort in Denmark to address the issues of motivation, compliance and relapse with a mobile application that enables and encourages the patient to carry out physical exercises at home. It provides the user with a platform for online consultation and appointment booking, video guides and a subjective tool to set personal goals and assess one's own performance by attaching notes to each exercise session. However, the application has no way of providing objective feedback to the patient whilst. That is, it provides neither a real-time monitoring function for use during a physical exercise; nor visual, auditory or haptic feedback to cue correctness. Much of this thesis is thus dedicated to addressing this "missing feature" with the *IdemoBelt* application.

Chapter 3 – Requirements Analysis

3.1 Initial analysis

The basis for the functional and non-functional requirements of my contribution to the *Lev Vel* project was drawn up during a meet-and-greet and introductory session at Helsingør Hospital. From the outset it had been deemed imperative that the development process be user-centric. In order to support the targeted age group of 45 to 65-year olds in leading good lives, it is paramount that one *understands* them and their needs. As such, in order to enrich our own understanding of the difficulties and challenges related to physiotherapy, the IdemoLab DELTA employees involved in the project were invited to spend "a day as a patient" at the hospital's Center for Spinal Disc Herniation (*Discuscenteret*); to try a series of dynamic stability exercises so as to build first-hand experience that could be translated into a meaningful and encouraging product and user interface (UI).

Dynamic stability exercises target a person's core posture muscles – located primarily in the back and abdomen – with the intention of strengthening those muscles, by prescribing that the person's trunk remains steady while he/she moves one or more of his/her extremities (feet and/or arms). During the exercises, the physiotherapists pointed out that they had no means of measuring how much the patient's spine moved; they relied on visual observation for directing the patient.

As such, a general and crucial aspect of the requirements was identified as some means to measure the orientation of the spine in three-dimensional space in real-time. The logical continuation of such a requirement was of course that the measurements needed to be transmitted to some device that could display them, and thus aid the physiotherapist in his/her judgment of the correctness of the movements conducted by the patient. These two requirements would serve well in a clinical setting to provide objective feedback – as long as the physiotherapist and/or patient observing the transmitted values understood their significance and could then adjust the movements appropriately based on them.

However, one of the cornerstones of the *Lev Vel* project remains the ambition to move as much of the interaction out of the hospital or chiropractor setting as possible, and into the home of the patient. For this reason, the product needed to provide a simple, intuitive and comfortable user experience, such that any physical sensor equipment had to be unobtrusive and uncomplicated to use (a simple on/off function was deemed to be the best solution, where possible); while the accompanying UI had to appear self-explanatory and present the measurements in a way that could easily be discerned and comprehended, even by less tech-savvy persons.

Furthermore, the product needed to have a persuasive and motivating characteristic. One of the physiotherapists involved in the project was quick to draw attention to the fact that a large portion of her patients tended to “cheat” with regard to the exercises she prescribed for them to carry out at home. The patients would carry out the exercises only in the first few weeks and last few days of the planned program; and then explain that they did not experience a significant improvement in their condition. She went on to point out that perhaps if the patients felt that they were being monitored or evaluated continuously, they would be much more adhesive to their programs by virtue of guilt or shame.

This investigative session, as well as a workshop conducted together with CSI shortly thereafter, laid the groundwork for the early sketches detailed later, in Section 4.1 of this report, and provided me with the necessary insight needed to outline the requirements of the project. These are discussed in the following Sections 3.2 and 3.3.

3.2 Functional requirements

The functional requirements, detailed in Table 3.2.1 below, have been divided into three priority classes according to the expected significance of their role with regard to the functionality of the application.

Primary requirements were deemed to be those that needed to be addressed in an early prototype; requirements that were crucial to demonstrating the concept. Secondary requirements refer to those that enhance the user experience to such an extent that the application could be tested in a real-life setting without significant expertise. Tertiary requirements describe those that were not prioritized; features that improve the user experience and accessibility, but are non-crucial with regard to the functionality of the application (it should, however, be noted that tertiary requirements, in this case, constitute requirements that should, wherever possible, be met before any clinical trial and/or market launch).

Requirement	Type	Description
Measure position & orientation of spine.	*Primary*	The sensor must be able to measure the position and orientation of the patient's spine in three-dimensional space.
Transmit the measurements to a mobile device.	*Primary*	The sensor must be able to transmit its measurements to an application on a mobile device in real-time.
Read data from the sensor.	*Primary*	The mobile application must be able to read transmitted measurements in real-time.
Display measurements.	*Primary*	The mobile application must have some means of displaying the measurements graphically, and in real-time.
Provide feedback on measurements.	*Primary*	A user must be able to monitor the correctness of a <u>dynamic stability</u> exercise based on the measurements, by means of auditory and visual cues.
Distinguish between different exercises.	*Secondary*	The mobile application must be able to list up to <u>seven</u> prescribed exercises by name and type.

Customize list of exercises.	*Secondary*	A user must be able to personalize the listed exercises by re-naming them and selecting their type.
Adjust exercise difficulty.	*Secondary*	A user must be able to adjust the difficulty level of an exercise.
Adjust sensor sensitivity.	*Secondary*	A user must be able to adjust the sensitivity of the sensor.
Store exercise session.	*Secondary*	A user must be able to store the measurements recorded over an exercise session for future reference.
Display performance over time.	*Secondary*	The mobile application should be able to display the stored scores, and display them graphically over time, for reference.
Compare different exercises.	*Secondary*	A user should be able to compare and contrast his/her performance in different exercises, for reference.
Store exercise settings.	*Tertiary*	A user must be able to store the settings (difficulty, sensitivity) for a specific exercise.
Comment on exercises.	*Tertiary*	A user should be able to attach subjective comments to stored exercise sessions (evaluating his/her performance, etc.).

Table 3.2.1 – Functional requirements

The table shows the functional requirements of the *IdemoBelt* application, divided into three priority classes. These requirements were constructed in close co-operation with the end users of the system.

3.3 Non-functional requirements

Given that the application discussed in this project intends to provide support for people suffering specific medical conditions, it is of course imperative that the application also meets a number of non-functional requirements. These are outlined in Table 3.3.1 below. Again, the priority classes described in Section 3.2 are employed.

Requirement	Type	Description
Comfort	*Primary*	The sensor must be physically unobtrusive and comfortable to wear during exercises.
Usability	*Primary*	The sensor and mobile application must be

		learnable, self-explanatory, and easy to use by the targeted users.
Persuasiveness	*Primary*	The application must elicit a positive emotional response from a user in terms of his/her willingness to use it persistently - a user must experience a personal gain.
Robustness	*Primary*	The sensor and mobile application must handle errors in such a way that they do not significantly impair the user experience during use.
Resource constraints	*Primary*	The sensor must be power-efficient; the mobile application must function in a satisfactory manner regardless of the mobile device used (with some leeway).
Performance	*Primary*	The sensor must utilize fast-forwarding of serial data and be able to transmit data in real-time; the mobile application must be able to read and display data in real-time.
Accessibility	*Secondary*	The application must be able to cater to people with physical impairments such as impaired sight or hearing.
Extensibility	*Secondary*	It must be relatively effortless to add or modify features and extend functionality.
Testability	*Tertiary*	The sensor and mobile application must support formal testing procedures.
Portability	*Tertiary*	The sensor and mobile application must be equally functional in the three major mobile environments (iOS, Android and Windows).
Price	*Tertiary*	The sensor and mobile application must be affordable and provide good value for money.

Table 3.3.1 – Non-functional requirements

The table shows the non-functional requirements of the *IdemoBelt* application, divided into three priority classes.

Chapter 4 – Design & Implementation

4.1 Initial sketches

4.1.1 Hardware component

Given that the sensor needs to be placed correctly onto the spine in order that the measurements be accurate, not to mention that the non-functional requirements dictate it be unobtrusive and comfortable when used, its shape and form were crucial design considerations. DELTA had considerable expertise at hand in this field, stemming from their ePatch® technology [31], so naturally one conceivable way of fulfilling the aforementioned requirement would be to embed the sensor into a durable plaster.

However, the Medical Devices Directive of the European Union dictates that any instrument, apparatus, equipment, software, material or other object used on its own or in combination; including software designed by the manufacturer specifically to be used for diagnostic or therapeutic purposes, which includes the correct use thereof, and which the manufacturer has specified for use by humans; needs to undergo stringent medical testing prior to its application involving human test subjects [33]. The ePatch® technology has undergone such testing.

The rapid prototyping nature, and thus the time constraints of the *IdemoBelt* project necessitated IdemoLab DELTA to bypass the directive by arguing that the sensor and mobile application are *not* used for diagnostic or therapeutic purposes; rather, it is, at present, described as a training aid used for motivational purposes and to assist in the correct conduction of exercises. One of the ways its application could temporarily be detached from 93/42/EEC was to introduce cloth between the sensor and human skin – this was verified through risk analysis with reference to the following documents: [34], [35] and [36]. The detailed results of this risk analysis are presented in the pre-assessment report in Appendix A.

The most logical and user-friendly way to do this was decided to be to embed the sensor into a woven belt with Velcro affixation. A rough sketch of the concept is illustrated in Figure 4.1.1 below; while in Section 4.1.2 I discuss some of the initial UI ideas that were developed.

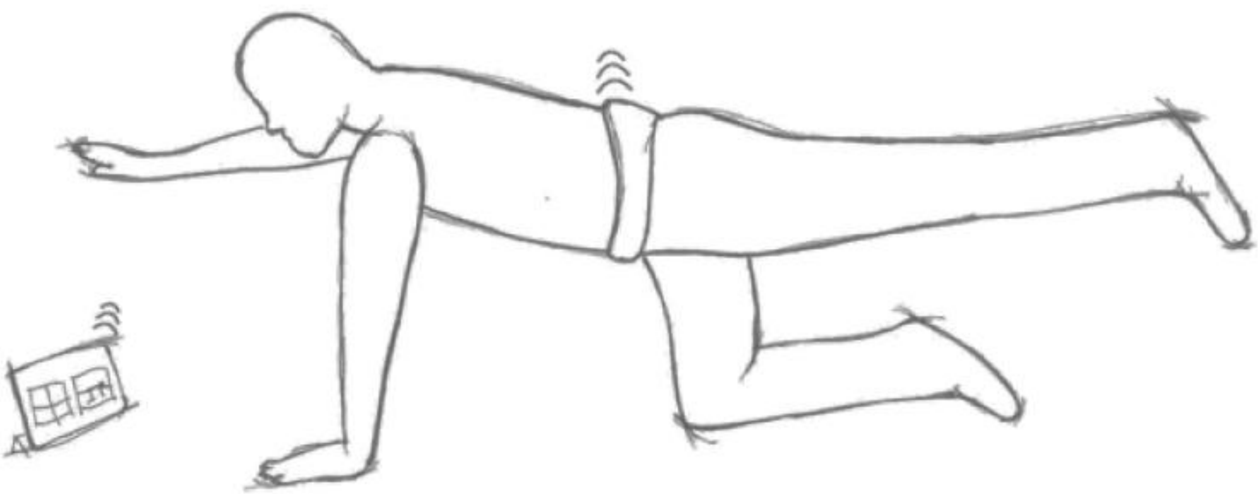


Figure 4.1.1 – Conceptual sketch of the system

Measurements from an accelerometer in the training belt are transmitted via Bluetooth™ using an RN-42 module, to an Android tablet or mobile smart phone, where the results are presented graphically to the user.

4.1.2 Affecting user behavior

During my research into COPD rehabilitation [4], I was particularly interested in the factors that influence the rate with which patients can be rehabilitated – at least to an extent that could be viewed objectively as an improvement in their quality of life. The recurring issue, pointed out by the physicians and physiotherapists involved in the *UNIK – Projekt KOL* project, was that an alarming number of patients had a tendency to prematurely quit their pulmonary rehabilitation programs due to demotivation derived from the inability to feel the gradual improvement in their condition (i.e. higher blood oxygenation). This sentiment was echoed by the physiotherapists advising this

Lev Vel project, who added that if the patients with back disorders felt the exercises that had been prescribed to them were difficult to carry out, then they were unlikely to muster up the motivation to carry them out regularly.

As a result, I felt compelled to investigate whether or not it might be possible to counter such motivational lapsing through engineering a clever user experience based on the *Fogg Behavior Model* [37] (see Figure 4.1.2) which postulates that motivation, ability and triggering must converge simultaneously in order to induce a desired behavior. In 2011, the model was selected as the framework for health behavior change by the World Economic Forum.

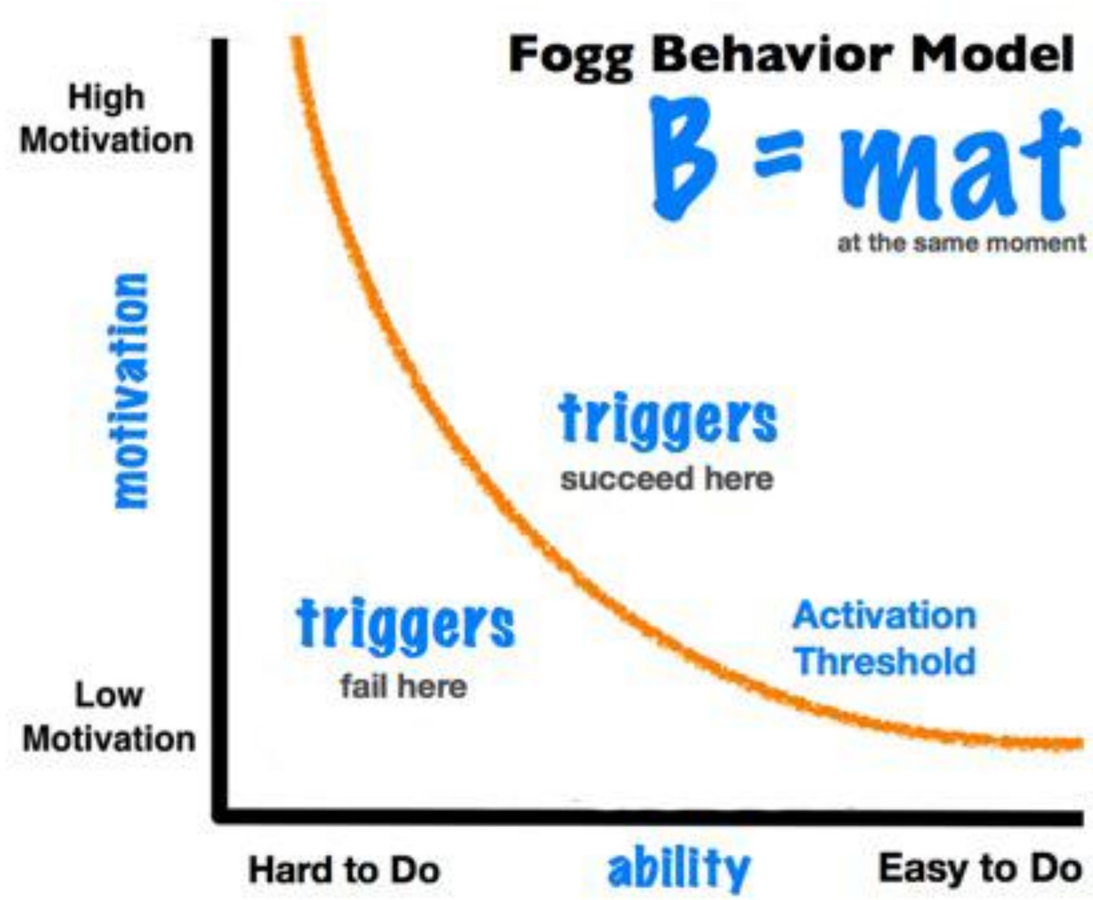


Figure 4.1.2 – Fogg Behavior Model

B. J. Fogg postulates that in order to attain a desired user behavior, a convergence of motivation, ability and triggering must take place; when the desired behavior does not occur, at least one of the three elements is missing.

B. J. Fogg's Behavior Model identifies the three core motivators as pleasure/pain, hope/fear, and social acceptance/rejection. In addition, he outlines six key factors that affect a user's perception of simplicity: time, money, physical effort, brain cycles, social deviance and the degree to which the user experience is non-routine. The three types of triggers for inducing desired behavior and the contexts that match user ability and motivation, are illustrated in Figure 4.1.3.

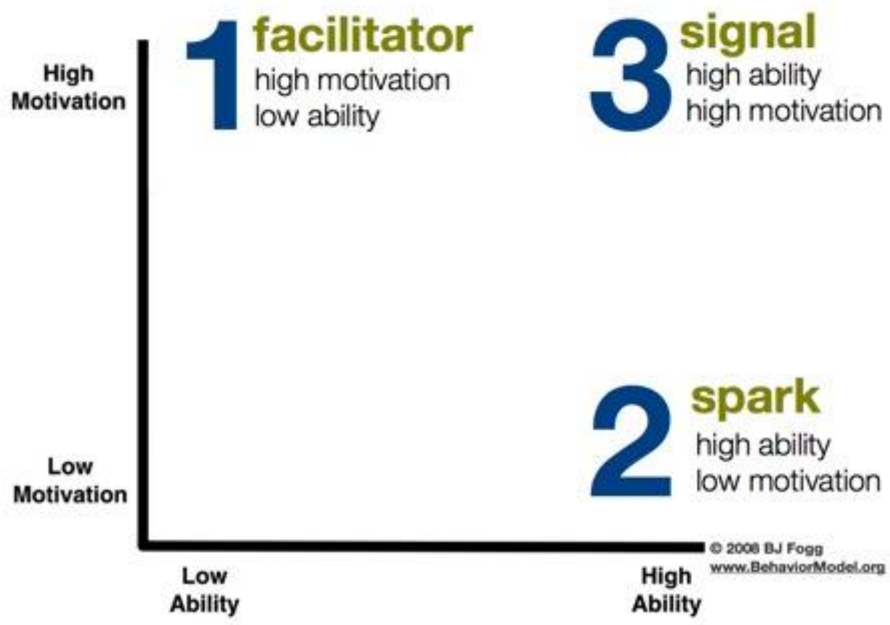


Figure 4.1.3 – Fogg's three triggers for behavior change

The triggers here are illustrated in relation to the axes of the Fogg Behavior Model in Figure 4.2.2. They should be understood in the context of cues, requests, offers, prompts and calls to action.

The application of the Fogg Behavior Model in this project is discussed in further detail in Chapter 6, where it is related to the results, in Chapter 5, of the real world pilot study conducted in conjunction with the *IdemoBelt* prototype. Its implications were considered, however, whilst sketching ideas for the UI, with the ambition that the prototype needed to find the optimal compromise between simplicity and motivation (i.e. sitting somewhere on a positive diagonal above the activation threshold).

4.1.3 User interface ideas

A key idea that arose during the participatory day at Helsingør hospital was the reflection of movement in physical space in some way. Quick brainstorming thereafter produced a number of ideas, most notably the use of individual bars with a dynamic substance "filling up" according to the magnitude of the given value, as illustrated in Figures 4.1.4 and 4.1.5.

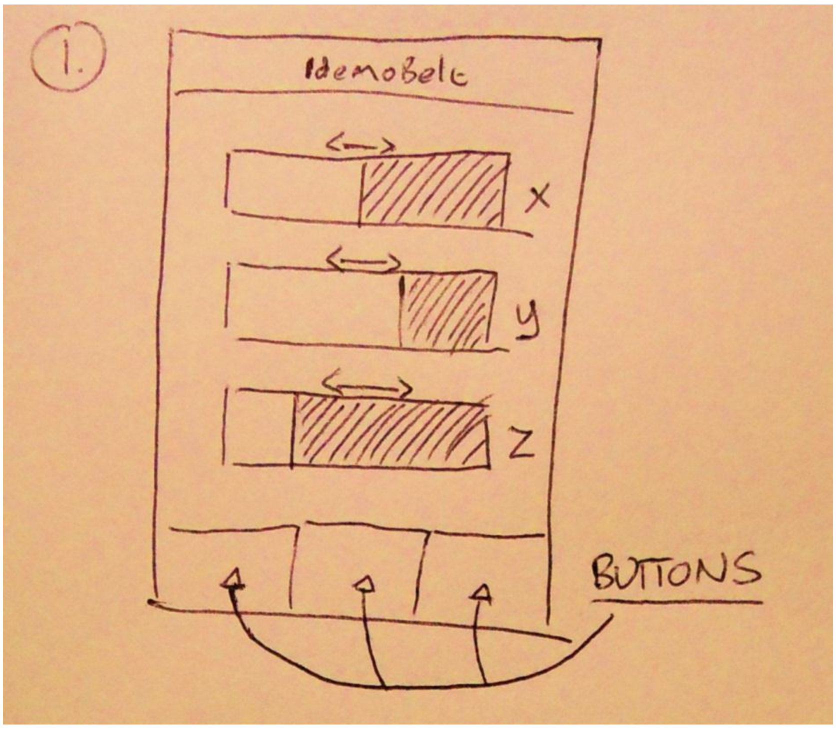


Figure 4.1.4 – Initial UI sketch with horizontal bars animation

The sketch in Figure 4.1.4 above was quickly developed with Ketai – an Android library for Processing – and presented to some of the physiotherapists and patients involved in the project. However, it was quickly determined that they had considerable difficulty understanding the meaning of the bars, let alone use them to assess the correctness of an exercise. In short, this form of presentation did not appear to be intuitive for the users. The sketch in Figure 4.1.5 remedied the issue somewhat, particularly so for the physiotherapists, for whom some ambiguity seemed to clear when direction was introduced to the bars. But there was no obvious way to represent the z-direction without creating a three-dimensional animation. The directional concept did, however, form the basis for the design used in the prototype described in Section 4.2, which makes use of an axis with an x-direction and an interchangeable y/z-direction.

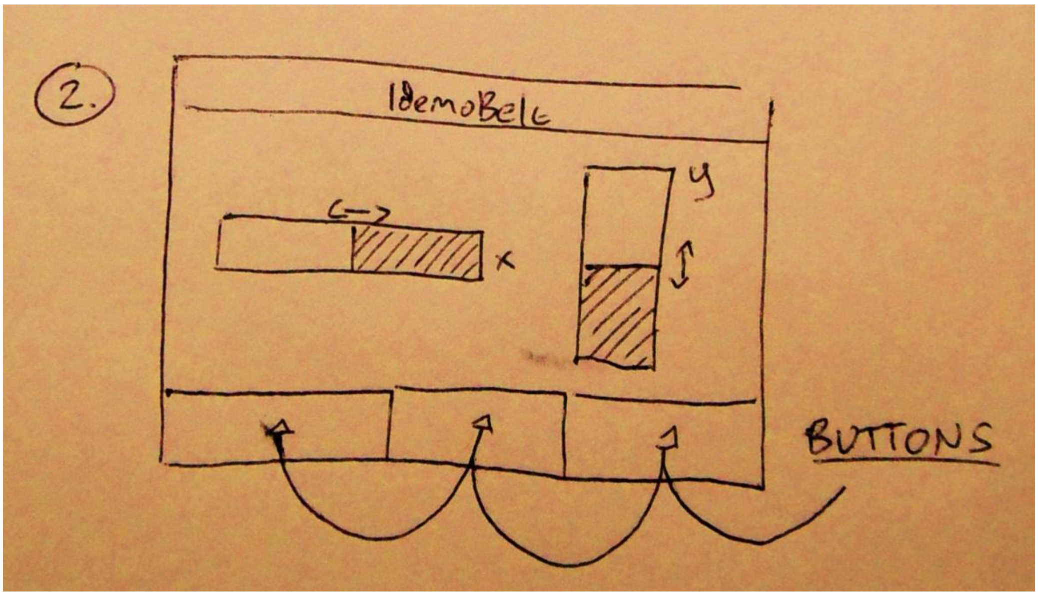


Figure 4.1.5 – Initial UI sketch with horizontal and vertical bars animation

The third concept was of an altogether different nature; an actual two-dimensional representation of the spine, which was to mimic the spine's curvature during an exercise. The idea was to create two adjustable, curved boundaries with the pretext that the user should attempt to carry out the movements in such a way that the spine image – the solid, bold line in Figure 4.1.6 would remain between them.

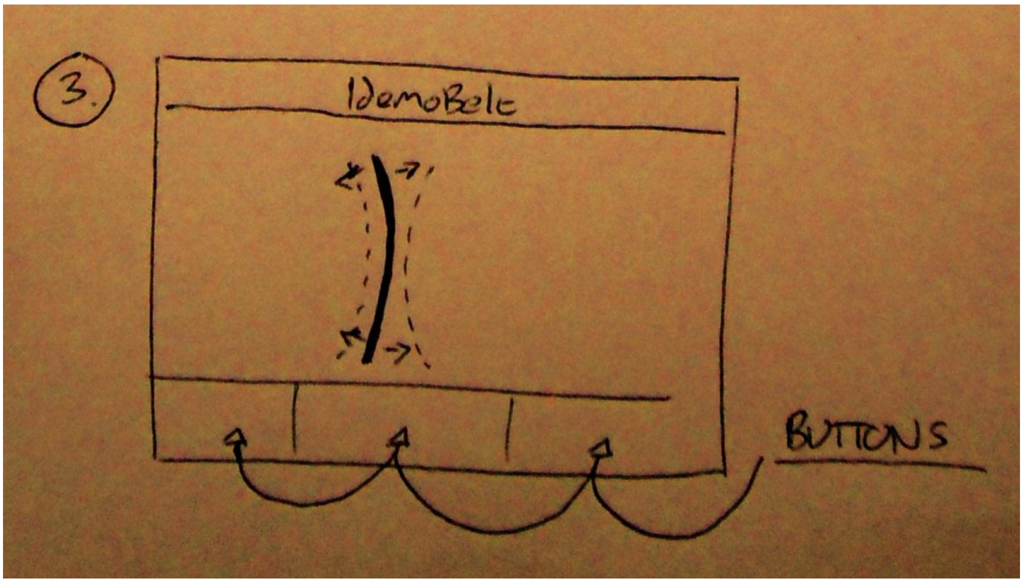


Figure 4.1.6 – Initial UI sketch with "bending spine" animation

Initially, the ambition was to feature both this animation and the aforementioned axis side-by-side in the UI, thus illustrating the correctness of an exercise in two distinct ways. However, because the latest version of the belt itself consists of just one sensor, as opposed to two in previous versions, there was no way to implement the trigonometric calculations necessary to determine the angle between two points on the spine. As such, this animation was discarded for the time being, though with the notion that it may be employed in future iterations.

4.2 Prototype

4.2.1 Overview

Development of a functional prototype for the *IdemoBelt* application constitutes a major component of this thesis. As explained in the previous Section 4.2, the prototype consists of a hardware and software component: a sensor – an Arduino based PCB connected to an RN-42 Bluetooth™ module, the schematic for which can be found in Appendix B, affixed to a textile belt (see Figure 4.2.1) – and the associated embedded software written in Processing, which is available in full in Appendix C; and a mobile application built on the Android platform, available on the accompanying CD and online (see Appendix D for details).

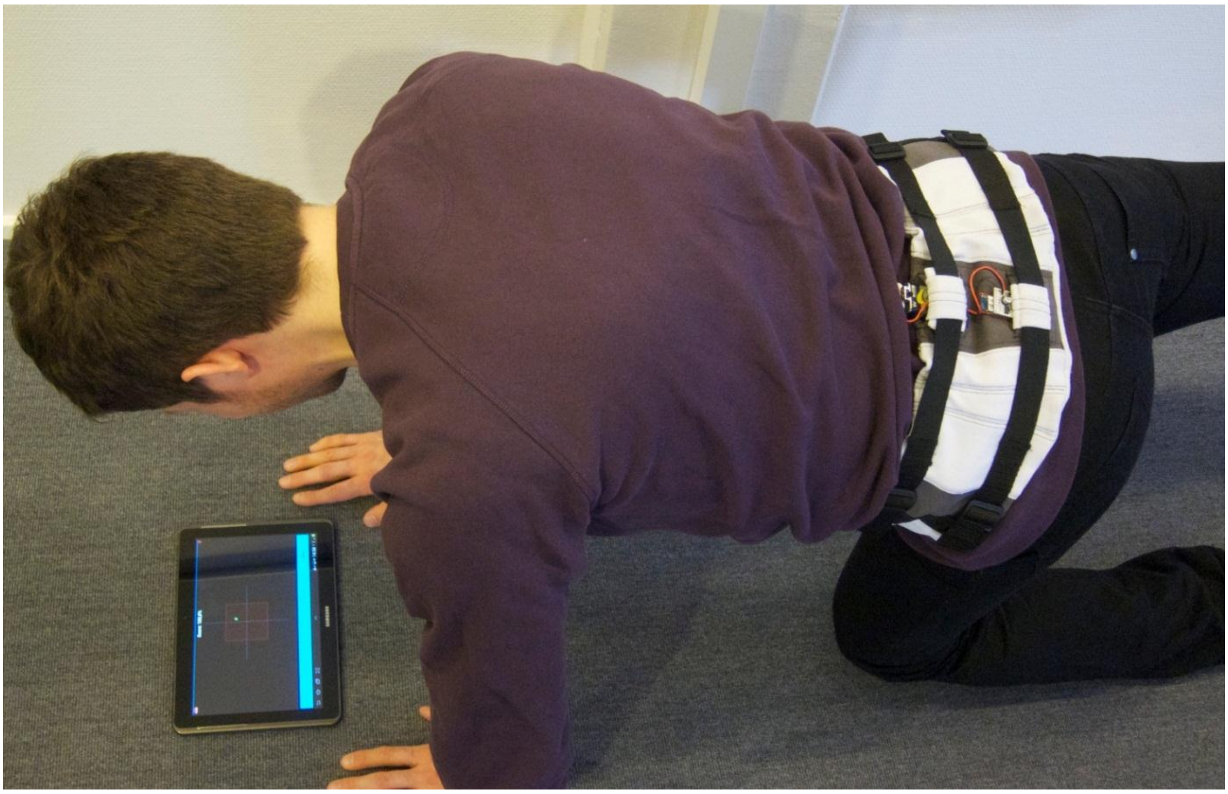


Figure 4.2.1 – Prototype implementation of the *IdemoBelt* system

The Processing program is quite simple. It includes the following five functions:

- `readADXL345();`
- `writeADXL345();`
- `initADXL345();`
- `setup();`
- `loop();`

The most important functions are `readADXL345()` and `loop()`; the former is used to read bytes from an ADXL345 accelerometer on the PCB, while the latter is the main loop responsible for formatting and serial output of the bytes. This output is of the form: `x,y,z<carriage return>`; it is transferred to the Bluetooth™ module from whence serial transmission to the *IdemoBelt* mobile application can occur. The remaining three functions are generic setup functions used for initialization operations, and to send requests to the accelerometer. They are not used in byte output per se.

IdemoBelt itself is an Android application divided into three Activities which provide the foundation for the UI:

- Exercises;
- Monitor; and
- Diary.

Each of these, in turn, makes use of standard and custom Views, which provide the actual graphical elements of the UI. The two custom Views employed are constructed in the following two Java classes implementing the Android View API:

- SquareAnimation; and
- PerformanceGraph.

Much emphasis has been placed on making the UI as simple and intuitive as possible, so as to eliminate the need for significant training in using the application.

4.2.2 Exercises

The Exercises Activity serves as the Launcher Activity of the application; it is the first screen presented to the user upon launching the application from the operating system and provides the interface to customize and distinguish between up to seven exercises. An early sketch of the application included a more traditional main Activity/welcome screen with navigation to the Activity shown here; however, because *IdemoBelt* is intended both as a standalone application, and as a special feature of the *BackTrack* application described in Chapter 2, it was decided that *IdemoBelt* should launch directly into a list of exercises for a better overall user experience.

One of the key factors in the *BackTrack* pilot project was that a prototype needed to be developed fast – which in turn necessitated a number of compromises to be made, both with regard to the program code and to the graphical elements that comprise UI of the application. Testing of the application was to be conducted with the assistance of nine volunteers actively receiving physiotherapy treatment for SDH related back pain, through up to seven different exercises (testing, including results is discussed in much greater detail in Chapter 5). Given that such testing had been pre-scheduled to take place at the end of November, 2012, the completion of a testable prototype needed to be swift, and as a result, some coding elegance had to be sacrificed in favor of functionality. For this reason, the Exercises Activity includes seven hard-coded, albeit fully customizable exercise options; the list is inextensible without further programming.

Upon installation onto a mobile device via the `onCreate()` method, the application generates a SQLite database in the physical memory of the device, with two tables:

EXERCISES		
Exercise ID	**Exercise name**	**Exercise type**
Integer, PK	String	String

Table 4.2.1 – Database table for storing exercises

PERFORMANCE HISTORY			
Session ID	**Exercise ID**	**Exercise name**	**Performance**
Integer, PK	Integer	String	String

Table 4.2.2 – Database table for storing performance scores

Table 4.2.1 contains the seven exercises listed initially with the default name and type in the Exercises Activity (see Figure 4.2.2). These are extracted from the database each time the application is started through the `onStart()` method, in order to facilitate continuity when the user alters the name and/or type of an exercise and exits the application. Table 4.2.2 is used by the Diary Activity, which can be accessed from this Activity via the Diary Button at the bottom of the screen.

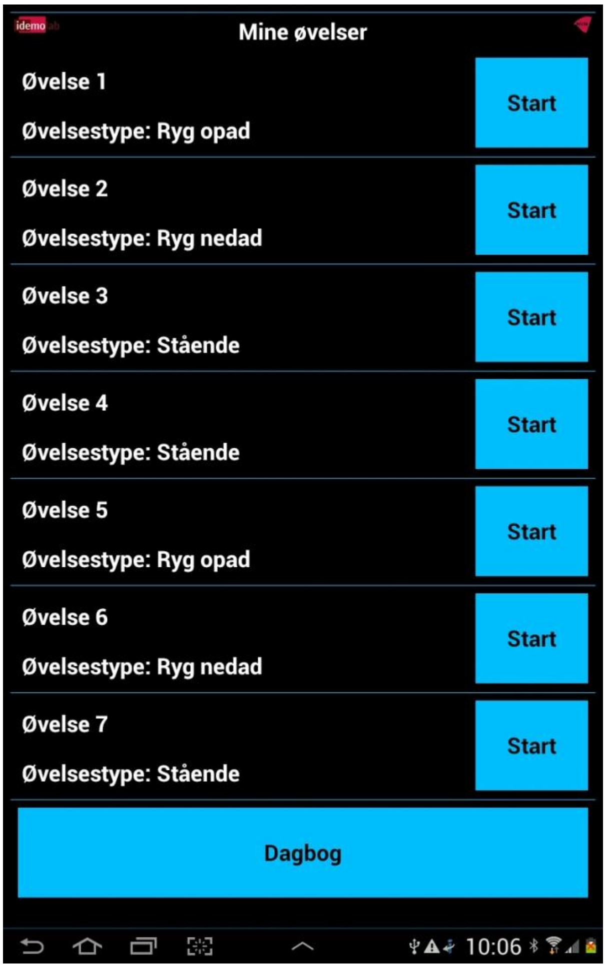


Figure 4.2.2 – Exercises Activity

This Activity serves as the main Activity/Launcher of the application. It presents the user with a list of customizable exercises, as well as navigation to the Diary Activity in Figure 4.2.7.

The exercises are listed in a stack of seven RelativeLayouts, which are native to the Android API. These contain a TextView each for the name and type of the exercise, as well as a masked Button, which facilitates an Intent to launch the Monitor Activity with the `startActivity()` method.

When the application is first installed, the `onCreate()` method ensures that an `onClickListener()` is bound to each TextView and Button. Touching either of the two TextViews will invoke a pop-up View, which allows the user to customize the name (left Figure 4.2.3) and/or type (right Figure 4.2.3) of the exercise in question. Touching the OK Button in either of these will replace the corresponding database values in the Exercises table with the new, user-defined values, and alter the contents of the TextView in question to reflect the changes.

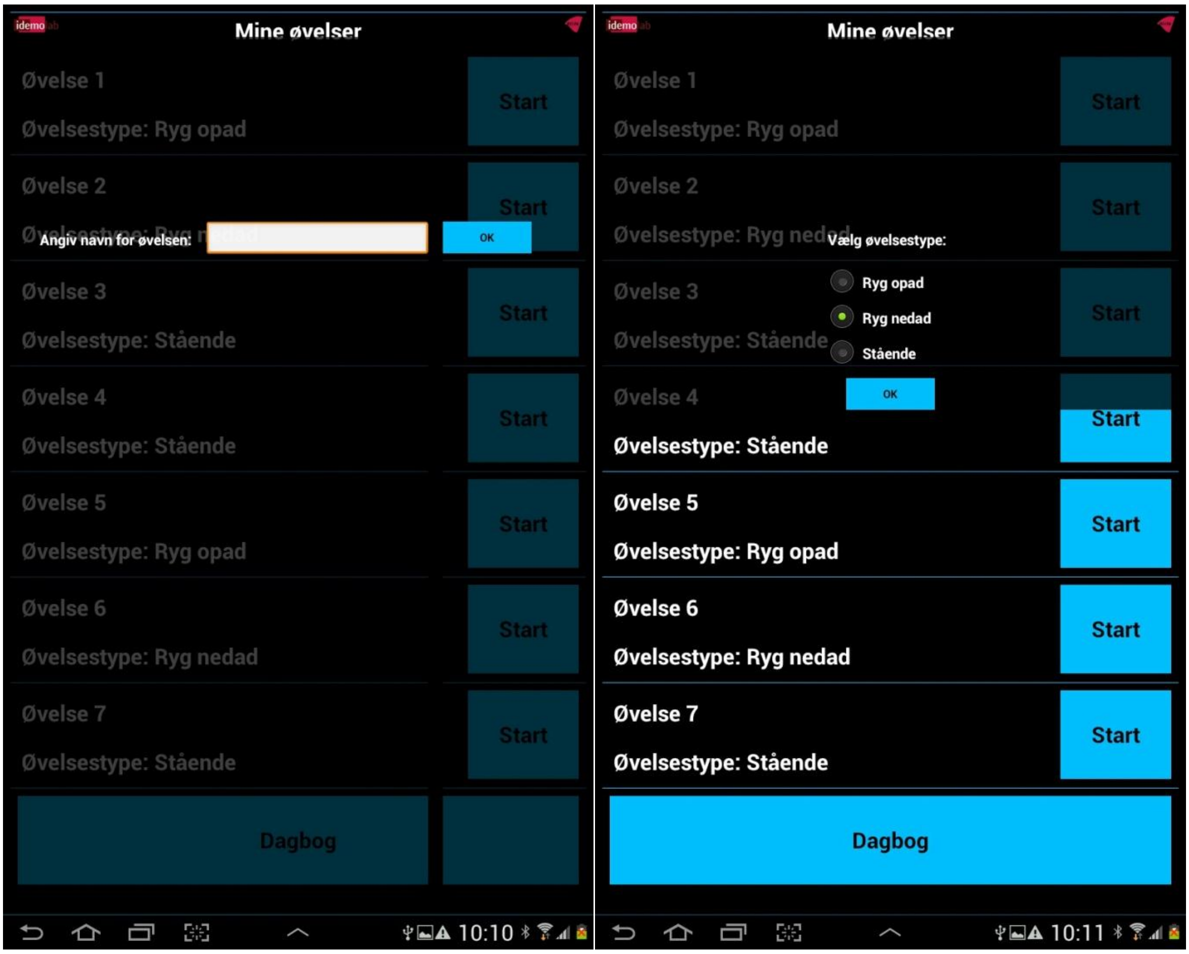


Figure 4.2.3 – Customization in Exercises Activity

Upon touching the *Exercise name* (shown on the left) and *Exercise type* (shown on the right) tabs, the user is presented with pop-up Views that enable him/her to customize the name and type of the specific Exercise. The type, in particular, is important, as the selection will dictate which axes the application will read from for the exercise in question.

The Exercises Activity also keeps static variables for the exercise ID (position of the exercise on the stack, in ascending order), name and type, as well as appropriate getter methods for these. When the user touches the Start Button corresponding to any of the seven exercises listed, the variables are set according to the current selection by checking which of the seven hard-coded

Start Buttons was pressed, thus determining the correct exercise ID, and getting the contents of the name and type TextViews. The Monitor Activity will then be launched. If the type has not been selected, the user application will use the x and y axis by default (this corresponds to the *Ryg opad* and *Ryg nedad* options).

4.2.3 Monitor

The Monitor Activity graphically displays performance graphically in real-time, and provides useful feedback to the user via visual and audio cue. It forms the backbone of the application, and as such, it is also the most programmatically complex. Its components are the SquareAnimation custom View, which contains a pair of axes, a luminescent rectangle and a dot; a performance indicator at the top; and a series of masked Buttons presenting further options to the user (see Figure 4.2.4 for reference).

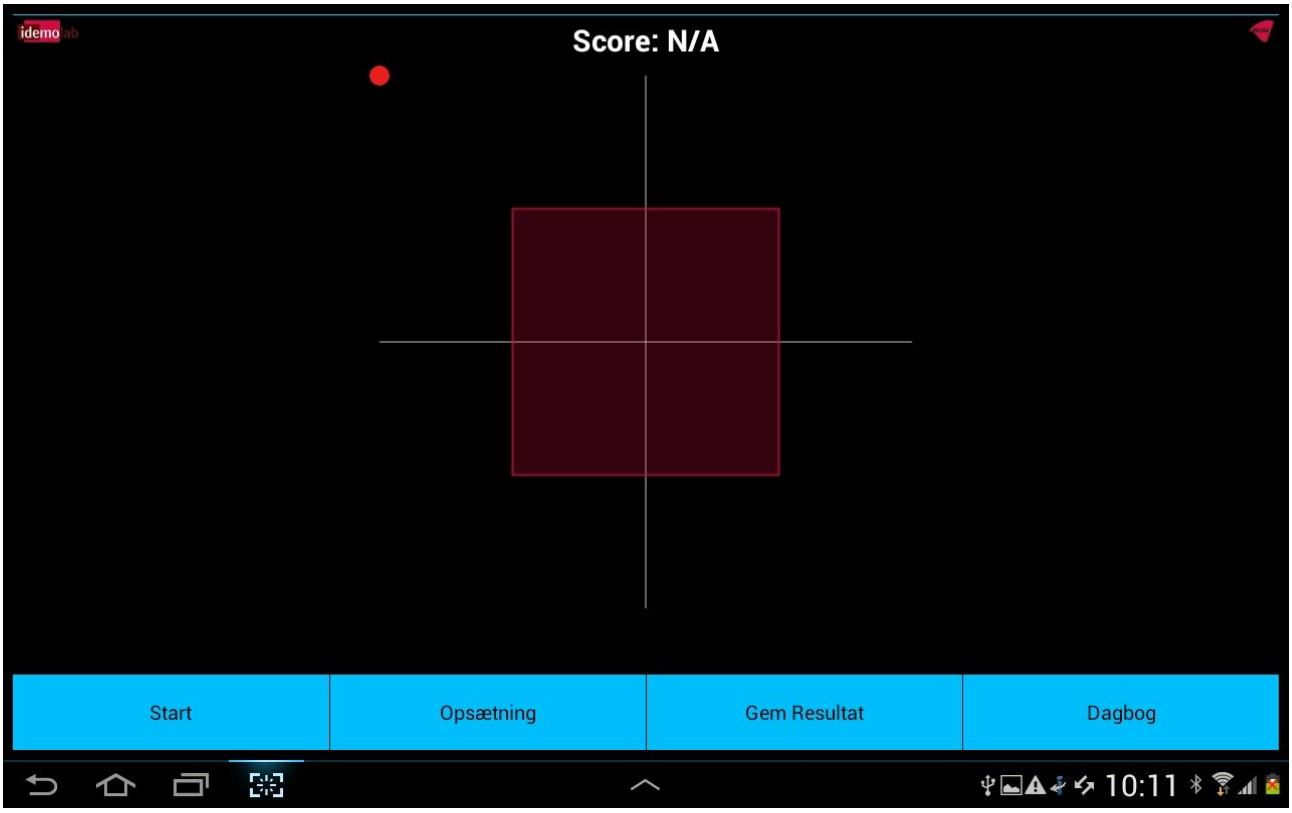


Figure 4.2.4 – Monitor Activity in disconnected mode

In disconnected mode, the dot, which indicates correctness during an exercise, defaults to the (0, 0) coordinate of the Android Canvas, and appears in red. The rectangle is approximately half the size of Canvas by default.

When this Activity is launched, the `onStart()` method invokes calls to the `getExerciseName()` and `getExerciseId()` methods of the Exercises Activity, in order to keep track of which exercise was started. In addition, the `onStart()` method performs a database query to get the corresponding exercise type from the Exercises table, and maps the type to an integer between 0 and 3. The Activity then sets its own private variables, most notably `exerciseName`, `exerciseId` and `exerciseTypeNo` according to the returned values. The `onStart()` method also initializes the sensitivity of the sensor to default, and sets the score at the top to N/A, before calling the `enableBluetooth()` method.

The role of that method is simply to turn on Bluetooth™ on the device, provided it has that functionality, by getting the device Bluetooth™ adapter with the `BluetoothAdapter.getDefaultAdapter()` method of the Android Bluetooth™ API and enabling it with the `BluetoothAdapter.enable()` method.

When the user touches the Start Button, a call is first made to the `connectBluetooth()` method. This method contains the hard-coded MAC address (of the form `00:06:66:4x:xx:xx`) and default UUID (universally unique identifier) of the RN-42 module, which are used to get a Bluetooth™ device object representing the sensor with the `BluetoothAdapter.getRemoteDevice()` method, and to create a Bluetooth™ socket for communication with the object using the `createRfcommSocketToService()` and `socket.connect()` methods.

Once the Bluetooth™ connection has been established, a call is made to the `establishInputStream()` method which, as its name suggests, creates a data input stream for the socket by execution of `socket.getInputStream()`. This method also contains calls to the `countDown()` and `readData()` methods, which can thus be called only once the input stream has been established.

The purpose of the `countDown()` method is to enable the user to get into the correct position or posture for an exercise by displaying a countdown from 5. It does this by creating a Handler instance, which posts numbers ranging from 5 to 1, as well as the expression “Go!” with a one second delay in an opaque pop-up View. This pop-up View disappears one second after the “Go!” string is posted together with a call to a method of the SquareAnimation class called `calibrate()`, which will be discussed shortly.

The dot, axes and luminescent rectangle at the center of the Activity stem from a separate class implementing the Android View API, dubbed SquareAnimation. This class is primarily composed of methods and settings that define the appearance of the custom View. Once the `readData()` method has been called, it instantiates a new Handler that facilitates reading from the data input stream in a separate Thread. Metaphorically, this method is the central nervous system of the *IdemoBelt* application. Its main component is a while-loop which reads bytes from the input stream and stores them in a temporary byte array. When the i-th byte in the array is determined to be the delimiter – in this case an ASCII encoded carriage return character (see Appendix C) – the bytes in the array are converted into a string, from which the individual x, y and z floating point values are parsed. The Handler then instantiates a Runnable used to execute the code responsible for telling the SquareAnimation View what the current position of the accelerometer is in physical space.

Figure 4.2.5 shows the graphical state of the Monitor Activity when the sensor is connected, and the reader Thread is running. The SquareAnimation View contains a method dubbed `updateDotPosition()`, which takes the x, y and z coordinates, as well as the `exerciseTypeNo` variable as input. It uses `exerciseTypeNo` to determine which two axes to read from through a conditional statement, adding each floating point number to a three-dimensional array (the position of the x-value in this array is constant; the position of the y- and z-values changes depending on which two axes are used). This array is then concatenated at the end of an ArrayList, which stores the recorded sensor positions. This is a rather inelegant coding decision, the purpose of which is to enable calibration by grabbing the last value in this array and using it as a reference point. An `invalidate()` method is then executed to trigger redrawing of the Canvas. The `updateDotPosition()` method is called every time a complete byte containing the x, y and z values is detected, parsed and passed into the SquareAnimation View (i.e. in real-time).

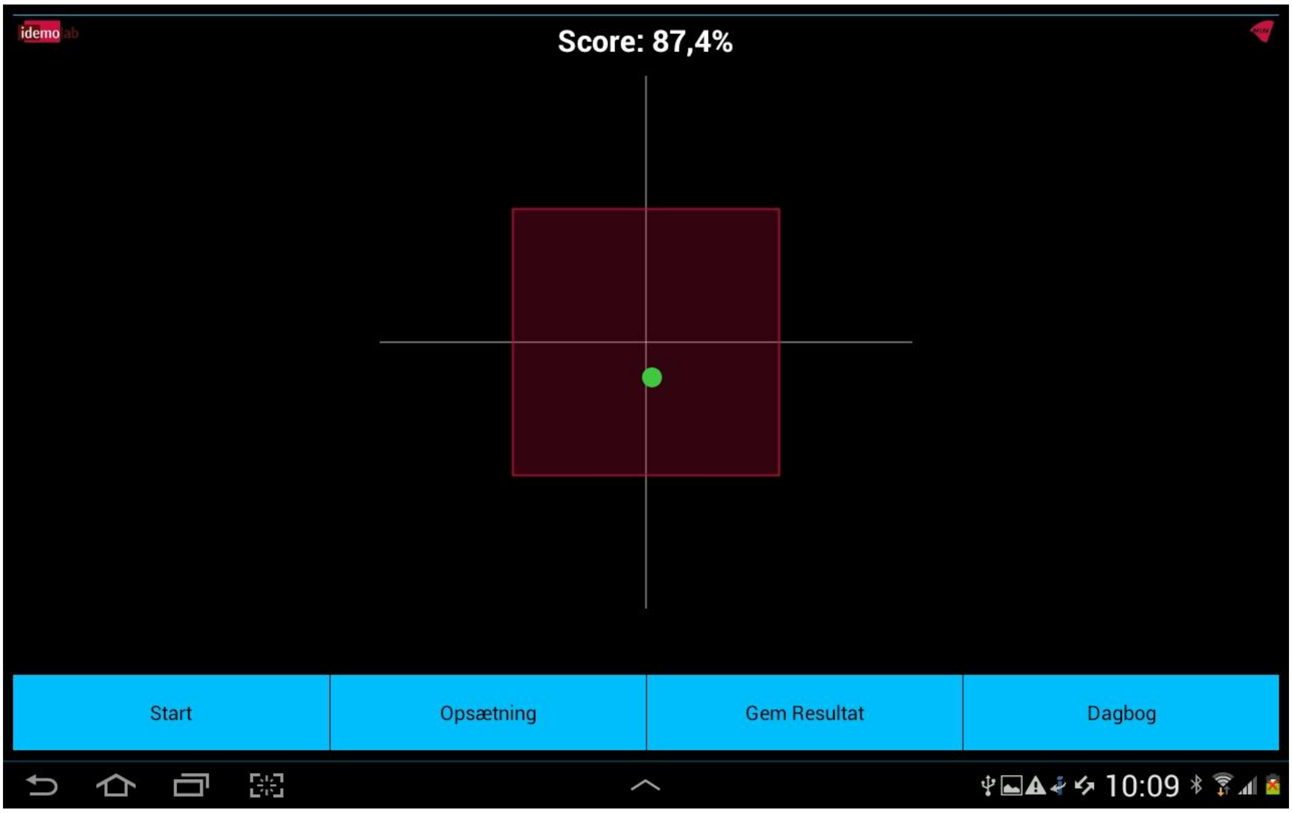


Figure 4.2.5 – Monitor Activity in connected mode

In connected mode, the dot initially calibrates to the (0, 0) coordinate on the illustrated axes, and uses this position as the basis for the dot's movements as the user carries out an exercise. It appears in green, when inside the rectangle, and red, when outside. The score at the top of the Activity indicates in real-time the percentage of time the dot remains within the rectangle boundaries – i.e. the score improves as the dot remains inside the rectangle, and worsens otherwise. Note that the Start Button changes its text to "Stop" in connected mode; however, the screenshot function on the Galaxy Tab 2 causes the Activity to stop temporarily, thus inducing disconnection.

Because the sensor uses fast-forwarding for added sensitivity to movements of the spine, I have introduced averaging to the *x*, *y* and *z* values. Before passing them to the SquareAnimation View, the values are injected into temporary arrays of size 10, added together, and divided by the size of the array; as shown by the chunk of code below:

```
// This conditional averages over 5 values read from the input stream in
order to counter excessive
// "shakyness" of the dot resulting from the fast forwarding on the Arduino
board
if (avgxval.size() < 10 && avgyval.size() < 10 && avgzval.size() < 10) {

        avgxval.add(x);
        avgyval.add(y);
```

```
        avgzval.add(z);

    } else if (avgxval.size() == 10 && avgyval.size() == 10 && avgzval.size()
    == 10) {

        float avgx = (avgxval.get(0) + avgxval.get(1) + avgxval.get(2) +
        avgxval.get(3) + avgxval.get(4) + avgxval.get(5) + avgxval.get(6) +
        avgxval.get(7) + avgxval.get(8) + avgxval.get(9)) / 10;

        float avgy = (avgyval.get(0) + avgyval.get(1) + avgyval.get(2) +
        avgyval.get(3) + avgyval.get(4) + avgyval.get(5) + avgyval.get(6) +
        avgyval.get(7) + avgyval.get(8) + avgyval.get(9)) / 10;

        float avgz = (avgzval.get(0) + avgzval.get(1) + avgzval.get(2) +
        avgzval.get(3) + avgzval.get(4) + avgzval.get(5) + avgzval.get(6) +
        avgzval.get(7) + avgzval.get(8) + avgzval.get(9)) / 10;

        mSA.updateDotPosition(avgx * ((10 + sensitivity) / 10), avgy * ((10 +
        sensitivity) / 10), avgz * ((10 + sensitivity) / 10),
        exerciseTypeNo);

        avgxval.clear();
        avgyval.clear();
        avgzval.clear();

    }
```

The resulting averaged *x*, *y* and *z* values are the input arguments of the `updateDotPosition()` method. The result is a less “twitchy” dot; its movements are smoother. The actual drawing of the dot takes place within the `onDraw()` method of the SquareAnimation View. Most of the code in this method concerns setting up the parameters of, and drawing the static graphical elements of the animation; the code snippet below is responsible for setting up and drawing the dot:

```
    // Dot position & size parameters
    float radius = 10; // Dot radius (Galaxy Tablet 2 10.1)
    //float radius = 5; // Dot radius (Galaxy S2)

    float circleX = (dot[0] - calibratedXVal) + 10;
    float circleY = (dot[1] - calibratedYVal) + 10;

    if (circleX < 0) {
        circleX = 0;
    }

    if (circleY < 0) {
        circleY = 0;
    }

    if (circleX >= rectRight || circleY <= rectTop || circleX <= rectLeft ||
    circleY >= rectBottom) {
        p4.setColor(red);
        badCount++;
```

```
            inside = false;
    } else {
            p4.setColor(green);
            goodCount++;
                inside = true;
    }

    // Draw the dot
    canvas.drawCircle(circleX, circleY, radius, p4);
```

Once the `calibrate()` method of the SquareAnimation View has been called at the end of the countdown, it sets two private variables in the class, `calibratedXVal` and `calibratedYVal`. These are important in "cheating" the coordinate system of the Android Canvas API, in which the (0, 0) position exists at the top left corner of the Canvas. The (0, 0) position of the axis exists at the center of the Canvas, and given that the *x*, *y* and *z* values transmitted by the sensor range approximately from -270 to 280, -270 to 270 and -280 to -260, respectively, the dot would be drawn outside of the Canvas for any values less than zero. The `calibratedXVal` and `calibratedYVal` variables represent the (x, y/*z*) position of the sensor at the time the "Go!" string is posted in the Monitor Activity, and by subtracting half the Canvas width (550 divided by 2) from these values, they are centered with respect to the axis. Calibration is done because the position of the accelerometer at the beginning of an exercise should naturally be considered as the reference point for any movements of the spine thereafter.

The *x* and *y* position of the dot, denoted by `circleX` and `circleY`, on the Canvas is thus the `calibratedXVal` and `calibratedYVal` variables subtracted from the *x* and *y*/*z* values set by the `updateDotPosition()` method (i.e. their position on the axis with respect to the calibrated zero position). These are used as input arguments for the `canvas.drawCircle()` method together with the radius of the circle (size of the dot) and the paint brush settings. To safeguard against values that may fall outside the aforementioned ranges (this can occur if the sensor is shaken violently, for instance), the code includes a conditional check that sets `circleX` and `circleY` to 0 in that case.

The other conditional statement involved in drawing the dot is used for feedback to the user on his/her performance. One form of such feedback is changing the dot's color to red when it exits the rectangle and green when it enters. The associated code is shown below:

```
if (circleX >= rectRight || circleY <= rectTop || circleX <= rectLeft ||
circleY >= rectBottom) {
        p4.setColor(red);
        badCount++;
        inside = false;
} else {
        p4.setColor(green);
        goodCount++;
              inside = true;
}
```

The SquareAnimation View keeps two private incremental variables dubbed `goodCount` and `badCount`, which are used to keep track of the number of times the dot's position was recorded to be inside and outside the boundaries of the rectangle. The View also contains the methods `getGoodCount()` and `getBadCount()`, which are called from the Monitor Activity in the Runnable instance of the reader Thread, cast to floating point numbers, and used to calculate the user's performance in real-time (the faction of time he/she has managed to keep the dot inside the rectangle boundaries) via the following code:

```
float goodCount = (float) mSA.getGoodCount();
float badCount = (float) mSA.getBadCount();
float performance = (goodCount / (goodCount + badCount)) * 100;
String perfAsString = String.format("%.01f", performance);

perfIndicator.setText("Score: " + perfAsString + "%");
boolean inside = mSA.getDotPosition();

if (inside == false) {
        mp.start(); // Loop the alarm beep to alert the user to the fact that
        the dot is currently outside of the rectangle
} else {
              if (lastInside == false) {
                    mp2.start(); // Play the "dot-enters-rectangle" sound to
                    alert the user to the fact that he/she has gotten the dot
                    inside the rectangle
              }
}
```

The performance is indicated as a percentage score at the top of the screen. In addition, the Monitor Activity uses the MediaPlayer class of the Android Media API to loop an alarm tone when the dot is outside the rectangle boundaries, and a single, more uplifting tone when the dot enters the rectangle.

When the user touches the Settings Button (*Opsætning*), a LayoutInflater will invoke a pop-up View, containing options related to the difficulty of the exercise: two SeekBars that enable the user to adjust the vertical and horizontal size of the rectangle in device independent pixels; as well as a SeekBar which allows the user to tweak the sensitivity of the sensor. This pop-up View is illustrated in Figure 4.2.6.

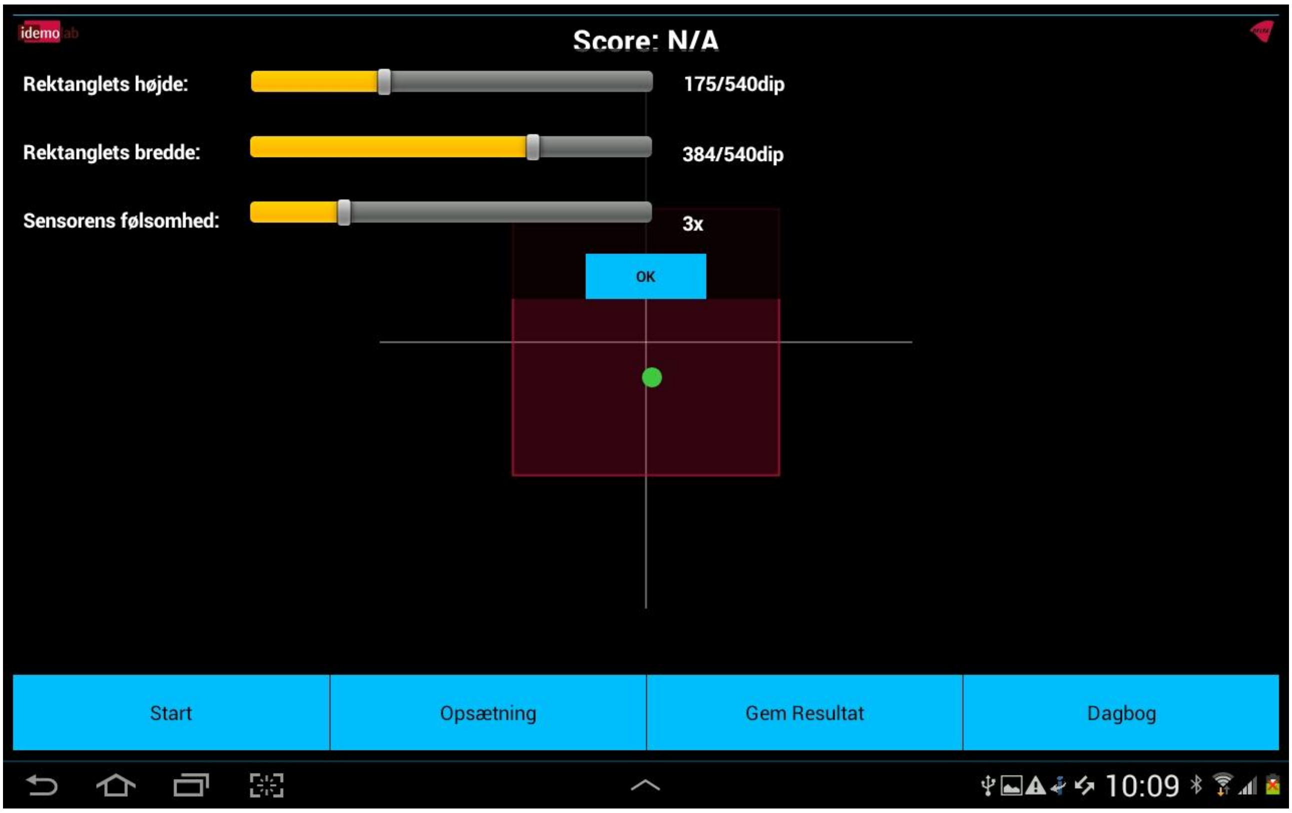


Figure 4.2.6 – Customization in Monitor Activity

The Setup button (*Opsætning*) reveals a pop-up View with options to customize the user experience – i.e. make the exercise easier or more difficult by adjusting the size of the rectangle. The sensitivity of the sensor can also be changed for a more or less responsive dot.

In order to simplify the experience for the user, the sensitivity bar *appears* to use a step size of 1, which would suggest that a sensitivity of *3x* should multiply each averaged sensor value by 3. However, initial experimentation quickly revealed that such multiplication was too drastic in reality. As a result, as the snippet of code below shows, the step size is actually 0.1, with a range from 1.0 to 2.0:

```
mSA.updateDotPosition(avgx * ((10 + sensitivity) / 10), avgy * ((10 + sensitivity) / 10), avgz * ((10 + sensitivity) / 10), exerciseTypeNo);
```

Once the user touches the OK Button in this pop-up View, the changes will be applied to the animation, as demonstrated in Figure 4.2.7.

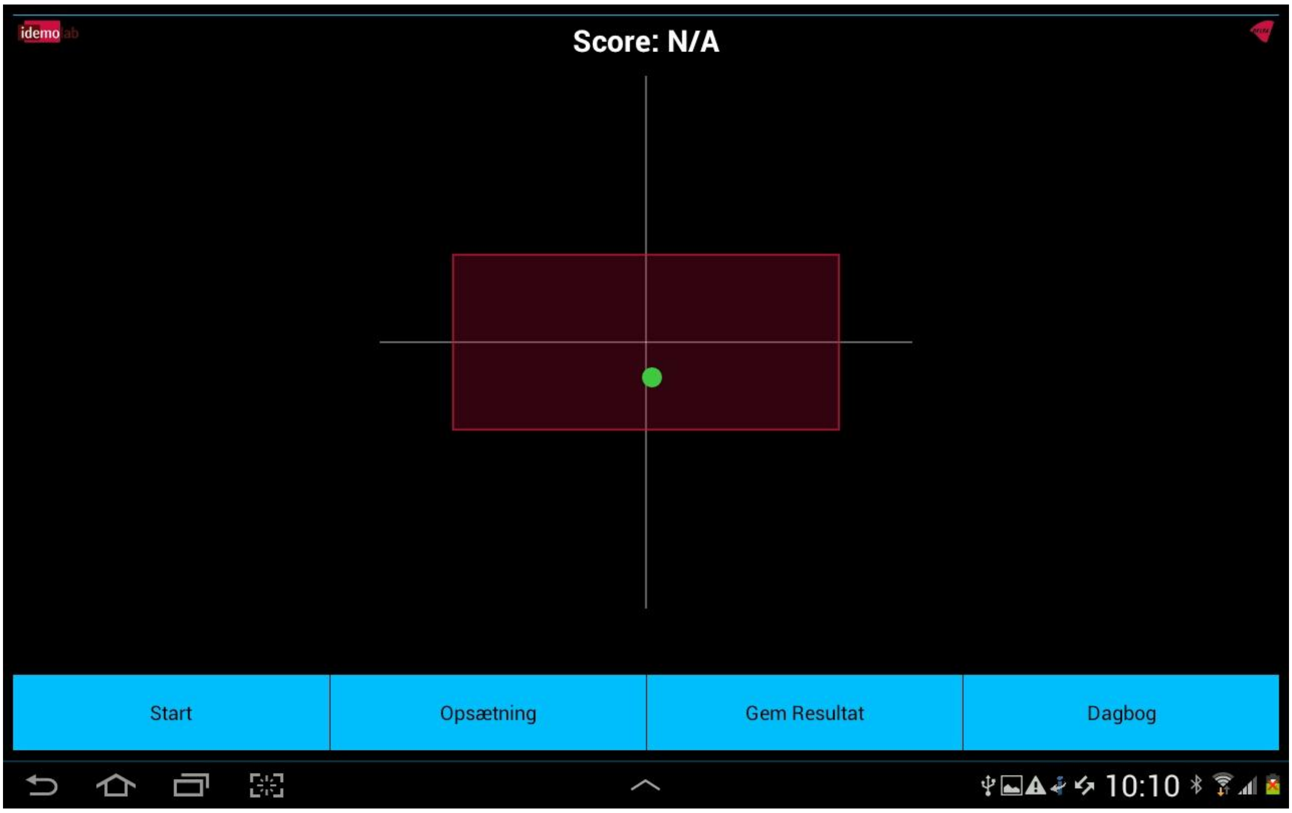


Figure 4.2.7 – Monitor Activity with resized rectangle
Here the rectangle has been made wider and shorter for an exercise which is most sensitive to changes on the y-axis (or z-axis, depending on the selected exercise type) so as to increase the difficulty of the exercise.

Touching the Save Result Button (*Gem Resultat*) will cause a call to be made to the `disconnectBluetooth()` method of the Monitor Activity, which takes steps to stop the reader Thread, close the data input stream, and close the Bluetooth™ socket. The `onClickListener()` bound to this button will then induce the `exerciseId`, `exerciseName` and `performance` variables to be inserted into the appropriate columns of the Performance History table in the database. If no exercise has been carried out, the score indicator will remain at its default value N/A, and any attempt to save the result will only result in the application informing the user that only valid scores can be stored via a Toast message at the top-center of the window. If a valid score was stored, the user can touch the Diary Button (*Dagbog*) to launch the Diary Activity, which shall be discussed in the following Section 4.2.4.

4.2.4 Diary

Although at the time of writing some of the key features intended for the Diary Activity have not been implemented, its overall role in the application is significant. It enables the user to graphically monitor his/her progress in a specific exercise over time, as well as to compare performance across the different exercises. The Activity itself is quite simple: its role is mainly to query the database for the user's performance history for each of the seven exercises; to pass this information to a custom View dubbed PerformanceGraph by calling that class' `addData()` method with input arguments `sessionIds`, `exerciseIds`, `exerciseNames` and `scores`.

PerformanceGraph itself comprises hard-coded methods to show and hide the individual graphs corresponding to each exercise, as illustrated by Figure 4.2.8.

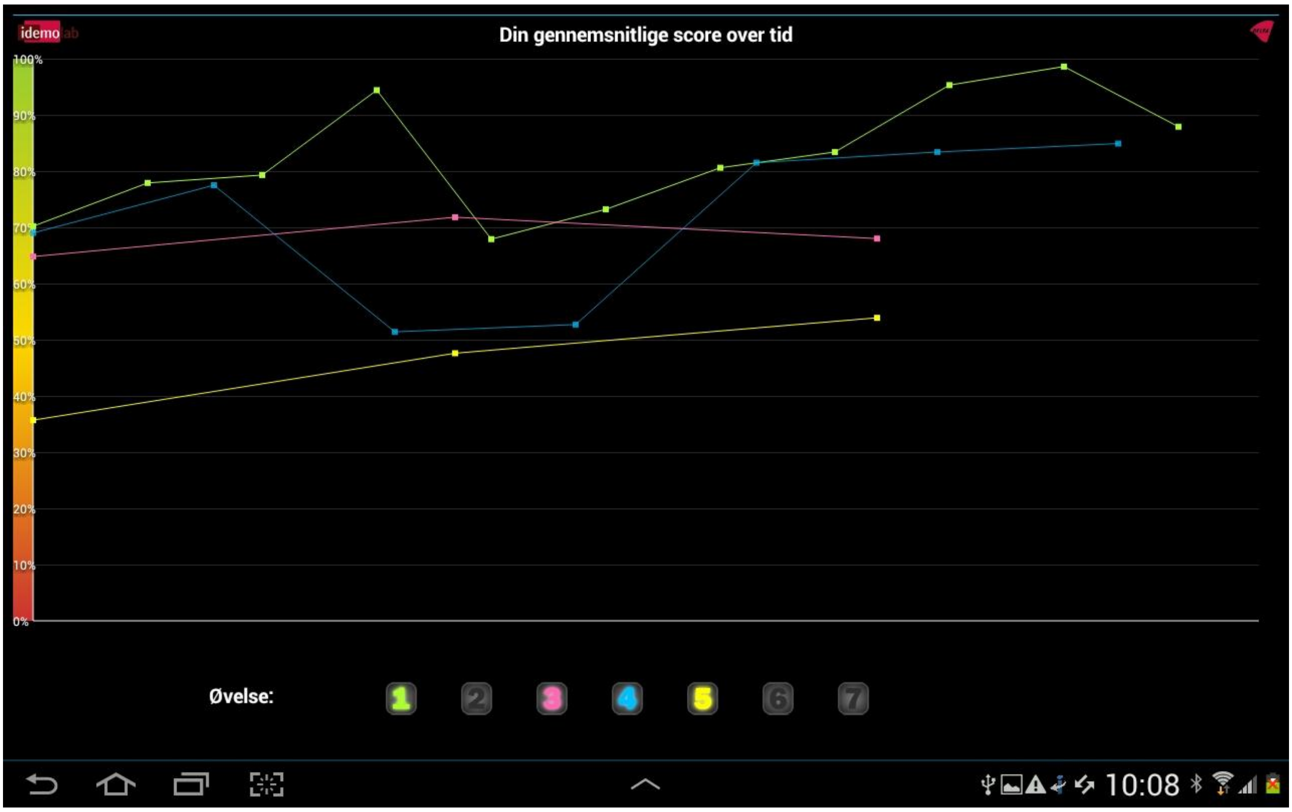


Figure 4.2.8 – Diary Activity

The graphs show the average performance (percentage scores from saved sessions, on the y-axis) of the user over time (x-axis). These can be displayed individually, or in any chosen combination by highlighting one or more of the checkboxes at the bottom.

Some of the key features still missing from this Activity are: the implementation of a scrollable graph so as to enable even spacing between the data points for every exercise (at present, each graph is drawn as a proportion of the width of the Canvas and the number of data points that exist for that exercise); pop-up Views that are invoked on focusing on the CheckBoxes that contain information about the exercise in question (name, type and average performance over time); and pop-up Views that appear when the user focuses on the data points (date, exact performance, and any notes the user will personally have made for them). Both of these features are conceptualized in Figure 4.2.9 below.

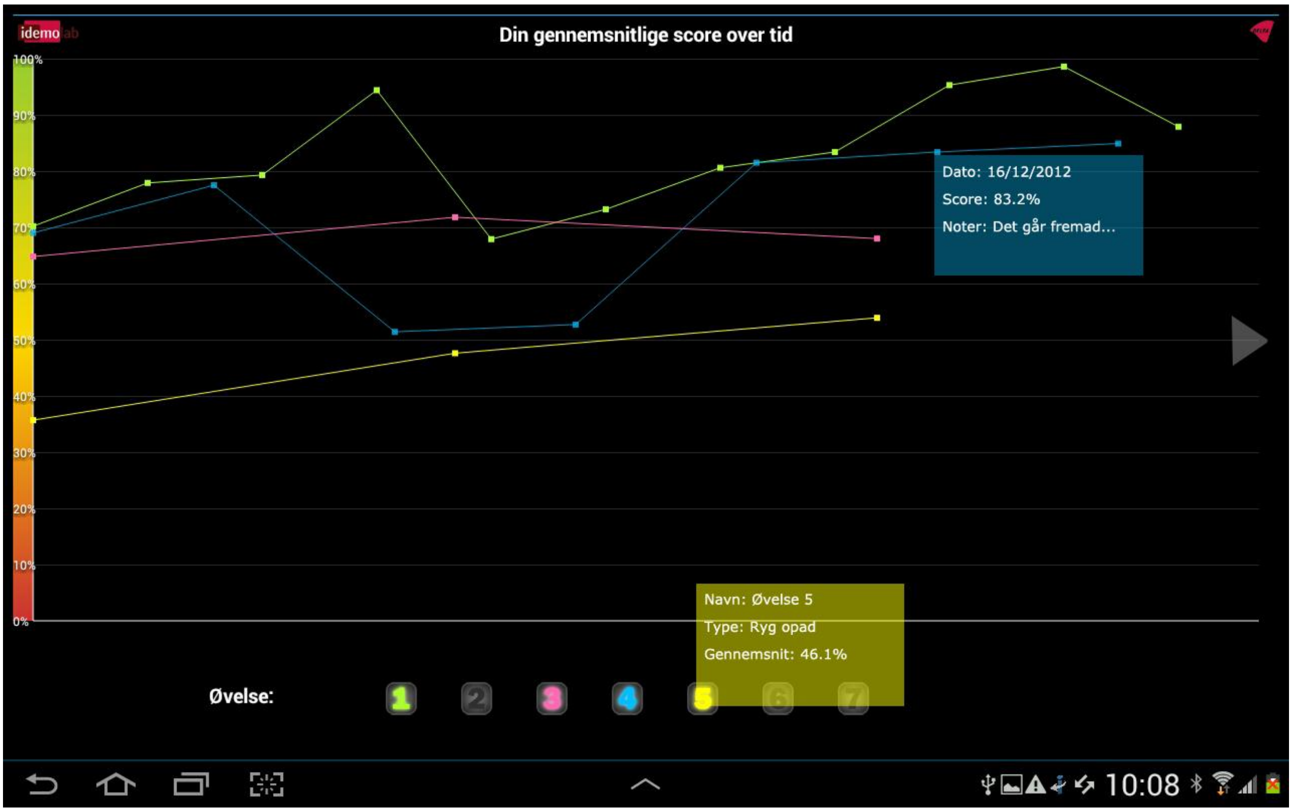


Figure 4.2.9 – Diary Activity with future improvements

This is a conceptual illustration of the pop-up Views that I intend to incorporate into the Diary in future iterations in order provide even more detailed feedback, and allowing the user to interact with the graph, and thus improve the user experience. Scroll buttons, such as the one on the right of the image, could also be added whilst making the Canvas horizontally scrollable.

Nonetheless, the current features were deemed sufficient to facilitate preliminary testing of the concept. Such testing is discussed in the following Chapter 5.

Chapter 5 – Testing

5.1 Testing strategy

Due to the prototype application described in this project stemming from a larger commercial project, and this thesis coinciding with an early prototyping phase in the development process, DELTA was able to provide me with a unique opportunity to test the feasibility of the application in a real world setting, and this way collect invaluable user feedback from the stakeholders involved in the project.

It should be emphasized that the testing in this context equates not to a clinical trial producing empirical results, but to an initial study of the characteristics and potential of the application in conjunction with the stakeholders. As such, the user feedback should be regarded as subjective; it is based on the personal experiences of the various users involved in the pilot study. Nonetheless, the feedback did provide a significant source for further development in terms of adjusting and refining the user experience.

The pilot test was conducted with a nine volunteer SDH patients in the targeted age group of 45 to 65 years from Helsingør and Gentofte hospitals, in co-operation with, and with observation by the physiotherapists involved with the patients, over the course of seven to ten days. Each of the seven was an active patient receiving physiotherapy for spinal conditions that necessitate regular dynamic stability exercises to rebuild core muscle strength in the back. The test was divided into three stages:

1. An initial introductory session (one per patient) at one of the hospitals, during which the patient was handed a 10.1-inch Samsung Galaxy Tab 2

and instructed in the use and functionality of the *IdemoBelt* application. I also encouraged the test subjects to make note of any issues they might discover, as well as to provide subjective feedback on the user experience.

2. A follow-up meeting in the patient's home on the following day – designed to ensure that the test subject was able to use the application independently, and that no significant technical issues had surfaced during initial use.
3. A "Go-Home" seminar, which encompassed representatives from each of the stakeholders detailed in Section 1.1 in an informal setting to discuss the potential of the *IdemoBelt* application (and the *BackTrack* project in general); to share user experiences; and to provide a debate platform for questions, concerns and further ideas regarding the project.

At the time of the testing, the application adhered to all of the primary functional requirements outlined in Table 1 of Section 3.2 and approximately half of the secondary functional requirements. The goal of the test was to seek answers to the following questions:

- Does *IdemoBelt* strengthen a patient's <u>ability</u> and <u>motivation</u> to carry out rehabilitating dynamic stability exercises?
- Does *IdemoBelt* help ensure that patients <u>persist</u> with, and complete their rehabilitation programs?
- Does *IdemoBelt* increase (or complement) the efficiency of <u>existing</u> training and exercise opportunities?

5.2 Results

As mentioned in the previous Section 5.1, the test results obtained are not quantifiable or conclusive; their validity cannot be proven empirically, and their origin should not be confused with formal testing procedures. Rather, the results represent consolidated feedback collected from the stakeholders during the two initial meetings and the "Go-Home" seminar.

Generally, the feedback from each group of stakeholders was positive. The patients in particular commented that they felt *"better equipped to be able to carry out the exercises at home"* and that it was *"easier to do the exercises* *correctly with the belt"*. Furthermore, a recurring theme in their feedback was that the test subjects felt motivated *"to collect data"* knowing that they would *"discuss it with their physiotherapist"* and that *"it* [was] *nice to have something concrete to talk about, when meeting with the physiotherapists."* [38]

The testing did, however, reveal a major issue inherent in the Arduino sensor used to measure the position and orientation of the test subject's spine in physical space – pointed out by multiple test subjects and physiotherapists. On each power-up or reset, the sensor engages a set-up mode lasting 60 seconds – programmatically this is the equivalent of a function that initializes variables, pin modes, external libraries, etcetera in the embedded program. Quite inexplicably, if the mobile device establishes a Bluetooth™ connection with the sensor during these 60 seconds, the connection will drop after only a few seconds, and the animation in the mobile application will consequently seize. The set-up mode is distinguished by a rapidly blinking LED on the sensor, and upon pinpointing it as the source of the connection drops, it was decided that until the next version of the hardware, the frequency of the blinking LED was to be used as the cue for when the sensor is ready for transmission, and thus when the test subject should place the belt around their lower waist/hips and press the "Start" button in the mobile application's UI.

The positioning of the sensor itself was found to be a point of confusion, as the physiotherapists at Gentofte hospital in particular, were keen to place it approximately in the crevice of the spine (i.e. where the spine bends inward). However, through trial-and-error it was quickly determined that the sensor should, in fact, sit at a position corresponding to the hips – this is where the most fluctuation in the accelerometer values was observed during the exercises (i.e. this is the position where the spine moves most).

Informal discussions at the "Go-Home" seminar also created some food for thought with regard to the UI of the mobile application itself. One of the suggestions aired was whether or not it could be useful to add haptic feedback as well as the existing visual and audio types. I did experiment with adding a vibrator that was activated when the dot fell outside of the rectangle perimeter to the application, but the collective – though subjective – opinion at IdemoLab seemed to be that its only contribution was to add an element of distraction – at least if implemented as a looping function like the existing alarm tone.

Furthermore, when a physiotherapist examined the application, she seemed somewhat confused by the role of the axis in the Monitor Activity's animation; she felt compelled to carry out movements related to an exercise in such a way as to ensure that the dot remained on, or in close proximity to the axis, whilst apparently overlooking the purpose of the luminescent red rectangle area – in reality, anywhere within this area constitutes correctness. After shedding some thought to this predicament, I decided that the best course of action for future development would be to add a feature into the settings which would enable the user to hide or show the axis – to add more support for customization.

Despite clarifying that the pilot study only incorporated developing a prototype that could assist patients suffering from back disorders and/or pain that can be relieved through dynamic stability exercises (revisit Section 3.1 for clarification), another physiotherapist also pointed out that the vast majority of his patients would need an application in which the idea was not to limit movement to a minimum, but instead to guide patients in carrying out a plethora of very specific movements correctly – i.e. support for non-stability exercises.

Worth mentioning here also is the skepticism with which one experienced physiotherapist regarded the use of technology such as the *IdemoBelt* application in physiotherapy. She felt that it would introduce an unnecessary learning curve and require too much of a time overhead for training, while she also felt that most of *her* patients would be reluctant to use cutting edge technology in the first place, given their lack of experience with devices such as smart phones and tablets.

Chapter 6 – Discussion

6.1 Evaluation of test results

The design of the *IdemoBelt* application discussed in this thesis is based on the Fogg Behavior Model (FBM) [37] (see Section 4.1.2) for affecting health change behavior, with the pretext of finding an optimal compromise between motivational aspects and ease of use. As mentioned in Chapter 5, the results of the pilot study should not be confused with empirical data that can be used as a business case for the application. Such empirical results would require thorough clinical testing, and a more advanced prototype than the one described in Section 4.3. The feedback from patients, physiotherapists and experts can, however, be used for an initial evaluation and validation of the prototype, and as a springboard for future development.

Interpreting the feedback in Section 5.2 reveals, at least to some extent, a successful application of the FBM. When instructed to adhere to dynamic stability exercise programs at home following orientation at a consultation by a physiotherapist, the nine patients involved in the *BackTrack* project allegedly felt ill-equipped to carry out the prescribed exercises correctly. However, when they were given the *IdemoBelt* application, users felt that the visual and auditory feedback gave them a better understanding of the correct way in which to do each exercise, or, at the very least, gave them additional motivation to continue *trying* to do them in the correct way. As mentioned in Section 2.1, the utilization of a game-like interface has been shown to motivate users to reach beyond their perceived physical capabilities [13]. The hypothesis is consolidated by game researcher Jane McGoniga: *"Gamers spend on average 80% of their time failing in game worlds, but instead of giving up, they stick with the difficult challenge and* *use the feedback of the game to get better. With some effort, we can learn to apply this resilience to the real-world challenges we face."* [39] It is possible that in trying to main as high a score as possible in the *IdemoBelt* application, the users had a similar experience. There are plans in motion to confirm this with detailed interviews with each, or at the very least some of the nine patients in the near future; however, those fell outside of the time frame for this thesis.

That many of the patients also explained they felt more encouraged to carry out the exercises when there was *"something concrete to talk about"* [38] at their next consultation with a physiotherapist, suggests that the Save Result Button in the application acts as a significant trigger; a prominent facilitator [37] which enables the user to perform the desired behavior with ease. It is effective, because this small behavior is likely – judging by the patient feedback – to lead the user to perform more complex behaviors (e.g. to monitor his/her performance over time using the Diary function, and to use the results as the basis for a more objective discussion, with a physiotherapist, of his/her progress with regard to relieving the disorder and/or pain) – that is, a natural chain of events that an effective Trigger puts into motion [40].

Although such potentially affected behavior needs to be validated through more in-depth clinical testing, there is certainly reason to believe that even in early prototype format, the *IdemoBelt* application *does* seem to strengthen a patient's ability and motivation to carry out physiotherapeutic exercises, and has the capacity to help them with compliance issues. If these two requirements are satisfied, then it can also be postulated that the amount of time and effort required for completing a given physiotherapeutic program can be reduced (by virtue of "outsourcing" the therapy to a patient's home), thus improving the efficiency of rehabilitation [9, 10].

6.2 Known technical issues

There are of course a number of technical aspects to the *IdemoBelt* application that demand improvement – beyond those detected via the testing outlined in Section 5.2. One of these is the versatility of the Diary function, which at present only has the capacity to display recorded data, and no means to interact with it. Pending improvements to the Activity were discussed in Section 4.2.4 and illustrated by Figure 4.2.9, and considering the significance of the Activity in terms of providing motivating feedback as well as a platform for objective analysis and discussion of a patient's performance, its continued development should be a priority. A key way to improve its functionality would be to add a pop-up View that is inflated upon touching the Save Result Button, containing the option to attach a personal note to the exercise session, with a subjective analysis of a patient's own feelings regarding his/her performance. This note could then be displayed as another pop-up View in the Diary Activity upon touching the appropriate data point in the graph, as suggested in Section 4.2.4. Furthermore, the graph clearly needs to be revamped with a more obvious reflection of time on the *x*-axis, so that the data points are positioned according to the dates of the exercise sessions, and not proportioned with respect to the amount of data points for each exercise. Such an improvement would most likely require the implementation of a scrollable View in the PerformanceGraph class, as well as of course annotation of the *x*-axis with dates.

Another issue that was uncovered post-testing is the lack of a Save Button in the Settings pop-up View, which dictates the size of the rectangle and the sensitivity of the sensor. At present, these settings need to be adjusted manually each time an exercise is carried out – otherwise the rectangle will assume its default size, and the sensitivity will be set to 1. The fix is relatively simple: extra columns in the Exercises table of the SQLite database storing the rectangle height and width, and the sensor sensitivity. The stored values could then be retrieved upon launching a particular exercise.

Coding elegance in general also offers room for improvement. Much of the application, at present, is not particularly flexible to changes. There is, for instance, no support as per yet for automatic Bluetooth™ discovery and pairing; the MAC addresses of each existing belt needs to be uncommented in the program code for the Monitor Activity (specifically in the `connectBluetooth()` method). In similar vein, the list of exercises in the Exercises Activity is currently limited to seven hard-coded exercises, and the same is true of the Diary Activity.

6.3 Future perspectives

The *IdemoBelt* application remains a work-in-progress, and DELTA has every intention to continue its development – both the belt and hardware, and the software. A breakdown of the immediate plans for the mobile application itself is given below:

- The Exercises Activity will be given a landscape orientation so as to avoid confusion incurred by window flipping.
- The TextView that appears in the pop-up View for customizing the name of an exercise needs to retrieve and display the *current* content of the TextView that contains the name of the exercise in the Exercises Activity window.
- Support will be added for the user to define which *axes* to read from in the sensor's accelerometer, as opposed to the three hard-coded *types* in use in the Exercises Activity at present.
- Support will be added for Bluetooth™ device selection in the Exercises or Monitor Activity – either by automatic discovery and pairing, or by providing a list of available MAC addresses (mapped, perhaps, to a unique identifier corresponding to each belt).
- An option to show/hide the axes in the SquareAnimation custom View will be added to the Settings pop-up View of the Monitor Activity.
- Additional functionality for saving the rectangle size and sensor sensitivity in the Settings pop-up View to the SQLite database will be added.
- A new pop-up View will be added for touch events on the Save Button in the Monitor Activity, with the option to attach a personal note to the given exercise session.
- Pop-up Views will be added for touch events on data points (date, score, personal notes) and the CheckBoxes (exercise name, type and average performance over time) in the Diary Activity.
- The PerformanceGraph custom View of the Diary Activity will implement a scrollable View for a more realistic representation of time on the *x*-axis.

Once these amendments have been implemented, the *IdemoBelt* application will theoretically be ready for clinical testing. The more long-term ambitions with the application include:

- Support for non-stability (N-S) exercises. This could be achieved by first recording the accelerometer values for the correct movement for an N-S exercise, and drawing a line based on these values. Some default buffer would be added around this line, and the resulting shape would then serve as the zone within which the dot should remain during execution of the exercise.
- Formulation of an algorithm with the capacity to calculate the optimum size of the rectangle (or arbitrary shape) and sensitivity of the sensor based on some parameters related to ability and difficulty.

Furthermore, the potential of integrating data from other sensors, such as GPS or the embedded accelerometer that most contemporary mobile devices house, as well as the web, will be investigated. There seems to be considerable promise in developing an application not just for monitoring, but also for prevention, so that users can avoid developing further back disorders.

One of the ways in which to introduce preventative functionality might be to utilize the device accelerometer to judge posture at regular intervals, and notify the user via auditory or haptic cues when his/her posture is judged to be sub-optimal. Such functionality could be particularly useful to users whose jobs entail much computer use. GPS and web data could then be integrated to suggest physical activities based on the user's location at specified intervals – imagine, for instance, a mobile application which determines that the user is not at work but has been sitting down for too long in a sub-optimal posture, and thus prompts him/her to go jogging in the next 30 minutes to avoid the rainstorm it has detected via web based real-time weather forecasts. The possibilities are vast.

Chapter 7 – Conclusion

In this thesis, I set out to design and implement a viable prototype mobile application that could be used for the physiotherapeutic home treatment of patients suffering from back disorders and/or back pain such as Spinal Disc Herniation or Spinal Stenosis that can be relieved with dynamic stability exercises. In order to understand the requirements for such an application, the entire process including analysis, design and testing was carried out in close co-operation with physiotherapists and SDH patients from the public hospitals of Gentofte and Helsingør in region Nordsjælland. From the outset of the design process, it was imperative that the application be constructed in such a way as to increase adherence and boost compliance [9, 10] by affecting user behavior through a well-designed UI based on the Fogg Behavior Model [37]; to relieve symptoms with increased rapidity and thus ease the pressure on the public healthcare system.

The preliminary testing that was conducted immediately in the wake of completing an early prototype did reveal the effect of some trigger above Fogg's activation threshold, as test subjects felt more encouraged to carry out their often difficult exercises when they were assisted by visual and auditory cues. In addition, both physiotherapists and the patients referred to the objectivity of the measurements as an excellent foundation for discussing the patients' progress at consultations. There seemed to be a definitive aspect or aspects in the proposed user interface that found a sound compromise between motivation and ease-of-use. The results were, however, inconclusive in the sense that they cannot be regarded as empirical due to the limited and informal nature of the testing procedure.

As body worn sensors and mobile applications utilizing them for self-help become more popular, focus is beginning to shift toward finding ways to change health-related behavior. P. Cipresso et. al [41] recently carried out a study based on psychophysiological measurements on the capacity of a mobile smart phone to affect user states and thus change their behavior with regard to stress management outside the clinical setting. They concluded that although the efficacy of mobile self-help approaches has been verified through several studies, there appears now to be a pressing demand for conducting in-depth studies that test the feasibility of mobile applications to actually elicit core affective states; something that the intriguing feedback outlined in Section 5.2 and discussed in Section 6.1 certainly confirms.

Appendix A
Pre-assessment report

Forhåndsvurdering

- ***Back Training Belt 1***

Projekt Lev Vel - FS
Træning, Rygtræning
Projektnr.: FU20051

Introduction:

The Training Belt is a part of the Lev Vel project, specifically the Preventive Self-Monitoring sub project. The purpose of the Lev Vel project is to contribute with the handling of the specific challenges of the society, in association with the ageing society.

The partnership consists of innovative companies including: DELTA, Center for Sundhedsinnovation, Mobile Fitness, and Welfare Solutions. The partnership also works in close collaboration with patients and health care professionals from hospitals.

The goal of the project is to further develop work begun in the pre-project phase of the Lev Vel project, namely to test, further develop and detail the concept - producing a proof of concept.

The Training Belt is a motivational tool, designed to accompany patients who are currently seeing a physiotherapist. These patients are prescribed specific exercises for their needs and would use the Training Belt to further motivate them to do their exercises more regularly. Patients are motivated through a feedback system wherein they are notified visually and audibly of the movement of their assigned exercises as given to them by their physiotherapist.

Note:
The Training Belt does not fall within the Medical Device Directive, since it is neither being used for a diagnostic or therapeutic use. Instead it is used as a motivational tool for already prescribed exercises.

Note: Dette skema udfyldes, når formål med en vurdering ikke er afprøvning af sikkerhed og ydeevne, men når der ønskes en vurdering af brugersegmenter, oftest i de tidlige design faser, inden yderligere design og udviklingsfremdrift. Dette omhandler artikler uden elektronik og funktionalitet, fabrikerede prototyper og allerede markedsførte produkter.

Det er en forudsætning for vurdering af brugersegmenter, at de involverede produkter/teknologier og materialer opfylder tænkte sikkerhedsaspekter, herunder biologisk evaluering af indholdsstoffer og materialer, der indgår samt at deltagere er anonymiseret frivillige og udøvere har faglig indsigt.

Udfyldt af:

Mikkel Leth Olsen / DELTA

Dato:

19. november, 2012

Partnere i denne undersøgelse

Center for Sundhedsinnovation	*Projektansvarlig*
DELTA, Mobile Fitness,	
WelfareSolutions,	
Helsingør Hospital, rygafdelingen	*Patient kontakt og sundhedsfaglig tilgang*

Den ansvarlige part ifm. undersøgelserne:

DELTA, Center for Sundhedsinnovation

Periode for undersøgelsen:

November 2012 – December 2012

Side 1 af 9

Forhåndsvurdering

- ***Back Training Belt 1***

Beskrivelse og Formål

Hvorfor denne session/undersøgelse? (Projekt, formål, problem område, hvad afsøges?)

The purpose of the Training Belt is to motivate the patients to do their prescribed exercises on a regular basis. Patients with back problems who are currently seeing a physiotherapist are the intended focus group of this study. The purpose of this study is to explore how training with an interactive training belt can motivate the patients in their training routine. Furthermore, we aim to explore how the patient feels about using the Training Belt in their home, and conducting exercises in their homes on a regular basis. Our primary areas of focus will be:

- Motivation - How does the Training Belt motivate patients with their exercises?
- Feedback - How could audio and visual cues help the patient to do their exercises?
- Design - How does the Training Belt function as a wearable addition to exercises with regards to size, material and adjustments?
- Future implementations: Would it be interesting for patients to see their exercises on a graph or trended in some visual way?

Målgruppe

Hvem er målgruppen? Hvilke persongrupper/hvilket segment? Hvordan er/skal deltagerne rekrutteres.

The target group will consist of five people who have been selected by their physiotherapist at Helsingør hospital for the purposes of this project. These five participants are between the ages of 20 and 60 years old and all have existing back problems for which they currently see a physiotherapist. Their physiotherapists have prescribed them specific exercises tailored to their specific needs, which are the exercises we are interested in studying for the purposes of this project.

Personer, der skal indgå i undersøgelsen

Kritisk masse, køn og alder angives.

There will be 5 patients involved in the study, of all genders, and all with existing back problems who are all currently seeing a physiotherapist.

For the specification and selection of users see the document "Undersøgelsesdesign"

Statistisk materiale og reproducerbarhed

Vurdering og begrundelse.

No data will be gathered for statistical use in this research. The purpose of this research is to conduct early usability testing focusing on use, shape and design and will include a small sample of users and physiotherapists.

Lokation og Varighed

Hvor og over hvor lang tid varer forløbet?

This study will take place during one week.

Each individual will conduct the study in their own home - they will take the Training Belt home, and do their prescribed exercises as per usual, whilst wearing the Training Belt. The patients will wear the belt during the time they do their exercises, typically 2 X 30 minutes per day.

Behandling af resultater

Hvordan orienteres deltagerne om sessionen? Hvad anvendes de resultater vi får (som vil være for "intangible til egentlig at kaldes "måledata", men nærmere er "indikationer") til?

The patient's physiotherapist, when selecting them for the study, will introduce the study to them. After that point, we will meet the patients, inform them of what our intended purpose of the study is, and what we would like them to do during the week. We will also give them our contact information and go through all documentation with them at this time. Furthermore, we will demonstrate how to use the Training Belt and ensure they are comfortable using it. During the week we will contact the patients over the phone to help them if they have any inquiries or problems.

We will also do a follow up interview with each patient to ask them about their experience and to find out how we could improve the Training Belt.

Udstyr

Hvad anvendes? (Indholdsstoffer, elektronik-dele, materialer – hvad vises og hvad kommer deltagerne evt. i berøring med?)

Back training belt

During the test the patients will be using a Training Belt. The belt will be worn outside of the patients' clothes. The belt is made from soft, stretchy, black and white fabric, and has been constructed to fit the body without interfering with the prescribed exercises.

Electronics

Inside the Training Belt is an electronic circuit board. The circuit board consists of electronic components such as an accelerometer, gyroscope, and magnetometer, to measure the movement of the lower back. Each component on the circuit board has been connected according to the specification of the individual component.

Attached to the circuit board is a CE approved Bluetooth module (model: RN-42 FLY-477) which is already CE / FCC Approved. These are attached via a male-female connector pin set.

The electronics uses a standard 9V battery of the brand Energizer.

On the battery connector is located a switch that turns the power on /off. Before and after using the belt, the user has to turn on and off the electronics using this switch. The users will be instructed about this during the instructions.

Feedback

The feedback to the patient will be given on a Samsung Galaxy Tab 2 Android tablet. The software giving the feedback has been tested and evaluated together with physiotherapists, physiotherapist teachers, back patients and doctors.

After turning on the belt, the user has to start the app on the tablet and select the correct exercise. They will be instructed about this during the initial instructions.

Risikovurdering

Hvilke risici er der i forbindelse med denne afprøvning - og hvordan stiller vi os i forhold til disse risici?

This risk analysis has been created on the basis of consultation from Jacob Steensen, an expert from DELTA.

Electronics and circuit board

An evaluation has been conducted on the basis of the constructed circuit board that is integrated into the training belt.
Given the electronics used, (circuit board, insulated wires, and integration in belt) it has been assessed that the risk of short circuit or outer influence on the product will have influence on the products safety or performance.

After discussions with DELTA's testing unit experts, it has been estimated that influence could include:

- A short circuit in the system (a result of electrical malfunction or addition of water to the system). At worst, the battery could become warm (but not hot) and would not cause the user any discomfort or danger.
- Since the electronics are housed inside of the fabric belt and worn outside of the user's clothing, there are two layers of fabric between the user and the device, shielding the device from sweat, which could be a source of water, or shielding the user from the battery or device, which could warm up slightly if a short-circuit occurs.

Battery

The electronics uses a standard 9V battery of the brand Energizer, connected to the circuit board with a standard connecter. Should the battery run out of power during the test phase, the users can easily replace the battery with a new one by themselves.
As explained above, precautions have been taken to prevent any foreseeable risks with regards to the battery being short circuited

Sound Level

We do not foresee a risk with regard to sound levels. Sound levels are determined by the patient, depending on how loud they set the volume on the tablet. The tablet has a maximum volume range as defined by the industry standard stereo speakers.

Software

There is a risk that the software disconnects from the circuit board, and thus not reacting to the patients' movement while they are doing their exercises. In other words, the patient will know from the alert that the software is no longer connected to the device, and furthermore, will not change their exercise routine in any way because of the software. The exercises will be the same; only the patient's ambition to do their exercises in the correct range of motion will be affected due to a momentary loss of connection until they reconnect the system.
There is also the risk that an unpredicted outside influence outer influence to the circuit board might send error prone data, resulting in the software not reacting to the belt's movement. The software has been constructed to alert the user if the connection is lost, and the patient will be told only to do the exercises in the way they have been instructed by the physiotherapists.

Fabric

The belt will be worn outside of clothes, and therefore it has been assessed that the risk of an allergic reaction due to contact with the fabric is at a minimum.

Belt

Where the electronics is integrated into the belt, there is a small, hard bump. This could be painful to put one's body weight on, and so we will instruct patients to wear the belt with the electronics facing upwards, so there is no risk of them laying on the electronics. Therefore the patients will be instructed to wear the belt with the electronics facing upwards, so they don't lay directly on the electronics.

Reliability
The Bluetooth module is CE/FCC approved and the circuit board is EMC tested. All components are being used in accordance to their specifications. Furthermore the circuit board and the battery are encapsulated in the belt.

The belt will only be used indoors, and has been assessed to be fully functional within normal indoor temperature range on 0°C - 40°C. The patients will be instructed not to get the belt wet, and to wear it outside of their clothes.

The product has not been fall tested, because it is expected to be used in a proper and careful manner. Should the product be dropped, or in another way be influenced by hits or hard handling, it has a risk of not functioning as expected, but will not pose a risk to the patient's health as outlined in the above risk analysis.

Execution
The study will be conducted so that the patients will be doing the exact same exercises that they otherwise would be doing, and in accordance to the instructions from their physiotherapist. The patients will be instructed on how to wear the belt correctly, and in a way that the belt doesn't interfere with the exercises. Furthermore, should the patients during the test have any questions about the exercises or the use of the belt, they will be provided with contact information for their physiotherapist and the person in charge of the research. Furthermore all the patients will be contacted by phone on the second day of using the belt, to see if they have any questions or issues.

Dermed anses risici i sikkerhed og etiske sammenhænge i denne undersøgelse for at være minimeret til acceptabelt niveau, når der tages hensyn til forpligtigelser indenfor området omhandlende medicinsk udstyr.

Data-opsamlings plan

Hvordan opsamles og bevares data/resultaterne fra forsøget?

After the study, we will examine the results of the study and find out where we can make improvements to the Training Belt. Furthermore, we will begin to think about the next phase of the project, and determine if it is worth pursuing in terms of helping patients to do exercises correctly.

When talking to patients, we will be using anonymised data and hypothetical situations to discuss potential usage situations with them.

All information will be anonymised throughout the study.

Forhåndsvurdering

- ***Back Training Belt 1***

Projekt Lev Vel - FS
Træning, Rygtræning
Projektnr.: FU20051

Denne side udfyldes FØR undersøgelsesforløbet igangsættes:

Udfyldelse og kvalitetssikring:

Udfyldt af (underskrift og dato):	
Morten Wagner *Afdelingsleder, IdemoLab* DELTA	Mikkel Leth Olsen, *Electronic Sketching Specialist, IdemoLab* DELTA

Kvalitetssikring og godkendelse:

Underskrives af:	Dato for godkendelse:
Underskrives, inden undersøgelsen igangsættes. Pernille Veje (Kvalitetschef, DELTA)	

Side 7 af 9

Forhåndsvurdering

- ***Back Training Belt 1***

Projekt Lev Vel - FS
Træning, Rygtræning
Projektnr.: FU20051

Denne side udfyldes EFTER forløbet af undersøgelsen:

Oplistning af indsamlede data og datamængder (beskrivelser, fotos etc.)
Konklusion Udfyldes, så snart undersøgelsesperioden er færdig. Opsummerer konklusion på de samlede undersøgelsesdesign, tilknyttet denne forhåndsvurdering.
Arkivering Originaldokument arkiveres hos DELTA. Kopi af dette dokument arkiveres hos projektledelsen af UNIK-KOL. Dette dokument følges op af og arkiveres sammen med samtykkeerklæring og undersøgelsesdesignet.

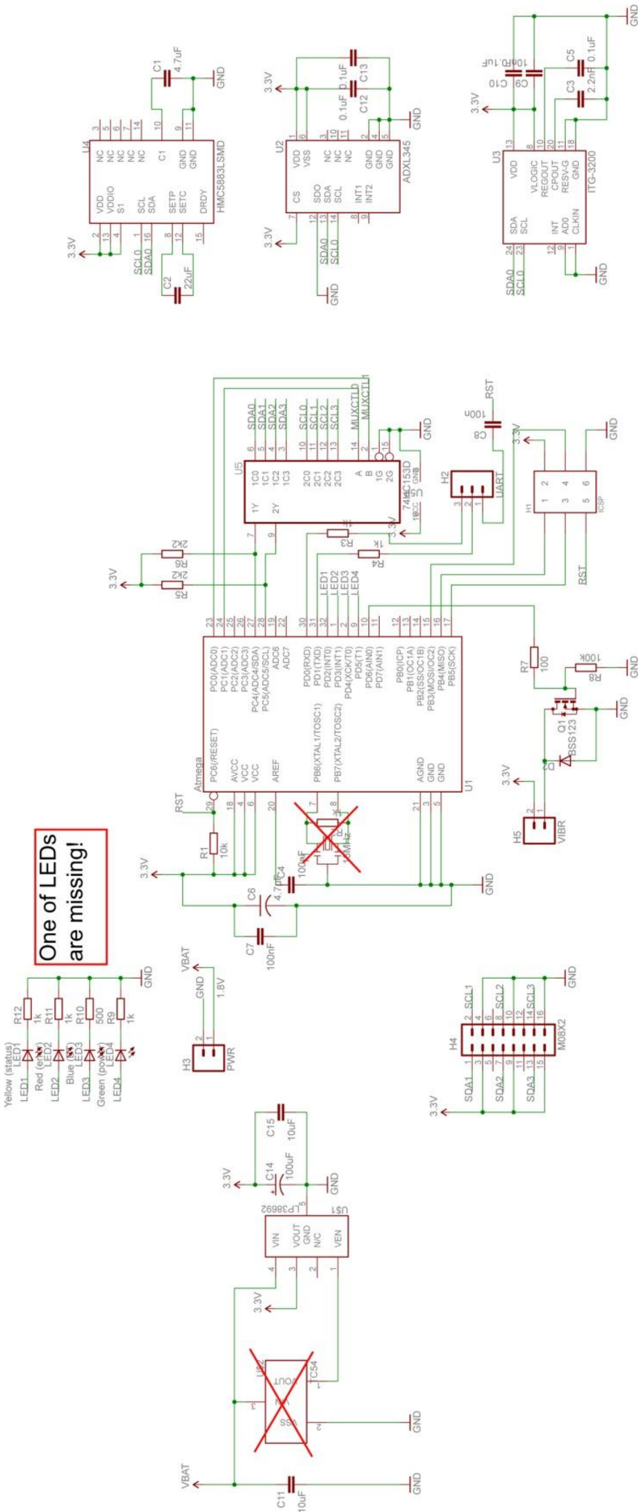
Udfyldelse og kvalitetssikring:

Udfyldt af (underskrift og dato):	
Morten Wagner *Afdelingsleder, IdemoLab* DELTA	Mikkel Leth Olsen, *Electronic Sketching Specialist, IdemoLab* DELTA

Side 8 af 9

Appendix B

Hardware schematic for sensor



Appendix C
Processing program for sensor

```
#include <twiMaster.h>
#include <Servo.h>
#include <I2cMaster.h>

#define ADXL345ADDR 0x53
#define ADXL345ADDR_READ 0xA7
#define ADXL345ADDR_WRITE 0xA6

// select software or hardware i2c (0) = hardware, (1) = software
#define USE_SOFT_I2C 0

#if USE_SOFT_I2C

#define SDA_PIN 18
#define SCL_PIN 19

// An instance of class for software master
SoftI2cMaster acc(SDA_PIN, SCL_PIN);

#else

// Instance of class for hardware master with pullups enabled
TwiMaster acc(true);

#endif

int yellowLED = 2;
int blueLED = 4;
int greenLED = 5;

boolean ledState = false;
unsigned long timer = 0;
int blinkTime = 750;

//------------------------------------------------------------------------------
/*
 * Read 'count' bytes from the ADXL345 starting at 'address'
 */

uint8_t readADXL345(uint8_t address, uint8_t *buf, uint8_t count) {
  // issue a start condition, send device address and write direction bit
  if (!acc.start(ADXL345ADDR_WRITE)) return false;

  // send the ADXL345 address
  if (!acc.write(address)) return false;

  // issue a repeated start condition, send device address and read direction
bit
  if (!acc.restart(ADXL345ADDR_READ)) return false;
```

```
  // read data from the ADXL345
  for (uint8_t i = 0; i < count; i++) {

    // send Ack until last byte then send Ack
    buf[i] = acc.read(i == (count-1));
  }

  // issue a stop condition
  acc.stop();
  return true;
}

//------------------------------------------------------------------------------
/*
 * write 'count' bytes to ADXL345 starting at 'address'
 */

uint8_t writeADXL345(uint8_t address, uint8_t *buf, uint8_t count) {
  // issue a start condition, send device address and write direction bit
  if (!acc.start(ADXL345ADDR_WRITE)) return false;

  // send the ADXL345 address
  if (!acc.write(address)) return false;

  // send data to the ADXL345
  for (uint8_t i = 0; i < count; i++) {
    if (!acc.write(buf[i])) return false;
  }

  // issue a stop condition
  acc.stop();
  return true;
}

//------------------------------------------------------------------------------
/*
 * simple "write" function that takes an address and a value to write
 */
uint8_t initADXL345(uint8_t address, uint8_t val) {
  // issue a start condition, send device address and write direction bit
  if (!acc.start(ADXL345ADDR_WRITE)) return false;
  // send the ADXL345 address
  if (!acc.write(address)) return false;

  // send data to the ADXL345
  if (!acc.write(val)) return false;

  // issue a stop condition
  acc.stop();
  return true;
}

//------------------------------------------------------------------------------

void setup()
{
  pinMode(yellowLED, OUTPUT);  //Yellow LED
  pinMode(blueLED, OUTPUT);    //Blue LED (very vague)
  pinMode(greenLED, OUTPUT);   //Green LED
```

```
  //pinMode(A0, OUTPUT);   //Multiplex input chooser
  //pinMode(A1, OUTPUT);   //Multiplex input chooser
  //digitalWrite(A0, LOW);
  //digitalWrite(A1, LOW);

  Serial.begin(9600);
  //Serial.println("Hello...");

  initADXL345(0x2d, 8);    //Set powermode with POWER_CTL byte
  initADXL345(0x2c, 0xF);  //Set bandwith rate
}

//---------------------------------------------------------------------------------

void loop()
{
  uint8_t r[6];
  readADXL345(0x32, r, 6);

  if(millis() - timer > blinkTime) {
    ledState = !ledState;
    timer = millis();
  }

  digitalWrite(yellowLED, ledState);

  int x = (r[1] << 8) + r[0];
  int y = (r[3] << 8) + r[2];
  int z = (r[5] << 8) + r[4];

  Serial.print(x);
  Serial.print(",");
  Serial.print(y);
  Serial.print(",");
  Serial.print(z);
  Serial.println();
}
```

Appendix D
Java code for the IdemoBelt Android application

The Java code and installer (*apk*) for the IdemoBelt mobile application is available on the attached CD, as well as online at DropBox via the following link: http://dl.dropbox.com/u/6052849/IdemoBelt%20Android%20app.rar.

The source code can be found in the *src* folder of the archive; the XML resources are contained in the *res* folder of the archive; the *apk* installer appears in the root of the archive.

Bibliography

[1] *LEVVEL*. Alexandra Institutet & Væksthus Hovedstadsregionen, n.d. Web. 21 Sept. 2012.

[2] T. L. Larsen. "Lev Vel." *Region Hovedstaden*. Region Hovedstaden, n.d. Web. 21 Sep. 2012.

[3] Kühne, Eva. *UNIK – Project KOL*. DELTA, n.d. Web. 5 Jan. 2013.

[4] A. Pertola. *Personalized mobile application for monitoring the heart rate and blood oxygen content of chronic obstructive pulmonary disease patients*. Department of Informatics and Mathematical Modeling, Technical University of Denmark. Nov. 2011.

[5] M. Davidsen, M. B. Koch and K. Juel. *De samfundsmæssige omkostninger ved rygsygdomme og rygsmerter i Danmark*. Statens Institut for Folkesundhed, Syddansk Universitet. May 2011.

[6] M. Mackie and K. E. Nielsen. *Statistisk Tiårsoversigt*, 52nd ed. Copenhagen, Denmark: Danmarks Statistik, 2011.

[7] A. L. Schmidt. "Hospitaler skal spare en milliard kroner". *Politiken*, 21. Jan. 2010. Web. 7. Jan. 2013.

[8] No author. "Hospitaler er tvunget til besparelser". *Politiken*, 17. Sep. 2009. Web. 7. Jan. 2013.

[9] N. Ashworth, K. Chad, E. Harrison, B. Reeder and S. Marshall. *Home versus center based physical activity programs in older adults*. Cochrane database of systematic reviews. 2005.

[10] C. Coley, Y. Li, A. Medsger, T. Marrie, M. Fine, W. Kapoor, J. Lave, A. Detsky, M. Weinstein and D. Singer. *Preferences for home vs. Hospital care among low-risk patients with community acquired pneumonia*. Archives of Internal Medicine, 1996. 156(14): p. 1565.

[11] J. M. Eklund, J. Sprinkle, S. Sastry and T.R. Hansen. *Information Technology for Assisted Living at Home: building a wireless infrastructure for assisted living*. In proc. of the 27th Annual International Conference of Engineering in Medicine and Biology Society, pp. 3931-3934. 2005.

[12] T. R. Hansen, J. M. Eklund, J. Sprinkle, R. Bajcsy and S. Sastry. *Using smart sensors and a camera phone to detect and verify the fall of elderly persons*. In proc. of the European Medicine, Biology and Engineering Conference. 2005.

[13] R. Aarhus, E. Grönvall, S. B. Larsen and S. Wollsen. *Turning training into play: Embodied gaming, seniors, physical training and motivation*. In Gerontechnology, vol. 10, no. 2, pp. 110-120. 2011.

[14] West Health Institute (2012). "West Health Institute Unveils New Kinect-based Physical Therapy Technology, Launches Research Studies with Naval Medical Center San Diego" [Press release]. Retrieved from: http://www.westhealth.org/news/press-release-west-health-institute-unveils-new-kinect-based-physical-therapy-technology.

[15] A .Stein. "Harnessing the black hole of health device data". *Smart Planet*. N.p., 27. Sep. 2012. Web. 7. Jan. 2013.

[16] V. M. Jones, R. A. G. Bults, D. Konstantas and P. A. M. Vierhout. *Healthcare PANs: Personal Area Networks for trauma care and home care*. In proc. of the Fourth International Symposium on Wireless Personal Multimedia Communications (WPMC), Aalborg, Denmark. 2001.

[17] V. Jones, A. van Halteren, N. Dokovsky, G. Koprinkov, J. Peuscher, R. Bults, D. Konstantas, I. Widya and R. Herzog. *MobiHealth: Mobile Services for Health Professionals*. In M-Health: Emerging Mobile Health Systems, pp. 237-46.

[18] V. M. Jones, R. Huis in't Veld, T. Tonis, R. B. Bults, B. van Beijnum, I. Widya, M. Vollenbroek-Hutten and H. Hermens. *Biosignal and Context Monitoring: Distributed Multimedia Applications of Body Area Networks in Healthcare*. In proc.

of the IEEE 10th Workshop on Multimedia Signal Processing, pp. 820-825, Cairns, Australia. 2008.

[19] P. Leijdekkers and V. Gay. *A Self-Test to Detect a Heart Attack Using a Mobile Phone and Wearable Sensors*. In proc. of the 21st IEEE International Symposium on Computer-Based Medical Systems, pp. 93-98. 2008.

[20] S. Fokkenrood, P. Leijdekkers and V. Gay. *Ventricular Tachycardia/Fibrillation Detection Algorithm for 24/7 Personal Heart Monitoring*. In proc. of ICOST on Pervasive Computing Perspectives for Quality of Life Enhancement, Nara, Japan. 2007.

[21] V. Gay and P. Leijdekkers. *A Health Monitoring System Using Smart Phones and Wearable Sensors*. In special issue on Smart Sensors in Smart Homes in International Journal of Assistive Robotics and Mechatronics (IJARM), vol. 8, no. 2. 2007.

[22] D. Boiroux, D. A. Finan, J. B. Jørgensen, N. K. Poulsen and H. Madsen. *ARX MPC for people with type 1 diabetes*. Technical University of Denmark. 2010.

[23] G. Bombardieri. “Device for the automatic insulin or glucose infusion in diabetic subjects, based on the continuous monitoring of the patient’s glucose, obtained without blood withdrawal”. U.S. Patent 4,633,878. 6. Jan. 1987.

[24] M. L. Blomquist. “Programmable insulin pump”. U.S. Patent 6,852,104. 8. Feb. 2005.

[25] I. S. Ross and L. H. Loeser. *Electroencephalographic findings in essential hypoglycemia*. In Electroencephalogr Clin Neurophysiol., vol. 3, no. 2, pp. 141-148. 1951.

[26] R. Elsborg, L. S. Remvig, H. Beck-Nielsen and C. B. Juhl. *Using the Brain as a Biosensor to Detect Hypoglycaemia*. Hypo-Safe A/S, Odense University Hospital, Sydvestjysk Sygehus Esbjerg, Denmark.

[27] C. B. Juhl, K. Højlund, R. Elsborg, M. K. Poulsen, P. E. Selmar, J. J. Holst, C. Christiansen and H. Beck-Nielsen. *Automated detection of hypoglycemia-induced EEG changes recorded by subcutaneous electrodes in subjects with type 1 diabetes*

– *the brain as a biosensor*. Medical Department, Clinic for Endocrinology and Diabetes, Sydvestjysk Sygehus Esbjerg, Denmark. 2010.

[28] L. S. Snogdal, L. Folkestad, R. Elsborg, L. S. Remvig, H. Beck-Nielsen, B. Thorsteinsson, P. Jennum, M. Gjerstad and C. B. Juhl. *Detection of hypoglycemia associated EEG changes during sleep in type 1 diabetes mellitus*. Department of Endocrinology, Odense University Hospital, Denmark. 2012.

[29] G. Hartvigssen, E. Årsand, O. K. Hejlesen and T. Nesbitt. *Could mobile diabetes management systems revolutionize patient care?* In Diabetes Manage, vol. 2, no. 3, pp. 181-183. 2012.

[30] I. Husain. “The Phone Oximeter peripheral measures pulse ox and respiratory rate, connects to iPhone”. *iMedicalApps*, N.d. Web. 14. Jan. 2013.

[31] R. G. Haahr, S. B. Duun, M. H. Toft, B. Belhage, J. Larsen, K. Birkelund and E. V. Thomsen. *An Electronic Patch for Wearable Health Monitoring by Reflectance Pulse Oximetry*. In IEEE Transactions on Biomedical Circuits and Systems. 2011.

[32] No author. *A Revolution in Health Monitoring – the ePatch platform brings intelligence to the world of health monitoring*. DELTA, Hørsholm, Denmark, N.d.

[33] Council Directive (EEC) 93/42/EEC of 14 June 1993 on Medical Devices.

[34] *ANSI/AAMI/IEC 62304:2006: Medical device software – Software life cycle processes* (2006). Arlington: Association for the Advancement of Medical Instrumentation.

[35] *ANSI/AAMI/IEC 62366:2007: Medical devices – Application of usability engineering to medical devices* (2007). Arlington, VA: Association for the Advancement of Medical Instrumentation.

[36] *CDRH 1497: Medical Device Use-Safety: Incorporating Human Factors Engineering into Risk Management* (2000). Silver Spring, MD: U.S. Department of Health and Human Services, Food and Drug Administration, Center for Devices and Radiological Health.

[37] B. J. Fogg. *A behavior model for persuasive design*. In proc. of the 4th International Conference on Persuasive Technology, no. 40, New York, NY. 2009.

[38] S. G. Striegler. “Gå-Hjem-Møde Om Fremtidens Rygtræning”. PowerPoint presentation. La Oficina, Frederiksberg, Denmark. 10. Dec. 2012.

[39] J. McGonigal. ”Gaming can make a better world”. Lecture, TED 2010, Feb. 2010.

[40] B. J. Fogg, Ph.D. "BJ Fogg's Behavior Model." *BJ Fogg's Behavior Model*. Persuasive Technology Lab, Stanford University, n.d. Web. 16 Dec. 2012.

[41] P. Cipresso, S. Serino, D. Villani, C. Repetto, L. Sellitti, G. Albani, A. Mauro, A. Gaggioli and G. Riva. *Is your phone so smart to affect your state? An exploratory study based on psychophysiological measures*. In Neurocomputing, vol. 84, pp. 23-30. 2012.