University of Bergen
Faculty of Social Sciences
Department of Information Science and Media Studies

Master Thesis

Supporting coach-player communication in organised sports teams with rockEnroll

Author:
Yngve Sundfjord

Supervisor:
Barbara Wasson

October 20, 2014

Abstract

Situations frequently occur where players on organised sports teams do not show up for scheduled activity without having notified the coach in advance, as it often requires extra effort from the coach. This happens despite the fact that players have a vast array of communication technologies available, and should be able to efficiently communicate their intended absence in a heartbeat.

This thesis is intended as a starting point for research on communication between player and coaches, and how information technology can contribute in this context. The goal of the research was to understand how players and coaches currently communicate, to investigate if a specialised, novel information technology could improve communication, and to provide design guidelines for future development of similar information technologies. In order to achieve the research goals, a prototype web application, rockEnroll, was developed.

The prototype was developed in two iterations. Web surveys were administered to investigate the current situation and gain a greater understanding of the user groups needs. Heuristic evaluation was used to improve upon the first version of the prototype, while a usability test was carried out after development had been completed. In addition, an extensive field study involved testing the prototype on real users.

The results showed that Facebook and SMS is currently the dominating technologies of communication, however, it was discovered that these current technologies seemed unsuitable to automatically aggregate information from players and provide a visual overview over attendance for an activity. The data collected did not conclusively establish that rockEnroll improved communication, but suggested that improved versions of rockEnroll has the potential to greatly improve communication. Making the technology the solution of least effort, designing an accompanying native or hybrid app and (or alternatively) providing Facebook integration is recommended in order for an extension of rockEnroll or another novel system to successfully improve communication.

Preface

Quite a few people influenced the work presented in this thesis and in some or another way helped me throughout the process, and for this I would like to extend my gratitude.

First and foremost, my supervisor **Barbara Wasson** deserve thanks beyond the ordinary. She showed her belief in the project by immediately taking me and my idea under her wings. She gave valuable feedback, some of it she provided during her vacation or amidst chaotic circumstances following a tropical storm in her hometown in Canada. She was also on her own initiative my proofreader, correcting the idiotic grammar mistakes I tend to make. Last, but not least, she showed patience, understanding, and even encouragement for my endeavours as a volleyball athlete on an international level, which ultimately caused me to postpone submission of the thesis. I consider this effort to by far exceed the expectations of a supervisor, and for this I am extremely grateful.

Also, thanks goes out to some of my fellow students for their feedback, positive influence and good companionship during the thesis work. Particularly I wish to thank my good friend **Eivind Mjelde** for fruitful discussions, lunches at Kvarteret and for making me chuckle even when I felt like throwing myself out the window of room 602 in frustration. In addition, I wish to thank the coffee machine on the 6th floor of the Faculty of Social Sciences for providing beverages that woke me up in the mornings.

I wish to thank everyone who took the the time to submit surveys, participate in tests and use the prototype during the trial period. In particular, **the coaches** for the three teams involved in the field study deserve sincere thanks for going out of their way to help me.

Last, I wish to thank my girlfriend **Maja**. Thank you for pretending to be interested in my blabberings about insignificant details in my thesis. Thank you for putting up with my physical and/or mental absence whilst working on the thesis. Thank you for calming me in times of frustration and lostness. Your encouragement and support was vital for my sanity, and for that I am truly grateful.

Contents

List of Figures

List of Tables

1 Introduction

In the increasingly technological world we live in, many communication technologies are available for people to use. Mobile phones allow for calls and text messages, and the emergence of smart phones extends this by allowing the user to write messages on social networks such as Facebook or with messaging services such as Whatsapp. This technology can be used practically anytime, anywhere, and allows both synchronous and asynchronous communication. In addition, more old-fashioned but ever popular technologies such as email are readily available to more or less everyone using a computer.

Yet despite possessing such a vast array of technologies enabling instant communication, situations still frequently occur where players on organised sports teams do not show up for scheduled activity without having informed the team's coach in advance. Such situations can be difficult for the coach, as the absence of players may greatly affect the plan for that particular practice or match. In addition, this element of instability from players over time requires an enormous amount of extra effort for coaches to keep track of who is attending activities. Over time, such unneccesary extra work is tiring for the coach. To further introduce the issue in question, a plausible scenario is presented. Scenarios are simple, but flexible and powerful design representations (Dix et al., 1998).

> *John is the coach of a football team. He wants to focus on the defensive aspects in today's practice. He writes down several drills aimed to improve these aspects and spends a considerable amount of time planning the session. When practice starts, four of the six defenders on the team are nowhere to be seen, without having let John know of their intended absence beforehand. As a result, the team will not benefit from the planned drills. John has to reorganise the practice, losing valuable time in the process. John is aware that this is a recurring problem, but he dreads the thought of sending individual text messages to players asking if they will be there or posting a question on the team's Facebook group every time there is an activity scheduled.*

With so many suitable technologies at hand, notifying a coach of intended absence at a practice or match can and should be done in a heartbeat. Still, players may be absent from training without having notified the coach. Why do players seem to neglect established, efficient technologies for communication? It appears that in order to improve the situation, a conscious effort must be made to overcome players' apparent reluctance to use available communication technologies to notify coaches of their absence. Not only must a novel technology inspire voluntary use, but it must do so on a variety of platforms, in a variety of settings. People are increasingly accessing the web and using communication technologies in urban environments such as bus stops and cafés using smart phones or tablets. This paradigm shift indicates that development of a web application must be in accord with dynamic user requirements that may widely differ depending on setting and device. For example, the conditions for user interaction with a technology using a desktop computer with a widescreen monitor in an office are completely different from those when using a smart phone while standing on a moving bus during rush hour. The user, however, can and should expect intuitive interaction with a user interface regardless of these factors.

This thesis addresses communication between players and coaches. After surveying the current communication situation, a prototype web application was developed, tailored for supporting communication between players and coaches in organised sports teams on all platforms. This prototype was used in a real setting and evaluated by several organised sports teams. The purpose of this was both to obtain a better understanding of how players and coaches currently communicate and to investigate if a specialised, novel information technology could improve communication. In addition, the thesis attempts to determine what particular functionality or quality in the prototype contributes to overcome the reluctance among players to communicate to their coaches, in order to provide design guidelines for future development of similar information technologies. The thesis is intended as a platform, a starting point for further research on the topic.

1.1 Research Questions

In support of this attempt to survey and improve the communication between players and coaches, three research questions have been formulated:

RQ1: *How is information technology currently supporting communication between players and coaches?*

RQ2: *Can a specialised, novel information technology improve communication between players and coaches?*

RQ3: *What guidelines can be given for designing an information technology that helps players overcome reluctance to communicate to coaches?*

1.2 Thesis Contents

The thesis is organised into 8 chapters. Chapter 2 introduces relevant theory and fields of research, while chapter 3 explains which methods for research, system development and data gathering were used. Chapters 4 and 5 describes the two development iterations, with data collection and analysis, as well as explanations for the many design decisions made. Chapter 6 documents the results of the summative data gathering methods that were carried out as an evaluation of the prototype. The research questions are discussed in chapter 7 based on results from the evaluation and related theory and research, while the final chapter summarises the research, discusses the limitations and weaknesses, and suggests future research.

2 Related Theory and Research

This research is situated at the intersection between information technology acceptance and user-centered Human-Computer Interaction design. Concepts, research and techniques relevant to the research are introduced in this chapter.

2.1 Information Technology Acceptance

Information technology acceptance is a field of research that concerns issues regarding how users come to accept and use an information technology. Researchers have found and since reiterated for decades that low adoption and use of IT by employees is a major barrier to successful IT implementation in organizations, an issue that has only become more severe over the years as IT is becoming more complex and more instrumental to a plethora of tasks within organizations (Venkatesh and Bala, 2008).

Several theoretical models has been developed in an attempt to describe user acceptance in IT. These user acceptance models are based on the same basic concept, as shown in figure 2.1. User reactions to using an information technology forms the intentions to use it, which in turn leads to actual use.

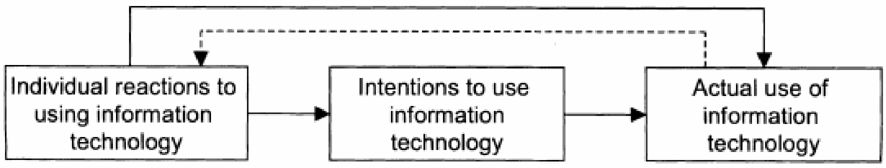


Figure 2.1: Basic concept underlying user acceptance models. (Venkatesh et al., 2003)

Two of these models will be introduced, namely the Technology Acceptance Model (TAM) and the Unified Theory of Acceptance and Use of Technology (UTAUT). In the presentation of TAM, the original TAM (Davis et al., 1989; Davis, 1989), TAM 2 (Venkatesh and Davis, 2000), and TAM 3 (Venkatesh and Bala, 2008) will be treated as one model. Although Bagozzi (2007) suggest that UTAUT is another extension of TAM, this model will here be presented separately. Various determinants and variables from these models are used to help explain or rationalise findings discussed in chapter 7.

2.1.1 The Technology Acceptance Model

The most influential and commonly employed theory attempting to provide an explanation of the determinants of computer acceptance is the Technology Acceptance Model (TAM) proposed by (Davis et al., 1989; Davis, 1989). The model was an extension to the Theory of Reasoned Action (TRA) proposed by Ajzen and Fishbein (1980); Fishbein and Ajzen (1975), a model derived from the social psychology setting for the prediction of behavioural intention.

TAM proposes two particular beliefs to be of primary relevance for computer acceptance behaviours (Davis et al., 1989; Davis, 1989).

- *Perceived usefulness* is defined as "the degree to which a person believes that using a particular system would enhance his or her job performance" (Davis, 1989, p.320).
- *Perceived ease of use* is defined as "the degree to which a person believes that using a particular system would be free of effort"(Davis, 1989, p.320).

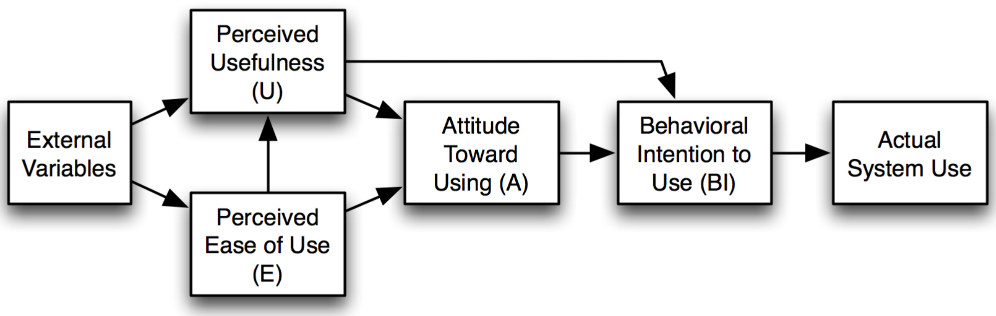


Figure 2.2: Technology Acceptance Model (Davis et al., 1989)

As can be seen in figure 2.2 above, the two determinants are influenced by *external variables*. If a user receives training in how to use a technology, or has access to well-made documentation, it will be easier for the user to learn the technology and subsequently accept it. Hence, training and documentation are examples of external variables that influences ease of use. The model also indicates that the perceived ease of use has an impact on the perceived usefulness. If a new technology is user friendly, it will also most likely be perceived as more useful by a user.

The two determinants, influenced by external variables, creates an *attitude towards using* the technology. This attitude, coupled with perceived usefulness, affects the user's *intention to use* the technology. This intention to use subsequently affects actual use of the technology. Perceived ease of use is not theorised to have a direct effect on the intention, as user friendliness will not help if the technology is not useful for executing the task at hand.

Many individual constructs have since been added to TAM, such as *trust, cognitive absorption, self-efficacy, disconfirmation, information satisfaction, top management commitment, personal innovativeness, information quality, system quality, computer anxiety, computer playfulness, and perceptions of external control* (Benbasat and Barki, 2007). As a more complete extension of the original model, TAM 2 was proposed by Venkatesh and Davis (2000). It introduced several new constructs. These were split in two categories; *social influence* and *cognitive instrumental processes*.

According to Venkatesh and Davis (2000), three interrelated social forces impinges an individual facing the opportunity to adopt or reject a new system. *Subjective norm* is how people around the user influence use of the new technology. *Image* is defined as the degree to which use of an innovation is perceived to enhance one's status in one's social system. *Voluntariness* is defined as the extent to which potential adopters perceive the adoption decision to be non-mandatory.

Venkatesh and Davis (2000) claim that people form perceived usefulness judgments in part by cognitively comparing what a system is capable of doing with what they need to get done in their job. Four determinants of perceived usefulness is theorised. *Job relevance* is defined as the perception of an individual regarding the degree to which the target system is applicable to his or her job, *output quality* is how well the system performs those tasks and *result demonstrability* is the tangibility of the results of using the innovation. In addition, TAM2 retains *perceived ease of use* as a determinant of perceived usefulness. Figure 2.3 below shows TAM2 and the relationships between determinants.

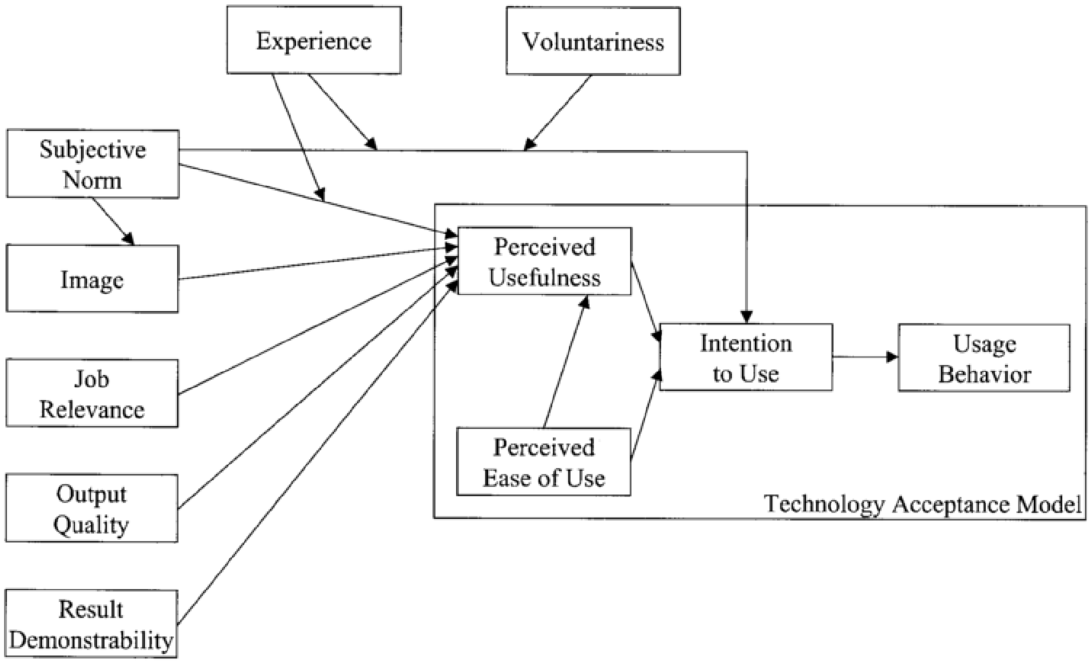


Figure 2.3: Proposed TAM2 (Venkatesh and Davis, 2000)

Despite its impact on technology acceptance research, shortcomings and limitations of TAM and "the abundance of literature that more or less replicate TAM's original message" (Benbasat and Barki, 2007, p.213) has been pointed out. Criticism of TAM has concerned both its methodology and its theoretical foundation. Methodological criticism has concerned TAM's use of students as participants, which cannot be generalised to the real world (Lee et al., 2003), and the self-reported use data which is claimed to be subjective and hence unreliable (Legris et al., 2003). Bagozzi (2007) pointed to the poor theoretical relationship that was formulated among the different constructs in TAM. Benbasat and Barki (2007) pointed out that

> *"study after study has reiterated the importance of PU, with very little research effort going into investigating what actually makes a system useful. In other words, PU and PEOU have largely been treated as black boxes that very few have tried to pry open."* (p.212)

They recommended to revisit TRA and redirect the focus towards examining different antecedents and different consequences in order to gain a greater understanding what influences adoption and acceptance in IT and to provide more useful recommendations for practice.

2.1.2 Unified Theory of Acceptance and Use of Technology

Venkatesh et al. (2003) reviewed the user acceptance literature and formulated a unified model that integrated elements from eight prominent models within this body of literature. This unified model is named the Unified Theory of Acceptance and Use of Technology (UTAUT).

In UTAUT, four constructs are theorised to play a significant role as direct determinants of user acceptance and usage behaviour. *Performance expectancy* is the degree to which an individual believes that using the system will help him or her to attain gains in job performance. *Effort expectancy* is the degree of ease associated with the use of the system. *Social influence* is the degree to which an individual perceives that important others believe he or she should use the new system. Finally, *facilitating conditions* is the degree to which an individual believes that an organizational and technical infrastructure exists to support use of the system.

UTAUT names four key moderators of these determinants, namely *gender*, *age*, *voluntariness*, and *experience*. Figure 2.4 shows by which moderator each determinant is moderated, and how the determinants affect *behavioural intention* and *use behaviours*.

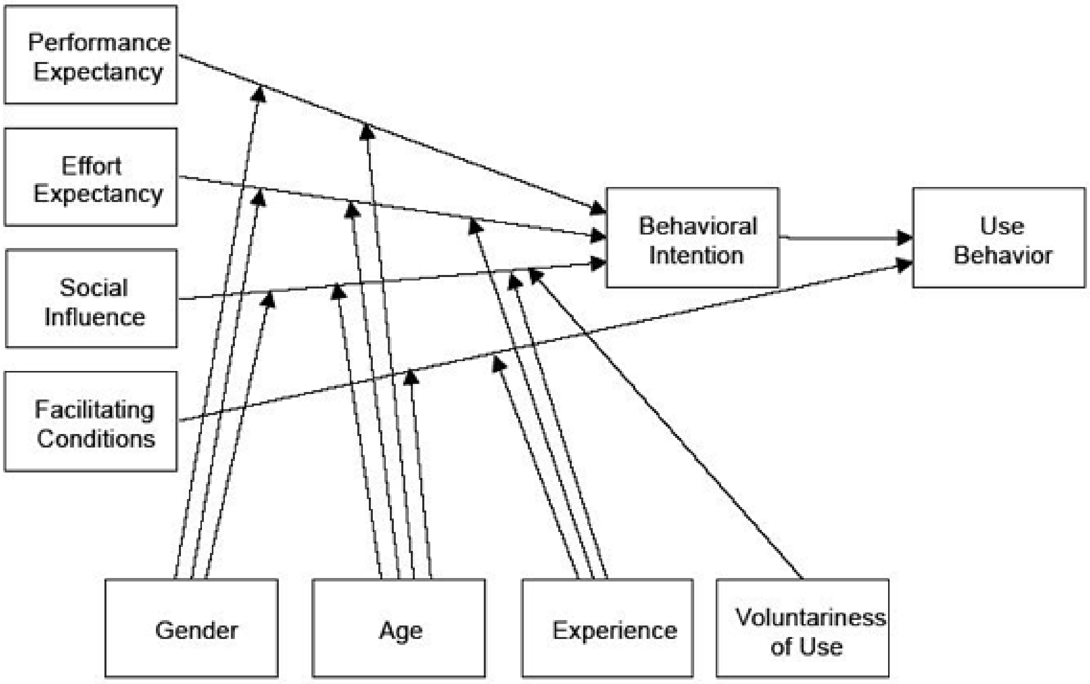


Figure 2.4: Unified Theory of Acceptance and Use of Technology (Venkatesh et al., 2003)

In addition, three constructs are theorised to not be direct determinants of intention. These are *self-efficacy*, people's judgments of their capabilities to perform a given task; *attitude towards using technology*, an individual's overall affective reaction to using a system; and *anxiety*, evoking anxious or emotional reactions when it comes to performing a behaviour, such as using a computer.

Venkatesh et al. (2003) claim that since UTAUT through empirical validation explains as much as 70 percent of the variance in behavioural intention, the practical limits of our ability to explain individual acceptance and usage decisions are being approached. Still, Bagozzi (2007) criticise UTAUT for contributing to the research field reaching a stage of chaos, claiming the model to be "a patchwork of many largely unintegrated and uncoordinated abridgements" (p.252) due to the large amount of independent variables it includes.

2.2 Lazy User Theory

Lazy User Theory is a theory that explains how users select what product or service to use when there are numerous possibilities (Tetard and Collan, 2009; Collan, 2007). It explains the selection process as the user selection of the solution that demands the least effort, an idea also found in physics, linguistics and informatics.

The lazy user theory of user selection uses two variables to define a set of possible products and services (henceforth labeled *solutions*). The *user need* is defined as an explicitly specifiable want that can be completely fulfillled, and this need can be tangible or intangible. Relevant for the theory are user needs that can be fully fulfillled by the utilization of solutions.

User state is defined by Tetard and Collan (2009) as the circumstances that surround the user at the moment when the user need arises. Examples of relevant circumstances are location, available devices, available resources, and available time. The user state may limit the set of solutions that fulfilll the user need to the set of possible solutions.

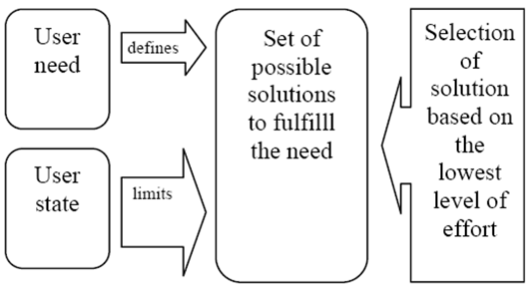


Figure 2.5: Lazy user theory of solution selection (Tetard and Collan, 2009)

Together, these two variables define the set of possible solutions to fulfilll the user need; the possible solutions can be material or immaterial objects and can be delivered by different products, devices, or services, depending on the need. The lazy user theory of solution selection assumes that the user will select the solution demanding the least effort, as shown in figure 2.5 above. A definition of *effort* provided by Tetard and Collan

(2009), is that effort is the amount of for example time, money, or energy or combination of these used to fulfill the need.

Two particular factors influence the decision of users to switch from one solution to another, namely *switching costs* and *learning issues*. Switching costs was defined by Thompson and Cats-Baril (1990, referenced in Tetard and Collan (2009)) as the costs associated with switching suppliers. Relevant examples of switching costs in this context is *learning costs*, the initial learning and *psychological costs*, the attachment to an old solution and the resistance to change. These are dynamic costs that may change over time. When switching to a new solution, the user will favor the solution where switching costs are minimal. An interesting aspect of switching costs is understanding why users switch. Relevant questions in that regard is which costs are the barriers to a possible switch, and what triggers switching.

Mentioned as an example of a switching cost, learning issues is an important factor influencing users in their choice of a new solution. Tetard and Collan (2009) identify 4 phases of learning in the solution adoption process:

- Pre-usage
- First time of use
- Early use
- Routine use

The pre-usage phase describes the point where the user needs information about a solution. In this phase, an expectation of how well the system performs and its usability may be created. Word of mouth or documentation of a solution may be sources of learning for a user. The expectations created in the pre-usage phase is put to the test when the user uses the solution for the first time. It is a critical point in the adoption process, because the first impression may result in the user accepting or rejecting the solution based on it, as it can be difficult to change a user's willingness to use a solution if he or she has built a negative attitude towards it. In early use, a routine in using the system will be built. The user learns more about how to use the solution by discovering new features and increases problem-solving ability. The last phase is routine use, and in this phase the user knows how to use the required functions without committing major errors.

Tetard and Collan (2009) argue that the theory has implications on the design of new products and services. For instance, designing solutions from the point of view of least effort may alter the focusing of resources. It is also claimed that niche markets for new products and services may be discovered through search and identification of user states where there are no devices that fulfill the user needs. Finally, the theory implies that a very small increase in ease of use may not justify the effort of new learning, hence users may not adopt new solutions unless the cost of learning is not fully refunded by advances in ease of use.

2.3 User-centered Human-Computer Interaction Design

Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them (Hewett et al., 1996). It is the region of intersection between psychology and the social sciences on the one hand, and computer science and technology on the other (Carroll, 1997). It is a multidisciplinary field, attracting professionals from many disciplines and incorporating diverse concepts and approaches.

Interaction design is a field related to HCI, but according to Rogers et al. (2011), differing in scope. HCI is usually considered a subset of interaction design. Interaction design has a broader focus, being concerned with the theory, research, and practice of the user experiences for all manner of technologies, systems and products. Interaction design is not just about the artifact that is produced, whether a physical device or a computer program, but about understanding and choosing how that is going to affect the way people work. Because of this, it may be better to think about *designing interventions* rather than systems or artifacts, as the product of a design exercise is to intervene with a current situation and change it to the better (Dix et al., 1998). At the heart of the discipline is *usability*, a quality attribute that assesses how easy an interactive object is to learn and use. The concept of usability will be introduced further below.

One of the most popular philosophies in HCI is *user-centered design*, a broad term that describes design processes in which end-users influence how a design takes shape (Abras et al., 2004). The user is placed at the center of the process (Rubin and Chisnell, 2008). Rogers et al. (2011) believe users and their goals, not just technology, should be the driving force behind development of a product. User-centered design comprises a variety of techniques, methods, and practices, each applied at different points in the product development lifecycle. Some of these techniques will be introduced below.

2.3.1 Usability Evaluation

When designing modern web pages and applications, end user satisfaction is a requirement. A considerable body of literature is aimed towards guiding designers towards achieving good *usability* in their designs. Rubin and Chisnell (2008) rely on this definition of usability: *"When a product or service is truly usable, the user can do what he or she wants to do the way he or she expects to do it, without hindrance, hesitation or questions*" (p. 4).

Nielsen (2012) name five quality components that define usability. 1) *Learnability*: How easy is it for users to accomplish basic tasks the first time they encounter the design? 2) *Efficiency*: Once users have learned the design, how quickly can they perform tasks? 3) *Memorability*: When users return to the design after a period of not using it, how easily can they re-establish proficiency? 4) *Errors*: How many errors do users make, how severe are these errors, and how easily can they recover from the errors? 5) *Satisfaction*: How pleasant is it to use the design? In addition to this, *utility* is one of many other quality measuring attributes. Utility refers to a system's functionality, and assesses whether the system does what the users need or not. Together, usability and utility determine the *usefulness* of a system (Nielsen, 2012).

There are a variety of techniques and methods that may test and improve usability. Below, three main usability evaluation approaches are introduced, namely analytical evaluation, field studies, and usability testing.

2.3.2 Analytical Evaluation

Rogers et al. (2011) explain that *analytical evaluation* consist of two categories of evaluation methods: *inspections* and *predictive models*, theoretically based models which are used to predict user performance. Here, only the concept of inspections will be explained further as this is the only category utilised in this research.

Usability inspection is the generic name for a set of methods that are all based on having evaluators inspect the interface. Typically usability inspection is aimed at finding usability problems in a design (Nielsen, 1994c). The most well known inspection method is the *heuristic evaluation*. This method involves having a small set of evaluators, one at a time, examine interface elements such as dialog boxes, menus and navigation structure and judge their compliance with the heuristics, which are predefined, recognised usability principles (Nielsen, 1995b). Subsequently, inspection reports from all the evaluators are combined to form the list of usability problems (Nielsen, 1994c). Advantages of a heuristic evaluation pointed out by Nielsen and Molich (1990) are its low cost, its intuitive nature, that it is easy to motivate people to do it, that it does not require advance planning, and finally that it can be used early in the development process.

When attempting to find out whether or not industry professionals would recognise serious user interface problems, Molich and Nielsen (1990) classified usability problems in accordance with nine principles that reflected their personal experience. These principles have since been refined in Nielsen (1994a) and expanded in Nielsen (1994d), resulting in a set of ten usability heuristics, which are displayed in table 2.1 below. The list of heuristics can be added upon in order to fit a certain field (Nielsen, 1994d).

Heuristic	Description
Visibility of system status	The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.
Match between system and the real world	The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.
User control and freedom	Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.
Consistency and standards	Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.
Error prevention	Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.
Recognition rather than recall	Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.
Flexibility and efficiency of use	Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.
Aesthetic and minimalist design	Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.
Help users recognize, diagnose, and recover from errors	Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.
Help and documentation	Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

Table 2.1: Table of heuristics (Nielsen, 1994d)

2.3.3 Field Studies

Field studies, according to Dix et al. (1998), takes the designer or evaluator out into the user's work environment in order to observe the prototype in action and how it is adopted and used by people in their working and everyday lives (Rogers et al., 2011). This is important since the way people interact with products in laboratories is different from how they interact with them in their messy everyday worlds. It is possible to get a better sense of how successful the product will be in the real world by evaluating how people think about, interact, and integrate it in the settings in which they will be used.

How field studies are conducted can vary greatly, both in terms of duration and method. Concerning duration, field studies can range from just a few minutes to a period of months, years, or even decades (Rogers et al., 2011). Particularly long-lasting field studies are called *longitudinal field studies*. Data may be collected by interviewing or observing people, and video, audio, and the researcher's field notes can record what occurs. In addition, participants may be asked to provide self-reported data, such as intermittently filling in dynamic forms with questions designed to provide the researchers with the desired information.

2.3.4 Usability Testing

In accordance with Rubin and Chisnell (2008), the notion of *usability testing* refers to a process that employs people who are representative to the target audience as testing participants to evaluate the degree to which a product meets specific usability criteria. The overall goal of usability testing is to inform design by gathering data from which to identify and rectify usability deficiencies. Despite its suitability to discover usability issues and thus improve the usability of a product, Rubin and Chisnell (2008) claim that a usability test fits into every phase of a development lifecycle. For instance, a *validation test* can be conducted late in the development cycle to confirm that problems discovered earlier have been remedied and that new ones have not been introduced. Hence, usability testing is valid as a summative data gathering method.

In usability testing, the product is placed in a controlled environment, where users' performance of pre-planned tasks are repeatedly measured (Rogers et al., 2011). Examples of tasks may be searching for information or navigating through menus. The information collected is often measured in time and number. For example, a typical measure of time is how long a user takes to complete a task, and a typical measure of number is how many errors a user makes trying to complete a task. What type of information to collect should be decided beforehand (Rubin and Chisnell, 2008).

The finding and selection of participants is an important aspect of usability testing. Effort should be placed on identifying participants whose background and abilities are representative of the product's intended users, as the results will otherwise be invalid and of little use (Rubin and Chisnell, 2008). With regard to how many participants to use, Rubin and Chisnell (2008) stress the importance of balancing the need for acquiring participants with practical constraints of time and resources. Although opinions are split (Spool and Schroeder, 2001), general consensus in research and industry seems to indicate that testing with five users is sufficient to generate valid results (Nielsen, 2000; Dumas and Redish, 1999; Rubin and Chisnell, 2008; Rogers et al., 2011).

2.4 Design Rationale

Design Rationale (DR) is defined by Jarczyk et al. (1992) as the explicit listing of decisions made during a design process and the reasons why those decisions were made.

MacLean et al. (1989) argues that to understand why a system design is the way it is, one also needs to gain understanding of how it could be different, and why the choices which were made are appropriate. The term Design Rationale is used to refer to this representation, which allows for the description of a design space rather than just a single artifact. A design rationale is a representation for explicitly documenting the reasoning and argumentation that make sense of an artifact.

The use of semiformal notation for structuring arguments about design decisions attracted much interest from researchers and industry, which led to the development of several different notations and support environments (Shum and Hammond, 1994). Building on a general argumentation format proposed by Toulmin (2003), examples of such argumentation-based approaches to design rationale include *Design Space Analysis* by MacLean et al. (1991b), *Issue-Based Information System* (IBIS) notation suggested by Kunz and Rittel (1970), and *Decision Representation Language* (DRL) by Lee and Lai (1991). The former example will be explained further below.

Design rationale is important because an artifact needs to be understood by a wide variety of people who have to deal with it. This variety of people ranges from those who design and build it, to those who sell and service it, to those who actually use it (MacLean et al., 1991b).

2.4.1 Design Space Analysis

Design Space Analysis, suggested by MacLean et al. (1991b) is an approach to representing design rationale. It is a method that is utilised for solving both specific design problems as well as wider problems, such as the general design approach to mobile devices, for which one or more options could be chosen as solution. It places an artifact in a space of possibilities and seeks to explain why the particular artifact was chosen from these possibilities. The semiformal notation it uses is called QOC, to represent the design space around an artifact. The main constituents of QOC are:

- Q - Questions - key issues for structuring the space of alternatives
- O - Options - possible alternative answers to the Questions
- C - Criteria - the bases for evaluating and choosing among the Options

Choosing among the various Options requires a range of considerations to be brought to bear and reasoning over those considerations. The most important elements for organising this reasoning are *Criteria*. As with the other parts of the design, appropriate criteria have to be invented by the designer. The relationship between Options and Criteria is an *Assessment* of whether an Option supports or challenges a Criterion. Figure 2.6 shows an example of reasoning with the QOC-notation, where the solid lines indicate that the criterion has been met, while stroked lines indicate that the criterion has not been met.

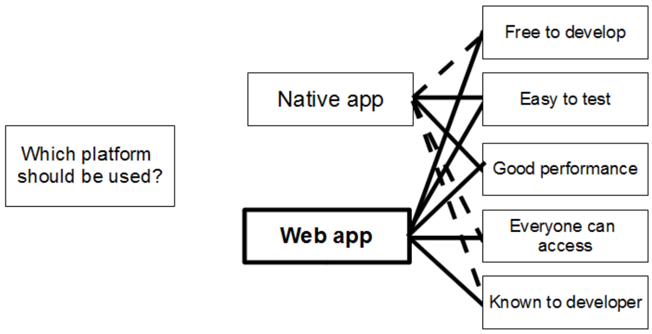


Figure 2.6: Example of a QOC-diagram for comparing platforms

Furthermore, MacLean et al. (1991b) states there are a variety of instances and issues of *justification*, the argumentation used to evaluate design alternatives. The QOC- notation can be extended with notation for such justifying arguments.

It is possible to justify by appealing to *empirical data*, *accepted theory*, or both (MacLean et al., 1991a). Certain elements may apply to certain existing work which supports a similar argument. Usually, no existing theory or relevant data will be available. The designers will have to construct an approximate explanation by formulating an ad hoc theory or collecting some "quick and dirty" data in order to produce a convincing Design Space Analysis.

An *analogy, model, or metaphor*, based on something outside of the design itself, is another form of justification. This use of analogy makes the technology transparent by exploiting the user's familiarity with other situations. Analogies used as justification draw attention to the parallel structures that are successful in the analogical domain. Ad hoc analogies may justify an Option by claiming similarity with some other domain.

They may also prove useful in helping eventual users learn about the system (MacLean et al., 1991a).

A *scenario* involves envisioning what it would be like to use the artifact being designed. They generate a context of use that emphasises variables not apparent from a static description of the artifact. Scenarios can justify a design in two ways. A scenario can justify a particular solution by demonstrating that an envisaged mode of use will work. Also, a scenario can evoke new Criteria that the design should meet. This is most likely to happen when a scenario shows flaws in a proposed solution (MacLean et al., 1991b).

2.5 Summary

This chapter has presented research in the field in technology acceptance by presenting two popular models used to explain user acceptance of IT, namely the Technology Acceptance Model (TAM) and the Unified Theory of Acceptance and Use of Technology (UTAUT). Design Rationale and the Lazy User Theory was explained. In addition, relevant research within Human-Computer Interaction (HCI) was presented.

3 Research Methodology

The thesis intends to achieve a greater understanding of how players and coaches communicate. It does so through surveying the current situation, by developing, testing and evaluating a prototype, and by presenting a set of guidelines aimed to aid development of similar information technologies. Several established research methods are employed in order to achieve this understanding, and to answer the research questions. This chapter presents an overview of which methods were used, and reason why they were suitable in this context.

A general description of the methodology is given through explanation of the concept of *Design-oriented Research*. *Design Science*, a framework for information systems research, is introduced. In addition, it explains the choice of *Design Rationale*, the *system development* methods, methods of *data gathering*, and how this data will be *analysed*.

3.1 Design-oriented Research

The research carried out in the thesis can be categorised as Design-oriented Research, a term introduced by Fallman (2003). Design-oriented Research should have knowledge of some sort as its main contribution, such knowledge that most likely would not be obtainable if design and the development of a prototype were not a vital part of the research process. Fallman (2003) compares this activity with other types of research: "In some ways, this resembles the way natural scientists may only be able to test a theory by first designing the tools with which to study the proposed phenomena" (p.231). Furthermore, while the resulting artifact is considered more a means than an end, the knowledge that comes from studying the artifact in use or from the process of bringing the product into being is considered the main contribution. The notion of Design-oriented Research is in contrast with *Research-oriented Design*, where the artifact is the primary outcome, which is a term believed to better illustrate the relationship that consultants, applied researchers, and designers from industry typically hold in relation to design in

HCI (Fällman, 2004).

The knowledge generated from the thesis research is a result of a Design-oriented approach to the research. While the knowledge is the main contribution, this knowledge could not have been obtained without the development and use of a design artifact.

3.2 Design Science

According to Hevner et al. (2004), two paradigms characterise much of the research in the information systems discipline: behavioral science and design science. While behavioral science is concerned with theories explaining or predicting human phenomena in relation to interaction with Information Systems, design science seek to extend the boundaries of human and organisational capabilities by creating new and innovative artifacts. In their widely cited paper, Hevner et al. (2004) propose a complementary use of the paradigms in informations systems research.

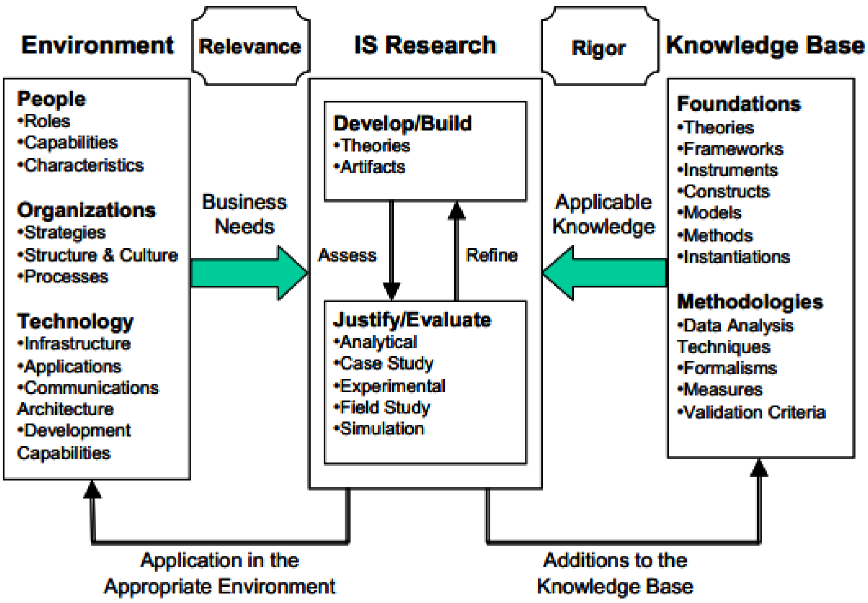


Figure 3.1: Framework for IS Research (Hevner et al., 2004)

Figure 3.1 above presents the framework for information systems research proposed by Hevner et al. (2004). The framework is particularly well suited for the research context, as the thesis context is a particular environment where the people, organisational structures, processes and existing technology or other factors is combined to form a need, or a "problem". Given a defined need, behavioral science addresses research through the development and justification of theories that explain or predict phenomena related to the identified need (Hevner et al., 2004). Hevner et al. (2004) also claim that technology and behaviour are inseparable in Informations Systems research, and suggest that researchers should *"engage the complementary research cycle between design science and behavioral science to address fundamental problems faced in the productive application of information technology"* (p.77). In addition to the behavioral theories developed particularly for this context, the thesis relies on technology acceptance literature, a field of research rooted in behavioral science.

Design science research engages in the activities of building and evaluating (March and Smith, 1995) artifacts that meet the established need. The knowledge base, containing knowledge attained by prior research and results from reference disciplines, provides foundational theories, frameworks, methodologies, etc. used in the build and evaluate phases (Hevner et al., 2004).

In addition to the framework, Hevner et al. (2004) propose seven guidelines in order to *"assist researchers, reviewers, editors and readers to understand the requirements for effective design science research"* (p.82). Table 3.1 below summarise these guidelines.

Guideline	Description
Design as an Artifact	Design-science research must produce a viable artifact in the form of a construct, model, a method, or an instantiation.
Problem Relevance	The objective of design-science research is to develop technology-based solutions to important and relevant business problems.
Design Evaluation	The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well-executed evaluation methods.
Research Contributions	Effective design-science research must provide clear and verifiable contributions in the areas of the design artifact, design foundations, and/or design methodologies
Research Rigor	Design-science research relies upon the application of rigorous methods in both the construction and evaluation of the design artifact.
Design as a Search Process	The search for an effective artifact requires utilising available means to reach desired ends while satisfying laws in the problem environment.
Communication of Research	Design-science research must be presented effectively both to technology-oriented as well as management-oriented audiences.

Table 3.1: Design Science Research Guidelines (Hevner et al., 2004)

Following a point made by Klein and Myers (1999, referenced in Hevner et al., 2004), judgement and discretion must be exercised in deciding whether, how, and which of these principles should be applied and appropriated in any given research project. Therefore, in this context the guidelines are used as helpful insights into the conduct of design science research rather than methodical steps requiring rigorous execution.

3.3 Development

Several methods in information systems research were employed during the development of the prototype. The methods concerned design process, design problem-solving and software development. These methods will be introduced further below.

3.3.1 Questions, Options and Criteria

Design Space Analysis, introduced in section 2.4.1, was used in the development process for suggestions of problem solutions. The QOC- notation was suitable to find potential solutions to a wide array of design problems, from high-level issues such as choice of development platform to problems on a low level in the development stack such as small details in the user interface.

Design problems solved with the QOC- method in this study do not contain explicit justification in the QOC- model. Despite the lack of explicit notation, awareness of justification was helpful in finding new, sound Criteria and subsequently the solutions to problems where these criteria were present.

3.3.2 Sketching

A helpful skill for interface designers and developers is to be able to rapidly sketch objects without paying too much attention to inappropriate details (Landay and Myers, 2001). Fallman (2003) also recognises the valuable role of sketching in design. The traditional view of sketching is as a way of externalising images already present in the mind of the designer, making it useful for communicating with other designers and customers as it provides a shared language that has no equivalent in ordinary, spoken language, but which allows designers to express themselves.

According to Fallman (2003), however, sketching should not merely be seen as a tool for communication. He underlines the importance of realising that sketching

> *"is not only a matter of one-directional externalisation from the mind of the designer onto paper: sketching is also reading and interpreting the sketch, explaining it and eventually rephrasing it. Sketching is hence a matter of externalising ideas and interpreting external representations as ideas (...) Sketching is hence not simply an externalisation of ideas already in the designer's mind, but on the contrary a way of shaping new ideas"* (p.230).

Sketching on paper was used extensively, particularly during the late phase of development, where attention was pointed towards user interface design. Most of the sketches were small and only concerned a certain element in the user interface, such as a menu bar or a table on a particular page.

3.3.3 System Development

Before commencing development of an information system, choosing a software development method is helpful. Core practices of the highly adaptive Kanban method as well as select agile practices were used to support development, combined in a way that suited the lone developer's situation. These practices, combined with the design science framework, formed the methodological platform for system development in the thesis. Such a multi-dimensional, multi-methodological approach complies with Nunamaker and Chen (1990) who propose that such an approach will yield good results in information systems research.

As mentioned above, core practices of Kanban were used in the development.

Kanban is a process tool that helps you work more effectively by, to a certain extent, telling you what to do (Kniberg and Skarin, 2010). Kanban comprises six core practices (Anderson, 2010). *Visualise the workflow* involves splitting the work into pieces, writing each item on a card and put on a wall. On this wall, named columns are used to illustrate where each item is in the workflow (Kniberg and Skarin, 2010). Figure 3.2 below illustrates an example wall.

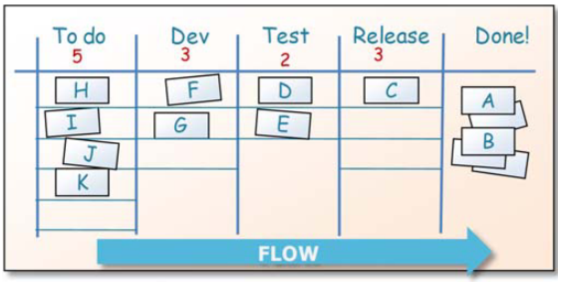


Figure 3.2: Kanban wall (Kniberg and Skarin, 2010)

The second practice is to *limit work in progress* by assigning explicit limits to how many items may be in progress at each workflow state. Third, by *managing flow*, the continuous, incremental, and evolutionary changes to the system and the positive or negative effects they cause can be evaluated. In addition, Anderson (2010) mentions *making policies explicit*, *implementing feedback loops* and *improving collaboratively, evolving experimentally* as the latter three of the core practices. However, only the three former practices were consciously used in the development of the prototype. As Kanban is not supposed to tell you everything that you need to do, but provide certain constraints and guidelines (Kniberg and Skarin, 2010), it was suitable for a solo developer on a project where it was hard to estimate how long each task would take to complete.

In addition, some established agile concepts and practices were incorporated into the software development method. The Scrum artifact *product backlog*, *"an ordered list of everything that might be needed in the product and is the single source of requirements for any changes to be made to the product"* (Schwaber and Sutherland, 2011, p.12), and was used on the virtual wall. In addition, the concept of *user stories* was used. User stories describe functionality that will be valuable to either a user or purchaser of a system or software (Cohn, 2004). User stories were used to represent tasks that had to be done during development, but were not used to its full extent as documented by for example Cohn (2004).

Also, the development was carried out iteratively and incrementally. Iterative and incremental development (IID) is applied by all agile methods (Larman and Basili, 2003), and appears to be a requirement for a method to be agile. Finally, an agile developer may once in a while require some time to investigate the unknown elements of a user story, for example a new technology, to be able to estimate the user story (Philippus, 2009). This activity is called a *spike*. In order to be familiar with new technologies when development commenced, an initial spike was conducted prior to the start of development.

3.4 Formative Data Gathering

Based on the purpose of the evaluation, Scriven (1967, referenced in Hartson et al., 2001) distinguishes between two basic approaches to evaluation: formative and summative. The concept of summative evaluation will be introduced later. Formative evaluation is carried out during development to improve a design and is used to find usability problems to fix so that an interaction design can be improved. This type of evaluation covers a broad span of design from early development to perfecting an almost finished design, providing information throughout the implementation process to help improve the means, or process, to accomplish objectives and aid interpretation of summative evaluation results (Hamilton and Chervany, 1981).

Not all techniques, however, that may aid a design before its completion are evaluative. Hence, labeling the methods used for this purpose *formative data gathering* is more appropriate than the original term. In an attempt to capture some initial requirements for the information system to be designed, and to survey the user group and their current situation and habits, questionnaires were used. During development, a heuristic evaluation was conducted in order to detect usability issues to be remedied and to consult experts for new ideas.

3.4.1 Questionnaires

Questionnaires are a technique for collecting quantitative demographic data and user opinions (Rogers et al., 2011). Questionnaires are useful for obtaining answers to specific questions from a large group of people, especially if this group is geographically scattered. Bryman (2012) refers to this technique as *self-completion questionnaire* or *self-administered questionnaire*, as it is completed by respondents themselves. Considering the researcher or interviewer is not present to resolve ambiguities or misunderstandings, clearly worded questions are important (Rogers et al., 2011).

Questionnaires being administered online, henceforth labeled *web surveys*, are becoming increasingly popular (Rogers et al., 2011; Bryman, 2012). They operate by inviting prospective respondents to visit a website at which the questionnaire can be found and completed online (Bryman, 2012). Benefits of a web survey include its ability to reach large numbers of people quickly and easily as well as provide immediate data-validation and faster response rates (Rogers et al., 2011).

To aid the identification of requirements for the information system to be designed and to survey current user habits, a web survey was created. This is in accord with Rogers et al. (2011), who claims that using web-based questionnaires when trying to capture requirements can be beneficial, for example by asking the target population about impressions and opinions about current tools and services. It may also attempt to gather more specific suggestions for the desired kind of tool or service.

3.4.2 Heuristic Evaluation

Although Nielsen's ten heuristics, presented in section 2.3.2, are undoubtedly helpful for a heuristic evaluation "out of the box", Rogers et al. (2011)) point out that some of these heuristics are too general for evaluating certain products, and that evaluators and researchers have typically developed their own heuristics by merging Nielsen's heuristics with other design guidelines, market research, and requirements documents for the specific product. A typical set of heuristics consists of five to ten items, as more than ten becomes difficult for evaluators to remember and fewer than five tends to not be sufficiently discriminating.

Following a careful assessment of this particular case, five principles were chosen as heuristics to guide the experts in their evaluation of the prototype that was designed. These principles were deemed most relevant, aided by industry professionals suggesting heuristics in a web context (Budd, 2007; Scott and Neil, 2009). The heuristics were the following:

1. Visibility of system status
2. Consistency and standards and match between system and the real world
3. Error prevention
4. Aesthetic and minimalist design
5. Help users recognise, diagnose, and recover from errors

A key, frequently asked question is how many evaluators are required to carry out a thorough heuristic evaluation (Rogers et al., 2011). Nielsen and Molich (1990) recommended using three to five evaluators. This recommendation was validated by Jeffries et al. (1991), who after using four evaluators in a comparison of different evaluation techniques, including usability testing, stated that: *"Overall, the heuristic evaluation technique as applied here produced the best results. It found the most problems, including more of the most serious ones, than did any other technique, and at the lowest cost"* (p.119).

Choosing the number of evaluators, several factors were taken into account, including time available, expert availability, system complexity, and perceived degree of system completeness. It was decided that three evaluators would be sufficient. The three evaluators recruited were fellow students with adequate experience in Human-Computer Interaction and interaction design, an obvious choice considering their appropriate expertise and their practical location in the immediate vicinity of the researcher.

3.5 Summative Data Gathering

As opposed to formative evaluation introduced above, summative evaluation is evaluation carried out after development to assess an artifact and is used to assess or compare the level of usability achieved in an interaction design (Scriven, 1967, referenced in Hartson et al., 2001). It assesses the success of a finished project (Rogers et al., 2011), which is a prominent desire with regards to answering research questions.

As with formative evaluation, the term summative evaluation is here expanded to the more appropriate *summative data gathering*, as not all methods employed are evaluation methods. To assess the success of the finished prototype, a traditional usability test was performed. The prototype was published and three organised sports teams tested it for a month. Field data was collected to see if the prototype had helped reduce the number of deviations from the coach's assumptions. In addition, to gain a better understanding of how the users perceived the prototype, qualitative interviews were conducted with select users, and a questionnaire was issued to the around 50 users who tested the prototype.

3.5.1 Usability Testing

As mentioned in the introduction of the concept in section 2.3.4, conducting a validation test late in the development cycle is an acceptable practice. The usability test was carried out after the development activities had been concluded. In accordance with literature, it was conducted in a controlled environment (Rubin and Chisnell, 2008; Rogers et al., 2011; Shneiderman, 1997). Although researchers appear to agree that testing five users is sufficient (Nielsen, 2000; Dumas and Redish, 1999; Rubin and Chisnell, 2008; Rogers et al., 2011), a total of 9 people participated, 5 men and 4 women, at ages ranging from about 20 to about 50. The number of participants was high due to the prototype functionality differed for coaches and players, and hence required a sufficient number of participants to test both the coach and player interface.

The participants in the usability test was encouraged to think aloud. The *think-aloud* *method* is a method that requires participants to talk aloud while solving a problem or performing a task. Traditionally applied in psychological and educational research on cognitive processes, the method has been appropriated to the field of HCI (Jaspers et al., 2004):

> *"Thinking aloud is a method that requires subjects to talk aloud while solving a problem or performing a task. (...) the think aloud method is a unique source of information on cognitive processes: it generates direct data on the ongoing thought processes during task performance. Thus, the think aloud method is a very direct method to gain insight in the way humans solve problems. It may be therefore used to know more about cognitive processes and to build computer systems on the basis of these insights"* (p.783).

3.5.2 Field Studies

In addition to the methods of evaluation outlined above, field data from the user's environment can add to the validity of the collected data. As mentioned in section 2.3.3, it can also improve perceptions on how successful the product will be in the real world. In accordance with literature (Rogers et al., 2011; Dix et al., 1998), a field study was carried out in order to acquire such data. The field study lasted for roughly two months.

During these two months, coaches were asked to register attendance deviation data for one month without their team using the prototype and one month while their team used the prototype. This activity was an attempt to obtain a quantitative measure on whether or not the prototype improved the predictability of attendance. The deviation data activity is further explained in section 6.2

In addition to the field data reported by coaches, a questionnaire was distributed online to all the players and coaches who used the prototype. The concept of questionnaires was introduced in section 3.4.1 and used as a formative data gathering method. However, here it will also be used as a summative data gathering method, as it may also be used in an evaluation context (Rogers et al., 2011; Dix et al., 1998). In addition to being a valid method of summative data gathering, questionnaires have been used extensively in technology acceptance research to survey which factors cause people to accept or reject information technology (Davis, 1989; Yi and Hwang, 2003; Venkatesh et al., 2003), and is hence suitable in the research context.

Questions in the web survey were phrased carefully and made heavy use of *likert scales*. Likert scales are questions that ask the user to judge a specific statement on a numeric scale (Dix et al., 1998) and are widely used in questionnaires for evaluating user satisfaction (Rogers et al., 2011) and to measure constructs in technology acceptance research (Davis, 1989; Yi and Hwang, 2003; Venkatesh et al., 2003). It is important to carefully choose granularity of scales used in this type of questions. A very coarse scale may give no room for varying levels of agreement (Dix et al., 1998), while people cannot be expected to discern accurately among points on a large scale (Rogers et al., 2011). Following advice from HCI literature, scales of 1 to 3 will be used in yes/no type answers, while 1 to 5 or 1 to 7 is an acceptable middle ground that have been used effectively in questions where respondents are asked to make judgments that involve like/dislike, agree/disagree statements (Dix et al., 1998; Rogers et al., 2011; Shneiderman, 1997).

3.5.3 Qualitative Interviews

Bryman (2012) claims that the interview is probably the most employed method in qualitative research. There are two major types of interviews in qualitative research, the *unstructured interview* and the *semi-structured interview*. While the unstructured interview is very exploratory, often goes into considerable depth and generally tends to be similar to a conversation (Rogers et al., 2011; Bryman, 2012), the semi-structured interview is slightly more rigid. Here, the researcher has a list of questions or fairly specific topics to be covered, but allows unplanned follow-up questions and for the conversation to "go where it may" (Lazar et al., 2010).

The semi-structured approach to interviews will be used with select users in an attempt to extract information that may enrich the answers of the research questions. When conducting an interview as a novice interviewer as is the case in this research, Roulston and Lewis (2003) identified several possible challenges for interviewers. The most prominent of them in this context is researcher bias and subjectivity. Participants in Roulston and Lewis (2003) recognised their own beliefs and subjectivities in the phrasing of questions and were surprised how evident their assumptions were. As the researcher is part of the user group, it is imperative to remain objective and not let biases and expectations determine the way questions are asked and replies are followed up.

Complex, in-depth data about how the users perceived the prototype is not easily obtained through questionnaires or usability tests. In order to complement the already gathered information with such in-depth data, it was decided that, as part of the field study, qualitative interviews should be held with participants from the different teams that used the prototype. It was found practical and sufficient to hold only one interview, mainly because the selected interviewee had served as coach for one team and a player for another team, which ensured the interviewee had rich experience from using the prototype, and could reflect on the use of it from both a player and a coach perspective. The interview questions was generated after looking at the other data that was gathered, and identifying what merited in-depth explanation. In the end, 9 questions were prepared for a semi-structured interview. The interview guide is found in appendix D. The interview was held in the interviewee's home, with no observers present.

3.6 Data Analysis

Whether the data collected during different kinds of evaluation is quantitative or qualitative, it needs to be analysed properly in order to provide strength to the answers to the research questions.

Quantitative data, that is data that is in the form of numbers, or that can easily be translated into numbers (Rogers et al., 2011), are often measured in *averages* and *percentages*. These two well known numerical measures, especially percentages, will most likely be used heavily in analysing the quantitative data collected. Regarding qualitative analysis, it is not as straightforward to analyse as quantitative data, but some simple techniques suggested by Rogers et al. (2011) will to different extents be employed to extract usable information from the data. *Identifying recurring patterns or themes* involves keeping clear and consistent records of what has been found, and closely describing patterns and themes that emerge. *Categorising data* is another way of analysing data. A categorisation scheme can be created, and the data collected can be categorised in accordance with this scheme. Categorised data can subsequently be used to answer the research questions. As analysing all qualitative data recorded may be infeasible, one possible solution to this problem is to *look for critical incidents*. This means identifying specific incidents that are significant or pivotal to the activity being observed, in either a desirable or an undesirable way.

An oft-discussed issue in methodology research, is how researchers can determine the extent to which a particular empirical indicator, such as a test or experiment, represents a given theoretical concept (Carmines and Zeller, 1979). Carmines and Zeller (1979) name *reliability* and *validity* as the two most basic concepts of empirical measurement. Reliability concerns the extent to which an experiment, test, or any measuring procedure yields the same result on repeated trials, while validity concerns if the method employed actually measures what it is intended to measure. The concept of reliability is considered most relevant for quantitative research, as ensuring validity in qualitative research subsequently ensures reliability (Seale, 1999). A frequent subject of discussion is these constructs' role in qualitative research. In their influential book, Lincoln and Guba (2001) name a set of criteria to ensure good quality qualitative research. The first criteria is *credibility*, concerning whether the data and method used to gather it is believable. Second, ensuring *transferability* gives readers sufficient information to be able to judge the applicability of findings to other settings that they know (Seale, 1999). *Dependability* is essentially a replacement of reliability and the previous definition of the concept is also suitable here. The final criterion, *confirmability*, is equivalent of a classic criterion of objectivity or neutrality, which concerns if the findings are neutral and the researcher unbiased.

Seale (1999) argues that if there is one thing that produces poor studies, it is a researcher who is blind to the methodological consequences of research decisions. *Methodological awareness* involves an enhanced capacity to anticipate a broad range of potential criticisms that may be made of a final research report, and can be achieved through almost any intelligent methodological discussion, as well as from critical reading of existing research studies. In the context of this research, merely exercising methodological awareness is deemed sufficient to maintain research rigour and ensuring valid research.

3.7 Summary

This Design-oriented Research uses the Design Science framework by building and evaluating an artifact. It uses sketching to convey design ideas and the QOC- method to rationalise design decisions. Core practices of Kanban and select agile methods were used in system development. Questionnaires and heuristic evaluation were used to improve a design and find usability problems. In the evaluation phase, usability testing and an extensive field study was conducted. This field study included recording attendance data in teams as well as a web survey and a qualitative interview. Figure 3.3 summarises the development.

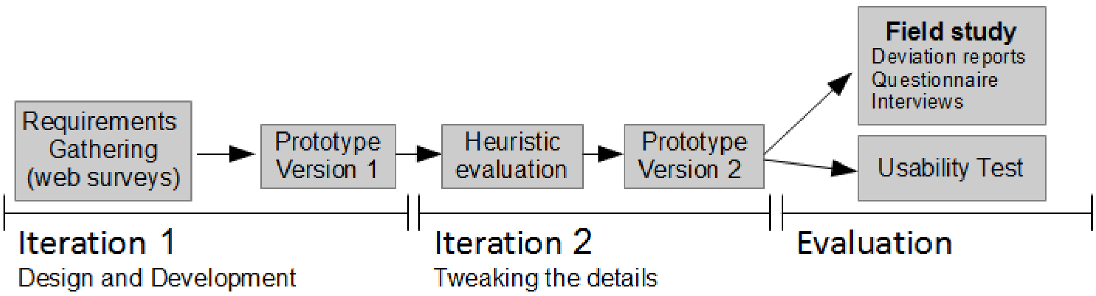


Figure 3.3: Overview of methodology

4 First Iteration - Design and Development

This chapter documents and describes the process of developing and designing the initial prototype of the web application. The vast majority of development work was done during this phase of development, which lasted for five months.

In addition to the design choices and why they were made, an array of aspects are elucidated, including the users' perceived situation before the prototype, sketching and defining requirements, as well as a more technical description of the prototype.

4.1 Surveying Current Situation

Experiences acquired through strong affiliation with various organised sports environments over the last 15 years highlight several important points about the current communication situation between players and coaches. Below, some of the most prominent perceived issues are mentioned.

Frequently, players on organised sports teams do not show up for the planned activity without having informed the team coach in advance. This can be difficult for coaches to handle as the absence of one or more players may greatly affect the plan for that particular practice or match. Over time, this element of instability from players require an enormous amount of extra effort for coaches in keeping track of who is attending activities.

Experiences also indicate that few teams have a strategy to maximise voluntary communication from players to coach. It seems that without applying pressure on them, players would rather refrain from communicating intended absence to coaches.

Also, it is perceived that few teams use only one common service as an established standard to communicate and handle intended absence. Currently, organised sports teams utilise a wide variety of communication technologies to keep the attendance situation predictable. Among the technologies that are perceived as the most commonly used are SMS, emails, and various means of communication on the social networking service Facebook, e.g through the integrated instant messaging system, or through posting on a team group's wall. This mix of technologies is disorganised and hard to administer for coaches.

The experiences outlined above can be summarised:

- Players frequently fail to attend without notifying the coach in advance.
- Most teams lack a strategy for maximising voluntary communication from player to coach. If given the choice, players would prefer to not communicate intended absence.
- Few teams have an established standard to communicate and handle intended absence, and rather use a mix of technologies, which is hard to administer.

These points, however, are merely educated guesses based on biased perceptions and observations made over several years. They require empirical confirmation by real players and coaches in order to be valid in research.

4.1.1 Web Surveys

A web survey, a data gathering technique introduced in section 3.4.1, was considered the easiest and most feasible way to gather as many responses as possible from a demographically wide and varied group of people as possible in order to investigate the current situation. Simultaneously, the survey could be used to gain a greater understanding of the user groups needs for the prototype.

Two separate web questionnaires (one for players, one for coaches and team leaders) were created. In total, there were 10 questions in the player questionnaire, with 9 of these being required questions. In the coach questionnaire, there were a total of 14 questions, where 10 were required. Relatively few questions were included in accordance with Bryman (2012), who claim that short questionnaires tend to get higher response rates. The first questions were standard demographic questions such as gender and age, followed by questions regarding the player or coach's perception of current tools and user habits. The experiences outlined above were used as basis for these questions, and strongly influenced the survey altogether. Additional questions attempted to gather more specific suggestions for functionality in a potential prototype. This was done both by asking participants to suggest their own features, and by proposing a set of features and let coaches choose which are desirable in a potential prototype. The proposed features originated from now outdated systems successfully used in the past as well as general observations and experience. Finally, one question in each questionnaire asked which technological platform are most commonly used by people in the target group.

The two questionnaires were posted on several websites and social media, most prominently Facebook. They were also sent by e-mail to a substantial number of receivers in the target group. The surveys were first posted and published on September 13th, 2013. Three weeks later, 287 responses had been submitted. Of these responses, 210 were from players and 77 from coaches. This number was satisfactory, especially the number of coaches taking the time to reflect on the issue and provide additional thoughts was very positive. The number of responses was assessed as a valid sample of the user group, and provided a good foundation for further development. The two web questionnaires can be found in its entirety in appendix A.

4.1.2 Web Survey Results

The two web surveys provided some valuable and interesting data. Perhaps the most interesting was that 42% of the players admitted that they have been absent from an activity without properly notifying the coach. Of these, 34 respondents reported it had happened that they had consciously avoided notifying the coach, either to avoid a conflict with the coach or because it is cumbersome to notify the coach every time they will be absent, while 48% had simply forgotten to notify the coach.

Facebook and SMS stand out as the most popular channels to convey messages of intended absence, being used by a combined 94%. The dominance of these two technologies was further confirmed by the coaches. Out of the 54 respondents who stated that their team used digital tools to aid communication, 30 used SMS and 42 used Facebook (the questions allowed multiple choices).

A significant number of coaches mentioned specifically that without a predictable situation, planning training sessions is difficult. Results showed that coaches consider the attendance situation surprisingly clear and predictable, with an average of 3.35 where 5 was the most predictable. However, when asked directly, 75% of the coaches consider it a problem that players do not notify them about intended absence, and 77% of coaches and 86% of players answered that a specialised tool for handling attendance is necessary for their team. This may indicate that coaches do in fact manage this problem relatively well, but that a specialised tool may simplify the administrative tasks and subsequently reduce the workload involved with coaching a team.

The initial system features proposed as alternatives in the survey were about equally desired. The respondents had the opportunity to choose multiple options. Notably, only 4% of the coaches participating in the survey currently use digital tools to keep statistics on activity attendance, while 64% considered such a system feature useful and/or necessary. As a combined 83% of all the participants considered seeing a list of attendees for an activity beforehand as useful and/or necessary, this appears to be the most important system feature to implement. In addition, a vast array of additional features were suggested in the free text field, most notably and frequent a feature enabling the coach to send short, customised messages to the players as a group or individuals. This corresponds with the fact that 57% of the coaches already use digital tools to communicate messages to the players, such as change in the start time or location of an activity.

Regarding devices, the player results also showed that the players are likely to use the prototype on all devices, with 67% most likely to use it on mobile devices and 33% on desktop computers. Among coaches, 52% of the participants mainly use a desktop computer when browsing the web, while 39% use a mobile phone. From these results, it can be concluded that the prototype must be consciously developed to be user friendly on all platforms, but that user friendliness for some of the administrative tasks for coaches may be prioritised lower as desktop computers are commonly used by coaches.

In summary, many players have been absent from team activity without notifying the coach in advance. Facebook and SMS are the most popular channels of communication. A vast majority felt a specialised communication tool was necessary, and felt seeing a list of attendees to an event was the most desirable feature. A potential prototype is required to work on all devices, as all types of devices are used by participants.

4.2 Designing the Prototype

In this section, the prototype is introduced. First, the technologies used in building the prototype is described. Furthermore, a general description of the prototyp is provided. Respectively, the back-end and front-end development process is documented, and specific design decisions made during these processes are explained and reasoned for.

4.2.1 Technological Choices

Prior to the creation of the survey it was decided that if the survey would document the need for a new information technology to accommodate the problems, the prototype of this technology would have to be made in the form of a web application. A number of reasons influenced this decision:

1. Developer experience.
2. Time constraints on the project.
3. Inclusion of users.

As a result, the questions in the web survey assumes that the new information technology will be a web application, leaving out questions or options for the respondents to choose their preferred platform for such a system. This decision is rationalised in a QOC-notation found in appendix B.

PHP

The first important choice that had to be made, was which programming language with which to build the web application. Although languages such as *ASP*, *JSP*, *C* and varieties of the latter are all frequently used for development on the web, the server-side scripting language *PHP*[1] was chosen because this language had been used in a similar web development project in the past, and was hence familiar. In January 2013, PHP was used on about 240 million Internet sites (Ide, 2013), which shows its suitability for web development. Among the known benefits of using PHP is that it is regarded as easy to learn, free, performance efficient and well documented with a large community, the latter being of utmost importance in this context.

CodeIgniter

To support the development of web applications, it is common to use a web application framework. Such frameworks may provide libraries for commonly used functionality, such as database access, session management and form validation. In addition, it promotes code reuse. A plethora of web application frameworks are available for use with PHP. *CodeIgniter*[2] was chosen as the framework to be used in the project, for a number of reasons. In addition to being fast, lightweight and well documented, CodeIgniter uses a model-view-controller (MVC) architectural pattern as the basis for programming. As shown in figure 4.1 below, the MVC pattern separates the job of different classes into

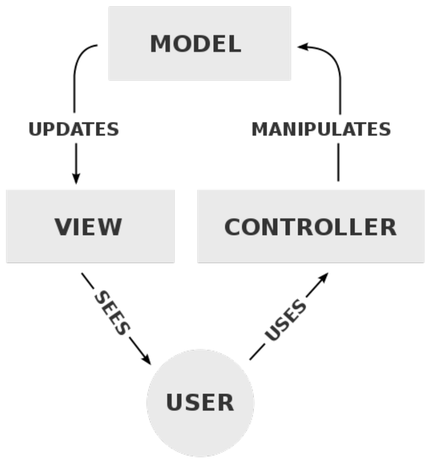


Figure 4.1: The MVC software architectural pattern

[1]www.php.net
[2]www.ellislab.com/codeigniter

clear and well-defined tasks, which in turn allows for more flexibility in the communication between classes. The model handles the data sources such as database data, the view is the presentation layer, the user interface, while the controller mediates requests between views and models and defines the business logic. The decisive factor for using CodeIgniter, however, was that it had been used in a project in the past, and was hence familiar.

MySQL

To handle the database that would be an integral part of the web application, the open-source relational database management system *MySQL*[3] was chosen due to its suitability for use in web applications. MySQL is considered easy to use in combination with PHP and CodeIgniter by utilising *XAMPP*, a widely used free and open-source cross-platform web server solution stack package. A great benefit of using MySQL is that it can be administered by *phpMyAdmin*, which is a free and open-source tool which provides users with a Graphical User Interface in which they can perform many of the most common tasks in database management such as creating, modifying or deleting databases, tables, fields or rows.

HTML

Since the application will be running on the Web, it is unavoidable that it uses *HyperText Markup Language* (HTML). HTML is the World Wide Web's markup language which structures and presents content on the Internet. It was created in 1990 and has been revised four times. HTML5 is the fifth revision, and this revision is currently at the candidate recommendation stage in the World Wide Web Consortium (W3C) Standard Formation Process. HTML5 accommodates modern websites by providing support for the latest multimedia, as well as enriching semantic content of documents.

CSS

Closely linked to HTML is *Cascading Style Sheets* (CSS), a styling language used to define the look and formatting of a web document. The style sheets created in CSS

[3] www.mysql.com

describe what layout, colours, fonts and other visual elements are presented on a web page. By providing a link to the CSS file in a HTML file, a particular design can be applied to an unlimited number of pages. Subsequently, if a change is made to a CSS file, these changes will occur on all the pages that use the CSS file. Hence, a developer can easily change the look of a web site. The CSS specifications are maintained by the W3C. The current version of CSS is version 3, or CSS3, and was developed by W3C's CSS Working Group.

JavaScript

Extending the utility of HTML5 and CSS3 on the web is *JavaScript*. This language is commonly used on the client-side on the web to interact with users. Examples of functionality that JavaScript typically provide, are page loading without reloading it, validating input in forms and animation of page elements. When using JavaScript for web development purposes, it is common to utilise a JavaScript library. Such a library simplifies the code with more efficient, shorter and more intuitive syntax. Although *Prototype* and *MooTools* are examples of good JavaScript libraries, the most popular library on the web, *jQuery*, was chosen for development.

Bootstrap

To reduce front-end development time, the framework *Bootstrap*[4] was used. Bootstrap is a collection of design templates enabling quick creation of websites and web applications. It is HTML and CSS-based and comprises templates for numerous interface components, as well as optional JavaScript extensions. The release of version 2 of Bootstrap marked the introduction of a grid system encouraging web developers to practice responsive design of web pages. *Responsive Web Design* is a term coined by Ethan Marcotte (2010), and the idea of his concept is that web pages should adapt to the device that the page is being viewed from, instead of redirecting a user to a different site for each different device used. In version 3 of Bootstrap, this grid system had evolved into a *mobile-first* system. Designing mobile-first involves developing sites first for mobile, then scaling up to suit

[4] www.getbootstrap.com

larger resolutions and sizes. Arguing for mobile-first development, Wroblewski (2012) pointed to predictions that mobile phones would overtake PCs as the most common web access device by 2013.

4.2.2 Prototype Description

The prototype, unceremoniously named rockEnroll, encompassed several features designed to aid communication between players and coaches. Below, the final features are listed, separated by user role.

Features for coaches:

- Create teams. Can be listed as coach of up to three teams at a time.
- Add recurring or standalone events, displayed in a calendar. Each occurrence of an event is called an episode.
- Edit and delete overlying events consisting of several episodes, or single episodes of overlying events.
- View statistics on attendance for a chosen period of time.
- See a list of attendees for each episode.
- Add and remove players and other coaches from teams.
- Edit team details.
- Change user name and email.

Features for players:

- Join up to three teams.
- See a list of attendees for each episode.
- Set their attendance status to attending or not attending.
- Change their name and email.

4.2.3 System Requirements and Sketching

The web survey results provides a basis for a formal requirement specification. Several of these requirements also occurred or were modified during the sketching process described below. The requirements are divided into two traditional categories, functional requirements and non-functional requirements, an approach recommended by Rogers et al. (2011). Functional requirements say what the system should do, while non-functional requirements concern constraints on the system and on the development process.

Functional requirements

1. Users can register either as a coach or a player.

2. Coaches can create teams, which players can join.

3. Coaches can create both recurring and one-time events/activities for a team.

4. Coaches and players can see these events, with which they can interact. Players can sign up for activities and hence make coaches aware of intended absence through this view.

5. Coaches can see continously updated statistics on player attendance. They can also track attendance from a specific period of time.

6. The system shall provide functionality enabling coaches to send short messages to all or select players in a team which the receiving players are made aware of through visible notifications.

Non-functional requirements

1. The system should be easy to use, even for users with a minimum level of user competence.

2. The system should be lightweight, allowing short loading time and low data use on mobile devices.

3. It should be easy to perform system tasks on all computers, tablets and mobile phones.

4. The system should be extensible, with possibilities for further development.

As mentioned above, the web survey generated interesting results regarding eventual requirements for the system. Although the three features proposed in the web survey were embraced by coaches and a plethora of suggestions for additional features were made, it was still not a straightforward task to assess which features were most important for the success of the system, and which features were feasible to develop taking the time available and the knowledge of the developer into account. Although it was used at several stages during development, sketching was particularly helpful in this stage of the development process.

Low-fidelity sketches on paper not only generated ideas that were useful when designing the user interface, but also contributed to the transparency of what core, back-end functionality was necessary for a certain design solution to work. This helped exclude unfeasible solutions that would not be realistic to achieve during such a limited period of development, or would take too much time measured against the importance and usefulness of the feature in the system. The result of the sketching sessions were a more complete list of functional requirements, as well as a starting point from which the design was evolved and improved. Figure 4.2 below shows examples of how an initial sketch was developed into a fully functional part of the prototype.

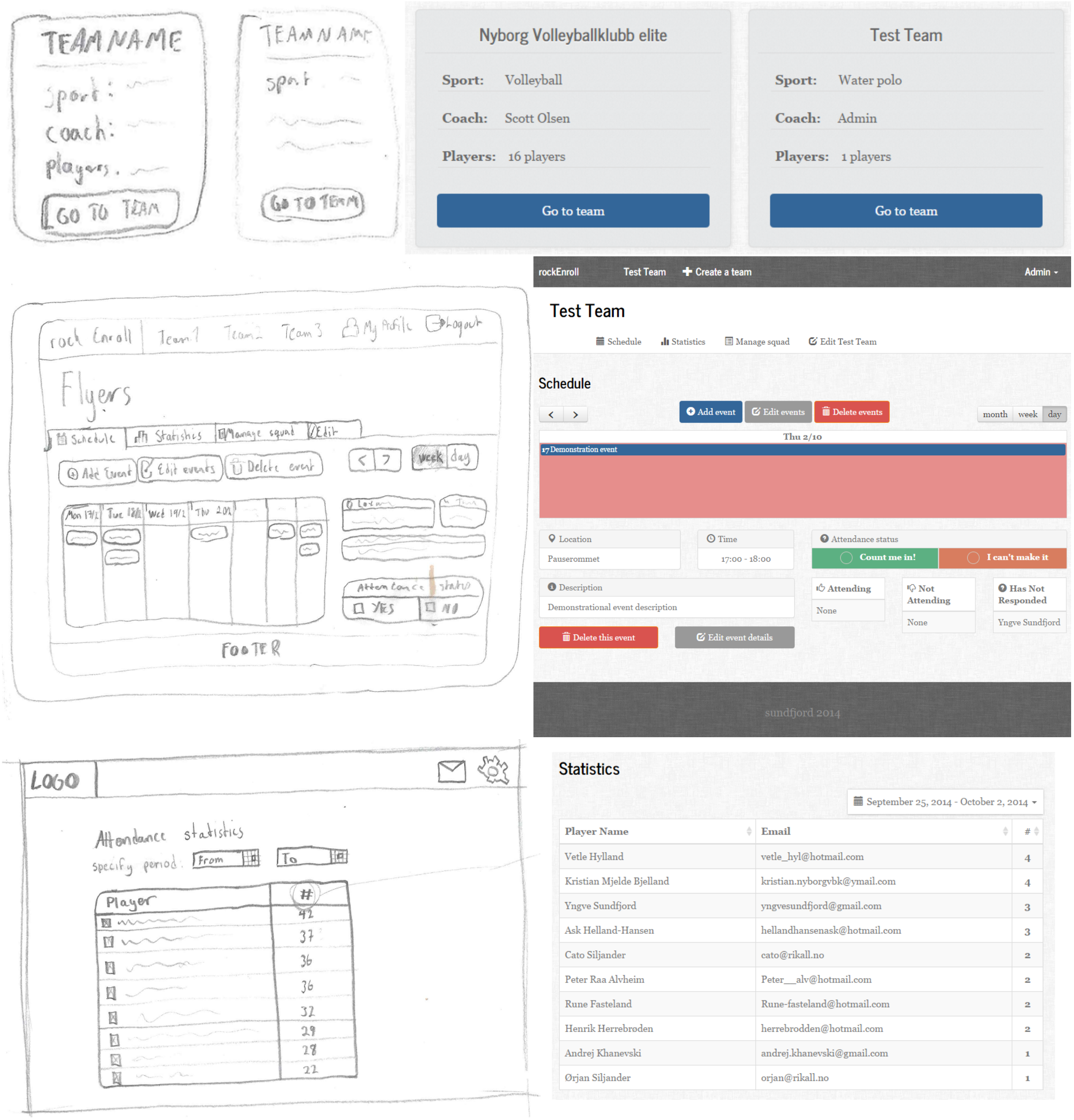


Figure 4.2: Paper sketches' impact on final design

The assessments that were made based on the survey results and during sketching, formed the basis of a relatively complete list of functional requirements. The written requirements were converted into user stories, and added to a backlog in the free, web-based project management application *Trello*.

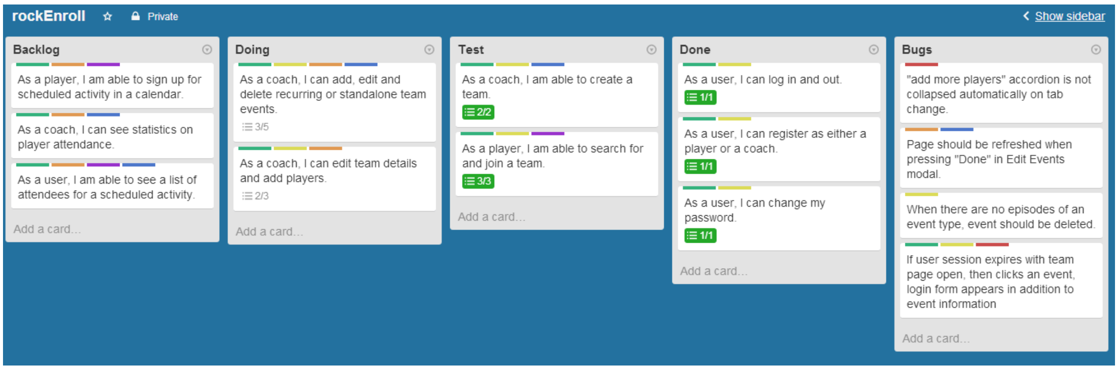


Figure 4.3: User stories organised in the project management application Trello

This management application was a suitable virtual alternative for conducting the Kanban practice of visualizing workflow, described in section 3.3.3. Once user stories had been made for all the proposed functionality, some of them were selected as tasks for the first increment. A choice was made to first focus on implementing and testing the back-end functionality first, and subsequently focus on designing an aesthetically pleasing, usable user interface. This made it easier to produce a more coherent visual theme for the application.

4.2.4 Back-end Development

A natural selection of tasks for the first increment was those related to login and registration functionality. As this is a very common feature for most web applications, it was relatively unproblematic to find an existing authentication library made for CodeIgniter, and use this as a starting point. The library *Tank Auth*[5] included common features such as password reset and password change. It did not support roles, however. The user story related to the functionality stated that users should be able to register either as a player or a coach. Hence, extension to the library was required. A field was added in the user database to distinguish players from coaches, and a simple check for which group they belonged to could easily determine which functionality and content the user

[5] http://konyukhov.com/soft/tank_auth/

was allowed to see.

The next increment involved providing users with coach privileges the ability to create a new team and editing the team details, as well as providing players with an option to join a team. This involved creating a table in the database for teams, and subsequently creating one junction table `is_coach_of` for the relationship between coach and team and one junction table `plays_for` for the relationship between player and team. When a coach created a team, a row was added in the junction table with the coach ID and the team ID to describe that that user was coach of that team. Similarly, entries in the `plays_for` table described that a player played for that team. Furthermore, coaches were given the ability to see a list of all players in the database, and add one or more to the team.

In the third increment, coaches were given the ability to add and edit team events. Players were given the ability to sign up for scheduled activity, and both user groups could see a list of attendees for these activities. The challenging aspect that made the functionality in this iteration the perhaps hardest to implement, was the fact that coaches should be able to add both recurring and standalone team events, as well as be able to edit and delete them. To support the non-functional requirement that the application had to be easy to use, it had to be easy for the coaches to edit details for both all events of a certain type, and one single event belonging to a certain type. To reflect this, a thoroughly planned database schema had to be developed. Once again, sketching was useful to visualise possible solutions, as seen below in figure 4.4.

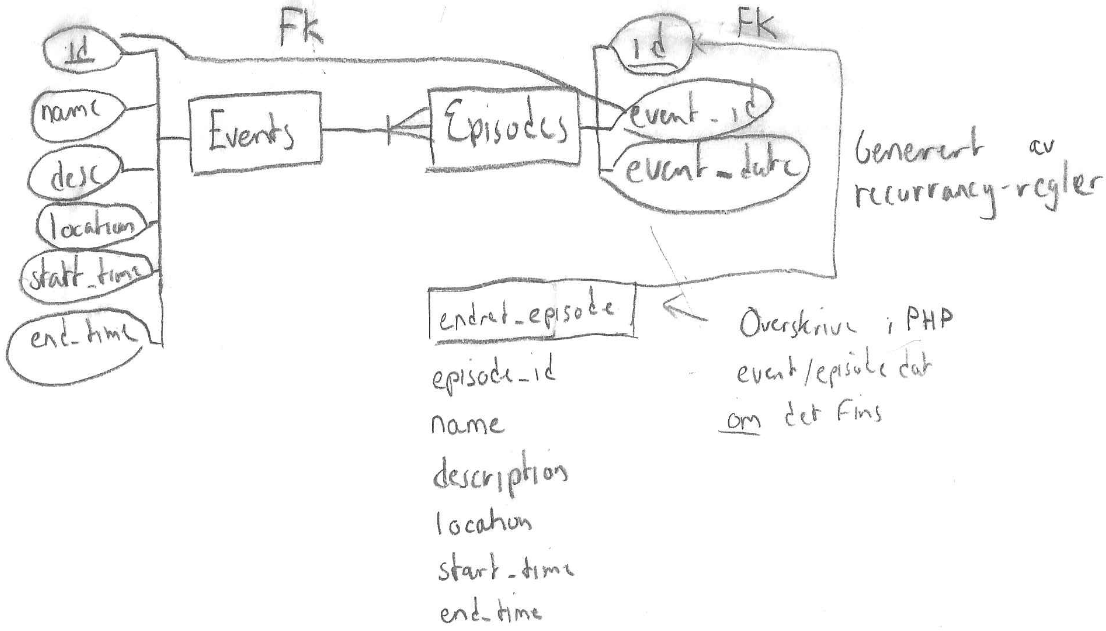


Figure 4.4: Database schema sketch

The final solution involved separating recurring and standalone events into two different concepts: Single events were termed *episodes* and the overlying information about each episode were termed *events*. Coaches could add an event with recurrence frequency daily, weekly or monthly, or a standalone episode. Regardless of frequency, the information about the event typed in by the coach would be saved in a table `events` in the database. A table `episodes` was made, and included a reference to `events`. Now, if a coach created for example an event with weekly occurrence for the next three weeks, three rows, one for each occurrence, would be created in the episode table. Recurrence logic was provided by utilizing a PHP Date Recursion library called *When*[6], and this ensured that correct dates would be calculated for each episode. Simultaneously, a row would be created in the events table with information about the event. The episodes table contained a field with the date of the episode, and a field that referenced the ID of the overlying event.

With this schema, coaches could edit all of the episodes of a certain event type, but also

[6]https://github.com/tplaner/When

edit the information about only one episode. If one episode was changed, information about it would be saved in fields in the episodes table similar to the ones in the events table, and subsequently overwrite the existing event information. If the coach changed the overlying event information again, the information about the specific episode would still overwrite this new information. This provided coaches with freedom to add, edit and delete events and episodes as flexibly and independent of each other as possible, while maintaining efficiency.

The second task in this increment was to provide players with the ability to sign up for the events. For this task, the `ID` fields of the user and episode database tables were used in a junction table named `attendance_status`. When an episode was created, a row was added in this table for each member of the team hosting the episode. In addition, a field `is_attending` was added, and this value could be 0, 1 or 2, where 0 corresponded to *has not responded*, 1 correspondend to *attending* and 2 corresponded to *not attending*. On user input the value would be changed. If players joined a team after team episodes had been created, the system would find all existing team episodes and add the player on the *has not responded* list. Similarly, if a player left a team, the player's attendance status would be removed from all episodes belonging to that team.

The third and final task in the increment was to let players see a list of attendees for the events. This was a rather straightforward task. When a user accessed an event, the attendance statuses associated with that ID were fetched by three model functions that requested each their attendance status from the database. This facilitated for a controller function to request the information from the model functions, apply some table styling to it, and concatenate the three tables into one variable, `echo` this variable and simply call the controller function in the front-end part of the application, the view.

At this point in the development, an assessment had to be made regarding which functionality was feasible to implement in the remaining time. After considering the options and the amount of work required to finish them, it was decided to implement a feature that allowed coaches to view statistics on event attendance, and let this be the only task in the final increment. This feature was implemented by adding a table `attendance_statistics` which only stored a user ID and an episode ID in each row. A function was created that checked for new rows in this table. This function was set up to run with 24 hour intervals aided by a job scheduler service[7].

After the work in this increment was implemented, back-end development had to cease in order to have sufficient time to develop a satisfactory user interface. Hence, the remaining planned features had to be shelved. This included an internal messaging system and a notification system, which was intended to provide support for a third feature, opportunity for players to notify coaches of intended absence. The final database schema, excluding some tables related to the Auth library, are shown in figure 4.5 below.

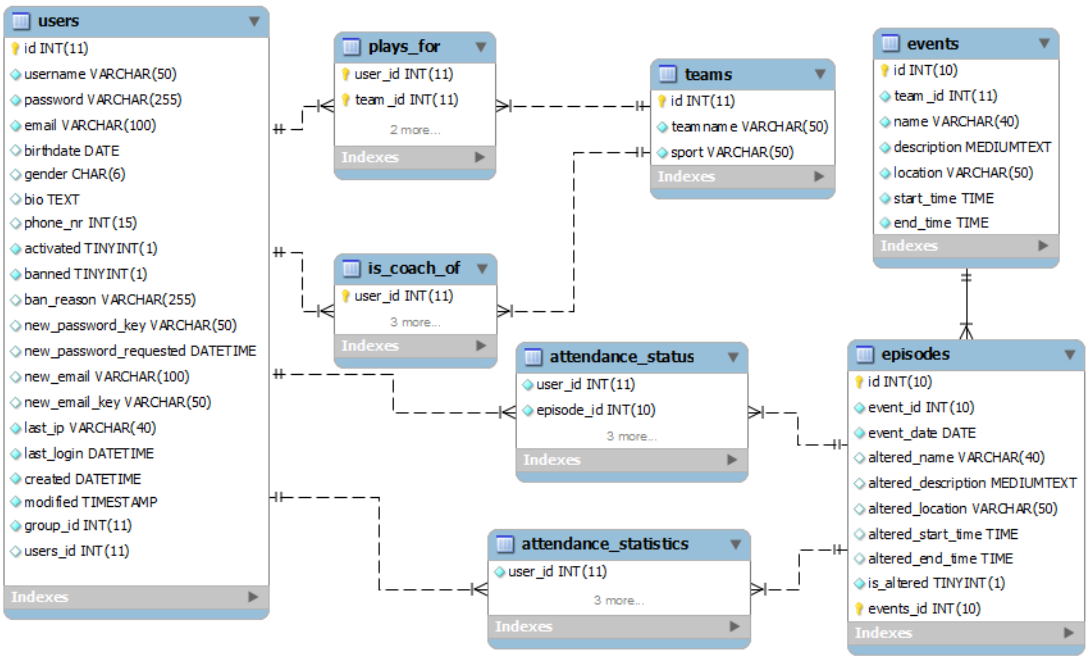


Figure 4.5: Final database schema in rockEnroll, excluding Authentication tables

[7]https://www.setcronjob.com/

4.2.5 Front-end Development

The user interface was split in three main sections, the home page, the team page and the profile page. Additional pages provided conventional login and registration related content. A common element in all of the three main pages was a horizontal navigation bar. This bar provided quick access to other pages. Notably, the bar contained dynamic links to the teams the user was registered as player of or coach of.

The home page consisted of a horizontally aligned list styled as boxes, each containing information about a team that the user belonged to and a button link to access the team page of that team, as seen in figure 4.6.

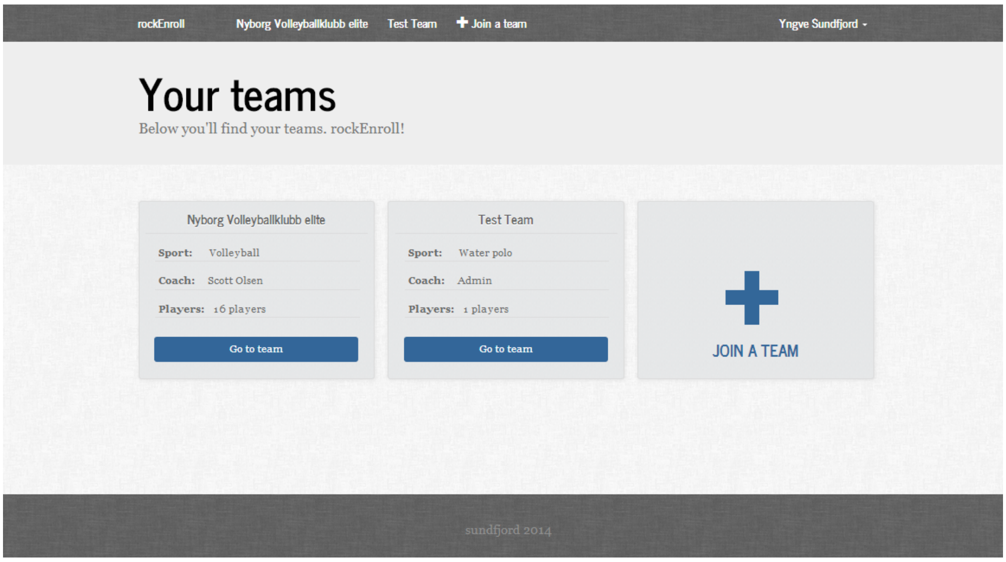


Figure 4.6: A home page in rockEnroll

The team page was the most complex page, as players and coaches had access to different information. For players, the team page consisted of a heading with the team name and two selectable tabs, *Schedule* and *About*. On the schedule tab, the player could see and interact with the most important feature of the web application, the calendar. The *About* tab hosted two tables, one containing a list of players and one containing the coaches. In addition, a button that allowed the user to leave the team. For coaches, the team page was more copious, as can be seen in figure 4.7 and 4.8. The coach schedule tab contained similar content as the player version.

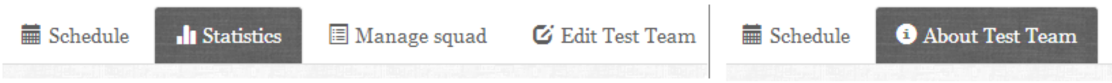


Figure 4.7: Tabs, in coach and player view respectively

A *Statistics* tab contained a table with three columns, player, email and a column with the number of events the player had attended. Above the table a date range picker was located, allowing the coach to pick a date range. Upon submission of the dates, the table updated automatically to show attendance from events occurring in the specified date range. A *Manage Squad* tab consisted of a list of players and a list of coaches. Coaches could select players or coaches by ticking a checkbox and remove them from the team by pressing a button. An accordion below these tables of members contained separate, paginated lists of all players and all coaches in the database, from which the coach could add members to the team. The last tab, the *Edit Team* tab, provided the coach with the opportunity to change the team name and sport, as well as deleting the team altogether by typing the exact team name and pressing a button.

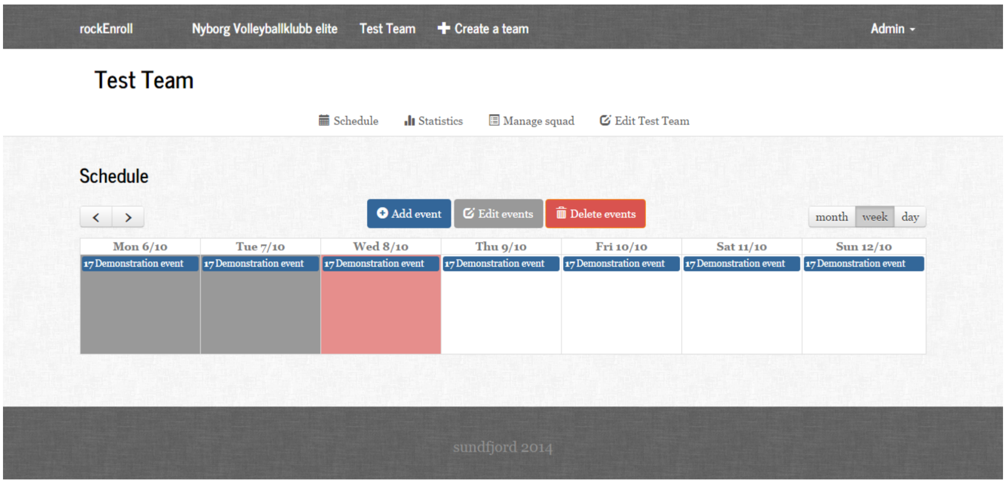


Figure 4.8: coach team page in rockEnroll

The profile page, accessible by clicking a conventional dropdown menu in the top right of the menu, was the same for both user roles, and offered users the opportunity to change their name and email.

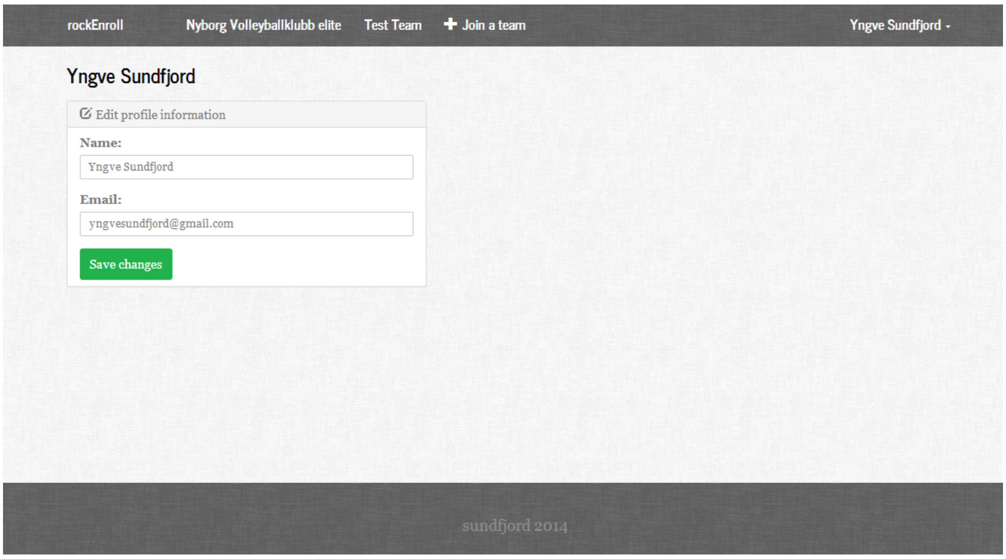


Figure 4.9: A profile page in rockEnroll

When development of the user interface began, some design choices had already been made. For instance, some simple tab functionality provided by Bootstrap was implemented early on in the back-end development process, to separate functionality. This tab functionality was kept in the final design, as no clear disadvantages were found. Normally, tabs can be unsuitable if the content to be loaded is of substantial size, as loading all the content at once may slow performance. The amount of data loaded, however, was not considered to be enough to cause an issue in this context.

Some of the rough paper sketches suggested that the navigation menu should be a vertically left-aligned sidebar. While building the back-end functionality, a very rough version of this design was used. When focus shifted to the user interface, a conscious consideration was made whether the menu should be a vertical sidebar as before or a horizontal menu on the top of the page. Both designs should be familiar to users as they are both a common element in web design, they both had a high visibility of the options available and were easy to implement. The crucial argument was that a horizontal menu were easier to design for mobile devices. A horizontal sidebar would require toggle functionality, as it would else occupy too much space. Due to the plethora of devices with widely varying screen dimensions, resolution and pixel ratio, it would also take more work to find menu widths that would accommodate all of the devices. Figure 4.10 shows the QOC reasoning for choice of navigation menu.

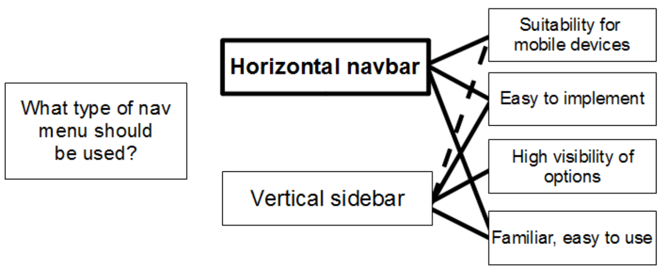
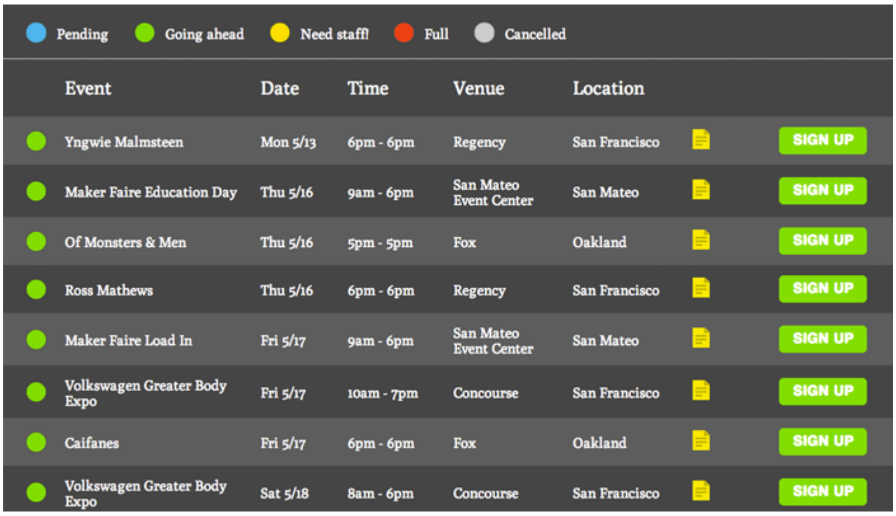


Figure 4.10: QOC reasoning for choice of navigation menu

Each episode had fields in the database for the date, start time and end time, and other relevant information. This information was fetched from the database, converted to a

JSON feed and read by *FullCalendar*, a jQuery plugin providing a full-sized calendar view of events. Users could switch calendar view between day, week and month. Thus, users were presented with a conventional, functional way of viewing events.


Pending Going ahead Need staff! Full Cancelled

	Event	Date	Time	Venue	Location		
	Yngwie Malmsteen	Mon 5/13	6pm - 6pm	Regency	San Francisco		SIGN UP
	Maker Faire Education Day	Thu 5/16	9am - 6pm	San Mateo Event Center	San Mateo		SIGN UP
	Of Monsters & Men	Thu 5/16	5pm - 5pm	Fox	Oakland		SIGN UP
	Ross Mathews	Thu 5/16	6pm - 6pm	Regency	San Francisco		SIGN UP
	Maker Faire Load In	Fri 5/17	9am - 6pm	San Mateo Event Center	San Mateo		SIGN UP
	Volkswagen Greater Body Expo	Fri 5/17	10am - 7pm	Concourse	San Francisco		SIGN UP
	Caifanes	Fri 5/17	6pm - 6pm	Fox	Oakland		SIGN UP
	Volkswagen Greater Body Expo	Sat 5/18	8am - 6pm	Concourse	San Francisco		SIGN UP

Figure 4.11: Example of events displayed in a table

An alternative was to have similar functionality to add, edit and delete events and episodes, but rather present the events in a table. An example of this approach done by Rock Medicine[8] is shown in figure 4.11 above. Nevertheless, the calendar solution was chosen as a result of a QOC reasoning as seen in figure 4.12 below.

[8]www.rockmed.org

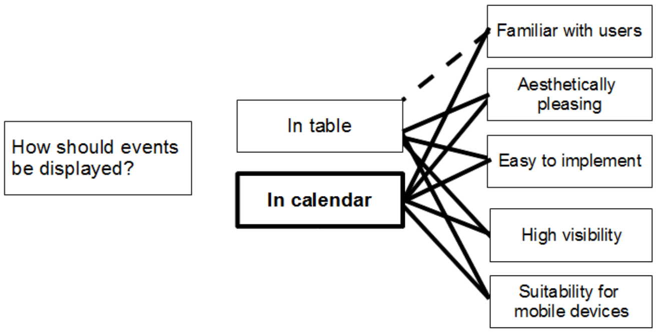


Figure 4.12: QOC reasoning for choosing how to display events

Back-end functionality allowing coaches to create events had been developed, which meant that the prototype would make heavy use of forms. Hence, a user friendly and straightforward solution for the display of forms had to be found. The two viable options identified were either implementing an accordion in the schedule tab containing the relevant forms, or providing buttons in the schedule tab that would open a modal with the relevant form inside. The latter was chosen, as a result of a QOC- reasoning shown in figure 4.13 below.

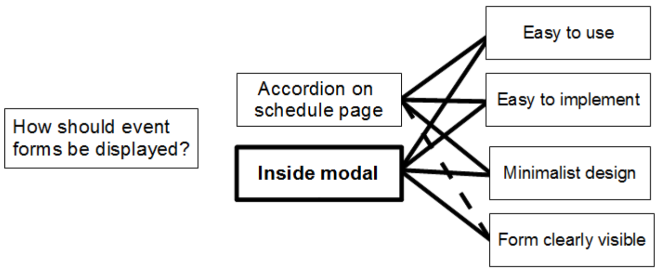


Figure 4.13: QOC reasoning for choosing display of forms

Furthermore, the form errors were placed below the form fields, not inline, which was the other alternative. This was because it would fit more easily with surrounding content, which was deemed more important than the aesthetic pleasure associated with inline form errors. Figure 4.14 below demonstrates an example of a form with errors.

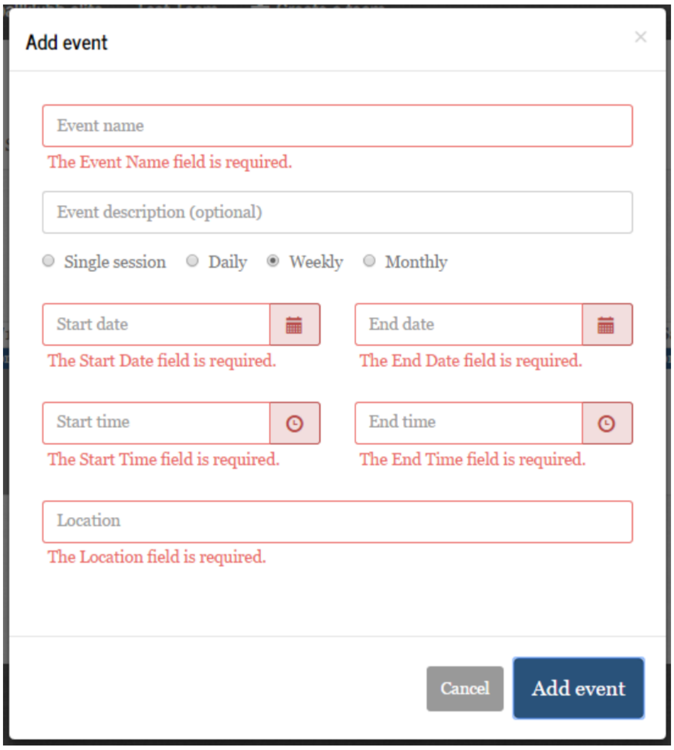


Figure 4.14: Form errors in rockEnroll

The two other tasks in this iteration was to provide players with the ability to sign up for the events, and to see a list of attendees for the events. The calendar plugin included a callback function that triggered an action when an episode was clicked on by the user. When clicked, it was decided that information would appear below the calendar rather than appear in a popover or in a modal window, as this is not an ideal alternative for mobile devices. When clicking the episodes, it was intended that the users should be presented with some basic information about the episode, an overview of attendees and a box containing two radio buttons offering options to click *attending* and *not attending*.

The completion of the back-end and front-end development described above marked the start of the second iteration.

5 Second Iteration - Tweaking the Details

This chapter documents the process of improving the prototype in the second iteration. This phase of development was considerably shorter than the first iteration, lasting three weeks. The results of the heuristic evaluation and the subsequent changes made to the user interface is thoroughly described.

5.1 Heuristic Evaluation

The second iteration started with a heuristic evaluation. This evaluation was carried out in order to uncover usability issues or missing or flawed functionality. Having been informed of the heuristics that the system was to be judged against, the evaluators were given mock tasks that demonstrated most of the functionality in the system. One of these tasks involved free browsing and exploration of the system. The evaluation was conducted on the researcher's computer, which was connected to a high-definition, widescreen monitor. Although the evaluator is supposed to perform the inspection of the interface alone, the researcher was present to observe the evaluation in order to *a)* record the evaluator's explicit comments about the interface, *b)* be available to assist in case of technical problems with the prototype or questions regarding the domain of the prototype, and *c)* record the time spent by the evaluator on each task. However, the responsibility for actually analysing the user interface was placed with the evaluator. The evaluators noted their findings in an issue column next to the task number.

After all evaluations had been completed, the noted findings were aggregated and categorised. The approach outlined is consistent with recommendations made by Nielsen (1994d, 1995b). Following the aggregation of findings, the usability issues discovered were examined and assigned a level of severity based on a discussion between the researcher and the evaluators. The possible levels of severity were a scale of 0 to 4, suggested by Nielsen (1995a).

0 = Not an issue

1 = Cosmetic problems

2 = Minor usability problems

3 = Major usability problems

4 = Usability catastrophes

Usability catastrophes are imperative to fix before the product can be released, while major usability problems should receive focus during the next iteration after release. Minor usability problems should be fixed once bigger problems have been fixed. Cosmetic problems need not be fixed unless extra time is available on the project. The rating of the problems allowed for the generation of a helpful priority list of which problems to rectify first in the second iteration.

5.2 Evaluation Results

As expected, the results of the heuristic evaluation mostly concerned pure usability flaws in the interface. Since the evaluators received no introduction into the domain of the prototype and were not members of the target group, their basis for commenting on the suitability of the functionality as well as suggesting further features was sparse. The evaluators used between 17 and 26 minutes to complete the tasks. The free exploration task accounted for the vast majority of this variation, suggesting the evaluators put different levels of effort into the evaluation. While one evaluator spent almost 10 minutes exploring the system and testing functionality, another evaluator considered 90 seconds to be enough time to execute this task adequately. Despite this variation, all evaluators pointed out issues none of the other two evaluators had discovered. Only four particular issues were mentioned by more than one evaluator. Of these, one issue was pointed out by all three evaluators. Most of the issues discovered received severity ranking 2 or 3, however, two issues was considered usability catastrophes.

The majority of the issues were simple to assign to an appropriate heuristic, indicating that the five heuristics chosen for this evaluation was sufficient to cover the issues in the prototype. Regarding *visibility of system status*, one evaluator felt that the success messages should be increased in size and pushed further down, making them more prominent when shown. When editing the details of an event, an evaluator commented that it was hard to understand that one had to click the *Save Changes* button in each event accordion before exiting the modal through the *Done* button, and that it was hard to know if the changes had been saved when the button was clicked. Also, it was commented that a loader image should show while the system was working to tell the user this.

The second heuristic was *Consistency and standards and match between system and the real world*. This heuristic was the one that the most issues fell under. This was unsurprising, as it is essentially two of Nielsen's heuristics merged into one. All three evaluators reacted negatively to the time picker used to help users select times. While most time pickers allows users to select times with one click, the picker used in the prototype broke platform conventions by being used in two steps, first choosing hour, and then minute. It was also remarked that the second tab on the player team page, called *About*, was not appropriately named and that a title more correctly reflecting the content of the tab should be identified. In addition, in the *Manage Squad* tab, a checkbox in the upper left corner of the player and coach tables could be clicked to select all players or coaches. However, once all records were selected, all of them could not be removed from the team simultaneously. Evaluators commented that this feature should either be fixed or removed altogether. Finally, one evaluator felt that the whole list element on the home page should be clickable, not only a button. This evaluator also thought that one should be logged in automatically after clicking the email activation link and receive a message about this.

Concerning error prevention, one evaluator, who committed an error when attempting to register, discovered that the choice of role was reset when the form was loaded again. The evaluator commented that this could cause the user to make another error, by only correcting the other errors and not noticing that the field had been reset, hence resubmitting the form containing another error. A second issue was discovered when an evaluator chose a full set of players and coaches to add to the team, only to discover that there was separate buttons for adding players and coaches, forcing the user to perform this task in two steps. A final issue in this category was that if event details were edited but not saved, the user could click done, not get prompted about saving the changes made and subsequently lose them.

It appeared that the evaluators found surprisingly few problems related to *aesthetic and minimalist design*, although one evaluator though the content should be separated in a better way. However, this evaluator did not explain in particular detail where in the prototype this was problematic.

Some features relied heavily on form submission. As a result of this, some issues were found that concerned the heuristic of *helping users to recognise, diagnose and recover from errors*. The evaluator who spent the most time exploring the prototype uncovered several situations where submitted forms did not get validated, but where the user got no error message explaining what went wrong. The evaluator considered these flaws a usability catastrophe that should receive high priority. Examples of this were that events could be made where the end time was set before the start time, and that the start time of the first episode of an event was allowed to be set before the current time. In addition, the evaluator felt that events should have a minimum duration.

Some issues were also discovered that could not easily be assigned to any of the heuristics. For example, the system duplicated events multiple times when the end date was set to a date in the subsequent year. The evaluator who was exposed to this bug assessed this as a catastrophic problem that had to receive high priority in the second iteration, although it was not an issue directly related to usability. Second, while attendance status had to be set by clicking a radio button on the left of the screen, two evaluators clicked on the head of the attendance table they wanted to be added to, as they did not immediately discover the functionality allowing them to change their attendance status. It was explicitly remarked that these radio buttons should be much more prominent in the design.

Finally, two evaluators commented that a user's teams should be listed on their profile page to provide additional overview.

5.3 Mending the Issues

Most of the issues discovered in the heuristic evaluation could be solved or at least improved through only one viable solution or through a quick fix. This rendered design reasoning with the QOC- method superfluous for these issues. As mentioned above, a list of issues sorted by the level of severity was used to determine what issues required the highest priority.

Naturally, the issues that the evaluators had considered a usability catastrophe was the first to be analysed. One of these perceived catastrophes, the bug that caused events to be duplicated multiple times, was fixed easily by correcting a small syntax error. The change meant to solve the second catastrophy was to add several additional rules to the event forms, disallowing for example end times that was set before start times, and episodes where the start time was set to time before the current time. As it was related, an issue perceived to be a minor usability problem was also attempted solved at the same time. For the form to be validated, events were required to have a minimum duration of 15 minutes. Figure 5.1 shows the new error messages.

Figure 5.1: Additional form errors

Five issues were considered major usability problems. Notably, the radio buttons used to change attendance status were relocated to above the attendance tables and styled with suitable colours and a custom radio button script, as shown below in figure 5.2 All this was intended to give a stronger indication of what clicking them would involve, as well as increasing visibility and to group the content more logically.

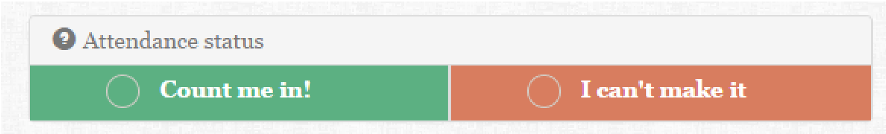


Figure 5.2: Increased prominence of attendance status radio buttons

When editing event details, the “save changes” button were made slightly larger so that the user would less easily risk losing changes by forgetting to click it although the size could not be increased excessively, as this could cause it to compete for attention with the “Done” button. In addition, when it was clicked, the button changed appearance for a few seconds, visually confirming that the changes had been saved, before reverting to its original state.

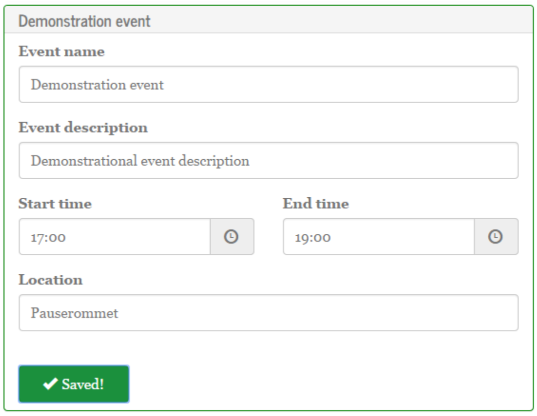


Figure 5.3: Visual confirmation of saved event data

In addition, the possibility of changing the criticised time picker was analysed, but the available time picker scripts that were more conventional and user friendly, were not compatible with the version of Bootstrap utilised in the prototype. A rewrite of one of these scripts would be necessary, and this was considered too time consuming for a non-catastrophic usability issue. It was decided that instead, the users would be instructed how the time picker functioned. Also, the role option chosen when registering was re-populated when there were errors in the form, and players and coaches already in a team could not be selected when users wanted to add more members to that team.

Seven issues were determined to be minor usability problems by the evaluators. These issues were generally attempted solved by following explicit or implicit recommendations by the evaluators. An example of an explicit recommendation was a comment that an overview of the teams that the player or coach is a member of should be added on the profile page. An example of an implicit recommendation was an issue that stated that the evaluator could not submit forms by pressing the return key. The implicit recommendation here was obviously to allow submission of forms by pressing the return key. In total, five of the seven issues were addressed. Two issues were looked into, but no changes were made as no better solution could be implemented in the time frame that was available. The *About* tab was not renamed, and players and coaches could still not be added simultaneously, as allowing this would require back-end changes to facilitate it.

Although comments about certain elements of the user interface occurred during the evaluation, the evaluators later retracted them, not regarding them important, or changing their mind about the suitability of the element on second thought. Hence, none of the issues recorded were labeled as cosmetic problems. Three issues were discovered by the evaluators, but discarded as not really an issue in the subsequent discussion.

With these tweaks, improvements and changes, in addition to some bug fixes and small code improvement, the prototype was considered complete.

6 Evaluation

This section describes the process and results from the evaluation of the prototype. It includes results from the usability test, the field study, and the qualitative interview.

6.1 Usability Test

As mentioned in section 3.5.1, the summative usability test was carried out at the end of development, in a controlled environment. After providing demographic data, participants were asked to rate their perceived proficiency with computer devices and browsing the web on a scale from 1 to 5. No user rated themselves lower than a 3 on either of these qualities. Thereafter, the participants were observed while performing a set of written, premade tasks. The tasks were performed on a laptop computer. There were two different sets of tasks, one for players, and one for coaches. Which set of tasks was given to which participant was random. The tasks aimed to ensure that the user tested all the main features of the prototype, which meant that the coach tasks were more numerous and time-consuming than the player tasks.

5 participants performed the player tasks and 4 performed the coach tasks. During the performance of tasks, time spent on each task was logged and a general description of errors made or other incidents requiring attention was noted. Video recording of user actions, software logging of keystrokes and mouse movements as well as obtaining more quantitative measures of performance are examples of additional recommended practices to collect data. The activities described above, however, were considered sufficient in the research context. This decision was also influenced by practical constraints at the test location, such as equipment and time available. The participants were, however, encouraged to think aloud. The think-aloud method was introduced in section 3.5.1. Comments from participants that were deemed relevant to the tasks were noted.

When the testing sessions were over, the data was analysed. In line with suggestions by Rubin and Chisnell (2008), *mean completion times* for each task was calculated, as well as the average total time spent by each participant. The mean time to complete is a rough indication of how the group performed as a whole (Rubin and Chisnell, 2008). In addition, notes were grouped by task and merged when they concerned the same issue. A summary of the testing of each set of tasks can be seen below in table 6.1 and 6.2.

PLAYERS		
Tasks	**Issues / comments**	**Mean time to complete**
1: Register to the system as a *player* and log in.	- 3 of 5 participants tried to log in before registered	2 min 6 sec
2: Join the team *Test Team*.		16 sec
3: Find the event scheduled for March 28th at 20.00. Change the attendance status to «attending».	- 4 of 5 participants clicked the «attending» table header	43 sec
4: Find the event on March 10th 15.00 and try to change the attendance status.	- All participants understood that this was not supposed to be possible when it was in the past.	39 sec
5: Change your profile name and email address.	- 1 participant didn't notice that profile changes were saved	1 min 1 second
6: Leave *Test Team*.	- 1 participant could not immediately find «leave team»	27 sec
7: Log out of the application.	No errors by any participants.	5 sec
AVERAGE TIME SPENT		**5 min 31 sec**

Table 6.1: Results from the usability test for players

COACHES		
Tasks	**Issues / comments**	**Mean time to complete**
1: Register to the system as a *coach* and log in.	- Register allows password of 4 characters, while login error rule allows 5 characters. - 1 participant accidentally clicked on register link when trying to log in. - 1 participant tried to log in before registering	2 min, 24 sec
2: Create a team with name and sport of your choosing.	4 of 4 participants were able to do the task without error	50 sec
3: Add the three players *Uncle Sam*, *Fred Bloggs*, and *John Smith* to the team.	- 2 of 4 participants struggled to find «manage squad» and associate it with the task. - 1 participant searched in coach table. - 1 participant tried to press «add coaches to team» button. - 1 participant remarked that accordion should be auto-opened. - 2 of 4 participants requested ability to be able to search several times, but to store each checked player in memory and press add once	1 min 32 sec
4: Add a recurring event with start date today and an end date and frequency of your choosing.	- 1 participant assumed start and end time was optional - 1 participant didn't notice timepicker showed 13.08 instead of 13.00	1 min 27 sec
5: Access the first occurrence of the event created in task 4, and edit the start time, end time and location.	- 1 participant tried to click event info panels to edit - 1 participant didn't notice timepicker showed 13.09 instead of 13.00 - Event name length rule is not the same in add event as in edit episode.	1min 43 sec
6: Delete the same specific event.	- 1 participant used the delete events button and deleted all events instead of one.	23 sec
7: Edit the details of all the events of the type you created in task 4.	- 2 of 4 participants did not find or understand the «edit events» button and instead edited details on each episode. - 2 out of 4 participant did not immediately understand clicking on event name accordion was necessary - 1 participant was confused when didn't see specific event when attempting to edit events	1 min 15 sec
8: Delete all the events you made in task 4.	No errors by any participants.	54 sec
9: Remove all the players in the squad. Add, and then remove *Test Coach* as coach.	- 1 participant tried to search in coach table	18 sec
10: Change the team name and sport to something of your choosing.	No errors by any participants.	40 sec
11: Delete the team you just renamed.	No errors by any participants.	30 sec
12: Log out.	No errors by any participants.	11 sec
AVERAGE TIME SPENT		**11min 50sec**

Table 6.2: Results from the usability test for coaches

Among the notable results, the users needed an average of 2 minutes and 24 seconds to complete the registration process, which was more than expected. However, 5 of 9 users thought a profile had been set up for them and that they could log in with any email and password, an issue that most likely would not have occurred in a normal use situation. Second, 4 of 5 player participants did not immediately understand how to change attendance status after they had clicked an event, and instead clicked the table headers. This was a surprise, considering the substantial changes that were made in order to remedy this issue following the heuristic evaluation. Also, a general impression was that coach participants struggled to understand the distinction between events and episodes, as some edited the details of each episode of an event when they were supposed to use the edit events button, while another participant thought he could edit a specific episode by pressing the edit events button. In addition, when trying to add players to the team, 2 coach participants thought it should be possible to search several times for different players, store each checked player in memory and then press add once, instead of searching for one player, checking it and adding it before searching again for the next player.

Considering the use context of the prototype includes both mobile and stationary use, conducting additional tests in a mobile context such as on the bus, as exemplified by Kjeldskov and Stage (2004), could provide fruitful additional results. Due to time constraints, being beyond the scope of the thesis, and the uncertainty associated with the conflicting effectivity results in such field testing (Kallio and Kaikkonen, 2005; Nielsen et al., 2006; Duh et al., 2006), such a test was not carried out.

6.2 Field study

The usability test highlighted some of the potential issues that could cause users to become dissatisfied with the prototype. However, little was still known about how real users in real environments would adopt the prototype and what issues in the prototype were the main barriers for using it. This section describes the activities in the field study that was carried out in order to acquire such data.

6.2.1 Deviation Data

On the confirmation page of the initial web survey for coaches described in section 4.1, respondents interested in trying the prototype were encouraged to contact the researcher. Surprisingly, coaches from as many as eight different teams reported an interest. Towards the end of the development of the prototype some months later, coaches from five of these teams were still interested. The coaches for these teams were instructed to note the number of players expected to attend prior to an event. Furthermore, they were instructed to note how many players actually attended the event, and subsequently report the number of deviations from the expected number in a Google Drive [9] spreadsheet accessible online. This task was performed for each event the team arranged for four weeks. Unfortunately, coaches from two of the teams did not perform this task adequately. When the development phase was completed and the prototype was ready for use, the coaches of the remaining three teams encouraged use of the prototype to all the players on the team. Using the prototype, the coaches were asked to repeat the task of reporting deviations in a spreadsheet for four additional weeks.

At the end of this period, the deviation data was compared and gave a purely quantitative measure on whether or not the prototype had been successful. The results of the field study were split.

[9] drive.google.com

TEAM 1

Deviation form

03.02.14 - 05.03.14

Activity date	Deviation #
04.02.2014	-1
06.02.2014	0
07.02.2014	0
10.02.2014	-2
11.02.2014	-2
13.02.2014	-1
14.02.2014	-2
17.02.2014	-2
18.02.2014	-5
20.02.2014	4
21.02.2014	0
24.02.2014	-1
25.02.2014	-1
27.02.2014	-3
03.03.2014	-2
04.03.2014	-2
AVERAGE	**1,75 deviations per activity**

05.03.14 - 05.04.14

Activity date	Deviation #
06.03.2014	-3
07.03.2014	0
10.03.2014	-2
11.03.2014	-2
13.03.2014	-1
14.03.2014	-1
17.03.2014	0
18.03.2014	-2
20.03.2014	-1
21.03.2014	1
24.03.2014	0
25.03.2014	-2
27.03.2014	0
28.03.2014	0
31.03.2014	0
01.04.2014	-2
03.04.2014	0
AVERAGE	**1 deviation per activity**

Figure 6.1: Detailed deviation data for Team 1

One team clearly had the most deviations of the three teams, both before and after the prototype was used. In this team, the prototype appeared to have a significant positive effect on attendance predictability. Before it was used, an average of 1.75 deviations per activity was reported. This number had dropped to an average of 1 deviation per activity in the second phase of reporting. This team also had the highest total amount of activities, 33. A detailed report of this team's deviations can be seen above in figure 6.1.

TEAM 2

Deviation form

03.02.14 - 01.03.14

Activity date	Deviation #
06.02.2014	0
13.02.2014	0
20.02.2014	1
24.02.2014	1
27.02.2014	0
03.02.2014	2
04.02.2014	1
10.02.2014	1
11.02.2014	0
17.02.2014	0
18.02.2014	2
24.02.2014	2
25.02.2014	0
14.02.2014	1
21.02.2014	1
AVERAGE	0,8 deviations per activity

01.03.14 - 01.04.14

Activity date	Deviation #
03.03.2014	0
04.03.2014	0
07.03.2014	1
10.03.2014	0
11.03.2014	1
13.03.2014	0
14.03.2014	0
17.03.2014	2
18.03.2014	0
21.03.2014	1
24.03.2014	0
25.03.2014	0
27.03.2014	1
AVERAGE	0,46 deviations per activity

Figure 6.2: Detailed deviation data for Team 2

The second team scored a considerably lower average of deviations per activity than the first team both before and after the prototype was introduced. An average of 0.8 deviations per activity in the first phase was followed by an average of 0.46 deviations per activity in the second phase, also a significant improvement. The second team had 28 activities in total, which was the second highest of the three teams. The second team's deviation report is detailed in figure 6.2.

TEAM 3

Deviation form

03.02.14 - 01.03.14

Activity date	Deviation #
02.02.14	0
05.02.14	0
06.02.14	0
12.02.14	1
13.02.14	0
16.02.14	1
19.02.14	0
20.02.14	0
23.02.14	0
26.02.14	0
27.02.14	0
AVERAGE	0.18 deviations per activity

01.03.14 - 01.04.14

Activity date	Deviation #
05.03.14	1
06.03.14	0
12.03.14	0
13.03.14	0
19.03.14	0
20.03.14	0
23.03.14	0
26.03.14	1
27.03.14	0
30.03.14	0
03.04.14	1
AVERAGE	0.27 deviations per activity

Figure 6.3: Detailed deviation data for Team 3

The third and final team participating in the field study had a total of 22 activities in this period, which was the lowest number of all three teams by a considerable margin. This low number of activity number seemed to affect predictability of attendance, as this team reportedly had very few deviations from the expected number of attendants. Although the difference was very marginal and hardly relevant, this team was the only team that reported a higher average of deviations per activity after the prototype had been introduced to the team. The full deviation report for this team can be seen in figure 6.3 above.

6.2.2 Evaluative Web Survey

In the field study, around 45 players and coaches created a profile and used the prototype for a month. This made each player a valuable source of data, having been able to form opinions and thoughts on the prototype through relatively frequent use. It was decided that an attempt would be made to extract some of this data by asking the users to complete a web survey.

The questionnaire consisted of 21 questions. In addition to demographic questions, one set of questions was designed to find the users' view on *perceived ease of use* in particular features of the prototype, while another attempted to capture feelings toward *perceived usefulness*. These beliefs are, as stated in section 2.1.1, deemed the two main beliefs relevant to acceptance of new technology (Davis, 1989; Davis et al., 1989). These questions were in the form of statements, to which respondents were asked to rate their level of agreement on a likert scale from 1 to 5, where 1 was "strongly agree" and 5 was "strongly disagree". This is a typical approach when measuring technology acceptance beliefs, demonstrated by for example Davis (1989) and Venkatesh and Davis (2000). Further questions asked the respondents whether or not they wanted to use the prototype in the first place, and if they wanted their team to continue using it beyond the test period, as well as other questions regarding how and why they used the prototype. Finally, two questions attempted to find out what other features the users would find useful to improve their opinion of the prototype.

Disappointingly, only 24 of the users responded to the questionnaire. Although this weakened the empirical strength, some interesting data was gathered. Regarding ease of use of various features the respondents were generally positive, with an average score on these questions of 1.86. This was consistent with the score from the question that measured the overall impression of ease of use in the prototype, which was 1.91. The exception here was the perceived ease of use on mobile devices, which scored a modest 2.29. Regarding usefulness, users were generally less positive. Questions about this belief scored an average of 2.42. Notably, with a score of 3.33, none of the coaches agreed with the statement that said that their team's use of the prototype increased the predictability of attendance. Over half the respondents strongly agreed to continue using the prototype, however, and when asked why they used the prototype, 67% responded that it was because it was useful for the team. Sadly though, almost a quarter of the respondents said they used it because they felt the coach had put pressure on them to do so.

Among the general questions, an intriguing result was that 64% of users admitted that they had not signed on or off for one or more activities. This result was confusing, as 88% claimed to have been positive towards using the prototype initially. In addition, 75%

thought it was annoying when others did not sign up. Regarding desired functionality for future versions, it was very clear that a mobile app was desired, as all but three respondents ticked this as a desired feature. The two other features proposed to the respondents were also attractive, however, with Facebook-integration being wanted by 33% and an internal messaging system by 46%. Five users elected to either suggest other features or elaborate on why a particular feature should be added in the final free text question. A full overview of the survey questions can be found in appendix C.

6.2.3 Interview

As announced in section 3.5.3, as part of the field study, a semi-structured interview was held with a person that had used the prototype as both coach and player. The first question tried to extract information about how the interviewee perceived the attendance predictability in the two teams he had been involved with. In the team he coached, the basis was that everyone attended, and if not, a text message should be sent to the coach who was leading the activity (there were two coaches in the team). The coaches did not want emails. If nothing was heard from a player, it was assumed they would show up. The respondent felt his players could be better at letting them know, but doubted they were particularly bad at it comparing it to many other teams. The players on the team the respondent played on were much worse at this task. The respondent felt there was no system in the reporting of absence, and as a result it was impossible to predict attendance. He felt it was very individual who were good at reporting absence and who was not.

After the prototype was taken into use, the respondent felt it was easier to get an overview over who was attending. He felt that it is very dependent on every member of the team using it, however, and that it had little purpose if some players did not use it.

A third question asked the respondent if he as coach had an impression that his players were satisfied with the prototype, and if he felt he had to pressure them into using it. The respondent explained that several players had trouble creating a profile. In the end, everyone had an account, but in the mean time, some had forgotten to use it, and some never received the verification email needed to activate the account. He could not definitely say anything about how the players felt about it, as he had not heard any specific feedback, other than the issues with registration. He felt he gave the players plenty of messages that the prototype had to be used, both verbally at pratices and on social media networks.

The fourth question concerned the respondent's perception of the prototype as a player. He thought it was very straightforward and easy to use, especially since he had experience using an old, roughly similar system. When he bookmarked the URL to the prototype, it was very fast to access. He emphasised that changing attendance status for several activities was too cumbersome, however, as one had to access each activity and change the status.

When asked about the general usefulness of the prototype, the respondent said that as it were today, it could not replace the arrangement they had today, where absentees are required to state the reason of their absence in a text message. With some modifications and improvement, however, he believed it could become extremely useful. He also thought that the concept of a system that gives an overview over attendees is very good.

Regarding ease of use, the respondent felt that the users had a lot of choices, and that this could be both positive and negative. In his opinion, ease of use could be increased by changing the prototype so that the players have an absolute minimum of functionality, things to do and choices in the interface. When the respondent was prompted to elaborate on his perception of ease of use in specific functionality, he said that there was nothing particularly negative, and that he thought that the general lack of colours, images, and fancy web design made it feel easier to use. Furthermore, the respondent was asked to voice his opinion on the mobile versions of the prototype. He admitted that he had struggled to use the prototype on the mobile phone as it was hard to interact, and generally felt that the smaller the device was, the harder it was to use it. He also thought the prototype loaded slowly when he was not connected to a wireless network. This added to the negative experience on mobile devices. He did not find this a problem on bigger devices, such as laptops. Also, he had not noticed any changes in his monthly mobile data use after he started using the prototype.

Some of the gathered data described in the previous sections indicated that players did not use the prototype regularly despite almost unison initial positivity towards it. Question 7 asked the respondent to identify possible reasons why. He believed the main reason to be that the players needed time to get used to using the prototype. He expected the situation to improve over time. He also believed that player attitudes and habits related to notifying coaches played a more important role in this, rather than specific aspects of the prototype. He claimed that the players were lazy. Despite mainly pointing to issues more associated with behavioural science, the respondent was nevertheless asked a follow-up question regarding specific weaknesses in the prototype that he felt could contribute to users neglecting it. Here, he reiterated that users could do too many things and suggested to make it even simpler, both for players and for coaches. The prototype should demand less frequent visits, for example by setting "attending" as the default attendance status. Since there are too many things to learn, people can not be bothered. Another follow-up question for the respondent was to explain what he felt could be done to increase usefulness and ease of use and subsequently increase use of the prototype. Fairly quickly, he suggested creating a mobile application which reminded players about activity, and requested their attendance status, which they could easily set by pressing a button with yes or no values. In an ideal situation, the initial registration process would be the only significant, time consuming task in the prototype.

Training has been suggested as one of the most important post-implementation interventions that leads to greater user acceptance and system success (Venkatesh and Bala, 2008). In addition, most user interfaces have sufficiently many features to warrant help or documentation (Nielsen, 1994b). The respondent believed, however, that neither training nor a help page could enhance the adoption and use of the prototype. He felt that he and several other users that he had talked to understood how to use it, and that this was not among the most pressing issues with the prototype.

7 Discussion

This chapter discusses the research findings in relation to the research questions. The first research question asked how information technology is currently supporting communication between players and coaches (RQ1). The second research question asked if a specialised, novel information technology can improve communication between players and coaches (RQ2). The third research question asked what guidelines can be given for designing an information technology that helps improve communication between players and coaches (RQ3).

7.1 Current support

The first research question was how information technology is currently supporting communication between coaches and players. A vital source of data in order to answer this question, are the initial web surveys introduced in section 4.1.1, which were designed to help gaining a better understanding of the needs of target users of the prototype. Several of the questions in both surveys concerned current use of information technology to convey attendance related communication, and the accompanying answers are hence highly relevant in the research question context.

Generally, 70% of the coaches participating in the survey stated that their team utilise some form of digital tool to organise team activities. Answers from both coaches and players strongly indicate that the social networking service Facebook and SMS are the most prevalent communication tools.

Fifty-five percent of players participating in the survey used SMS, while 36% used Facebook. Communication using these technologies was to inform coaches and fellow players of their intended absence. This was exemplified by the players on the team that was coached by the interview respondent, who were obliged to send an SMS when they could not attend an activity, stating the reason for their absence.

The most popular technology among coaches for organising team activities seemed to be Facebook, as 46% of the coaches in the survey reported that they used it. The use of Facebook seemed closely linked to the opportunity it provides to publish messages on the wall that all players can see, as 57% stated that tools were used to convey messages to players about changes in time, location or similar of an activity. One coach explained that they relied heavily on Facebook because "everyone" used it. Thirty-nine percent of the coaches used SMS for organising, and it is most likely used this widely because it is a suitable technology for players to notify the coach of intended absence, and 49% of the coaches that stated they used tools for this task.

As expected, these results indicate that communication between players and coaches is well supported by current information technologies. Almost half of the players who admitted that they on at least one occasion had not notified coach of their intended absence, however, claimed they had simply forgotten to do it. It appears current technologies do not sufficiently take into account the forgetful nature of players. In addition, it seems that current technologies support delivery of individual messages from player to coach well, but do not support aggregation of messages that makes it easier to get an overview. This claim is backed by the fact that 78% of coaches and 88% of players wanted the ability to see a list of attendees for an activity, while only 27% of the coaches reported that they used information technologies for this task. This may indicate that for this task current technologies are unsuitable.

7.2 rockEnroll support

The second research question asked if a specialised, novel information technology, in this case rockEnroll, can improve communication between players and coaches. The research activities relevant to answer this question are mainly the final, evaluative usability test and the field study activities, but also data from the initial web survey is drawn upon to discuss this research question.

Analysing the quantitative deviation data related to attendance yields no conclusive results, but the general tendency appears to be that the prototype increased attendance predictability. Two out of the three participating teams recorded a significant decrease in deviations per activity, while one team recorded a minimal increase.

Two of the three teams reported a decrease of around 43% in deviations per activity after the prototype was introduced, a result that suggests that use of the prototype increased attendance predictabilityconsiderably. It is difficult, however, to determine how much of this decrease can be attributed specifically to the use of the prototype and what can be attributed to the generally increased focus and awareness on attendance predictability and absence reporting during this period. In addition, in the third team, deviations per activity increased. Out of the three teams participating, however, this team had by a considerable margin the least data on which to rely. The team arranged the lowest number of total activities and had the fewest players on the roster. In addition, the number of deviations per activity were by far the lowest of the three teams both before and after the introduction of the prototype. Hence, the increase of 0.09 deviations per activity is so minimal that the situation for this team after the introduction of the prototype can be considered unchanged.

The results of the evaluative web survey could not further enhance the perception that the introduction of the prototype had improved communication between player and coach. None of the coaches responding to the survey directly agreed that the prototype had improved attendance predictability, a strong indicator that the prototype had not been successful in improving the communication between players and coaches in the view of the coaches.

Results in the initial web survey (described in section 4.1.2) had 42% of the players admitting that they had been absent from an activity without properly notifying the coach. In the evaluative web survey, 64% of users admitted that they had not signed up for one or more activities. These results are not directly comparable, as the initial question asks about absence reporting using any technology, while the latter asks about simply reporting attendance status using the prototype. The results indicate, however, that introducing the prototype did not help to correct the forgetful and/or careless behaviour of players regarding attendance reporting. The goal of providing players with a common platform to communicate attendance status seems to have failed, with players continuing to use SMS and Facebook as alternative or additional platforms of communications. Hence, it can be argued that rockEnroll has not improved communication between players and coaches, but rather exacerbated it by adding another platform of communication into the equation, dispersing attendance information even more than before. A few respondents explicitly shared this view, by answering that they were initially negative towards using rockEnroll because it meant having to use another system. Many players are already avid technology users, and probably already use many services that require registration and login. This may be a barrier from using the system.

Nonetheless, over half the respondents strongly agreed to continue using the prototype, and 67% responded that the reason they used it was that it was useful for the team, indicating that this view is not shared by all players, and it may indicate that they felt it improved communication. It should be remarked that since only about half the total user base responded to the survey, it is possible that the users who responded were among the more dutiful and enthusiastic as they filled out the web survey, and the less enthusiastic, who did not respond to the survey, would most likely vote less positive for continue using the prototype. Hence, there is a chance that the total impression among the users may be more negative than the results indicate. In the initial web survey, 77% of coaches and 86% of players stated that they felt a specialised tool for handling attendance is necessary for their team. With such an overwhelming majority of respondents acknowledging the need for a specialised, novel information technology, it seems to be only a matter of altering and developing rockEnroll until it satisfies the users need for both functionality and simplicity before it can improve the situation.

Data collected from the qualitative interview supports some of the arguments made above. The interviewee said that the prototype as it was today could not replace the technologies they already used for communication. This indicates that the prototype did not improve communication between him and his players. He highlighted, however, the ability to see a list of attendees as an improvement. This element of improvement is negated if players forgets or neglects to use the prototype, causing inconsistent and unreliable lists of attendees. In addition, the coach mentioned that he saw it necessary to remind the players repeatedly about using the prototype. The prototype was intended to eliminate the need for this sort of additional communication from the coach. It appears the introduction of the prototype may have had the opposite effect. The need for such extra reminders, however, may diminish over time, as users gets increasingly familiar with the prototype and develops a habit of using it, like reported in Davis et al. (1989) and described in Tetard and Collan (2009). This was a point stressed by the interviewee as well. A few players stated that they used rockEnroll because they felt pressured by their coach to do so. Agarwal and Prasad (1997) found that initial usage of a system may be influenced by perceptions of voluntariness, but later in the process, people will continue to use the system only if they find it useful. Since external pressure has an impact on adopter's initial acceptance behaviour, it may justifiably be applied to players in order to force them to try out the system and determine how useful they think it is. Further research evaluating use of the prototype over a longer period of time is required to provide more data on this topic.

Results from the both the evaluative usability test and the field study indicates that lack of usability or low perceived ease of use was not the main barrier for using the prototype. No serious usability flaws were found in the final usability test and none were reported by the real users during the trial month, despite being encouraged to report any bugs, unexpected behaviour, or other errors in functionality. Ease-of-use factors scored relatively highly among the same respondents in the evaluative web survey. In addition, no specific, critical issues were raised by the interviewee in the field study, despite being asked directly. Smaller issues were mentioned that may have contributed to the users becoming less inclined to using rockEnroll, possibly in combination with other factors reported in prior research to influence acceptance, such as low perceived usefulness (Davis, 1989), individual differences (Agarwal and Prasad, 1999) and gender (Venkatesh and Morris, 2000). High ease of use has been reported to facilitate early adoption, but becomes less important than perceived usefulness in the post-adoption stage (Adams et al., 1992). Keil et al. (1995) labeled systems that are generally easy to use but lack the power needed to be considered truly useful "toys", and stated that such systems stand very little chance of lasting acceptance. Davis et al. (1989) reported that *"[o]ver time, as users learned to effectively operate the word processor, the direct effect of ease of use on behavioural intention disappeared"*(p.998). As the use context for rockEnroll is on the web, usability is an absolute necessity since leaving is the first line of defense when users encounter a difficulty (Nielsen, 2012).

Regarding perceived usefulness, the features implemented in rockEnroll appeared to be either not advanced enough nor simply not a good fit with what the players and coaches wanted. It seems clear that features in future versions of rockEnroll must become even more useful for the task, as perceived usefulness is commonly regarded as a vital determinant for acceptance of information technology (Keil et al., 1995; Venkatesh and Morris, 2000; Igbaria et al., 1995). In the evaluative web survey, the relatively modest average score of 2.42 on a 1 to 5 scale where 1 was most useful indicates room for improvement in this regard. The extent of this improvement in potential new versions will be vital for a succesful propagation of rockEnroll in the future.

It may appear that even a prototype that achieves both high perceived usefulness and ease of use may not be enough, as long as it is based on expecting a forgetful, neglecting user group such as players to use it consistently. Eighty-eight percent of the users responding to the evaluative web survey stated they were initially positive towards using the system initially. This is futile, however, when the remaining 12% and most likely many of the users who did not respond to the survey, may not wish to use rockEnroll or any other similar system. The interviewee placed part of the blame for not using the prototype on the players rather than any intrinsic quality in the prototype. The system will likely never achieve perfect predictability based on rockEnroll's features alone, as a strong culture for reporting absence within the team will be the deciding factor. In addition, the interviewee contradicted himself by suggesting to have less functionality in order to make the prototype easier to use, while he earlier in the session implicitly suggested implementing another feature to improve the usefulness of the prototype. This sort of ambivalence from users suggests it may be a difficult task to find the right balance between functionality and simplicity in an information technology such as rockEnroll, as they themselves do not appear certain what they want.

The general impression, however, is that there is big potential in increasing predictability in attendance and simplifying the process of reporting attendance status. The results generated from the evaluative research activities and observations made during the field study showed plenty of promise for a potential successful propagation of such an information technology into many organised sports environments and teams. They also showed that the balance between functionality and simplicity must hit the absolute sweetspot for both players and coaches in order for the technology to be accepted among users. This may be difficult due to the forgetful/careless nature of many players as well as their inability to know what they want from the system. In addition, modern technology users become increasingly used to and adept at using technologies, elevating the threshold for accepting new ones. It seems clear that this elevated threshold must be passed in order for rockEnroll to be embraced by users.

7.3 Guidelines for support

The third research question asked what guidelines can be given for designing an information technology that helps overcome reluctance among potential users.

The old saying "all roads lead to Rome" is a suitable metaphor in this context. The modern interpretation of the ancient saying is that there are many ways to reach the same outcome. The desired outcome is improved communication between players and coaches, and there seems to be many different ways to reach this outcome aided by information technology. To extend the metaphor, however, all these roads leading to Rome comprise certain building blocks. The guidelines presented in this section can be looked upon as such building blocks, providing a foundation for a way to Rome.

7.3.1 Solution of Least Effort

The first such a building block is that the potential system *must* be least effort of all the technologies with which one can perform the necessary tasks. It must be the universally least effort fulfillment to players needing to communicate absence to their coaches. This applies regardless if the team policy is to justify their absence with a written text or if they only need to click “not attending” to an activity in a checkbox. Seeing how existing technologies already require little effort, this is a considerable challenge due to today’s demanding technology users.

Assuming the user had bookmarked rockEnroll in their web browser or on their home screen, rockEnroll demonstrably required 4 touch or click actions to set attendance status for an activity regardless of device. Despite this relative efficiency, several users expressed frustration with how “cumbersome” it was and how “much effort” it demanded. Based on feedback from the respondent in the interview, player functionality should be kept to an absolute minimum to reduce possible actions and subsequently reduce effort required. These perceptions suggest that a potential system must be built free of both cognitive (options, possible actions) and physical effort (touches/clicks needed to perform an action) to an even greater extent than rockEnroll. The improvement should be considerable, following a point made by Tetard and Collan (2009), who claimed users may not adopt new solutions unless the cost of learning a new technology is not fully refunded by advances in ease of use.

As steps on the way of becoming the solution of least effort, a few fairly specific sub-guidelines are proposed.

Minimal registration effort

An effort should be made to keep the effort required to register an account as minimal as possible, er even optional. The registration process has been described by for example Wroblewski (2008) as a potential use barrier, as having to fill out a form before you can access the content is annoying to increasingly fickle users who most likely may find plenty of other options to get their task done. The final usability test showed that the register process took almost two and a half minute on average to complete, and this time should be reduced considerably to increase acceptance. An increasingly popular approach in design industry is labeled *lazy registration* or *gradual engagement*. This approach allows the user to learn what a system does without having to go through a sign-up form first (Wroblewski, 2008). Instead, the sign-up should be only one part late in the process of introducing users to a system (Jovanovic, 2009), or even ask for data along the way until the user is a member of the site. In addition it is recommended to, if possible without compromising security, ommit common register form fields such as CAPTCHA and the "confirm email" and "confirm password" fields to speed up the registration progress.

Finally, measures should also be made to ensure that the system is equipped with a well-functioning and visible "remember me"-checkbox as shown in figure 7.1 so that the user does not have to type their login details each time they wish to use the system. Most modern web browsers are equipped with password managers which saves passwords used in login forms and automatically fills them upon the user's next visit to the site, or when the user types the first few letters of their username or email. The system should facilitate for this functionality in the login area.

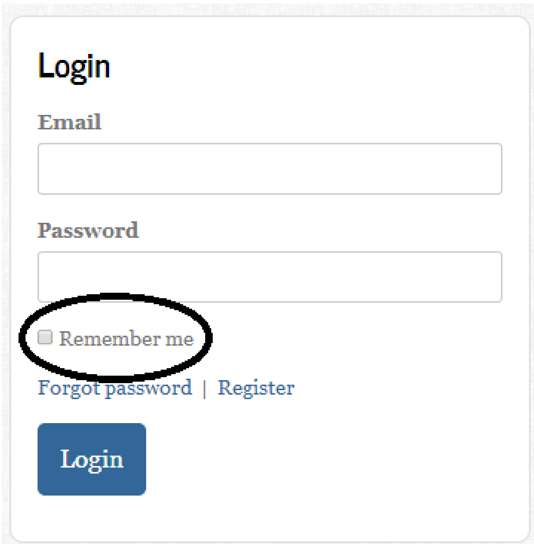


Figure 7.1: "Remember me" functionality in rockEnroll

Batch actions

The design of the potential system should focus on efficiency through batch actions, so that both players and coaches can minimise the time required to use the system. Particularly, the most important and common player actions should be optimised, as coaches are most likely inclined to spend more time using the system. The most prominent example in rockEnroll was when players wanted to set their attendance status. The solution chosen for this action did not allow players to set their attendance status for multiple activities at a time. Instead, players had to click on each activity and then set their attendance status. This was criticised by the respondent in the interview. Players should be able to perform actions such as this in batches, for example by marking the activities one wants to interact with and then press a button to set a particular attendance status on all the chosen activities.

When coaches wished to add players or coaches in the user database to their team, they could browse the player and coach databases in a table. The table was *paginated*, divided into pages, so that only a certain number of entries in the table would be shown on each page. The problem occurred when coaches selected for example, players in one paginated page in the player table, then switched to another paginated page and selected more players. When they pressed the "Add player" button, the players selected in the first paginated page were overwritten, and only the players selected in the last paginated page were added to the team. This meant the coaches had to perform the action multiple times to add all the desired players. Ideally, all players selected should be stored in temporary memory and when all were added, this memory would be reset.

Simplicity

The respondent in the interview highlighted the simplicity of the interface as a positive factor in rockEnroll, as it allowed users to focus on the important elements. Therefore, it is suggested to avoid implementing potential disturbances in the user interface to the greatest extent possible. According to design literature, examples of disturbances are too many features, too much contrast in elements and unaligned elements (Tate, 2009; Walkin, 2009).

Performance

The final sub-guideline is related to system performance. During the testing period, several users mentioned that rockEnroll sometimes loaded slowly, especially on mobile devices. This was confirmed by the respondent in the interview, who experienced loading issues when not connected to a wireless network. Lederer et al. (2000) identified slow speed of downloading or viewing Web pages as a key ease of use problem. Knowing Nielsen (2012) regards leaving the first line of defense when users encounter a difficulty, it seems clear that the system must be designed for good performance. In addition, Nielsen (1993) considers 1 second the limit for the user's flow of thought to stay uninterrupted, even though the user will notice the delay. Grigorik (2013) is even more demanding, claiming that visual feedback should be given within as little as 250 milliseconds in order to keep the user engaged.

Several techniques exist for optimising performance, including *minification and compression of resources*, *image sprites* and *responsive imaging*. Optimising real performance is important, but extra efforts should focus on *perceived performance*, the user's perception of how quickly and smoothly a task appears to be performed.

7.3.2 Native App

Second, designing a native app for all platforms to accompany the web content is strongly recommended. rockEnroll is a responsive web application, designed to be usable on any device. Performing the most central functionality, as has previously been explained, can demonstrably be accomplished through few touches on a mobile device. Despite the effort to accommodate mobile users, almost every single one of the respondents to the evaluative web survey requested a native app. The necessity of developing a native app does not arise because it is essential to the functionality of the prototype, but because the user perceives it as necessary to use the prototype on mobile devices. They appear to have little faith in performing tasks in rockEnroll through web browsers on their phones and tablets.

Native apps have a few important advantages that may greatly impact the effort required. The opportunity to use push notifications reminding them to take action makes sure the users do not always have to remember to use the system themselves, which relieves the user's required cognitive effort, which in turn takes the system closer to being the solution of least effort. In addition, native apps have better performance and functionality is less limited than in web apps, both of which can lessen the effort needed to use the system. As native apps can be costly to develop and maintain, an alternative solution may be a hybrid app (web apps that are compiled into a native app shell), but such apps also struggle with performance when compared to a native app. It should be stressed, however, that it is also recommended that the potential system does not *depend* on the native app, but allowes tasks to be performed without using it if users wish to do so. Some users may be satisfied using the responsive web app, do not wish to install a native app, or do not own a smart phone at all.

7.3.3 Facebook Integration

The third guideline may possibly serve as an alternative to the two first, rather than an addition to them. Some users complained that using rockEnroll meant that they had to sign up for another service and consistently visit it and log in. As a remedy, one of the coaches suggested integrating the service into an existing one that everyone already used, such as Facebook. If all relevant tasks could be performed in the Facebook interface instead of visiting a separate page, the chances of them remembering to use it would increase. It would negate the need of a least effort solution, as it would become part of a solution that users already make an effort to use.

In addition, this solution would negate the need for a novel native app, as Facebook already has a popular, well-developed app on every conceivable platform. This, however, presupposes that the developers can achieve the same level of functionality through the Facebook API that they can in a standalone web and native application. Obviously, this guideline may be applied to other relevant social networks, but few are so widespread and has such a sufficiently sophisticated API to achieve this kind of integration.

The guidelines suggested are summarised in table 7.1 below.

Guidelines	Description
Solution of Least Effort	The system must provide the universally least effort fulfillment of user need, regardless of user state. Sub guidelines: - Minimal registration effort - Batch actions - Simplicity - Performance
Native App	A native or hybrid app should accompany the web content, but using the system should not depend on using this app as it may be unsuitable for some users.
Facebook Integration	As many tasks as possible should be possible to perform in the Facebook interface. May serve as an alternative to the two former guidelines.

Table 7.1: Design guidelines for systems supporting player-coach communication

7.4 Summary

This chapter has discussed the current communication situation between player and coach, whether or not an information technology can improve the communication, and proposed guidelines for design of a potential future information technology.

Currently, Facebook and SMS are the dominating technologies of communication between players and coach as they are suitable for direct communication. Current technologies, however, seemed unsuitable to aggregate information from players and provide a visual overview over attendance for an activity.

The data collected did not conclusively establish that rockEnroll improved communication, however, the data collected suggests that an improved, extended version of rock-Enroll or another novel information technology has the potential to greatly improve communication.

Making the technology the solution of least effort, designing an accompanying native or hybrid app, and (or alternatively) providing Facebook integration is recommended in order for an extension of rockEnroll or another novel information technology to successfully improve communication.

8 Conclusion

This chapter summarises the thesis, discusses limitations and weaknesses in the research, and suggests future research.

8.1 Thesis Summary

This thesis describes the development of a prototype designed to improve communication between players and coach in organised sports teams. Using design research, design rationale, and user-centered design, the responsive web application rockEnroll was developed and tested before being trialled by real users for a month.

The development phase lasted roughly 6 months, consisting of two iterations with design and testing. After initial online questionnaires were administered, the current situation could be surveyed. Based on the results, requirements for the prototype were formulated, and development commenced. Many of the design decisions was rationalised throughout. The second iteration began with a heuristic evaluation. After the second iteration, the prototype was evaluated through a usability test and a field study. Through these methods of evaluation, data relevant to answer the research questions was gathered.

The findings after analysing the data indicate that Facebook and SMS are currently the dominating technologies of communication between players and coach as they are suitable for direct communication, but seemed unsuitable to aggregate information from players and provide a visual overview over attendance for an activity. The data collected did not conclusively establish that rockEnroll improved communication, but suggests that an improved, extended version of rockEnroll or another system has the potential to improve communication. Last, making the technology the solution of least effort, designing an accompanying native or hybrid app, and (or alternatively) providing Facebook integration is recommended in order for an extension of rockEnroll or another novel information technology to successfully improve communication.

8.2 Research Contributions

This thesis serves as a platform for further research on communication between players and coach in organised sports teams, but touches upon and contributes to several other fields of research, namely Technology Acceptance, Lazy User Theory and Human-Computer Interaction. The study has added further empirical strength to the claim that perceived usefulness is the most important determinant in user acceptance of technology. The lazy user theory in a use situation has been strengthened and the chapters related to interface design are applicable for web application usability in general.

Furthermore, the research resulted in a functional prototype, rockEnroll, which is a complete system currently running live[10] for anyone to use.

8.3 Limitations and Weaknesses

The main limitation imposed on the thesis work, especially related to the development of the prototype was time constraints, as the thesis was due to be completed in two semesters.

This limitation may have at least partly contributed to some of the notable weaknesses in the research:

- The quantitative deviation data was self-reported. This is a commonly reported weakness in technology acceptance literature, known to be subject to the common method bias, which distorts and exaggerates the causal relationship between independent and dependent variables (Lee et al., 2003).
- The researcher was closely involved with the environment in two of the three test teams before research commenced. This may have caused bias from the users. They may have felt an urge to "help" the researcher or felt additional pressure to use rockEnroll being personally acquainted with the researcher, rather than using

[10] www.sundfjord.com/rockEnroll

it in a natural, independent way.

- The lack of rigorous data analysis, explained in 3.6, may have caused weaknesses in the data material used to answer research questions, subsequently reducing validity of the research.

8.4 Future research

This single-case thesis has been written as a starting point for further research on the topic, as recommended by Benbasat et al. (1987), who claim that single-case projects are most useful at the outset of theory generation, and recommend subsequently carrying out a multiple-case study. Hence, following this advice, a multiple-case study should be carried out. Such a study should include testing other systems attempting to improve communication, such as Teamer[11], Teamstuff[12], and ITeamSport[13]. Such an approach may yield valuable empirical data on the topic.

To fully explore the potential of rockEnroll to improve communication between players and coaches in organised sports teams, the prototype must be further developed based on feedback from users and the recommendations made in section 7.3, and re-tested by real users in a longitudinal field study of greater duration than in this thesis. In order to ensure more valid data, further research should not make use of self-reported data from coaches or similar. Instead, the researchers should observe use of the system through analytical tools and surveillance, as well as personally observing the actual attendance without letting players know they are being observed. Generally, future studies should be more methodologically rigorous.

[11] www.teamer.net/
[12] www.teamstuff.com
[13] www.iteamsport.com

References

Abras, C., Maloney-Krichmar, D., and Preece, J. User-Centered Design. In Bainbridge, W. S., editor, *Encyclopedia of Human-Computer Interaction*. Sage Publications, 2004. URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.94.381.

Adams, D., Nelson, R., and Todd, P. Perceived usefulness, ease of use, and usage of information technology: a replication. *MIS quarterly*, 1992. URL http://www.jstor.org/stable/249577.

Agarwal, R. and Prasad, J. The role of innovation characteristics and perceived voluntariness in the acceptance of information technologies. *Decision sciences*, 1997. URL http://onlinelibrary.wiley.com/doi/10.1111/j.1540-5915.1997.tb01322.x/abstract.

Agarwal, R. and Prasad, J. Are individual differences germane to the acceptance of new information technologies? *Decision sciences*, 1999. URL http://onlinelibrary.wiley.com/doi/10.1111/j.1540-5915.1999.tb01614.x/abstract.

Ajzen, I. and Fishbein, M. Understanding attitudes and predicting social behaviour. 1980. URL http://www.citeulike.org/group/38/article/235626.

Anderson, D. J. *Kanban: Successful Evolutionary Change in Your Software Business*. Blue Hole, 2010. ISBN 9780984521401.

Bagozzi, R. The Legacy of the Technology Acceptance Model and a Proposal for a Paradigm Shift. *Journal of the association for information ...*, 2007. URL http://search.ebscohost.com/login.aspx?direct=true&profile=ehost&scope=site&authtype=crawler&jrnl=15369323&AN=25513239&h=UANHoH0xr3ZSCi5Cd4HnydCDapPxqwf9Z+1acA4W/Th7/geO0n48AdiTYgKd3EBKIglKyvnM/cvbsINe/v7Jag==&crl=c.

Benbasat, I., Goldstein, D., and Mead, M. The case research strategy in studies of information systems. *MIS quarterly*, 1987. URL http://search.ebscohost.com/login.aspx?direct=true&profile=ehost&scope=site&authtype=crawler&jrnl=02767783&AN=4679735&h=MYu2WOv76xmuo61yTCY+U2UxEU8ltQcdyAPcXgL3jHzSmdl5G+KwDRkE07Y3PsqHqvm2g5G8ugfr8fyaRuvqhw==&crl=c.

Benbasat, I. and Barki, H. Quo vadis TAM?, 2007. ISSN 1536-9323. URL http://aisel.aisnet.org/jais/vol8/iss4/16.

Bryman, A. *Social research methods*. 4th editio edition, 2012. URL http://www.google.com/books?hl=en&lr=&id=vCq5m2hPkOMC&oi=fnd&pg=PP2&dq=bryman+social+research+methods&ots=CJUiPo9SrE&sig=cfeNrLSxszGMcWyOIsB_2711cbg.

Budd, A. Heuristics for Modern Web Application Development, 2007. URL http://www.andybudd.com/archives/2007/01/heuristics_for_modern_web_application_development/.

Carmines, E. and Zeller, R. *Reliability and validity assessment*. 1979. URL http://www.google.com/books?hl=en&lr=&id=BN_MMD9BHogC&oi=fnd&pg=PA5&dq=carmines+and+zeller&ots=_MT4H9XyXc&sig=SQNXadO3gqKHBZ6AglkyuwlAnpc.

Carroll, J. M. Human-computer interaction: psychology as a science of design. *Annual review of psychology*, 48:61–83, January 1997. ISSN 0066-4308. doi: 10.1146/annurev.psych.48.1.61. URL http://www.ncbi.nlm.nih.gov/pubmed/15012476.

Cohn, M. *User stories applied: For agile software development*. 2004. URL http://www.google.com/books?hl=en&lr=&id=SvIwuX4SVigC&oi=fnd&pg=PR15&dq=user+stories&ots=Vo_ec6oROO&sig=MOpWTcZWEtQAmfEpw9Kz-d6ScI8.

Collan, M. Lazy User Behaviour. 2007. URL http://mpra.ub.uni-muenchen.de/4330/.

Davis, F. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly*, 1989. URL http://www.jstor.org/stable/249008.

Davis, F., Bagozzi, R., and Warshaw, P. User acceptance of computer technology: a comparison of two theoretical models. *Management science*, 1989. URL http://pubsonline.informs.org/doi/abs/10.1287/mnsc.35.8.982.

Dix, A., Finley, J., Abowd, G., and Beale, R. *Human-computer interaction.* Prentice-Hall, Inc., 2nd edition, January 1998. ISBN 0-13-239864-8. URL http://dl.acm.org/citation.cfm?id=275608.

Duh, H. B.-L., Tan, G. C. B., and Chen, V. H.-h. Usability evaluation for mobile device. In *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services - MobileHCI '06*, page 181, New York, New York, USA, September 2006. ACM Press. ISBN 1595933905. doi: 10.1145/1152215.1152254. URL http://dl.acm.org/citation.cfm?id=1152215.1152254.

Dumas, J. and Redish, J. *A practical guide to usability testing.* 1999. URL http://www.google.com/books?hl=en&lr=&id=4lge5k_F9EwC&oi=fnd&pg=PR5&dq=usability+testing+dumas&ots=vpld7Bh4xJ&sig=gm72ITUxZHu3xUULVouv9wgRzDs.

Fällman, D. Design oriented-research versus Research-oriented Design. *..., CHI 2004 Workshop on Design and HCI, Conference ...*, 2004. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.2.5468&rep=rep1&type=pdf.

Fallman, D. Design-oriented human-computer interaction. In *Proceedings of the conference on Human factors in computing systems - CHI '03*, page 225, New York, New York, USA, April 2003. ACM Press. ISBN 1581136307. doi: 10.1145/642611.642652. URL http://dl.acm.org/citation.cfm?id=642611.642652.

Fishbein, M. and Ajzen, I. Belief, attitude, intention and behavior: An introduction to theory and research. 1975. URL http://trid.trb.org/view.aspx?id=1150648.

Grigorik, I. *High Performance Browser Networking: What every web developer should know about networking and web performance*. 2013. URL http://www.google.com/books?hl=en&lr=&id=KfW-AAAAQBAJ&oi=fnd&pg=PR2&dq=High+Performance+Browser+Networking:+What+Every+Web+Developer+Should+Know+about+Networking+and+Web+Performance&ots=saX9YIy1xI&sig=xHgbVtc_WmP_TnKyU5ARWFuZZ88.

Hamilton, S. and Chervany, N. Evaluating information system effectiveness-Part I: Comparing evaluation approaches. *MIS quarterly*, 1981. URL http://www.jstor.org/stable/249291.

Hartson, H. R., Andre, T. S., and Williges, R. C. Criteria For Evaluating Usability Evaluation Methods. *International Journal of Human-Computer Interaction*, 13 (4):373–410, December 2001. ISSN 1044-7318. doi: 10.1207/S15327590IJHC1304_03. URL http://dx.doi.org/10.1207/S15327590IJHC1304_03.

Hevner, A. R., March, S. T., Park, J., and Ram, S. Design Science in Information Systems Research. *MIS Quarterly*, 28(1):75–105, 2004. ISSN 02767783. doi: 10.2307/249422. URL http://www.jstor.org/stable/25148625.

Hewett, T. T., Baecker, R., Card, S., Carey, T., Gasen, J., Mantei, M., Perlman, G., Strong, G., and Verplank, W. ACM SIGCHI Curricula for Human-Computer Interaction : Front Page and Full Table of Contents, 1996. URL http://old.sigchi.org/cdg/index.html.

Ide, A. PHP just grows & grows, 2013. URL http://news.netcraft.com/archives/2013/01/31/php-just-grows-grows.html.

Igbaria, M., Iivari, J., and Maragahh, H. Why do individuals use computer technology? A Finnish case study. *Information & Management*, 29(5):227–238, November 1995. ISSN 03787206. doi: 10.1016/0378-7206(95)00031-0. URL http://www.sciencedirect.com/science/article/pii/0378720695000310.

Jarczyk, A., Loffler, P., and Shipmann, F. Design rationale for software engineering: a survey. In *Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences*, volume ii, pages 577–586 vol.2. IEEE, 1992. ISBN 0-8186-2420-5. doi: 10.1109/HICSS.1992.183309. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=183309.

Jaspers, M., Steen, T., Bos, C., and Geenen, M. The think aloud method: a guide to user interface design. *International journal of medical ...*, 2004. URL http://www.sciencedirect.com/science/article/pii/S1386505604001820.

Jeffries, R., Miller, J. R., Wharton, C., and Uyeda, K. User interface evaluation in the real world. In *Proceedings of the SIGCHI conference on Human factors in computing systems Reaching through technology - CHI '91*, pages 119–124, New York, New York, USA, March 1991. ACM Press. ISBN 0897913833. doi: 10.1145/108844.108862. URL http://dl.acm.org/citation.cfm?id=108844.108862.

Jovanovic, J. 10 UI Design Patterns You Should Be Paying Attention To. *Smashing Magazine*, 2009. URL http://www.smashingmagazine.com/2009/06/23/10-ui-design-patterns-you-should-be-paying-attention-to/.

Kallio, T. and Kaikkonen, A. Usability testing of mobile applications: A comparison between laboratory and field testing. *Journal of usability studies*, 2005. URL http://lib.tkk.fi/Diss/2009/isbn9789522481900/article6.pdf.

Keil, M., Beranek, P. M., and Konsynski, B. R. Usefulness and ease of use: field study evidence regarding task considerations. *Decision Support Systems*, 13(1):75–91, January 1995. ISSN 01679236. doi: 10.1016/0167-9236(94)E0032-M. URL http://www.sciencedirect.com/science/article/pii/0167923694E0032M.

Kjeldskov, J. and Stage, J. New techniques for usability evaluation of mobile systems.

International Journal of Human-Computer Studies, 60(5-6):599–620, May 2004. ISSN 10715819. doi: 10.1016/j.ijhcs.2003.11.001. URL http://dx.doi.org/10.1016/j.ijhcs.2003.11.001.

Kniberg, K. and Skarin, M. Kanban and Scrum, Making the Most of Both. *Lulu.com*, 2010. URL http://www.infoq.com/resource/minibooks/kanban-scrum-minibook/en/pdf/KanbanAndScrumInfoQVersionFINAL.pdf.

Kunz, W. and Rittel, H. *Issues as elements of information systems.* 1970. URL http://www.cc.gatech.edu/fac/ellendo/rittel/rittel-issues.pdf.

Landay, J. and Myers, B. Sketching interfaces: toward more human interface design. *Computer*, 34(3):56–64, March 2001. ISSN 00189162. doi: 10.1109/2.910894. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=910894.

Larman, C. and Basili, V. Iterative and incremental developments. A brief history. *Computer*, 36(6):47–56, June 2003. ISSN 0018-9162. doi: 10.1109/MC.2003.1204375. URL http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=1204375.

Lazar, J., Feng, J., and Hochheiser, H. *Research methods in human-computer interaction.* 2010.

Lederer, A. L., Maupin, D. J., Sena, M. P., and Zhuang, Y. The technology acceptance model and the World Wide Web. *Decision Support Systems*, 29(3):269–282, October 2000. ISSN 01679236. doi: 10.1016/S0167-9236(00)00076-2. URL http://dx.doi.org/10.1016/S0167-9236(00)00076-2.

Lee, J. and Lai, K. What's in design rationale? *Human–Computer Interaction*, 1991. URL http://www.tandfonline.com/doi/abs/10.1080/07370024.1991.9667169.

Lee, Y., Kozar, K. A., and Larsen, K. R. The Technology Acceptance Model: Past, Present, and Future, 2003. ISSN 1529-3181. URL http://aisel.aisnet.org/cais/vol12/iss1/50.

Legris, P., Ingham, J., and Collerette, P. Why do people use information technology? A critical review of the technology acceptance model. *Information & management*, 2003. URL http://www.sciencedirect.com/science/article/pii/S0378720601001434.

Lincoln, Y. and Guba, E. Naturalistic inquiry. 1985. *VALLES, M. Técnicas*, 2001. URL http://scholar.google.no/scholar?q=lincoln+and+guba+1985&btnG=&hl=en&as_sdt=0,5#3.

MacLean, A., Young, R. M., and Moran, T. P. Design rationale: the argument behind the artifact. *ACM SIGCHI Bulletin*, 20(SI):247–252, March 1989. ISSN 07366906. doi: 10.1145/67450.67497. URL http://dl.acm.org/citation.cfm?id=67450.67497.

MacLean, A., Bellotti, V., Young, R., and Moran, T. Reaching through analogy: a Design Rationale perspective on roles of analogy. *Proceedings of the SIGCHI ...*, 1991a. URL http://dl.acm.org/citation.cfm?id=108869.

MacLean, A., Young, R., Bellotti, V., and Moran, T. Questions, Options, and Criteria: Elements of Design Space Analysis. *Human-Computer Interaction*, 6(3):201–250, September 1991b. ISSN 0737-0024. doi: 10.1207/s15327051hci0603\&4_2. URL http://dl.acm.org/citation.cfm?id=1456153.1456155.

March, S. T. and Smith, G. F. Design and natural science research on information technology. *Decision Support Systems*, 15(4):251–266, December 1995. ISSN 01679236. doi: 10.1016/0167-9236(94)00041-2. URL http://dx.doi.org/10.1016/0167-9236(94)00041-2.

Marcotte, E. Responsive Web Design, 2010. URL http://alistapart.com/article/responsive-web-design/.

Molich, R. and Nielsen, J. Improving a human-computer dialogue. *Communications of the ACM*, 33(3):338–348, March 1990. ISSN 00010782. doi: 10.1145/77481.77486. URL http://dl.acm.org/citation.cfm?id=77481.77486.

Nielsen, C. M., Overgaard, M., Pedersen, M. B., Stage, J., and Stenild, S. It's worth the hassle! In *Proceedings of the 4th Nordic conference on Human-computer interaction changing roles - NordiCHI '06*, pages 272–280, New York, New York, USA, October 2006. ACM Press. ISBN 1595933255. doi: 10.1145/1182475.1182504. URL http://dl.acm.org/citation.cfm?id=1182475.1182504.

Nielsen, J. Response times: The 3 important limits. *Usability Engineering*, 1993. URL http://scholar.google.no/scholar?hl=en&q=response+times+the+3+important+limits&btnG=&as_sdt=1,5&as_sdtp=#0.

Nielsen, J. Enhancing the explanatory power of usability heuristics. *Proceedings of the SIGCHI conference on Human ...*, 1994a. URL http://dl.acm.org/citation.cfm?id=191729.

Nielsen, J. *Usability engineering*. 1994b. URL http://www.google.com/books?hl=en&lr=&id=DBOowF7LqIQC&oi=fnd&pg=PP1&dq=help+documentation+nielsen&ots=Bk05VRIVwT&sig=S37Dm7rBCCJBNjviQ_6nWWvo5oA.

Nielsen, J. Severity ratings for usability problems. *Papers and Essays*, 1995a. URL http://katsvision.com/canm606/session_2/M2_reading4.pdf.

Nielsen, J. Usability inspection methods. In *Conference companion on Human factors in computing systems - CHI '94*, pages 413–414, New York, New York, USA, April 1994c. ACM Press. ISBN 0897916514. doi: 10.1145/259963.260531. URL http://dl.acm.org/citation.cfm?id=259963.260531.

Nielsen, J. *Heuristic evaluation*. John Wiley & Sons, Inc., June 1994d. ISBN 0-471-01877-5. URL http://dl.acm.org/citation.cfm?id=189200.189209.

Nielsen, J. How to Conduct a Heuristic Evaluation. *nngroup.com*, 1995b. URL http://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/.

Nielsen, J. Why You Only Need to Test with 5 Users, 2000. URL http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/.

Nielsen, J. Usability 101: Introduction to Usability, 2012. URL http://www.nngroup.com/articles/usability-101-introduction-to-usability/.

Nielsen, J. and Molich, R. Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems Empowering people - CHI '90*, pages 249–256, New York, New York, USA, March 1990. ACM Press. ISBN 0201509326. doi: 10.1145/97243.97281. URL http://dl.acm.org/citation.cfm?id=97243.97281.

Nunamaker, J. and Chen, M. Systems development in information systems research. In *Twenty-Third Annual Hawaii International Conference on System Sciences*, volume iii, pages 631–640. IEEE Comput. Soc. Press, 1990. doi: 10.1109/HICSS.1990.205401. URL http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=205401.

Philippus, E. Architecture Spikes. 2009. URL http://www.improvement-services.nl/Articles/ArchitectureSpikes.pdf.

Rogers, Y., Sharp, H., and Preece, J. *Interaction design: beyond human-computer interaction*. 3rd edition, 2011.

Roulston, K. and Lewis, J. Learning to interview in the social sciences. *Qualitative Inquiry*, 2003. URL http://qix.sagepub.com/content/9/4/643.short.

Rubin, J. and Chisnell, D. *Handbook of usability testing: how to plan, design, and conduct effective tests*. Wiley Publishing, Inc., 2nd editio edition, 2008. URL http://www.google.com/books?hl=en&lr=&id=LyLSSg_a5kQC&oi=fnd&pg=PR7&dq=handbook+of+usability+testing&ots=jOvWw35qXK&sig=8tPlIVOXKYUEXmw1Yt_gGAfaqTU.

Schwaber, K. and Sutherland, J. The scrum guide. *Scrum Alliance*, 2011. URL http://www.scrum.org/Portals/0/Documents/ScrumGuides/Scrum_Guide.pdf.

Scott, B. and Neil, T. *Designing web interfaces: Principles and patterns for rich interactions*. 2009. URL http://www.google.com/books?hl=en&lr=&id=1qyJdxba6qgC&oi=fnd&pg=PR5&dq=Designing+Web+Interfaces+Principles+and+Patterns+for+Rich+Interaction&ots=3rnU872bqU&sig=qw7TeeW_bfkUYzznk1YEPgpGAX0.

Seale, C. Quality in qualitative research. *Qualitative inquiry*, 1999. URL http://qix.sagepub.com/content/5/4/465.short.

Shneiderman, B. *Designing the User Interface: Strategies for Effective Human-Computer Interaction.* Addison-Wesley Longman Publishing Co., Inc., 3rd editio edition, July 1997. ISBN 0201694972. URL http://dl.acm.org/citation.cfm?id=523237.

Shum, S. B. and Hammond, N. Argumentation-based design rationale: what use at what cost? *International Journal of Human-Computer Studies*, 40(4):603–652, April 1994. ISSN 10715819. doi: 10.1006/ijhc.1994.1029. URL http://www.sciencedirect.com/science/article/pii/S1071581984710299.

Spool, J. and Schroeder, W. Testing web sites. In *CHI '01 extended abstracts on Human factors in computing systems - CHI '01*, page 285, New York, New York, USA, March 2001. ACM Press. ISBN 1581133405. doi: 10.1145/634067.634236. URL http://dl.acm.org/citation.cfm?id=634067.634236.

Tate, T. Minimizing Complexity in User Interfaces, 2009. URL http://www.smashingmagazine.com/2009/10/07/minimizing-complexity-in-user-interfaces/.

Tetard, F. and Collan, M. Lazy User Theory: A Dynamic Model to Understand User Selection of Products and Services. In *2009 42nd Hawaii International Conference on System Sciences*, pages 1–9. IEEE, 2009. ISBN 978-0-7695-3450-3. doi: 10.1109/HICSS.2009.287. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4755755.

Toulmin, S. The Uses of Argument. 1958. *Cambridge: Cambridge UP*, 2003. URL http://scholar.google.no/scholar?hl=en&q=the+uses+of+argument+&btnG=&as_sdt=1,5&as_sdtp=#3.

Venkatesh, V. and Bala, H. Technology acceptance model 3 and a research agenda on interventions. *Decision sciences*, 2008. URL http://onlinelibrary.wiley.com/doi/10.1111/j.1540-5915.2008.00192.x/full.

Venkatesh, V. and Davis, F. A theoretical extension of the technology acceptance model: four longitudinal field studies. *Management science*, 2000. URL http://pubsonline.informs.org/doi/abs/10.1287/mnsc.46.2.186.11926.

Venkatesh, V. and Morris, M. Why don't men ever stop to ask for directions? Gender, social influence, and their role in technology acceptance and usage behavior. *MIS quarterly*, 2000. URL http://www.jstor.org/stable/3250981.

Venkatesh, V., Morris, M., Davis, G., and Davis, F. User acceptance of information technology: Toward a unified view. *MIS quarterly*, 2003. URL http://www.jstor.org/stable/pdfplus/30036540.pdf?&acceptTC=true&jpdConfirm=true.

Walkin, B. Managing UI Complexity, 2009. URL http://www.brandonwalkin.com/blog/2009/08/10/managing-ui-complexity/.

Wroblewski, L. Sign Up Forms Must Die, 2008. URL http://alistapart.com/article/signupforms.

Wroblewski, L. Mobile first. 2012. URL http://scholar.google.no/scholar?hl=en&q=luke+wroblewski+mobile+first&btnG=&as_sdt=1,5&as_sdtp=#0.

Yi, M. Y. and Hwang, Y. Predicting the use of web-based information systems: self-efficacy, enjoyment, learning goal orientation, and the technology acceptance model. *International Journal of Human-Computer Studies*, 59(4):431–449, October 2003. ISSN 10715819. doi: 10.1016/S1071-5819(03)00114-9. URL http://www.sciencedirect.com/science/article/pii/S1071581903001149.

Appendix A Initial Web Surveys

Edit this form

Spørjeskjema for spelarar i organiserte idrettslag

I samband med ein masteroppgåve i informasjonsvitenskap ved Universitetet i Bergen vil det bli utvikla ein web-applikasjon som skal hjelpe spelarar med å gi konkret beskjed om planlagt fråver frå planlagte trenings- og kampaktivitetar. For å få oversikt over dagens situasjon, vanar og oppfatningar ber me deg som spelar på eit slikt lag å hjelpe oss ved å ofre nokre minutt til å svare på nokre enkle spørsmål. Undersøkinga bør normalt ikkje ta meir enn 5 minutt.

* Required

Kjønn *

- Mann
- Kvinne

Alder? *

- 0-15 år
- 15-20 år
- 20-30 år
- 30-50 år
- 50+ år

Kva idrett utøver du? *

Du kan velje fleire enn eitt alternativ

- Fotball
- Handball
- Basketball
- Volleyball
- Other:

Dersom du ikkje har mulighet til å komme på trening, korleis gir du vanlegvis beskjed til trenar/ledar for ditt lag? *

- Via SMS
- Via e-post
- På Facebook
- Eg ber nokon andre på mitt lag eller foreldrene mine gi beskjeden vidare
- Gjennom eigen programvare/web-applikasjon som laget bruker
- Eg gir ikkje beskjed

Har det nokon gong hendt at du har uteblitt frå trening utan å gi konkret beskjed? *

- Ja
- Nei
- Hugsar ikkje

Dersom ja på førre spørsmål, kva var grunnen til at du ikkje ga beskjed?

- For å unngå konflikt med trenaren, "det er lettare å få tilgivelse enn tillatelse"
- Eg gløymte det
- Det er tungvint å måtte seie frå kvar gong eg ikkje kjem på treing
- Trenaren har sagt det ikkje er nødvendig
- Other:

Føler du at du og laget ditt kan ha nytte av eit spesialisert verktøy på nett for å gi beskjed om oppmøte og eventuelt fråver? *

- Ja
- Nei

Dersom dette verktøyet skal vere på nett, kva plattformer er det sannsynleg at du kjem til å bruke det på? *

Du kan velje fleire enn eitt alternativ

- [ ] PC
- [ ] Nettbrett / iPad
- [ ] Mobiltelefon

Er det ønskeleg for deg som spelar å kunne sjå ei oversikt over kven som kjem på ei trening? *

- Ja
- Nei

Kva er vanlege grunnar til at du ikkje kjem på trening? *

- [ ] Sjuk
- [ ] Skade
- [ ] Opptatt med lekser/studier
- [ ] Vekkreist/opptatt med jobb
- [ ] Other:

Submit

100%: You made it.

Never submit passwords through Google Forms.

Edit this form

Spørjeskjema for trenarar/lagleiarar i organiserte idrettslag

I samband med ein masteroppgåve i informasjonsvitenskap ved Universitetet i Bergen vil det bli utvikla ein web-applikasjon som skal hjelpe trenarar/lagleiarar for organiserte idrettslag med å organisere og planlegge trenings- og kampaktivitetar. For å få oversikt over dagens situasjon, vanar og oppfatningar ber me deg som trenar/lagleiar for eit slikt lag om å hjelpe oss ved å ofre nokre minutt til å svare på nokre enkle spørsmål. Undersøkinga bør normalt ikkje ta meir enn 10 minutt.

* Required

I kva idrett trenar/leiar du eit organisert lag? *

- ○ Fotball
- ○ Handball
- ○ Basketball
- ○ Volleyball
- ○ Other: ______

Kva slags rolle har du i laget? *

Du kan velje fleire enn eitt alternativ

- ☐ Trenar
- ☐ Lagleiar/Oppmann
- ☐ Other: ______

Kva aldersgruppe høyrer laget du trenar/leiar til? *

- ○ 0-10 år
- ○ 10-15 år
- ○ 15-20 år
- ○ 20-23 år
- ○ Seniorlag (alle aldrar)

Kor ofte hender det at spelarar på ditt lag er fråverande frå trening utan å gi konkret beskjed? *

"Svært ofte" vil vere fleire spelarar, fleire gongar i veka

	1	2	3	4	5	
Svært sjeldan	○	○	○	○	○	Svært ofte

Ser du på dette som eit problem for deg som trenar/leiar? *

- ◯ Ja
- ◯ Nei

Dersom ja på førre spørsmål, nemn grunnar til at dette er problematisk

Nyttar ditt lag digitale verktøy for å organisere treningsaktivitet? *

- ◯ Ja
- ◯ Nei

Dersom ja på førre spørsmål, kva verktøy nyttar ditt lag?

Du kan velje fleire enn eitt alternativ

- ☐ Facebook events
- ☐ Doodle
- ☐ Tekstmelding
- ☐ Annan programvare/web-applikasjon
- ☐ Other:

Kva del av organiseringa vert desse digitale verktøya nytta til?

Du kan velje fleire enn eitt alternativ

- ☐ Kartlegge antal spelarar som kjem på trening i forkant
- ☐ Motta beskjed om eventuelle fråver
- ☐ Føre statistikk over treningsoppmøte
- ☐ Gi beskjedar om endring av treningstid, stad e.l
- ☐ Other:

Med dagens situasjon, kor forutsigbar er oppmøtesituasjonen? *

Dvs, er det lett for deg som trenar å planlegge øktene ut frå kven og kor mange som kjem på trening? I kor stor grad har du til einkvar tid oversikt over kven som skal komme på ein planlagt aktivitet?

	1	2	3	4	5	
Svært uoversiktleg	◯	◯	◯	◯	◯	Svært oversiktleg

Føler du at du og laget ditt har behov for eit spesialisert verktøy på nett for å gi beskjed om

oppmøte og eventuelt fråver? *

- ○ Ja
- ○ Nei

Kva plattform nyttar du hovudsakleg Internett på? *

Surfing, organisering, sosiale medier osb.

- ○ Stasjonær/bærbar PC
- ○ Nettbrett
- ○ Mobiltelefon

Kva eventuell systemfunksjon ser du på som nyttig/nødvendig for deg som trenar/leiar? *

Du kan velge fleire alternativ

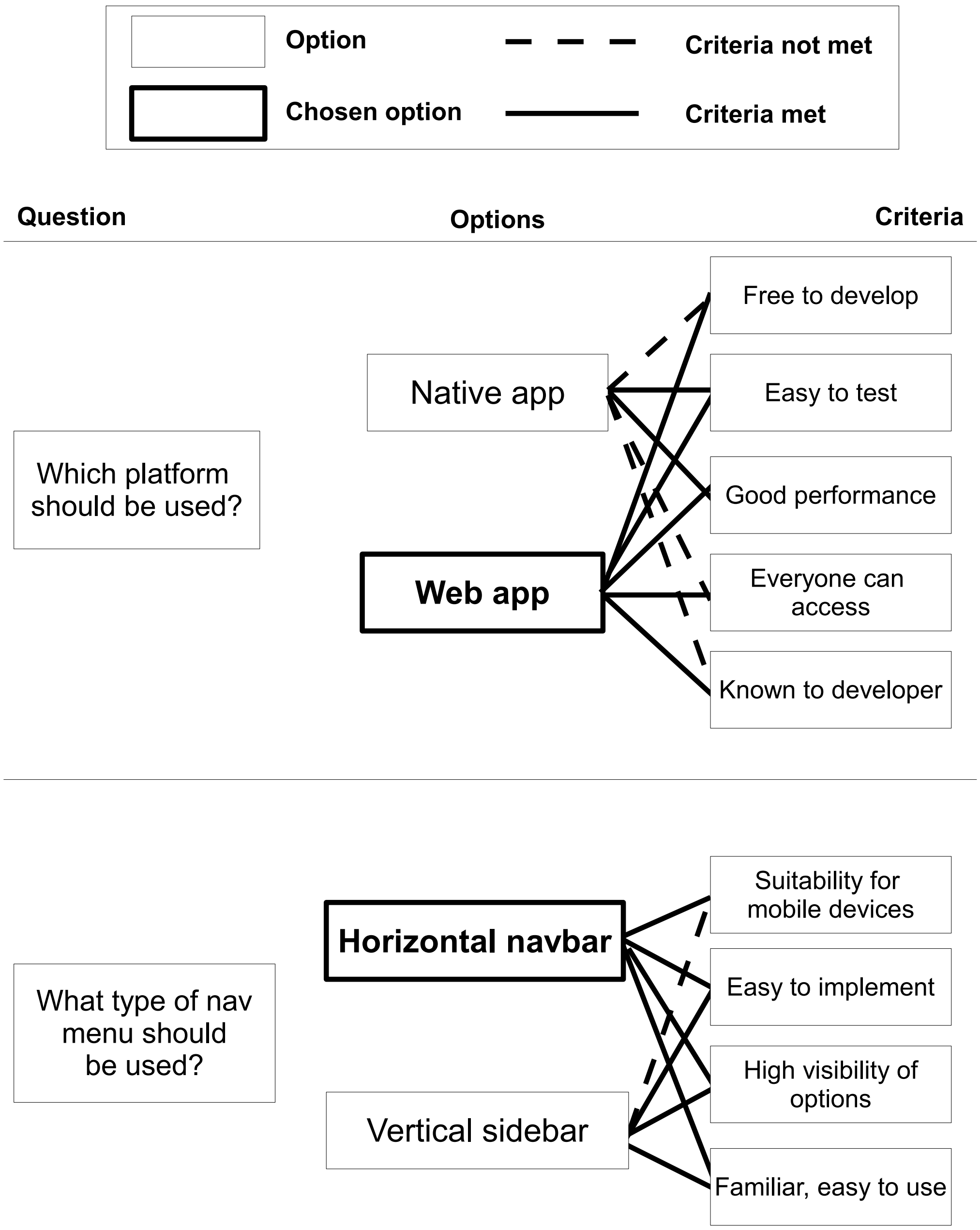
- ☐ Å i forkant av treninga kunne sjå ei liste over dei som kjem på trening
- ☐ Å få varsling samt grunnlag frå dei som ikkje kjem
- ☐ Å halde statistikk over treningsoppmøte

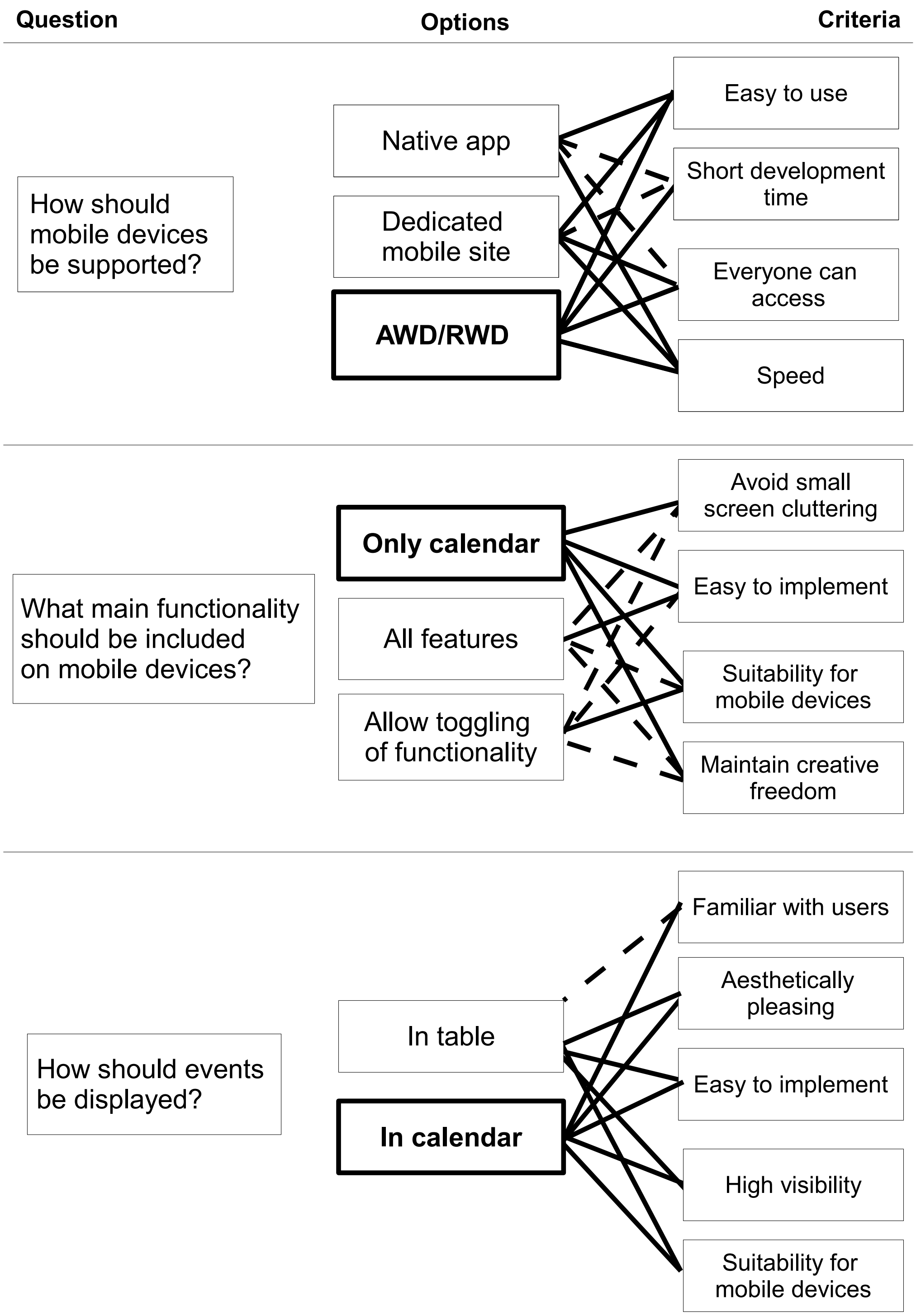
Ser du for deg andre funksjonar i eit slikt system som kan vere nyttige for deg som trenar?

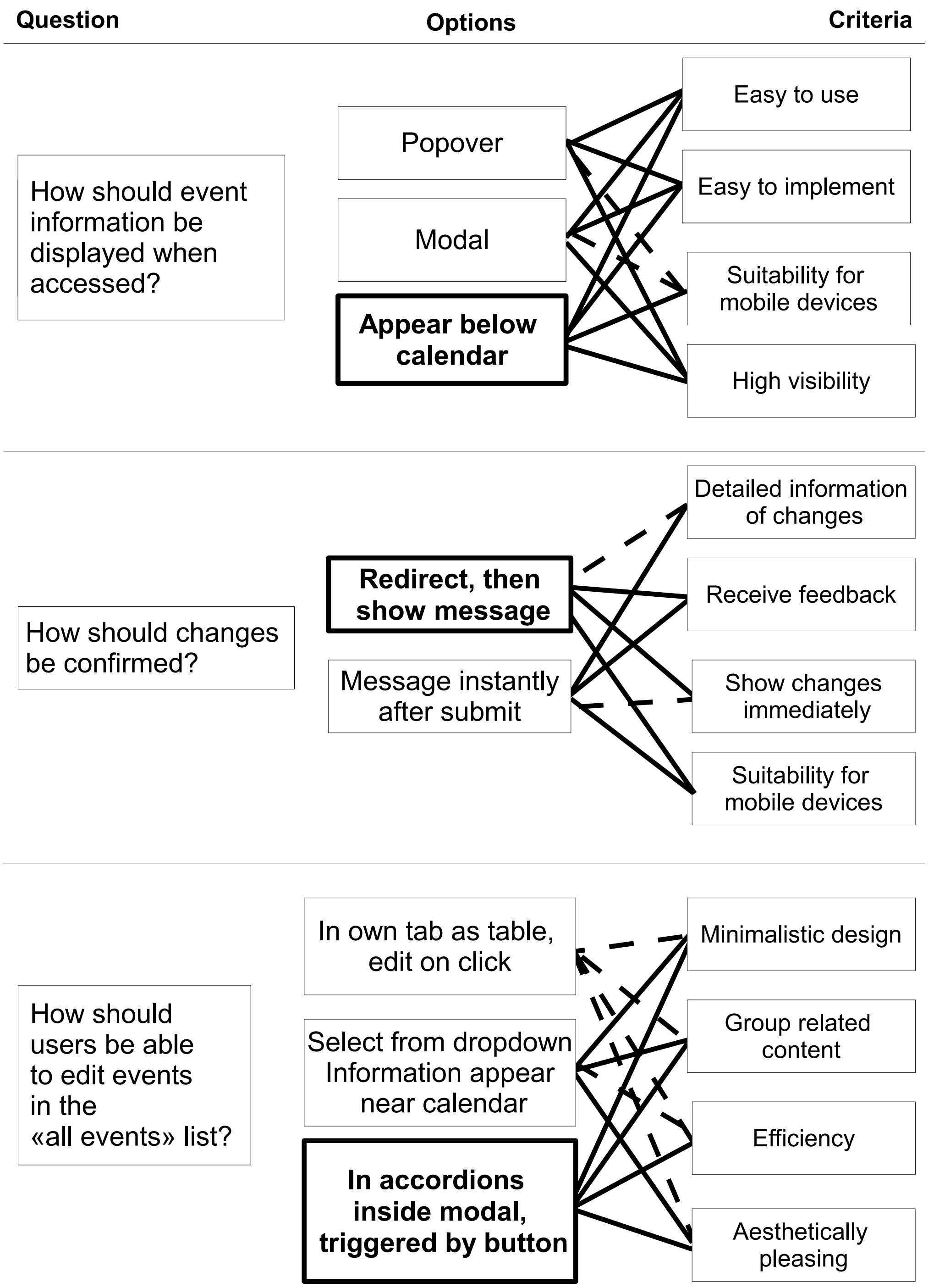
Submit

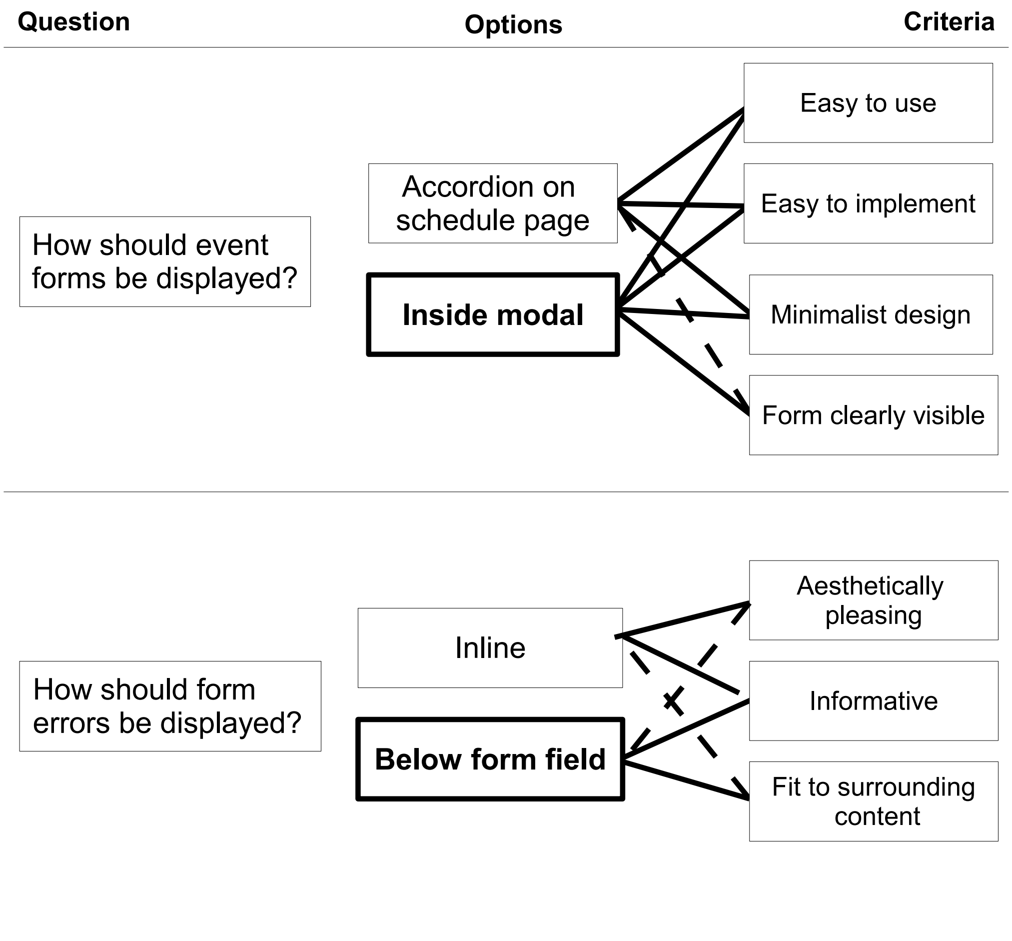
Never submit passwords through Google Forms.

Appendix B QOC representations




Question
Options
Criteria
How should mobile devices be supported?
Native app
Dedicated mobile site
AWD/RWD
Easy to use
Short development time
Everyone can access
Speed
What main functionality should be included on mobile devices?
Only calendar
All features
Allow toggling of functionality
Avoid small screen cluttering
Easy to implement
Suitability for mobile devices
Maintain creative freedom
How should events be displayed?
In table
In calendar
Familiar with users
Aesthetically pleasing
Easy to implement
High visibility
Suitability for mobile devices


Question
Options
Criteria
How should event information be displayed when accessed?
Popover
Modal
Appear below calendar
Easy to use
Easy to implement
Suitability for mobile devices
High visibility
How should changes be confirmed?
Redirect, then show message
Message instantly after submit
Detailed information of changes
Receive feedback
Show changes immediately
Suitability for mobile devices
How should users be able to edit events in the «all events» list?
In own tab as table, edit on click
Select from dropdown Information appear near calendar
In accordions inside modal, triggered by button
Minimalistic design
Group related content
Efficiency
Aesthetically pleasing


Question
Options
Criteria
How should event forms be displayed?
Accordion on schedule page
Inside modal
Easy to use
Easy to implement
Minimalist design
Form clearly visible
How should form errors be displayed?
Inline
Below form field
Aesthetically pleasing
Informative
Fit to surrounding content

Appendix C Evaluative Web Survey

Edit this form

Evaluering av webapplikasjon

Les du dette, betyr det at du har vore ein av dei som har brukt min web-applikasjon rockEnroll den siste tida. Eg vil gjerne at du svarar på nokre søte små spørsmål vedrørande din opplevelse av web-applikasjonen. Det bør ikkje ta mange minuttane.

* Required

Kjønn *

- Mann
- Kvinne

Alder *

- 0-15 år
- 15-20 år
- 20-30 år
- 30-50 år
- 50+ år

Rolle i laget? *

- Spelar
- Trenar
- Spelande trenar

Brukarvennlighet

Dei neste påstandane vedrører din oppfattelse av brukarvennligheten til web-applikasjonen.

Å lære seg korleis ein bruker web-applikasjonen var enkelt for meg *

	1	2	3	4	5	
Særs einig	○	○	○	○	○	Særs ueining

Det var raskt og enkelt å registrere seg. *

	1	2	3	4	5	
Særs einig	○	○	○	○	○	Særs ueining

Det var raskt og enkelt å finne og bli medlem av laget mitt. *

	1	2	3	4	5	
Særs einig	○	○	○	○	○	Særs ueining

Det var raskt og enkelt å finne fram i kalenderen *

	1	2	3	4	5	
Særs einig	○	○	○	○	○	Særs ueinig

Det var raskt og enkelt å melde seg opp til eller av frå ei hending

	1	2	3	4	5	
Særs einig	○	○	○	○	○	Særs ueining

Web-applikasjonen var enkel å bruke på mobil og nettbrett
Svar på dette spørsmålet dersom du brukte applikasjonen på desse einhetane

	1	2	3	4	5	
Særs einig	○	○	○	○	○	Særs ueinig

Totalt sett var web-applikasjonen enkel å bruke *

	1	2	3	4	5	
Særs einig	○	○	○	○	○	Særs ueinig

Nyttighet

Dei neste påstandane vedrører din oppfattelse av nyttigheten til web-applikasjonen.

For trenar: Mitt lags bruk av web-applikasjonen gjorde oppmøtesituasjonen meir forutsigbar

	1	2	3	4	5	
Særs einig	○	○	○	○	○	Særs ueinig

For spelar: Bruk av web-applikasjonen gjer at eg får gitt beskjed om planlagt fråver meir effektivt samanlikna med andre teknologiar.

	1	2	3	4	5	

Særs einig ○ ○ ○ ○ ○ Særs ueinig

Web-applikasjonen sine funksjonar passa til formålet *

	1	2	3	4	5	
Særs einig	○	○	○	○	○	Særs uеining

Eg ønskar at me skal halde fram med å bruke web-applikasjonen. *

	1	2	3	4	5	
Særs einig	○	○	○	○	○	Særs ueinig

Generelt

Her følgjer generelle spørsmål om din bruk av web-applikasjonen

Var du positiv til å bruke web-applikasjonen når laget ditt begynte å bruke den? *

- ○ Ja
- ○ Nei

Dersom nei, kvifor?

- ○ Dei teknologiane som finst tilfredsstiller behovet
- ○ Det er stress med enda ei innlogging, enda eit system å forholde seg til
- ○ Eg syns ikkje det er viktig å sei frå om eg kjem på trening eller ikkje
- ○ Other:

Hendte det at du ikkje meldte frå om du kom på ei eller fleire økter? *

- ○ Ja, fleire økter
- ○ Ja, ei økt
- ○ Nei

Var det eit irritasjonsmoment når andre på laget ikkje svarte? *

- ○ Ja
- ○ Nei

Kvifor brukte du hovudsakleg web-applikasjonen? *

- ○ Fordi det var enkelt og raskt å bruke
- ○ Fordi det var nyttig for laget
- ○ Fordi trenaren la press på oss til å bruke det

Other:

Kunne du tenkt deg ein eller fleire av dei følgjande funksjonane?

Du kan krysse av fleire boksar

- App for smarttelefon
- Facebook-integrasjon
- Internt meldingssystem
- Other:

Har du andre forslag til funksjonalitet som kunne forbetra din opplevelse av web-applikasjonen?

Ingen idèar er dumme

Submit

Never submit passwords through Google Forms.

Appendix D Interview Guide

1. Kva følte du om forutsigbarheten i oppmøtesituasjonen før rockEnroll blei tatt i bruk?

2. Kva følte du om forutsigbarheten i oppmøtesituasjonen etter det blei tatt i bruk? Både som trenar og spelar.

3. Som trenar, hadde du eit inntrykk av at spelarane dine var nøgd med rockEnroll? Følte du at du måtte legge press på dei for at dei skulle bruke det?

4. Som spelar, var du nøgd med rockEnroll? Kva var bra, kva var ikkje bra?

5. Kva syns du spesifikt om nytteverdien til rockEnroll? Kva er din oppfatning av kor eigna rockEnroll i utgangspunktet er for å kunne gjere oppmøtesituasjonen meir forutsigbar?

6. Kva syns du spesifikt om brukarvennligheten i rockEnroll?
   - Kva syns du om kalenderen og generell funksjonalitet?
   - Kva er din oppfatning av mobilversjon?
   - Kva er din oppfatning av hurtighet (lastetid, databruk, osv.)?

7. Fleire spelarar rapporterte at dei ikkje brukte systemet jevnlig, sjølv om dei aller fleste svarte at dei i utgangspunktet var positivt innstilt til å bruke det, kvifor trur du det var slik? -
   - Kva kan vere grunnar til at dei ikkje brukte det, f.eks kva var den største svakheten til systemet, som du trur kan ha vore medverkande til at folk ikkje har brukt det så mykje som først forventa/håpt?
   - Kva kunne vore gjort for å forbetre rockEnroll, som kunne vore med på gjere systemet meir nyttig for oppgåva og auke bruken?
   - Kva kan du *ikkje* bruke rockEnroll til i per i dag?

8. Korleis trur du eventuell introduksjon/opplæring i korleis ein bruker systemet eller meir hjelp/dokumentasjon integrert i systemet hadde påvirka bruken?

9. Er det noko anna du vil sei som omhandlar oppmøtesituasjonen og rockEnroll?