Designing a BCSS for mobile devices: an application to help the chronically tardy
Abstract

The purpose of this thesis is to study the process of developing a behaviour change support system (BCSS) which help users to manage and utilize their time. Proper time management is an important asset for a successful life. However, most of us have difficulties in using our time efficiently due to various reasons. Some of the reasons include procrastination and becoming overoptimistic about completing tasks in short periods of time. On this study, an iOS based application is developed based on BCSS theory. The application can be used by anyone who wants to improve their efficiency in time management. The intention of the system is to help users identify and improve their weaknesses on time management through using the application for a longer period. BCSS is one area of persuasive technology research, and it is defined as information systems designed to form, alter or reinforce attitudes, behaviours or an act of complying without using deception, coercion or inducements. BCSS are applied in various sectors such as health, wellness and environment. It is also important to note that Persuasive System Design model (PSD) is used as the main construct to develop our system. PSD provides a recent and technology oriented persuasion framework that helps to design, develop and evaluate BCSS. This project is based on a Design Science research method through developing an artefact. In the end, the system is evaluated qualitatively.

Keywords
Behaviour change support system, Persuasive system, Persuasive technology, punctuality

Supervisor
Profesor, Harri Oinas-Kukkonen
Foreword

I would like to thank, Professor Harri Oinas Kukkonen for giving me the opportunity and guidance to research work. I would like to express my utmost gratitude Piiastiina Tikka (Ph.D.) for continuous support and guidance while conducting this study.

I am grateful to the University of Oulu and Department of Information Processing Science for providing me an opportunity to study my master degree.

Bereket Kahsay Woldemicael

Oulu, November 15, 2015
## Contents

Abstract ................................................................................................................................. 2  
Foreword ................................................................................................................................. 3  
Contents ................................................................................................................................. 4  
1. Introduction ....................................................................................................................... 5  
2. Background ....................................................................................................................... 7  
   2.1 Behaviour change support systems .............................................................................. 7  
   2.2 Persuasive Systems Design ......................................................................................... 9  
   2.2.1 Persuasive system postulates ................................................................................. 9  
   2.2.2 Persuasive system context ....................................................................................... 10  
   2.2.3 Design of System Features ..................................................................................... 11  
   2.3 Mobile based timekeeping application ...................................................................... 12  
3. Research Method .............................................................................................................. 14  
   3.1 Design Science in information system ........................................................................ 14  
   3.2 Design Science Research Methodology (DSRM) ....................................................... 15  
   3.3 Implemented research methodology ........................................................................ 17  
4. Software Artefact .............................................................................................................. 19  
   4.1 Persuasion context ....................................................................................................... 19  
   4.1.1 The Intent ............................................................................................................... 19  
   4.1.2 The Event ............................................................................................................... 19  
   4.1.3 Strategy ................................................................................................................. 20  
   4.2 Software features ....................................................................................................... 20  
   4.3 Software development process ................................................................................... 23  
   4.3.1 Software Requirement ............................................................................................ 23  
   4.3.2 Software architecture .............................................................................................. 24  
   4.3.3 Software implementation ........................................................................................ 25  
   4.4 Artefact demonstration ............................................................................................... 29  
   4.4.1 Activity .................................................................................................................. 29  
   4.4.2 Practice .................................................................................................................. 32  
   4.4.3 Tips and Hints ........................................................................................................ 37  
5 Evaluation ......................................................................................................................... 38  
   5.1 Objective of Evaluation ............................................................................................. 38  
   5.2 Evaluation setting ....................................................................................................... 38  
   5.3 Evaluation of artefact ................................................................................................. 39  
   5.3.1 GQM Evaluation .................................................................................................... 39  
   5.3.2 Desirability Test ...................................................................................................... 44  
   5.4 Summary .................................................................................................................... 44  
6 Discussion ............................................................................................................................ 46  
7 Conclusion .......................................................................................................................... 48  
References ............................................................................................................................. 50
1. Introduction

Nowadays smartphones, the web and other related technologies provide new ways to create, access and share information. As a result, numerous applications are developed to enhance our day to day activities. These technologies have the power to influence the way we live our lives. Thus, they are not neutral (Oinas-Kukkonen & Harjumaa, 2009). Behavioural change support systems (BCSSs) are one of the fruits of these modern technologies. BCSS are intentionally developed to form, alter, or reinforce attitudes, behaviours, or acts of complying avoiding deception or coercion (Oinas-kukkonen 2013).

BCSSs can be very useful to prevent future medical problems. As a result, it could also have an impact on economic and public healthcare systems (Oinas-Kukkonen & Harjumaa, 2009). Modern web and mobile technologies create a wide range of opportunities for developing applications that are a target to bring behavioural changes. For example, some healthcare applications are intended to help people to adapt a healthier lifestyle. Researchers have already found encouraging results in areas such as health, wellness and increased environmental awareness (Segerståhl et al., 2010; Oinas-kukkonen, 2013). For example, BCSS which promote weight loss (Lehto & Oinas-Kukkonen, 2015) and helping to avoid sleep disorder (Langrial et al., 2012), an application to create awareness in green energy (Midden & Ham, 2009).

Persuasion is vital for BCSSs (Lehto & Oinas-Kukkonen, 2015). However, persuasion is a challenging process; it requires various techniques and strategies to ensure its success (Segerståhl et al., 2010). Oinas-Kukkonen & Harjumaa, (2009), introduce a model for development of persuasive system called Persuasive Systems Design (PSD). The model contains postulates and strategies which guide in developing a persuasive system. The purpose of this thesis is to study the process of design and development of a BCSSs, through developing a mobile based application which intends to help people adapt/acquire a better time management skill. The system is develop based on mainly PSD model, O/C matrix and other related theoretical constructs.

Efficiently managing one's time is important to everyone in achieving their goals in life. However, most of us have difficulties in managing our time properly due to various reasons. Some of the reasons include procrastination, becoming overoptimistic of completing tasks in short periods of time and being easily distracted (or losing focus). Proper time usage does not require special skill. Anyone can develop punctuality skill, provided proper measures are taken.( DeLonzor, 2003)

This thesis work utilizes a design science research method (DSR). DSR involves an iterative process of design and evaluation to develop a functional artefact to solve a research problem. There are various types of frameworks that can be used while implementing a design science research. One of such frameworks is the design science research methodology framework that illustrated by Hevner et al., (2004), and Peffers et al., (2007) is applied in this case.

The contribution of the thesis work is a mobile-based artefact that is developed based on the concept of BCSSs and also the documentation of the application. The resulting artefact is published in Apple store and is freely available. The artefact can be utilized
by anyone, with an iPhone or iPad, who wants to improve their time management skills. In addition, this thesis can be used as a reference for those who want to peruse a study related to behaviour support systems, or intend to develop further/improve the application.

This thesis is structured into seven chapters. The next chapter follows up with the general overview in the introduction and discusses theoretical background of BCSS and the PSD model. The third chapter introduces the research method utilized to conduct the research. The artefact’s design and development process are illustrated in the following, fourth, chapter. Chapter five discusses evaluation and user's reflection about the artefact while the sixth chapter covers the findings of the research. The seventh chapter closes up by describing the conclusions and limitations of the thesis work.
2. Background

In this chapter, the theoretical backboard of study is discussed. The chapter presents behaviour change support system, PSD model and O/C matrix that are the main construct to develop BCSSs, and lastly the mobile-based time keeping application is introduced.

2.1 Behaviour change support systems

Information technology creates a platform for the innovation of important technological tools for our day-to-day activities. And these tools are not neutral because they cause behavioural change amongst users (Oinas-kukkonen, 2013). There is an incidental, even accidental behavioural or attitudinal changes occur to us while using those tools (Berdichevsky & Neuenschwander, 1999). BCSSs are one of the outcomes of information technology, and they are intentionally designed to help people to achieve a certain behavioural, attitudinal or both changes. BCSSs might resemble a scenario on how teachers' personal behaviour could contribute to behavioural or attitudinal changes of their students. Oinas-kukkonen (2013), define BCSS as follows: “A BCSS is a socio-technical information system with psychological and behavioural outcomes designed to form, alter or reinforce attitudes, behaviours or an act of complying without using coercion or deception”. (p. 1225)

The interest of BCSSs includes methodologies, approaches, usage of technologies and possible outcomes. One of the most prominent sectors of BCSS is health. Various applications are developed to help people to adopt a healthier lifestyle. For example Nike+ website has implemented various techniques to encourage users for more exercising (Oinas-Kukkonen & Harjumaa, 2009), AppDextro an Android mobile application that helps to avoid procrastination by letting users control their usage on their mobile applications (Ansaldo, 2014).

BCSSs are one research part of persuasive technology (Oinas-kukkonen 2010, 2013). Fogg 2003 define a Persuasive technology as an interactive technology that is intentionally designed to change peoples' behaviour or attitude without coercion or deception. Persuasive technologies cover disciplines such as social psychology, communication and persuasion, computer science and information systems (Lehto & Oinas-kukkonen, 2015). Persuasive technologies are applied in online commerce (Kaptein and Eckles 2012), user training (Forget et al. 2008) and health sectors (Chatterjee and Price 2009; Lehto, & Oinas-Kukkonen, 2015).

Previously conducted researches on behavioural theories provide multiple approaches for conducting a successful persuasion (Consolvo et al., 2009). For example, Goal-Setting Theory (Locke & Latham, 2002) which describes how people respond to different types of goals that tend to encourage them. Transtheoretical Model (Prochaska et al., 1992) which describes the stages of human behaviour. Transtheoretical Model includes Precontemplation: no intention to change in future, Contemplation: seriously considering changing but not yet taking action, Preparation: ready to take action, Action: achieved the desired behaviour, Maintenance: achieved the desired behaviour consistently. Cognitive consistency theory (Fraser et al., 2001) which states that people
like their views about the world to be organized and consistent, and they likely to follow on the action that could help them to achieve their mental structure in this way. Information processing theory (McGuire, 1973) provides six steps for conducting persuasion, which are information presentation, attention, and comprehension of the arguments, yielding to the position presented, retention and action in compliance with the new position. The elaboration likelihood model (Petty & Briñol, 2011) which elaborates the relation between persuasion and attitude. In computer science fields, the technology acceptance model (Davis, 1989) also provides a model that states how user accept and use technology. Fogg 2003 & 2009 also provides various models including Fogg Behaviour Model (FBM) which states factors that affect behavioural change. While these behavioural sciences are important in developing a persuasive technology, the design components are ignored. The Persuasive System Design model presented on (Oinas-Kukkonen & Harjumaa, 2009) provides a recent and technology oriented persuasion framework that help to design persuasive systems and to list software requirements (Torniing & Oinas-Kukkonen, 2009).

Oinas-kukkonen (2013) introduces outcome/change (O/C) matrix which elaborates the behavioural changes that can be achieved with BCSSs. In addition, the matrix describes the outputs that reflected in end-user due to the respective changes. In O/C matrix, the changes categorized into three divisions which are change in act of complying (C-change), change on behaviour (B - Change) and change on attitude (A-change). The C-change simply tries to influence the user to comply with the system. For example, if a user has to measure blood pressure continuously in a certain span of time but doesn’t have proper motivation to perform this task. The C-change system tries to motivate the user to perform measurement task in appropriate time. The B-changes are more enduring behavioural changes. For instance, in the previous example if system able to change user’s behaviour, so that user have the motivation to perform measurement task without the system intervention illustrates B-change. The A-change systems help the user to make attitudinal changes. For example, system helping user successfully avoid addiction behaviour.

The outcome of the easier changes complements to the outcome of the harder changes. The outcome of C-change could latter result to bringing B-change, and the outcome of the B-change system could result in A-Change.

In Table 1, illustrates outcome/change matrix. From O/C matrix, the stakeholders who want to develop BCSS a can choose appropriate O/C to their system.

Table 1: Outcome/change design matrix (quoted from Oinas-kukkonen, 2013)

<table>
<thead>
<tr>
<th></th>
<th>C-Change</th>
<th>B-Change</th>
<th>A -Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>F-outcome</td>
<td>Forming an act of complying (F/C)</td>
<td>Forming a behaviour (F/B)</td>
<td>Forming an attitude (F/A)</td>
</tr>
<tr>
<td>A-outcome</td>
<td>Altering an act of complying (A/C)</td>
<td>Altering a behaviour (A/B)</td>
<td>Altering an attitude (A/A)</td>
</tr>
<tr>
<td>R- outcome</td>
<td>Reinforcing an act of complying (R/C)</td>
<td>Reinforcing a behaviour (R/B)</td>
<td>Reinforcing an attitude (R/A)</td>
</tr>
</tbody>
</table>
The O/C matrix elaborates outcomes which happen in the end-user due to the CBA-changes. The potential voluntary outcomes are categorised into three which are formation, alteration and reinforcement (FAR). Formation outcome (F-outcome) refers to the formation of a new behaviour or attitude that does not exist before in a user. Alteration outcome (A-outcome) refers increasing user’s response to a certain behaviour. For example, minimize alcohol consumption, increasing exercise. Reinforcement outcome (R-Outcome) refers to strengthening the current respectable behaviour so that it won’t change easily.

The other important issue that needs to be considered in BCSSs is ethics. The BCSSs by definition should be designed avoiding coercion, deception. For example, in certain system dialog, if both “Ok” and “Cancel” buttons perform a similar task, then the system is considered both deceptive and coercive. In addition, BCSS system should avoid monetary inducement. It should simply guide the end user to the desired behaviour (Oinas-kukkonen, 2013).

To summarize the BCSSs are transformative which deliberately designed to make an influence on peoples' cognition or emotion (Oinas-Kukkonen, 2013). It helps to transfer one mind state to the other planned mind state. Developing BCSSs cannot only be interactives of the technology or the user interface but it needs consideration of various things, such as technical, cultural, and personal goals.

2.2 Persuasive Systems Design

Oinas-Kukkonen & Harjumaa, 2009 has developed Persuasive System Design model (PSD) which help for designing and evaluating the BCSSs. The PSD model contains seven postulates, application context and design of system features, the following subsections discuss the PSD model.

2.2.1 Persuasive system postulates

Oinas-Kukkonen & Harjumaa, 2009, define seven postulates which are important ideologies to develop a persuasive system. The seven postulates are:

1. Information technology is never neutral
2. People like their views about the world to be organized and consistent
3. Direct and indirect routes are the key persuasion strategies
4. Persuasion is often incremental
5. Persuasion through persuasive system should be always open
6. Persuasive systems should aim at unobtrusiveness
7. Persuasive systems should aim at being both useful and easy to use

The first postulate states that: Information technology is never neutral. Currently, we use information technology products such as web and mobile in our day to day life. As a result, technology influences our behaviour in one way or another, whether intended or not and whether we are aware of it or not.

The second postulate states that People like their views about the world to be organized and consistent. Users are likely to persuade on the systems, which could help them to commit to their desired goal. For example, a user might have an interest in doing exercise. If the system helps the user to commit to her/his interest, such as sending a text
message and reminding exercising time, then the user more likely to comply with the system. The user might be unable to perform her/his intended goal due to psychological inconsistency. If the system supports the user to realize those inconsistencies, the user might be motivated to bring change in behaviour or attitudes.

The third postulate states that direct and indirect routes are the key persuasion strategies. People who have the knowledge, thoughtful and positive can be approached by a direct route, whereas people who are less thoughtful can be approached by indirect routes such as using cues or stereotypes. For example, if the person has less knowledge about the system, unlikely to be motivated to comply with the system. In contrary if a user has knowledge most likely to be interested and easy to comply with the system.

The fourth postulate states that Persuasion is always incremental. Instead of enforcing the user to do everything at one time, a persuasive system should follow an incremental approach. For example, if the a system, designed to make a behavioural change in adapting healthy eating habit, initially the system should simply encourage the user to include vegetables in his meal, and then gradually increase the amount to the correct portion.

The fifth postulate states that Persuasion through persuasive system should be always open. The mission of the persuasive system should be clear to the user. If the user does not understand the message, it might mislead the user and could fail to achieve the intended behavioural or attitudinal changes.

The sixth postulate states that Persuasive systems should aim at unobtrusiveness. The persuasive systems should avoid disturbing users while doing their main tasks. In addition, it should consider the situation where the user is not able to respond to systems, for example, suggesting to exercise while the user is sick.

The seventh postulate states that Persuasive systems should aim at being both useful and easy to use. Similar to any software systems, the persuasive system should be developed with a good usability.

2.2.2 Persuasive system context

Persuasion context compromises where the persuasion takes place and includes intent, event and strategy.

The Intent in the PSD model includes the persuader and the change type. Persuaders are stakeholders who decide the system components. The change type includes behavioural or attitudinal change. The change type also determines the level on which the persuader puts effort to develop the persuasive system.

The event includes use context, user context and technology context. The use context refers to the problem domain in which the system intend to tackle. On the other hand, the user context refers to an individual. To study a user context, it is required to know user's goals, abilities, motivations, pre-existing attitudes, and other characteristics related to the individual that could help the user to achieve persuasion. The technology context refers to the technology usage. Some conditions of the persuasions could bond to the technology usage (Torning & Oinas-Kukkonen, 2009). For example, there might be the difference in persuasion between a mobile and desktop application.
The strategy includes message and route. The message could be form, content or both which is chosen by persuader to transfer the intended transformation. The message could be transferred through dialog system or course of the game. The persuader selects the type of media which is appropriate for the content. The route for persuasion includes direct, indirect, or both approaches. The direct route is less argumentative, and it can be applied for those who have better knowledge. Indirect approach relays on the number facts and figures which can be applied for those who have less knowledge or interest. Both approaches can also be applied at the same time in a persuasive system.

2.2.3 Design of System Features

Oinas-kukkonen & Harjumaa 2009, provide a system features which a persuasive system should adopt in order to increase its success in persuasion to the desired behaviour. Those features are also known as persuasive elements that are primary task support, computer-human dialog support, system credibility support, and social support. The persuasive elements provide a wide range of features for developing the persuasive system, from which a system designer could choose the most appropriate and feasible ones for the desired system.

Primary task support provides the guidance how to approach the desired behaviour changes. And it contains seven parts which are reduction, tunnelling, tailoring, personalization, self-monitoring, simulation and rehearsal. Reduction states breaking down the complex behaviour into simpler units. Tunnelling means guiding user step by step to intended behaviour change. Tailoring can be implemented by identifying and providing classified information according to user’s need and potential. Personalization states offering content targeting individuals, so that a user feels as if a system is developed for her/himself. Self-Monitoring states providing tool which user monitor her/himself. Simulation states providing means which user able to understand the cause and effect regarding to a certain behaviour. Rehearsal states system could support to rehearse the desired behaviour.

Computer-Human dialog support features are feedbacks, and its purpose is to motivate the user to perform the primary tasks. Computer-Human dialog support includes praise, rewards, reminders, suggestions, similarity, liking and social role. This can be implemented on persuasive system in the following way. Praising and rewarding user while achieving goal, reminding user to perform his/her target task. Appropriate suggestion to increase persuasion. Creating Similarity between user and system, for example system can use user’s nicknames for achieving this feature. Liking can be achieved by creating visually attractive interface. Social role can be implemented by linking system to outside interface so that professionals able to provide to the user feedbacks on the desired behaviour changes.

System credibility support category states making the system in general credible and includes trustworthiness, expertise, surface credibility, real-world feel, authority, third-party endorsement and verifiability. Trustworthiness can be achieved by providing unbiased truthful information. Expertise states that system should reflect its expertise on the subject matter. Real world feel can be achieved by letting a well-known organization to support the system. Authority states system feels more credible if it refers to authorities. Third party endorsements can be achieved by letting third party endorse the system. Verifiability states system could be more credible if verified by outside source.
Social support states system should motivate users by leveraging social influence. Social support category includes social learning, social comparison, normative influence, social facilitation, cooperation, competition and recognition. Social learning describes system should include access where a user able to learn from other users. Social comparison can be achieved by letting compare users among themselves on the desired behaviour change. Normative influence means engaging third person influence, such as peer comment. Social facilitation can be achieved through letting user users to learn from other users’ progress. Cooperation among users is also one motivation factor. Competition can be achieved through creating competition among users. Recognition states giving user recognition by others when achieving the desired behaviour change.

2.3 Mobile based timekeeping application

We all need to manage properly our time to be able to address our conflict demands. However, most of us have difficulty managing their time properly, and some consistently suffer from time management. According to Vozza 2014, study conducted in San Francisco, 20% of America population are chronically late. People who are often late is not because of they need it, or they don’t value others time, but lateness became a habit for them and difficult for them to avoid it. Time management skill is often related to the cultural background. However, there are common problems that most of us share. Someone could improve time management skill if the appropriate measure is taken (DeLonzor, 2003).

Brett & Kate (2012), lists some common reasons that make us often late, such as Misperceive the passage of time, underestimate on how things take time, procrastination, being easily distracted, need an external deadline to get motivated, Enjoy the satisfaction of rushing to beat the clock.

Misperceive the passage of time: One of the cause’s lateness on some people is underestimating the passage of time. For example, someone might need to go somewhere in afternoon and getting ready to leave 11:45. In meanwhile want to read news and check messages in social medias for about 10 minutes, but when s/he look at the clock already 20 minutes elapsed and begin running around with panic and trying to get out of home.

Underestimate on how things take time: Some people tend to allocate consistently less time than the actual time they need to complete their task. One simple example could be; someone might think the journey to work takes for him 30 minute when in fact the journey almost takes 45 minutes. Such thinking can come from a tendency of being overoptimistic as regard to time.

Procrastination is also one of the most common problems that cause time management difficult. Most of us procrastinate time to time, but some people tend to procrastinate consistently, and it is difficult for them to avoid this habit. Being procrastinate has its unpleasant consequence such as, unable to perform at full capacity due to postponing tasks, health problem caused by stress (Dewitte & Schouwenburg, 2002; Tice & Baumeister, 1997; Blouin-Hudon & Pychyl, 2015). On a study conducted on students by Schouwenburg & Groenewoud (2001), students tend to procrastinate when exam time is far, and they become tempted more on social events than their study. This behaviour causes stress on students when exam date approaches. In addition, it affects their performance on exams. The internet could also be one cause that often leads people to procrastinate. According to a report from Websense ®Inc. in 2005, cyber-
loaing cost the U.S. businesses $178 billion, because employees leave their main tasks aside and waste their valuable time surfing the internet on things that are unrelated to their job (Yan & Yang 2014).

Bing easily distracted: being easily distract or easily losing focus is one of the reason that causes lateness. For example, someone might need to complete a certain important task. While doing her/his task an email notification comes in her/his mobile. After checking email and s/he want also to check Facebook, meanwhile plenty valuable working hours elapsed without the person noticing it.

The mobile based time keeping application is an application developed for iOS based devices to help users to adopt an efficient time management skill. The app is designed to be utilized in day-to-day activities. The main desire of the application is to help user to be able to identify own weakness while using the application so that user able to take action on her/his problems. In the app self-reflection O/C matrix and PSD model is used as the main theoretical constructs to develop the application. We have utilized some selected features from primary task support, computer-human dialog support and system credulity to shape the application to be more persuasive.
3. Research Method

This chapter discusses a design science research methodology that we adopt for the purpose of carrying out this research. The first section briefly discusses the theoretical background of design science research, followed by an elaboration of design science research implementation in information system research. Finally, steps for conducting this research work are described.

3.1 Design Science in information system

The main promises of an information system research are to resolve problems that are found in the organization, people and environment. Design science is one of research methodology in information system, and its purpose is to create an artefact that define ideas, practices, technical capabilities and products. Those artefacts add value to the organization or solve problems that are found in people or environment. (Hevner et al. 2004)

Hevner et al., 2004 argue that to conduct a significant design science research in the information system (IS), it is crucial to utilize behavioural science. A behavioural science research seeks to develop and justify theories that explain or predict organizational and human phenomenon. Hevner et al. 2004 further developed a conceptual framework for understanding, evaluating and executing design science research in IS. The framework combines behavioural and design science. The framework contains Environment, IS research and Knowledge base. The environment includes people, organization and existing or planned technologies. IS research is a stage where the build/develop and justify/evaluate took place. The knowledge base contains foundation and methodologies. Foundations are previously conducted IS researches and methodologies are guidelines that help to justify/evaluate during IS researches.

The business need or a research problem comes from the Environment (people, organization, technology problems). To solve those problems existing knowledge (previously conducted IS research and existing technologies) are applied to build/develop artefact. Artefacts in design science research can be constructs, models, methods or instantiations. Constructs are vocabulary and symbols. Models are a simulation that represent the real world. Methods are processes that solve the problem which includes formula, mathematical algorithm, textual descriptions or a combination of those. Instantiations are implemented systems. (March & Smith, 1995; Hevner et al., 2004)

In design science research, in the development process two important dichotomies that must be conducted (Hevner et al., 2004). The dichotomies are design as process (set of activities) and design as a product (artefact). Design as a process contains a collection of activities which a researcher performs while conducting research. Design as a product involves the design of the actual artefact.
Hevner et al., 2004 list the following seven principles for performing a design science research in an information system.

1. Design of an artefact: The outcome of design science research should be a viable artefact in the form of constructs, model, methods or instantiations. This artefact solves the problem in a certain domain.
2. Problem relevance: The design science should be commenced to address business need of organization, people or technology by providing relevant artefact.
3. Design Evaluation: The evaluation is an important part of the research, and it must be executed rigorously with a well-known evaluation method. While performing the evaluation, it is important to define metrics and methods for gathering data. Evaluation is performed to access artefact’s functionality, completeness, consistency, accuracy, performance, reliability and usability.
4. Research contributions: Design science research should contribute clear and tangible contribution in the form of foundation and/or methodologies.
5. Research rigor: A design is commenced in iterative design and evaluation.
6. Design as a search process: The design should be commenced a process.
7. Communication of research: Design science research must be presented with the language both to technology-oriented, and nontechnology-oriented audiences understand.

3.2 Design Science Research Methodology (DSRM)

Peffers et al., (2006), developed a model for carrying out a design science research. The model helps researchers to follow the consistent procedure for producing and presenting a design science research. The model contains six consecutive activities which are problem identification and motivation, objectives of solution, design and development, demonstration, evaluation, and communication.

The first activity is **Problem identification and motivation**: in design science it is vital to specify the research problem and justify the benefits of the research. Doing this provides two advantages, first researcher and audiences are motivated to commit and find out the solution, and second it helps audiences to understand researcher's reasoning.

**Objectives of the solution**: the objective comes from the problem specification. The objective solution can be quantitative or qualitative. In qualitative solutions, the new artefact is produced to support problem. Whereas in the quantitative once, the aim is to produce artefacts which are advanced than the current one.

**Design and development**: in this activity the actual artefact is developed. The artefact could be construct, model, methods and/or instantiation.

**Demonstration**: demonstrates the artefact with the appropriate method as it solves the problem specification. The demonstration method can be experimentation, case study, proof or simulation, or other appropriate means of demonstration. While demonstration knowledge resources are needed to demonstrate the artefact effectively.

**Evaluation**: In evaluation the artefact is measured against the problem specification on how well it supports. To carry out this activity, relevant metrics and techniques are needed. Depending on the result, The researcher can decide whether to iterate back to design or development stage, if the artefact doesn't support the problem specification.
well, or proceed to the communication and leave further improvements to subsequent projects.

**Communication**: communication is the final stage of the design science research. This activity includes communication of problem, artefact utility and novelty, the rigors of the of the artefact design to the researcher and other relevant audiences. The communication requires its own discipline, for example in scholarly research publication the literature review, research method, development process and data collection, analysis and result, discussion and conclusion must be documented.

![Design science research process model](image)

Figure 1. Design science research process model (Peffers et al., 2006)

As Figure 1 illustrated, there is no need to follow activities in sequential order. The researchers could start at any point of the first four activities and proceed outwards. For instance, if the research is done based on the suggested future researches or observations of the problem, the activity could start from problem identification and motivation and then outwards. If the business need is already defined by consultants then, the research could start from the objective of the solution. The research could start from design and development if the artefact already exists to solve other problem but not thought to solve the intended one. Lastly, the research could start at demonstration if it is thought to extend the artefacts scope and go back rigorously. (Peffers et al., 2006)
3.3 Implemented research methodology

We have implemented DSRM (see Figure 2) to conduct this study. Due to the nature of our research problem, Problem centred approach is chosen as starting point to carry out the study. This section discusses the activities performed in the research process.

![Figure 2. Implemented DSRM](image)

Problem centered approach in BCSS is one of the prominent research areas in an information system. Oinas-Kukkonen, 2010, 2013 & Oinas-Kukkonen and Harjumaa (2009) has provided a recent framework to develop BCSS. This study uses those approaches as starting point to carry out research.

Problem identification and motivation: The problem involves to study the process BCSS, and the motivation for the app is, in addition, helping people to be more punctual and study the success of the persuasion principles in terms of change in end-user behaviour and attitude.

Objectives of Solution: the objective is to develop fully functional iOS app based on BCSS principles.

Design and Development: The development of the artefact followed the persuasive and formal software development process. To give the artefact a persuasive feature, we have utilized both PSD model and O/C matrix while designing and developing our artefact. O/C matrix provides a framework to analyse the desired behavioural changes of the system. Moreover, it is used to analyse outcomes reflected on the end user due to respective behavioural changes. O/C matrix is utilized in on this research as a laid principle during the initial stage of development of the process. The PSD model also provides postulates, context, and design features. PSD’s postulates are laid principles mostly used to list non-functional software requirements. PSD’s context is used to
analyse conditions where persuasion takes place, and PSD’s design features is used to gear the artefact to be more persuasive. In addition, we followed Apple’s iOS Human Interface guidelines while designing user interfaces.

Demonstration: The app first tested in simulator and device. The final artefact (app) is published on the Apple’s app store. After that the app is evaluated by users for about a week.

Evaluation: Evaluation is carried out by collecting user’s reflection on the artefact. For this purpose, we prepared a questioner which contain Likert-based question, open question and collection of adjectives. The questions is distributed through email. We have utilized goal question metric to analyse the evaluation. The details about the evaluation documented in chapter 5.

Communication: The entire research process including the literature review, research method, software development process, evaluation, and the conclusion is documented in this thesis.
4. Software Artefact

This chapter discusses the software artefact and its development processes. The artefact is developed for any individual who has come to realize that s/he has a problem with their time keeping and wants to improve time management skill. The artefact is developed based on BCSS concept. We called it RightOnTime and it is a native iOS app freely available in Apple store for iPhone and iPad.

This chapter discusses artefact design and development process. It begins analyzing the persuasion context, followed discussing software features with correlation to PSD’s design feature. The next section address software requirement and its implementation and the last section conclude with artefact demonstration

4.1 Persuasion context

Before the development of the application, we analysed the persuasion context, which includes intent, event and strategy.

4.1.1 The Intent

Acknowledging the intent requires addressing persuader and intended change type in the application. Persuaders are information processing science researchers, in the University of Oulu, OASIS department who want to study the development of BCSS. Our intention is to study the development of BCSS with domain problem in time management. In section 2.3 some reasons are mentioned that often cause tardiness, such as becoming over-optimistic for completing certain tasks over a certain period, becoming unrealistic while listing schedule, procrastination. Those problems are persistently affecting some people and causing lateness or in some case causing chronic lateness. The outcome of the study is to develop an application with BCSSs concept where users utilized it on a routine daily activity. Our intention is for the beginning user to comply with the system, and able to perform tasks that system require. As the user continue using the application for a longer period to bring behavioural change on user’s time management skill. The intended outcome that we want to bring to the end user is the formation of new behaviour.

4.1.2 The Event

User Context: The application targets any individual who wants to develop their time keeping skill regardless of their profession. It could particularly be more useful to those who has difficulties in using their time properly even to accomplish their routine daily tasks. For example, people who are consistently late by underestimating the actual time on their task, people who suffer from procrastinating, cyberloafing and others.

Use Context: The application can be utilized in day-to-day activities. For example, practicing timing for routine daily routine activities. Thus, by using the application users could identify their weaknesses and strengths through own evaluation.

Technology context: The application is developed for portable iOS devices, and it runs on both iPhone and iPad. Since the application is based on portable devices, users
usually have their device around, it is pretty handy to use the application almost everywhere they move.

4.1.3 Strategy

The strategy in the PSD model includes analysing of the message and route of persuasion. The application will be built on the principles of direct persuasion, where the features are geared towards relying on the user’s reason, intellect, and personal internalization (analysis) of the effects of the change process. The persuasive message contains suggestion which transferred in the form of reminders, and a self-reflection data that will be constructed on user’s own evaluation.

4.2 Software features

At the initial stage of research work, we have collected resources which explain about time management. Our interest areas include on topics, such as an effective use of time, common challenges which often cause people to be unpunctual. In addition, we have conducted an informal survey on few user to get their perception of time management. After completing this process, we came up with three different features to implement in our application. We have called this software features Activity, Practice and Tips and Hints. In practice and Activity, a self-reflection (Ploderer et al., 2014) both in-action and on-action approach are heavily used to support self-monitoring on the desired behaviour changes. We have utilized PSD model to gear our application to be more persuasive. Some selected features of primary task support, computer-human dialog support and system credibility support are applied in our application. Social support features are beyond the scope of this study. Thus, we haven’t implemented any features related to social interaction.

Practice

The purpose of practice features is to create a time-awareness on users on how long actually tasks take a time to complete. One of the problems that cause lateness is allocating less time for tasks than the actual time needed to complete (DeLonzor, 2003). For example, some individuals often assume to complete a certain task within a few minutes, in fact, the task often takes for them a longer time than they expected. Practice is targeted to create awareness on such kinds of problem. The application requests the user to practise timing for least one of routine daily activities. After user practising timing for a task can make a self-evaluation. On this feature, two parameters implemented for self-evaluation purpose, which are Satisfaction and Timeliness. On Satisfaction, the user makes self-evaluation depending on how happy s/he is on her/his performance and on Timeliness user makes self-evaluation depending on her/his timeliness. After making a self-evaluation, the user can keep data on the device and later can navigate back to the saved data to learn how her/his performance went through in the span of times.

Activity

The purpose of Activity feature is to help the user to organize and properly allocate time for daily activities. As discussed in section 2.3, there are various problems which often causes lateness if else chronic lateness on some people. Some of problem includes becoming over-optimistic while listing schedule, procrastination, being easily distracted and miss-perceive the passage of time. Those problems persistently happen on some of us and some people even if this problem shown consistently, they often assume as
tomorrow will be better instead of taking action on their behaviour (Vozza 2014). The Activity feature is targeted to tackle that kind of problems. This feature encourages the user to pre-plan and allocate time for daily activities. Just like Practise feature, the Activity feature also has two parameters for self-evaluation purpose. The application reminds user through notification to evaluate performance and timeliness after each activity is completed. The activity also reminds a user to review her/his self-evaluation at the end of the day.

**Tips and Hints**

The purpose of Tips and Hints feature is to provide the user with tips and hints about efficient time management. We composed messages which would be enough for about a week. The user gets three random tips and hints throughout a day.

We have used some selected design features from the PSD model in our application. Table 2 lists the selected design features implemented in our application.

Table 2: PSD design features implemented on the app.

<table>
<thead>
<tr>
<th><strong>Primary Task support</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduction</td>
<td>A behavioural change towards an efficient time management is a generic task. The task need be divided into a set of tasks so that a user able to perform the needed activities. In our application, we have reduced tasks such as “Activity”, and “Practice”. And each task designed in way user able to perform easily.</td>
</tr>
<tr>
<td>Self-monitoring</td>
<td>The user can do self-monitoring depending on self-assessment data (timeliness and satisfaction). The application also reminds the user to perform such tasks.</td>
</tr>
<tr>
<td>Rehearsal</td>
<td>The application provides a feature which user practices timeliness so that user can compare own expectation to the actual time needed to complete the task.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Computer-human Dialog support</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Reminders</td>
<td>In the application, different reminders implemented which remind the user to perform activities such as, to practising timeliness, to make self-assessment and review self-assessment data.</td>
</tr>
<tr>
<td>Suggestion</td>
<td>The app provides three tips and hints which suggest about punctuality though out the day.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>System credibility support</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Trustworthiness</td>
<td>In every section of the application its purpose clearly described. In addition, in “About” section a detail information is provided, such as the purpose of the application, about app developers and their contact address.</td>
</tr>
<tr>
<td>Expertise</td>
<td>The application is developed based on scientific literature review.</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------------------------------------------------------------</td>
</tr>
<tr>
<td>Surface Credibility</td>
<td>The app is developed based on iOS human interface guideline. Thus, similar practice is applied while designing UI.</td>
</tr>
</tbody>
</table>

**Social support:** No feature implemented.

In all features our intentions user to learn gradually her/his weakness on time keeping. It is required user’s reason, intellect, and personal analysis of the effects of the change process. The next section discusses the how those features implemented on the application.
4.3 Software development process

As standard software development, we have started the development process by listing out the functional and non-functional requirements. The requirement in software engineering includes identifying stakeholders and their need, formulation and documentation of the requirements, design and implementation of software (Nuseibeh & Easterbrook, 2000).

4.3.1 Software Requirement

The stakeholders in our application include users who want to practice punctuality and the system, which do persuasion behalf of persuader (System developers). The use case diagram in figure 3 shows user and system responsibility on the application.

![Use case diagram](image)

**Figure 3. Use case diagram**

On the *Activity*, the user should be able to insert, modify, delete her/his daily activities and should be able to evaluate and save his/her performance (Timeliness, satisfaction). The System should remind the user to perform a self-evaluation and review self-evaluation. However, the system shouldn’t disturb user’s main task. Therefore user should be able to prefer when to get reminders.
On Practice, the user should be able to add, delete and modify tasks that user wants to practise. In addition, the user should be able to change estimation time for completing the task. After completing practising for a certain task, user should be able to evaluate performance (Timeliness and Satisfaction) and save to database. On this feature, the system should remind the user to perform time-awareness practice daily.

The last part is Tips and Hints. In this feature, the system provides tips and hints randomly. However, the user shouldn’t be disturbed with notification. The user should prefer when to get reminders.

4.3.2 Software architecture

As any complex structure software also needs a foundation, which ensures software reliability and future success. Software architecture is the process where the technical and operational requirements are defined while enhancing software quality such as performance, maintainability, and manageability (Clements et al., 2010). Deciding the software architecture involves a series of decisions in various factors, and each factor contributes on all over the success of the software application. Software architecture involves selection of structural elements and interfaces as well as the collaboration among those elements.

In the coca-core application (coca is application development interface for Apple) model view controller (MVC) is a chosen design pattern for application development (Apple Inc 2013). Thus for developing our application we adopt MVC design pattern.

![MVC design pattern](figure4)

Figure 4. MVC design pattern (Adopted Apple 2013)

MVS design pattern (figure 4) classifies application objects into three, which are model view and controller. The model handles objects related to data; view handles objects related to user interface and controller is play mediator role between view and model object. In our application, the model object is utilized to encapsulate data related to calendar’s event, Local notification, and user’s setting. Those data are persistent in our application, and it has no explicit relation to the application’s view object. The interaction between application’s model and view objects take place through controller object.

The purpose of model object is to encapsulate the data. In our application; there are three persistent data storages which are device’s database (SQLite), Calendar and Local notification. In the SQLite user’s setting and user’s self-evaluations data are stored. iOS application framework provides access to iOS calendar database. We have utilized iOS calendar to store data related to user’s events. In Local notification data such as notification time and notification message are kept.

The purpose of the view object is to provide an interface to the user. All graphics and interaction are handled by view object. The user can view and update model objects.
However, view and model objects do not have an explicit connection. All interaction between view and model object is handled through controller object.

Controller object plays an intermediary role between application’s view object and model object. It learns updates from a user and notifies the update to model object. It also used as a tunnel to display model object to the user. In overall MVC pattern helps to have a less or no interaction between user interface and data. This helps to broaden scalability, maintainability of the code. Whenever there is a change on view or model object, one’s change doesn’t affect others change. The next section discusses how this architecture is implemented on the application.

4.3.3 Software implementation

The software is implemented in Objective-C programming. Objective-C is the primary language for developing iOS and OX software (Apple Inc 2014). Objective-C follows the syntax of primitive C language; it also has object-oriented feature. In addition to plain objective C-programing language, we have also utilized different Apple’s foundation framework to develop our application.

As shown in Figure 5, the application contains 13 Classes. The name of the classes contain a suffix VC (view controller) at their name, contains the implementation of the View and Controller. Other Classes: EkCalender, NotificationManager and DBManager handle data related to calendar access, notification and user inputs respectively. The LineChart class handles the graph-drawing part of the application. Table 3, explains the detail responsibilities of each class on the application.

Figure 5. Class Diagram
<table>
<thead>
<tr>
<th>Class Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActivityVC</td>
<td>The Activity class responsible for handling user’s daily activities. Whenever a user adds, modifies or deletes an event, it will inform the NotificationManager, so that it add, modify or delete data related notifications.</td>
</tr>
<tr>
<td>SelfEvaluationVC</td>
<td>This class handles user’s self-evaluation. It provides an interface, which user can evaluate tasks on “Activity” and save it to device’s database.</td>
</tr>
<tr>
<td>SummaryVC</td>
<td>This class contains view objects for drawing summary chart and date-pickers (see Figure 10) which user get an overview own evaluation data. In addition, the class contains controller methods for this view objects.</td>
</tr>
<tr>
<td>PracticeListVC</td>
<td>This class handles task-lists that user wants to practise timing. It provides an interface where user add, delete and modify task-list (see Figure 11).</td>
</tr>
<tr>
<td>PractiseVC</td>
<td>This class handles the part which user practise timing. The view part contains, date-picker, timer, start/stop button, and a button which links to the summary page (see Figure 12). In addition, the class contains methods which controller these view components.</td>
</tr>
<tr>
<td>EvaluationForPracticeVC</td>
<td>This class handles self-evaluation for practice. It contains view elements: sliders for both timeliness and satisfaction parameters and buttons which user able to save data (see Figure 13).</td>
</tr>
<tr>
<td>SummaryForPracticeVC</td>
<td>The class contains view objects for drawing summary charts for selected task (see Figure 14)</td>
</tr>
<tr>
<td>EK Calendar</td>
<td>The class provides access to iOS Calendar. It helps to retrieve and store events on iOS calendar.</td>
</tr>
<tr>
<td>NotificationManager</td>
<td>The class handles all notification related objects that the application creates, modifies and deletes.</td>
</tr>
<tr>
<td>UserSettingVC</td>
<td>This class contains user’s preferences. The preference includes user able to choose the summary time, “continuous” mood and time for evaluating the activities (see Figure 15).</td>
</tr>
<tr>
<td>LineChart</td>
<td>This class handles chart drawing</td>
</tr>
<tr>
<td>DBManager</td>
<td>DBManager gives access to SQLite database, to save and retrieve data, such as the self-evaluations and user’s settings.</td>
</tr>
<tr>
<td>TipsandHints</td>
<td>The tips and hints stored in the application in JSON format. This class helps to retrieve data from the JSON file.</td>
</tr>
</tbody>
</table>
Activity

Figure 6 illustrates the flow diagram for *Activity feature*. When a user adds activity (tasks that user wants to be on time) the system schedules a notification depending on the user’s setting. If user’s setting is in “continuous” mode, then the system schedules a notification that remind the user to evaluate a task when it the task completed. If the user’s setting on “non-continuous” mode, the system will schedule a notification at user’s chosen time for evaluating the tasks. The system also schedules a notification which remind the user to summarize self-evaluation whenever there is data to for summary.

![Activity feature flow diagram](image)

Figure 6. flow diagram for Activity feature
Practice

Figure 7 shows the flow diagram for *Practise feature*. In this feature, the application is implemented to request user daily to practising time-awareness. To perform practice, first user adds tasks to the task list and change the default estimation time. Starting and stopping the timer, the user can practice how actually a task takes to complete. After completing practising, the user can evaluate performance (timeliness and satisfaction) and save to database. The estimation time that user modified will be saved on the database. The user can navigate back to self-evaluation data and get an overview of own performance at any time. System graphically present using chart how the performance went through in the span of times.

![Activity diagram for Practice feature](image)

Figure 7. Activity diagram for Practice feature
4.4 Artefact demonstration

This section demonstrates the software artefact. The application has four main section separated by a tabular interface, and it contains Home, Practise, Self-Evaluation and Setting sections (see Figure 8). From one page to the other page can be navigated through tab button.

4.4.1 Activity

**Home:** Fig 8 shows the home page of the application and its interface inherits iOS calendar. In this page, the Activity feature is implemented. The user can add delete or modify the daily activities that s/he want to perform throughout the day. On the top of the page, there is a brief description, which can be useful especially for the first time users.

![Figure 8. Home page of the app](image)

The “+” sign on the home page links to the page where user adds or modify items on the list. The user can also delete an item by flipping it to right or left.
**Self-Evaluation:** In this section user performs self-evaluation for daily activities. The system reminds the user to perform self-evaluation task. Figure 9 shows the first page of Self-Evaluation. The page contains the lists of activities including their status which shows whether the activity evaluated or not. The user can make a self-evaluation by tapping on an item. Two kinds of evaluation (Satisfaction and Timeliness) implemented on the application, and each of them is scaled from 1 to 5. In Timeliness, the user evaluates how happy in addressing her/his task according to schedule. The scale 1 represents for horrendously late and the scale 5 represent for comfortably on time. On Satisfaction, user can evaluate his/her performance on timelines. Satisfaction also scaled from 1 to 5. The scale 1 represents for poor performance on timeliness and 5 represents for good performance on timeliness, which shows the user is happy addressing his task. After completing the self-evaluation, the user can keep data using the Save button.

![Figure 9. Self-evaluation](image_url)
**Summary:** The *Summary* section is found under Self-Evaluation page. The Summary button on Self-Evaluation section links to Summary page (see Figure 10). In this page User can get an overview of her/his performance. It uses Self-evaluation (Timeliness and Satisfaction) data to draw the chart. Summary page has two date pickers. The date pickers used to extract a data within the range of time, which helps user to get an overview how his performance went through within a range of times.

![Summary page](image)

**Figure 10. Summary page**
4.4.2 Practice

In this section, the Practice feature is implemented. To use feature first add items (tasks) using the plus button (Figure 11). Once user added an item on the list, and it will be saved in the database.

![Figure 11. Add Item on Practise](image-url)

Practise your daily activities: how long does it really take to have breakfast or to commute to work each day. Use the "+" button on top right corner to add an activity to the list. After adding an activity and your estimate of how long you should spend on it, tap the activity on the list and record your time. You can repeat this activity every day, if you like, to see how you make progress. Check your results and progress by tapping the "i" button next to each activity on your list.
**Practice with timer:** Tapping on the item links to the page where user practises timing. The application usually set a default time for completing the task, but the user can change estimation time. Start/stop button starts and stops timer respectively. The timer is implemented to count downwards until estimation time is over and it will start to count upwards with red colour for the delay. When stop button hit, the evaluation button becomes active, and the user can make self-evaluation for task.

![Practise with timer](image)

Figure 12. Practise with timer
Self-Evaluation for practice: The evaluation page (Figure 13), holds estimation time, the time taken for completing the task and delay time. The self-evaluation has two parameters (satisfaction, timeliness). On Satisfaction, the user makes self-evaluation depending on how happy s/he is on her/his performance and on timeliness user makes self-evaluation depending on her/his timing performance. Both evaluation types are scaled from 1 to 5 and 1 represents for not happy with performance and shows user need re-evaluate his/her estimation time and 5 represents very happy with the performance that means the user is completed task according to estimation time. After evaluating performance, the user can save evaluation using “Save” button.

Figure 13. Practise Evaluation
**Summary for practice**: user can navigate back to self-evaluation records to overview timing performance. The detail button, shown in Figure 14, links summary page. The summary holds time stamps and self-evaluation chart.

**Journey to work**

Practise your daily activities: how long does it really take to have breakfast or to commute to work each day. Use the "+" button on top right corner to add an activity to the list. After adding an activity and your estimate of how long you should spend on it, tap the activity on the list and record your time. You can repeat this activity every day, if you like, to see how you make progress. Check your results and progress by tapping the "i" button next to each activity on your list.

![Summary Self-evaluation for practice](image)

Figure 14. Summary Self-evaluation for practice
**Setting:** Setting page includes about section and preferences for application’s notifications as shown in figure 15. The about section contains a description of the application. Continuous helps to control evaluation time. If the continuous mode set on, the user gets a notification which reminds to evaluate task at the end of each activity. If it is switched off, then the user will get a notification at his preference time. The second is the summary time the application reminds user’s to review the summary at user’s preference time.

Figure 15. Application setting
4.4.3 Tips and Hints

The user gets tips and hints about time management as shown in Figure 16. If in Setting, if the continuous mode is on, then the user will get three notification randomly at a random time, if it is not then the user will get a notification at preference time.

Figure 16. Tips and Hints
5 Evaluation

In DSR evaluation is crucial, it demonstrates artefacts utility, quality and efficiency (Hevner et al., 2004). Evaluation is carried out rigorously on the development process, and finally on the artefact (Pries-Heje et al., 2008). Vaishnavi & Kuechler (2004), states that DSR must be evaluated to identify its contribution to technology, people or organizations. In addition, evaluations help to identify weakness and strength of the artefact, as well as to find out any indication that could need future development.

This chapter discusses the evaluation process carried out on the artefact. The chapter covers objective of the evaluation, followed by experiment setting that includes participant’s background and evaluation method, and finally evaluation result and its analysis.

5.1 Objective of Evaluation

The main objective of this thesis is to study the process the design and development of a BCSSs by developing mobile application in domain area of time management. The main goal of the application is helping tardy people to bring behavioural change towards better punctuality as they continue using the application for a longer period. The development of BCSS includes methods, process and technology usage, and thus we have used the O/C matrix and PSD model to design and develop, as well as evaluate our application.

The objective of this evaluations is to access how those persuasive components (PSD design features) reflected on the app. Also, to find out weakness, strength, user perspective on the application, and to suggest any possible improvements for future development.

5.2 Evaluation setting

The experiment is based on a moral evaluation of participants, and it is conducted with the help of five students. Three of evaluators are information processing science students from University of Oulu, and the other two are information technology students from Helsinki Metropolia UAS. All of the participants do not have expertise in BCSS.

We first provide the evaluators a manual page that explains how to use the application, and then we have requested them to use the application for five days before answering the questionnaires. Questionnaires are prepared in Google forms, and the link were sent to each participant via email.

The questioner contains both open and closed (Likert scale) questions and desirability test. Likert question scaled from 1 to 5. 1 stands for completely disagree, 5 stands for completely agree. We adopted the questions from previously conducted researches (Karppinen et al., 2014; Lehto & Oinas-Kukkonen 2015). The open question is placed below the Likert questions, so that participants after rating the Likert question, they can explain their thoughts freely.
The third type of evaluation is desirability testing. To conduct this test, we have provided participants with lists of adjectives. Participants are free to choose five adjectives that they think could best explain the application. ISO definition of usability (9241-11) has three major elements for measuring usability, which are effectiveness, efficiency, and satisfaction. According to Barnum & Palmer 2010, Usability evaluations give good result in addressing effectiveness and efficiency issues, but may not do so well in addressing user satisfaction. In addition, the usability tests are not able to measure the user intangible experiences such as fun, enjoyment (Bendek & Miner, 2002). The desirability test initially studied by Microsoft Research group. The research team has conducted their research and brainstorm and listed 118 adjectives. The adjectives contain 60% negative and 40% positive. In our evaluation, we have used the adjectives that came from Microsoft (Barnum & Palmer 2010). In this test, we provided 118 adjectives, and we have requested them to select five adjectives among them.

5.3 Evaluation of artefact

This section discusses the analysis of the participants’ evaluation. To analyse Liker and the open question we have employed the goal question metric (GQM) method. A GQM is one of the methods adopted in software industries to analyses effectiveness of a software product. Goal refers objective of a certain particular feature in the software and question characterize the achievement or assessment of the goal. Metric are data that are associated with the question. Finally, the desirability test is analysed to find out user’s perception on the artefact.

5.3.1 GQM Evaluation

In our application, we have implemented some selected software features from primary task support, computer-human dialog support and system credibility support. To synthesise Goal question metric, we have categorized application’s feature to the respective correspondence of PSD design feature as illustrated from Table 4-6.

**Primary task support:** The implemented features in primary task support includes reduction, rehearsal and self-monitoring. Hence, the application should divide the desired behaviour changes into a set of tasks in which user able to perform. In addition, it should guide a user to the desired behaviour through self-monitoring and rehearsal.

Table 4: goal question metric for primary task support.

<table>
<thead>
<tr>
<th>Goal</th>
<th>Question</th>
<th>Metric</th>
</tr>
</thead>
<tbody>
<tr>
<td>The system should divide the abstract desired behaviour change</td>
<td>Does application help the user to rehearse the desired behaviour change?</td>
<td>Measured participant’s point of view.</td>
</tr>
<tr>
<td>into a set of tasks that user able to perform.</td>
<td>Does the application increase the effort in reaching the desired behaviour?</td>
<td></td>
</tr>
<tr>
<td>System should help user to learn own weakness regards to time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>management by self-monitoring and rehearsal</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Summary: On survey indicated figure 17- 18 most evaluation participants provided on average a positive response to the application’s primary task features. Some of the respondents replied that the application requests to allocate time for daily activities. They responded, allocation of time for tasks helped them to use time more efficiently. One of the participants answered; “Application helps as a reminder of an event routine. Also, one can ascertain if excess or lesser time is being spent on a certain task. Which in the future could help to reschedule time allocation for a particular task”. In addition, most of the participants answered that application could be helpful if it used for longer period. In contrary, some of the participants pointed that tasks that application require could be tedious to perform it daily.

Figure 17. Perceived effectiveness primary task support

Figure 18. Perceived effectiveness primary task support
**Computer-human dialog support:** On these application’s feature, it should help the user to perform the main or primary task. The chosen design features are reminders and suggestion. The purpose of reminders is to help the user to comply system and suggestions are to increase user’s knowledge about punctuality.

Table 5: goal question metric for computer-human dialog support.

<table>
<thead>
<tr>
<th>Goal</th>
<th>Question</th>
<th>Metric</th>
</tr>
</thead>
<tbody>
<tr>
<td>➢ System should help user to comply with system and perform tasks, such as allocating proper time for tasks, practising timeliness for tasks, and performing self-evaluation tasks</td>
<td>➢ Does the app motivates to perform required tasks in the application</td>
<td>➢ Measured participant’s point of view.</td>
</tr>
<tr>
<td>➢ System should increase user’s knowledge regards to punctuality</td>
<td>➢ Does system increase user’s knowledge on the intended behaviour change?</td>
<td></td>
</tr>
</tbody>
</table>

**Summary:** On the survey Figure 18-19, most of the participants provided an affirmative response to application’s computer-human dialog support features. Most of respondent replied that applications reminder helpful to perform tasks like: to evaluate activities and review self-evaluation. Some respondents also answered contents such as self-monitoring, and tips and hints could increase knowledge with regards of punctuality. However, one of the participants mentioned as reminder disturbed him and didn't have any effect supporting his main task.

Figure 18. perceived effectiveness: computer-human dialog support
Figure 18. perceived effectiveness: computer-human dialog support

**System credibility support:** On system credibility support the selected design features are trustworthiness, usability, expertise, surface credibility. Among these features, we have assessed mainly trustworthiness and surface credibility. Hence, the application should provide trustworthy information, and the user should feel secure while using application and also the application should be easy to use.

Table 4: goal question metric for system credibility support.

<table>
<thead>
<tr>
<th>Goal</th>
<th>Question</th>
<th>Metric</th>
</tr>
</thead>
<tbody>
<tr>
<td>The system should provide truthful and unbiased information.</td>
<td>Does the system provide user truthful and unbiased information?</td>
<td>Measured participant’s point of view.</td>
</tr>
<tr>
<td>The system should be secure, avoid vulnerabilities.</td>
<td>Does user feel secure while using the application?</td>
<td></td>
</tr>
<tr>
<td>Application functionalities should be easy to learn and use.</td>
<td>Is it easy to use and quickly learn application’s functionalities?</td>
<td></td>
</tr>
</tbody>
</table>

**Summary:** Figures 19 -22, shows survey response for a question directed to analyse applications system credibility features. Most of the participants answered application content is clearly described, and it is easy to understand. Most of them also agree as it was easy to interpret charts that come from self-evaluation records. However, they mentioned also on the first day finding the application’s functionality was difficult.

Most of the respondents feel secure, and they answered that they are not worried about giving any information the application requires. Their reason was since the application passed through Apple’s security measurement before it released on the Apple store.
In my opinion, it is easy to understand the information, the content presented on the app

Figure 19. participants perceived effectiveness: system credibility support

Using the app is straightforward

Figure 20. participants perceived effectiveness: system credibility support

In my opinion, I feel secure while using the app.

Figure 21. participants perceived effectiveness: system credibility support
5.3.2 Desirability Test

The final evaluation conducted by participants is desirability test. To perform this test, we requested participants to select five adjectives from among 116 adjectives provided along with questioner. Also, to provide a general comment about the application. Figure 22 shows the selected adjectives by participants.

Among 25 selected adjectives, usable, personal and clear each selected four times. Innovative is selected three times. Helpful and Accessible are selected two times. Relevant, ordinary, effective, creative complex and boring are selected one time.

The overall four of participants though the application is usable, personal and clear. One of participates selected complex, boring, ordinary, personal and yet clear. The adjective Personal has a positive implication for our application since it is one of the important features of the PSD model. Adjectives innovative and creative are also implied novelty of application. Clear, relevant and accessible implies some participants are able to understand the application content easily. On contrary boring, complex and ordinary implies are usability not inclusive all participants.

Participants on their final comment stated that the application could be useful with regular usage. However, they also stated more functionalities need to be implemented, and some suggested to add more graphics and interactivity on the application.

5.4 Summary

To summarize, the overall reflection users on the app functionality is affirmative. Most of the testers have agreed that the application could support to bring the desired behavioural changes if it is used for longer period. Testers also pointed out some issues that need to be addressed. Usability is an issue on the application. For most testers especially on their first day, it is difficult to find different application's functionalities. From our observation, the application also lacks appealing interface that could attract users to use application. Some also noted that the tasks that the app require might be tedious to perform in day-to-day activities.

For further development, one of the methods to address procrastination is giving priority for more important tasks. In our application, this kind of feature is not implemented.
However in future development this could be implemented in a manner, such as implementing functionalities that users be able to rearrange task lists depending to their priority. Also, to be able to give a colour and sound coding (reminders) according to their importance. These could help to catch user’s attention on the most important tasks.

In our application the summary part, it just simply shows chart depending on user’s self-evaluation and leave the judgment to a user. However, it could be improved in the way rewarding and acknowledging for good performances and suggesting for improvement for poor performances.
6 Discussion

This section discusses findings of the study during the design, development and evaluation processes.

This thesis studies the development of a BCSS to help people to adapt a better management skill. Using time efficiently is one of the important issues which determine success whether in education or working life. However, most of us have difficulties managing time efficiently due to various reasons, such as procrastination, becoming over-optimistic while listing schedule, unable to allocate proper time for our tasks and other problems. Some people are a victim of chronic tardiness due with those problems. The intention of the system to help users to be able to identify their weakness and be able to take measure while using the application in their day-to-day activities.

For designing and developing our application, we have used PSD model and O/C matrix. The PSD is a valuable tool while carrying out this research. PSD model contains postulates, context and design feature. PSD’s postulates provide the key assumptions that are needed to be considered in the persuasive system. They are mostly used in our application to list non-functional software requirements. For example, the sixth postulate is one of the non-functional requirement in our application. The postulate states that persuasive systems should aim at unobtrusiveness. In our application, the postulate reflected on limiting applications reminders so that users not disturbed from the main task.

PSD’s context provides a solid starting point for developing BCSSs, and it contains intent, event and strategy. Acknowledging intent requires analysing the change types and determining the persuader in the system. O/C matrix provides a unique approach for analysing outcome and change types in BCSSs. We have employed O/C matrix for analysing the change types. PSD’s event discusses circumstances in which a persuasion took place, and it includes analysing of user, use and technology context. We have utilized the concept of PSD’s strategy to analyse how the message is transferred to the user and to determine the route of persuasion. The persuasion context in general utilized at the initial stage of research work.

The design feature primary task support, computer-human dialog support and system credibility support helped to list software requirements and shaping software functionalities to be persuasive.

In this study, we utilized the DSRM framework that is illustrated Hevner et al., (2004), and Peffers et al., (2007). The framework provides a unique approach to carrying out the research by combining a behavioural science and design science. It gives six major activities that need to carry out while conducting design science research. Since our research combines both behavioural and design science, the DSRM is an appropriate method to carry out our research.
Technology usage

The emerging of portable devices such as mobile phones and tablets are one of the important tools for developing BCSSs application because users often hold if else always a mobile or tablets and this help to constantly obtain user’s attention. Moreover, the current mobile devices come with a suitable hardware with higher processing capability as well as complex software (operating system) which are suitable for developing full-fledged functional software.

Our application is developed for iOS devices, including iPhones and iPads. We have utilized Objective-C and various iOS frameworks. The iOS framework comes with various valuable software components which help easily develop complex application. In our application, we have used iOS frameworks such as Local notification for implementing reminders, SQLite for storing user’s data, EKCalendar for accessing a calendar that come along with operating system and various Cocoa and Cocoa Touch frameworks to develop the user interface.

Apple also provides a guideline to develop user interfaces called iOS human interface guidelines (Apple Inc, 2015). The guideline helps to follow Apple's standard for developing user interface components, such as buttons, textbox, and other common practices. We have followed Apple's human interface guidelines for developing the interface in our application.

We have used MVC design pattern for developing our application. MVC is a chosen design patterned by Apple to develop iOS application. MVC pattern minimizes helps to have a loose connection between model and view object. Hence, when there is needed an interface or data type change one's change doesn’t affect others change.

The final artefact tested by users. We have used a goal question metric along with the PSD model. Our main interest in the evaluation was to test the persuasive components that are implemented in the application. Most of the evaluation participants has responded that the application can be useful, helpful if it is used a longer period. They have also pointed more functionalities, interaction and graphics needed.

In summary, we have studied the development of BCSS with domain problem in time management. We have used PSD model and O/C matrix a theoretical construct for designing and developing our application. Hopefully, this research used as a reference for those who want to develop a BCSSs.
7 Conclusion

This thesis studies the development of a mobile based behaviour change system. The system developed with the concept of BCSSs and applied design science research methodology to developed native iOS application that help people to adapt better time management skill.

The theoretical background of covers topics BCSSs and persuasive system development. Since BCSS are a persuasive system, methodologies to develop persuasive system can also be applied to develop a BCSSs. Persuasive system development involves disciplines such as social psychology, communication and persuasion, computer science and information systems. The prior studies carried out the behavioural system, such as Goal-Setting theory, Cognitive consistency theory, Information processing theory, elaboration likelihood model and in computer science fields, such as technology acceptance model, are used as theoretical background to in persuasion. O/C matrix and PSD model provide a unique approach developing BCSSs. The PSD model provides a recent technics and approaches for developing a persuasive system. O/C matrix also elaborates outcome and change types caused by BCSS.

We have utilized design science methodology to carry out the research. Design science research methodology utilizes an iterative process of design and evaluation to solve problems that are found in the environment, which includes people, organization or technical problems. Design science utilizes existing knowledge to solve the research problem and create a new artefact. Our initial entry point to develop BCSS that help people to develop a better time management skill. We have utilized theoretical methods of persuasive system design as existing knowledge to develop our artefact. The final artefact tested by users. Their reflection documented by analysing the software artefact’s weakness and strength.

We have developed our artefact the mainly based on PSD model and O/C matrix. We begin our development by first acknowledging PSD postulate, and then analysing the persuasion context and design features. We have documented the software requirements and software architecture utilizing UML diagrams. And finally we demonstrated the actual artefact.

The evaluation is conducted with five persons. After evaluation participants using the application for five days, we provide them questioner which contains Likert, free forms, and collection of adjectives. The evaluation provided a useful information on app weakness, strength and features that need further development. Generally most of the evaluators give a positive feedback on app’s functionalities. Some testers pointed issues on the artefact that need to be addressed. Their reflection is documented in this thesis.

In the discussion, we discussed the findings of the study. The contemporary mobile technology is coming with suitable hardware and software that help to develop easily full-flagged interactive software system. We reviewed the benefit of the implemented a design science methodology. We have also acknowledged the PSD model and O/C matrix as important theoretical constructs for developing BCSSs.
One limitation of this research is that, since the development is in an early stage the ability of the application to alter behaviour is not measured. We had few number of testers, and the test focus was on the functionality of systems, such as whether the application capable of helping the desired task, usability and security. The application needs to be developed further adding more graphics and additional functionalities and should be tested longer period, to measure the persuasiveness and capable of bringing a behaviour change on the end user.
References


