UiT The Arctic University of Norway

Faculty of Science and Technology
Department of Computer Science

Storing and representing smart nudges in a user profile

Marius Johan Mæland

INF-3981 Master's thesis in Computer Science - June 2020

For my friends and family, thank you for the support.

“It is better to be drunk than dead.”
–Eyvind Hellstrøm

Abstract

Globally, both adults and adolescents are insufficiently physically active. Lack of physical activity is one of the leading risk factors for non-communicable diseases worldwide. The smart nudge project's goal is to develop a system that nudges people to both be more physically active and choose environmentally-friendly transportation. The research in this thesis focuses on physical activity and how to create a user profile that supports smart nudging.

Nudging is a method used to influence the behavior and choices of people without forbidding any options. Smart nudging is a further development of nudging, by profiling the users to see how different users are influenced, personalized nudges for each user can be built.

The research in this thesis use applied research to design a user profile that supports smart nudging. Information from prior research is combined to determine what data to store in the user profile. For a user profile to support smart nudging, it has to contain personal information about the user, such as behavior, and preferences on nudges. To learn user preferences, nudges given to the user in the past are stored so it can be utilized to calculate preferences on different nudge components. A user profile design is created. The design is based on the information to store such as nudge metadata, user preferences, and how the system around the user profile should work. The user profile design is used to implement a demonstrator that demonstrates how a user profile can be formatted and stored in a database.

Acknowledgements

Firstly, I would like to thank my supervisor Randi Karlsen for your advice and feedback during this project.

Thanks to Lars Karlsen and Raymon S. Hansen from the Open Distributed Systems (ODS) research group for good talks and discussions.

Thanks to my brother Ruben Mæland for reading this thesis and providing feedback.

Finally, I would like to thank my family and friends for all the encouragement and support.

Contents

List of Figures

List of Tables

List of Abbreviations

API Application Programming Interface

EU The European Union

GDPR General Data Protection Regulation

GPS Global Positioning System

HTTP Hypertext Transfer Protocol

JSON JavaScript Object Notation

ODS Open Distributed Systems

PSD Persuasive System Design

SLR Structured Literature Review

UiT University of Tromsø

URL Uniform Resource Locator

WHO World Health Organization

XML Extensible Markup Language

1 Introduction

1.1 Motivation

The primary motivation for this research is that many people are struggling to be physically active. World Health Organization (WHO) discovered that globally, around 23% of adults above 18 years were not sufficiently active in 2010[21]. By not being sufficiently active, the risks of getting cancer, heart disease, stroke, and diabetes increases by 20-30%[22]. To be able to participate in the research and development of a system that can help people be more physically active and improve people's health and life quality motivates me to conduct this research. This research is a part of a smart nudge project that aims to provide users with nudges that motivate them to be more physically active and choose environmentally-friendly transportation.

Nudging is a method used to influence people's behavior and decisions. It can be be used to influence people's subconsciousness, alter the context a decision is made in, or give a short motivational text to influence people's behavior[23]. Smart nudging is a digital form of nudging that aims to motivate people to choose environmentally friendly transportation or increase physical activity[17]. A smart nudge has to be dynamic to provide tailored nudges for each user based on user preferences and behavior. To tailor smart nudges accurately to different users, the system generating these nudges have to be personalized. To personalize a system, a user profile is needed. The user profile should contain the users' preferences and behavior to know how each user is best influenced.

1.2 Purpose

This thesis aims to investigate and demonstrate how information about prior nudges can be presented in a user profile. When designing a nudge, prior nudges is useful for the system to see how the user reacts to different nudges. By storing this information, the nudge design component can utilize it to build accurate and personalized nudges for each user and influence them as much as possible.

1.3 Goal

This thesis is a part of the smart nudge project that aims to give personalized nudges to the users. In order to personalize nudges to the users, the system has to learn the preferences of the users to know how different users are influenced.

The main research question of the thesis is:

> How can a user profile support smart nudging?

Some subquestions have been identified to help answer the main research question.

Subquestion one:

> What smart nudge information should be included in the user profile?

Subquestion two:

> How should the smart nudge user profile be represented?

Subquestion three:

> How can previous smart nudges be represented in the user profile?

The structure of the user profile should fit the needs of the system that generates nudges. An essential aspect of the nudge design system is that it should dynamically adapt to behavioral change caused by earlier nudges. The user profile should, therefore, enable the system to access information about how previous nudges were received and how the user reacted to them.

1.4 Challenge

The main challenge of this thesis is to define how nudges should be represented and stored in the user profile and how to represent weights of different nudge components describing how relevant they are for the different users. The nudges a user has received should be stored to learn how the user reacts to different nudges. To store historical nudges in a suitable representation makes it easier to evaluate and learn how different users are influenced. To learn the preferences of different users are essential when building nudges that best influence each user.

1.5 Methodology/ Methods

When conducting academic research, the research methods are essential to plan and steer the research. The research becomes more structured by creating a plan of what research methods to use to get the desired results.

Anne Håkansson has written about the portal of research methods and methodologies for research projects and degree projects[13]. The portal of research should help define the path of the research, meaning what methods to use to get the desired results.

The research in this thesis is a *qualitative* research, and in-depth insight into smart nudging, and how smart nudges can be stored and represented in a user profile.

The next step of the research portal is the research methods. The research method of this thesis is *applied* research. This thesis solves a practical problem and builds on prior research done on topics covered by this thesis. Prior research help decide what information is relevant to store in the user profile and how to design the user profile. A representation can be made based on the design. When the design and representation are decided, a demonstrator can be created from the design.

The research approach of this research is the *inductive* approach. This research is looking at how a smart nudge can be stored in a user profile based on the information Dalecke presents in his thesis about different nudge components[6]. Also how to represent the relevance of different nudges are to the different users, which is the user preferences on nudges.

The research strategy used in this research is *case study*. The experiment in this research is an empirical study on how a user profile can support smart nudges and discover how the user profile should be divided into different components to represent different nudges and how relevant they are for the user.

1.6 Contribution

The main contribution of this thesis is a user profile design that supports smart nudges and reflects user preferences. Metadata from prior nudges are stored to be used for calculating user preferences on different nudge components. The preferences calculated from prior nudges are tied to weights representing how to influence a user so the smart nudge system can accurately build nudges for each user in the future. The activities a user performs are stored to find the preferred activities. When storing information about activities performed, it can be used as motivation by presenting the progress in physical performance to the user.

A demonstrator is implemented to demonstrate how a user profile can be represented in JavaScript Object Notation (JSON) format and how it can be structured for storage in a database.

1.7 Delimitations

When creating a personalizing system that stores personal data about the users, user privacy is essential. The smart nudge system relies on collecting personal information from the users and analyze it to detect user preferences and behavior. The work in this thesis investigates how a user profile can support smart nudges and how the smart nudges can be represented in the user profile. The work does not focus on the privacy and protection of user data. However, since it is such a central topic when storing private data, it is briefly discussed, and a few approaches to protecting user privacy are presented.

1.8 Context of project

This research is conducted within the nudge project in the ODS research group is working on in the University of Tromsø (UiT). The nudge research project's goal focuses on nudging people to change behavior towards green transportation and physical activity to gain better health. To achieve this, a system that collects user data such as interests, preferences, and behavior is needed. The idea is to use the collected data to tailor smart nudges to each user towards a nudge goal.

The work in this master thesis builds on the work I did in my capstone project in the fall of 2019[19]. The capstone project presents information sources that provide relevant information to store in a user profile, which contains relevant information used for nudging users to increase physical activity. Other topics presented in the capstone project are data the user profile should contain, which sources can be used to gather the information, and a proposal on how a user profile can be presented[19].

1.9 Outline

The outline of the thesis is as follows:

Chapter 2 - Background
This chapter gives an introduction to nudging and related fields of work. It also provides information about user profiles for personalized content to each user, what information user profiles contain, and different representations of user profiles. This chapter also covers user privacy and the General Data Protection Regulation (GDPR) rules to have in mind when working with private user data. Lastly, relevant work done in the smart nudge project is presented.

Chapter 3 - Related work
The chapter describes the steps used to find work that relates to the research conducted in this thesis. The results of the search and the related work are presented. Similarities and differences in the related work and the work done in this thesis are presented.

Chapter 4 - Methods
The chapter describes the methods used to conduct the research and describes the steps of this research from start to end.

Chapter 5 - Design
The chapter presents dynamic smart nudges, and different media approaches

to provide nudges to users. It presents the content that should be stored in the user profile to accurately represent user preferences, behavior, and nudges given to the user in the past. A user profile design and structure are created based on the information to store. When designing the user profile, a few changes in the smart nudge system architecture was enlightened, and these changes are presented. The process of updating the user profile are discussed along with ideas on how to calculate the different preferences. Lastly, privacy measures to keep the privacy of users by design are discussed.

Chapter 6 - Implementation
The chapter explains the experiment architecture to implement. The framework and chosen tools to do the experiment and why these tools are chosen are presented. Lastly, implementation details on how the experiment is implemented are presented. The demonstrator are explained with input and output. Lastly, how to use the implemented back end are explained and requirements for the data to send it.

Chapter 7 - Evaluation
The chapter evaluates the result of the experiment, the chosen solution is discussed along with other approaches and observations made from conducting this research.

Chapter 8 - Conclusion
The chapter concludes the research and results of this thesis.

2

Background

This chapter presents the necessary background information needed to conduct this research. It starts with an introduction of nudging, the different forms of nudging, and the ethics of nudging. Then the smart nudge system architecture is presented, followed by a presentation about a related field of work. User profiles for personalizing systems are presented with the data to store and different user profile representations. User privacy in personalizing systems is presented, followed by the relevant General Data Protection Regulation (GDPR) rules. Lastly, how to design dynamic nudges and smart nudge specifications are presented, followed by a smart nudge example.

2.1 Nudge

Thaler and Sunstein[23] introduced the term nudge, and defined it as:

> *...any aspect of the choice architecture that alters people's behavior in a predictable way without forbidding any options or significantly changing their economic incentives*[23].

Nudging is used to give people better economics, a longer, healthier, and better life. In many cases, people make bad decisions that they would not have made if they had paid full attention to the options available and possessed complete information, unlimited cognitive abilities, and complete self control[23]. Choice architecture is an important topic in nudging that Thaler and Sunstein discuss in their book. Choice architecture is the context and environment where people make decisions. By changing the context and environment, people's choices can be influenced[23]. If people want to smoke or eat unhealthy, they will not be forced to choose a healthier lifestyle, because unhealthy food or cigarettes will not be removed. However, there will be nudges trying to make people live a healthier life. In some countries, cigarette boxes have pictures of consequences caused by smoking. Consequences like yellow teeth, ruined lungs, or text that says "smoking kills" to scare people not to smoke. People can freely choose to ignore these nudges because the nudges preserve freedom of choice by stating the consequences instead of removing any options.

Thaler and Sunstein present examples of nudging. One example is from schools where the cafeteria arranged the food in specific ways to influence what food the students chose. Different arrangements could be made based on what the cafeteria wanted to achieve, by arranging fruit at eye level where it is easier to spot and arrange the unhealthy food like chocolate, on places where the students have to move the sight around to spot. They could arrange the food to make the students eat healthier, maximize profits, or try to make the students choose what they would choose on their own[23]. Another example is from the airport in Amsterdam. The authorities have etched a picture of a black house fly in the urinal. This makes the men using the urinal focus on the target and aim at the fly. The results were a lesser mess around the urinals[23].

2.1.1 Digital nudging

Schneider et. al.[24] have brought nudging to the digital world with the same goals as offline nudging, to influence peoples decisions and behavior. They define digital nudging as:

> *The use of user interface design elements to guide people's behavior in digital choice environments[24].*

Digital nudging is guiding users' choices by presenting the choices in certain ways[24]. Digital nudging influence the decision of a user at the moment the user is to make a decision. Highlighting price reductions or displaying reviews are known to have a substantial effect on the users' decision[24]. One of the techniques used in digital nudging is the decoy effect. The decoy effect increases the attractiveness of a product by presenting an option alongside with an unattractive option. By using this effect, any reasonable user chooses the intended option[24].

2.1.2 Smart nudging

Karlsen and Andersen[17] introduce the term smart nudge. Smart nudging is a further development from digital nudging, and the difference is that smart nudging has the goal of providing tailored nudges for each user in the current situation. Karlsen and Andersen define smart nudging as:

> *...digital nudging, where the guidance of user behavior is tailored to be relevant to the current situation of each individual user[17].*

The term smart nudge is used in the context of influencing people's decisions towards behavior change in terms of a healthier lifestyle or choosing environmental friendly transportation. When influencing people's behavior towards a goal, the behavior will change over time, and the nudges should be based on the user's current preferences and behavior. Therefore Karlsen and Andersen believe that tailored nudges are more likely to succeed than a non tailored pre defined nudge[17]. To be able to tailor an accurate nudge, a wide variety of data is needed. The data has to be analyzed according to each user's goal and context. Then a user can be informed and nudged[17].

2.2 Ethics of nudging

Nudging is about influencing the choices and behavior of people indirectly. When trying to influence people's behavior, it immediately raises ethical concerns about manipulation. Because of the ethical concerns, it is essential to discuss the ethics regarding nudging and present the main arguments for and against nudging. Renaud and Zimmermann[16] present an overview of the arguments both for and against nudging[16]. One of the main arguments for nudging focuses on choice architecture and its constant presence. The information guiding and steering peoples' choices is present, either by choice or circumstance. Structuring the information for a good purpose is not immoral or unethical[23]. The main argument against nudging raises concerns about autonomy and how nudges work fundamentally. Nudges influence people's decisions, often without their knowledge or consent. This makes nudges inherently unethical, regardless of good intentions[16].

Transparency is one ethical issue of nudging. Do people know that the context around a choice is constructed purposely with a particular structure? People should know when the context is manipulated and for what reason[26]. A nudge should be fully transparent by having full disclosure of its presence to be ethical[27]. Luc Bovens[4] says that a nudge has the most substantial impact in the dark when people do not know about it and the purpose of the nudge[4]. Bovens is challenged on the statement that nudges work best in the dark. Some studies provide experiments with nudges that have different levels of transparency, and the results reveal that transparent nudges work just as well as the non transparent nudges[5]. Many nudges do not satisfy the above mentioned degree of transparency. Simple nudges that primarily influence the automatic system and subconsciousness of people will probably not be transparent to those exposed to the nudge[12].

Ethics concerning this research and the smart nudge system are discussed in section 5.8.

2.3 Smart nudge system architecture

Karlsen and Andersen have created a high level proposal of the architecture for a smart nudging system. Figure 2.1 illustrates the smart nudge architecture. The nudging process is a set of different tasks, where each task is one component in the figure[17].

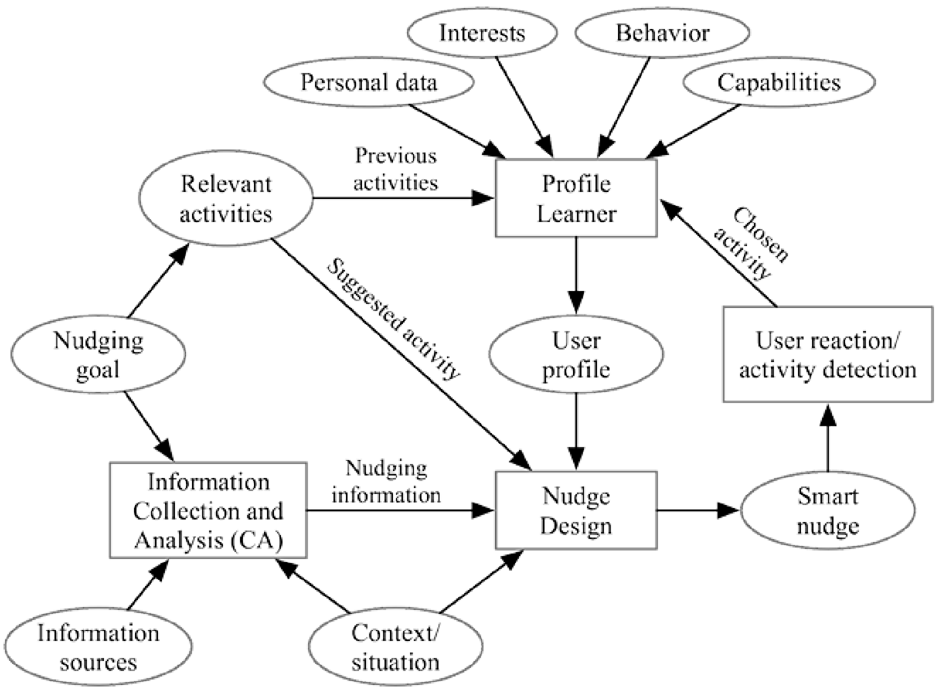

Figure 2.1: Illustration of the smart nudge architecture [17]

The nudge architecture starts with the *Nudging Goal*. The goal determines what information should be collected and what activities that are relevant to propose. *Information Collection and Analysis(CA)* collects relevant information for creating the nudge, Karlsen and Andersen calls this *Nudging Information*. The nudging information provides practical information about a suggested activity, weather, and environmental conditions to motivate the user. To tailor a personalized smart nudge to each user, a *User Profile* is essential. The *Profile Learner* generates the user profile, based on personal information that describes the user. The profile learner should monitor the users' previous chosen activities to learn each user's activity preferences and behavior. The *Nudge Design* component is where all the collected information is combined to create a personalized nudge to the specific user. The *User reaction* detects the users' reaction to a nudge and feeds the result of the nudge to the profile learner. Both the nudge and the user reaction to the nudge should be stored to adjust future nudges for the better[17].

2.4 Persuasive system design

The work in this thesis focuses on smart nudging, but there is another field of research that pursue the same goals as nudging, namely influencing people towards behavior change. Persuasive System Design (PSD) is a research area that aims to persuade users into a wanted behavior[28].

PSD is closely connected to digital nudging, as digital nudging is changing the environment in which a choice is evaluated. While PSD aims to influence user behavior through the user interface of applications and web pages. The PSD-model has been established to consider the constraints of digital persuasion. Figure2.2 illustrates the PSD-model.

The intent includes the designer of the system and the intended behavior the system is to cause[28].

The event contains the use context, user context, and technology context. This includes the specific situation in addition to the user's individual goal. The technology used can itself be bound to some aspects of persuasion. For example, persuasion through computers and mobile phones might be different[28].

The strategy contains the analysis of the different approaches to influence the users through messages or actions. Two elements in the strategy are message and route. The message is the form and content to provide for influencing. The route contains either a few solid arguments or some facts[28].

Persuasive systems are closely related to nudging, especially digital nudging. Many of the principles in persuasive systems can be seen as nudges. The main difference between persuasive systems and nudging is that nudging focuses

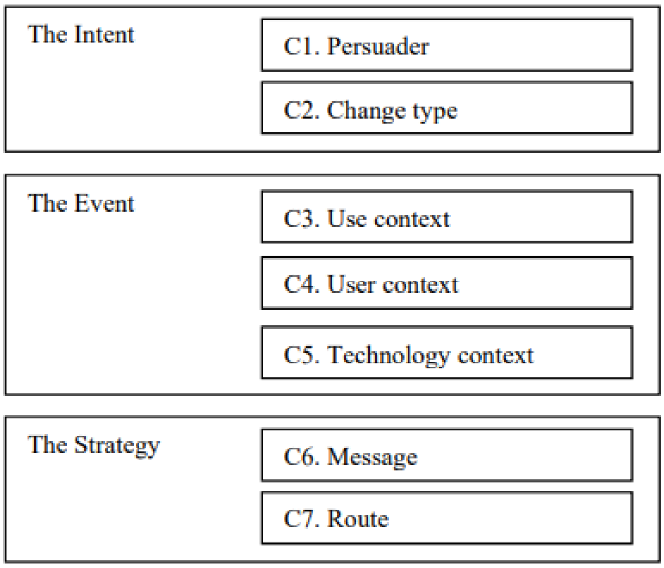

Figure 2.2: Core components in the persuasive system design model [28]

on freedom of choice, while persuasive systems do not specifically focus on it. Smart nudging differs from persuasive systems because profiling of the user is a big part of smart nudging. Smart nudging still follows many of the same principles as persuasive systems.

2.5 User profile

When creating a personalized system that provides personalized content to the users, the user profile is one of the key components. A user profile contains relevant information about the user, such as user preferences and behavior patterns. Most of the research conducted on user profiles is on personalizing on the web. While the research in this thesis focuses on the user preferences and behavior in both physical activity and what types of nudges best influence each user. The two areas are quite different, but some methods used for personalizing on the web are transferable to nudging for physical activity.

2.5.1 User profiling

When generating and maintaining a user profile, there are four design choices to make. These choices include *profile representation*, *generating an initial profile*, *capture feedback* from users and *profile learning*[9].

- *Profile representation*. The first step is to decide the representation of the profile. The profile should have the form of a general universal representation method, for example, representing the user profile as Extensible Markup Language (XML) or JavaScript Object Notation (JSON) documents.

- *Generating an initial profile*. The second step is to find a suitable technique to generate an accurate initial profile. Information for the initial profile could be gathered from questionnaires, setting default values, or gathering information from sources that might provide relevant information.

- *Capture feedback*. Step three is to capture feedback from the user. The feedback can be gathered through small surveys, monitoring users, or make the users fill the information manually in a form.

- *Profile learning*. The fourth step is to gather all the information collected and learn user preferences. The system can learn user preferences by using calculations to represent statistics about user behavior or machine learning techniques to capture patterns in user behavior.

2.5.2 User profile content

User profiles content are based on the functionality of the application using the user profile. In general, user profiles usually contain information like the user's name, birth date, gender, e-mail address, phone number, and home address. Some applications also ask the users to add identity certificate, employer, and business contact information[9, 1].

In personalizing systems, the goal is to provide relevant content to each user, and the user profile is one of the key elements for this kind of system. In a personalizing system, the user profile stores the same information as mentioned above, like a general user profile. The user profile should represent the interests, preferences, and behavior of the user in addition to the general information. By providing this information to a system, the system can give users tailored content or recommendations based on information from the user profile[9]. Information from the user profile is often used to categorize users in groups of users with the same preferences. In that way, one can assume that a recommendation that worked for some users in a group could work for other users in the same group.

Andersen, Karlsen, and Yu present some main dimensions that user profiles should contain to support smart nudging. These dimensions are listed in

Table2.1 [3, 17].

Table 2.1: Main user profile dimensions (from[3, 17])

Dimension	**Description/examples**
Personal data	Gender, age, nationality and preferred language
Cognitive style	The way in which the user process information
Device information	May be used to personalize presentation of information
Context	The physical environment where the user processes information
History	The user's historical interactions
Behavior	The user's behavior pattern
Interests	Topics the user are interested in
Intention/Goal	Intention, goals or purposes of the user
Interaction experience	The user's knowledge on interacting with the system
Domain knowledge	The user's knowledge of a particular topic
Nudging history	The user's reactions to previous nudges
Capabilities	The user's psychological and physical capacity to engage in activities

2.6 User profile representation

The work in this thesis builds on work done in my capstone project[19], where user profile for personalizing smart nudges was the main topic. The capstone project researched user profile representation, the information to store, and sources for the information to store. Some of the information presented in the capstone project is used in this section.

The information to store and the purpose of the user profile is a significant factor when deciding the user profile representation. Different structures can be chosen based on different usages. Mostly user profiles for personalizing systems are used to personalize the content on the web. When personalizing content on the web, the content is the different pages. User preferences are gathered by evaluating the pages a user spends time on. The sections below present three common user profile representation structures. The three different representations are keyword profiles, semantic networks profiles, and concept-based profiles [10]. Typically, these representations are used to discover user behavior and preferences on the web to recommend relevant content to each user accurately.

2.6.1 Keyword Profiles

The most common representation of a personalized user profile is sets of weighted keywords[10]. The keywords can either be provided manually by the user or automatically extracted from, for example, web documents. Each keyword has a weight where the weights are a numerical representation of the user's interests in different topics. The keyword representation of a user profile is one of the first to be explored for personalizing purposes. The user profile can be represented as a keyword vector and can be compared to other keyword vectors. Keyword vectors can be extracted from web pages to find relevant pages based on the user's keyword profile.

A problem with the keyword profile is the polysemy problem[10]. The polysemy problem is caused by homonyms, which are words with more than one meaning. If the user profile weight a word that has two different meanings, for example, a crane, it can be either be a bird or a machine used at construction sites. By weighting the word crane with no other context, it is hard for a system to know which one is relevant to the user and makes it hard to personalize content to the user accurately.

2.6.2 Semantic Network Profiles

The semantic network profile is based on the keyword profile but further developed to avoid the polysemy problem. In the semantic network profile, each word is a node, and each node has a weight representing relevance to the user, similar to the words in the keyword profile. The difference is that nodes have relations to other nodes that are weighted based on co occurrences of the two words. By weighting pairs of words, it helps to detect how relevant the different compositions of the words are for the user[10].

2.6.3 Concept based profiles

The concept-based profiles are similar to the semantic network-based profiles since conceptual nodes and relationships between nodes represent them both. The nodes in the concept-based profile represent abstract topics considered attractive to the user instead of specific words or related words. The concept-based profile is also similar to the keyword-based profile in that it often is represented as vectors with weighted keywords. The difference between the keyword profile and the concept based profile is that the features represent concepts instead of words or sets of words. There are several approaches to find how much a user is interested in a topic. The basic approach is to use a numerical value or weight to represent how relevant a topic is for the user. [10].

2.7 Privacy and data protection

When building a system to gather and store private information about the users, privacy and data protection are important topics to evaluate in order to preserve user privacy, and protect the stored data. The following sections will present user privacy and the GDPR rules that applies to this research.

2.7.1 User Privacy

Today, many different software applications provide services to users, such as online shopping, social networking, and physical activity monitoring[25]. These applications often collect data from the user. Sometimes an application collects data the users do not expect the application to collect based on the functionality of the application. This can make users feel that their privacy is invaded[25]. When users are installing a new application, they have the opportunity to read through the terms of use, which includes the required permissions. Installation can be canceled if the user does not agree with the terms. The terms should include what data is collected and how the data is handled. However, the terms of use are often very long, with much text formulated in a way that makes it hard to understand.

Studies show that most users tend to accept the permissions an application is asking for without reading the terms of use, and blindly trust the application[8]. When the permissions requested are accepted without reading them, the application may collect data the users do not expect to be collected, from the functionality of the application. Or the application may use or handle the data in a way the user does not agree with. When data is collected that the user do not expected, the users' privacy is compromised[25]. There are some solutions to minimize the mismatch between the behavior of the application and the users' expectations. The software developers can consider user expectations and try to develop the application's data collection as close as possible to users' expectations. One approach is to make the application transparent by informing the users about what data is collected and why the application needs this data in a short, accurate message when a feature is activated[25].

Privacy in applications can be compared to the transparency of nudges mentioned above. Indeed, the default permissions could be considered a nudge itself. If the users know what data the application collects and what the data is used for, they can choose whether or not they want to use the application. However, most importantly, the privacy of the users is not invaded without them knowing.

2.7.2 General Data Protection Regulation

The research of this project is in happening Norway, and Norway is following the The European Union (EU) directives, which means the GDPR have to be taken into account. GDPR is a set of regulations to harmonize data privacy laws across Europe. GDPR makes it easier for users to know what data are collected and how they can control the information stored about them. Users can request to have all data stored about them deleted if desired[11].

Since this research is focusing on a user profile that should store personal information about users, it is necessary to look at the rules that relate to the research.

In Chapter 2, Art.5 of the GDPR rules, the processing of personal data is in focus. The rules state that the data should not be abused and that users should know what data are collected and why it is needed. Also, the rules state that the data shall only be collected for specified and legitimate purposes. Only the relevant and necessary data to serve the purpose of the processing of data shall be collected[11].

Chapter 2 Art.6 states that the user has to give consent to the processing of the user's data for one or more specific reasons for the processing to be lawful[11].

In chapter 3 of the GDPR rules, the rights of the user are in focus. In Art.15, the rules apply to the right of access to the data by the user. The user has the right to get confirmation whether or not personal data about the user are being processed. Where personal data about a user are processed, the user has rights to[11]:

- Gain access to personal data
- Know the purposes of the processing
- Know the categories of personal data concerned
- Know the period for which the personal data are stored, or, if not possible, the criteria used to determine that period
- Demand erasure of personal data or restriction of processing of personal data

Art.16, in chapter 3, states that the user has all rights to obtain rectification of any inaccurate personal data concerning the user.

In chapter 3, Art.17 rules on erasing personal data ('right to be forgotten') are presented. The user shall have the right to get the personal data erased based on this reasons[11]:

- If the personal data are no longer necessary concerning the purposes which they were collected or otherwise processed
- If the user withdraws consent on the processing, and where there is no other legal ground for the processing
- If the personal data have been unlawfully processed

Art.20, in chapter 3, states that users have the right to receive all the personal data concerning the user, which the user has to provide to the application. The data delivered to the user should be structured in a common machine-readable format[11].

In chapter 4, Art.25 states that the application that is collecting personal information should be implemented with security and privacy measures by design. The necessary measures, like data minimization and necessary safeguards, should be integrated into the processing of personal data. The application should, by default, only use the necessary data to perform a process.

The rules and regulations presented above are discussed concerning the work done in this thesis in section 5.6.

2.8 Dynamic nudge design

When generating personalized smart nudges, the nudges have to be dynamically designed with different nudge components to influence each user accurately. The goal is to create specific nudges personalized to each user based on the user's preferences. The following sections present details about how to divide a nudge into different components and the specifications of the different nudge components. Lastly, there are presented an example of a nudge built from different components.

2.8.1 Smart nudge components

When nudging different users, presenting the most effective nudge for each user is one of the steps towards personalized nudges. A nudge can be selected from a set of predefined nudges. But to provide tailored nudges to each user, the nudges should be dynamically generated. Dalecke[6] presents how to dynamically generate a nudge by dividing the nudge into five different categories. The five different categories Dalecke[6] divide the nudge in to are activity, content, motivation, effect, and presentation.

The *activity* is the desired behavior of the user and the most important part of the nudge. The activity of the nudge helps to decide what information is relevant to include in the nudge. By clearly stating the desired behavior a nudge wants from a user, the transparency of the nudge is increased[6]. Examples of nudge activities can be a nudge towards:

- a walk
- a hike in the mountain
- a ride on the bicycle

The *content* is the information that should be presented in the nudge. Each user should get customized information based on the proposed activity. The content presented should make it easier for the user to perform the activity proposed[6]. The information can be:

- weather conditions
- inform that the ski trail newly prepared
- inform about what time a group session at the gym starts

Psychological *effects* can be used as a way of formulating a nudge, or it can be a piece of extra information that helps to motivate the user, this is to make the nudge more attractive and try to influence the users better. Dalecke presents motivation and psychological effects as two different nudge components. However, they overlap as the motivation is defined as extra information that should motivate the user, while some of the effects also provide motivational information. Further, in this thesis, motivation is included in the psychological effects. Dalecke[6] gives an overview of different psychological effects that can be used in nudges. Dalecke[6] presents some effects tied to the topic of mobility and transportation. Since this project is focusing on better health through physical activity, some effects that can be used regarding physical activity are introduced and put in the context of physical activity.

- *Loss aversion* is one of the effects used to affect one's emotions to make a choice. People often feel a higher emotional effect of losing something than acquiring it[15]. Loss aversion can be tied to physical activity by looking at progress in the performance of a user and inform that they lose the progress if they do not maintain it.

- *Framing* is an effect that architecture the context a choice is made in. The context a decision is made in is essential for the outcome. Two statements that state the same point can have a very different impact. Framing the fuel used by fuel per distance instead of distance per fuel can have a high impact[15]. This effect can be tied to physical activity by framing a particular disease by using a public figure of stating how many people have the disease because of lacking physical activity.

- A *commitment* can be used to make the user feel that they have committed to a program[6]. For example, an exercise plan can be formed, and the user can be reminded of the commitment.

- *Simplification* is an effect that can be used to simplify a choice where there are many alternatives. In the case of going for a run, the choices are, how far should the run be, what route should be chosen that suites a specific person. With simplification, a suitable route can be proposed to the specific user, and information like distance and approximate time to walk or run the route can be provided to the user[23].

- *Social norm* is an effect that can be used to influence people directly. People often follow social norms, which can be seen as what the majority defines as ideal behavior. There are social norms for everything, how a person should look and behave are two examples. By knowing how the majority behave, this information can inform the ones who deviate from the majority, the ones who deviate are highly likely to change because people want to be normal and not deviate from the majority. Thaler and Sunstein[23] talk about the spotlight effect, which is an effect where one person feels like he is in the spotlight because he deviates from the majority and therefore feels uncomfortable, which increases the influence of social norm.

Presentation is at least as important as the content of a nudge. The presentation can increase the impact a nudge does at a user. The presentation is more guidelines for a nudge than a category[6]. Dalecke[6] presents some guidelines on presentation regarding the topic of mobility, and these guidelines are also usable regarding the topic of physical activity. These guidelines are:

- To greet the user can invoke a personal connection to the user
- Emotional messages can be used and may make the user more engaged and more likely to follow the nudge
- Stating that a nudge did work for the user may make the user feel obligated to follow another nudge
- Content can be used to generate a quality score. The score can be used to change the message. Weather can be nice, great or perfect to use a certain activity

2.8.2 Nudge example

In Dalecke's thesis, he presents an example of how a nudge can look like based on the nudge components[6]. In the example on what a nudge can look like, Dalecke first presents the different nudge components are, then all the components are connected to a comprehensive nudge. An example of a nudge can look like this:

- **Greeting:** Hi Ola
- **Activity:** you should go for a run
- **Content: Weather:** the weather is sunny and will be all day, temperatures will stay around 17 degree Celsius, perfect to **activity**
- **Content: Route:** I have looked up a route for you
- **Effect: Loss Aversion:** If you go for a run you will not loose your physical progression

The nudge example presented to a user: *Hi Ola, you should go for a run to get some exercise. The weather is sunny and will be all day. Temperatures will stay around 17 degrees Celsius, perfect to go for a run. I have looked up a route for you. If you go for a run, you will not lose your physical progression*.

2.8.3 Dynamic design

When dynamically generating personalized nudges by combining nudge components, the right components have to be used to generate accurate nudges based on user preferences. To determine the components to use when generating a nudge, all components should be weighted based on how the user reacted to prior nudges. The process of weighing nudge components to learn the user's preferences is essential to personalizing nudges. Section 2.8.1 present different categories the nudge components can be divided into. A nudge should contain components from several of these categories. A nudge should be generated to influence the user's behavior towards a goal, and the nudge components have to be chosen to serve this purpose[6].

3

Related work

To find work related to the research in this thesis, a Structured Literature Review (SLR)[29, 18] are conducted to create a list of studies relating to the subject area researched in this thesis. The SLR is divided into three phases, phase 1 is the planning phase, this is where you define the object of review, find proper research questions for the research to answer and generate a search query based on the key terms in the research questions. Phase 2 is when examining the results from running the query and choose the relevant articles. In phase 3, the relevant information is presented.

Phase 1: The main research question for this thesis is:
How can a user profile support smart nudging?

To help answer the main research question some subquestions have been identified. These subquestions are:

- What smart nudge information should be included in the user profile?
- How should the smart nudge user profile be represented?
- How can previous smart nudges be represented in the user profile?

The research questions is analyzed to extract key terms, synonyms, or other terms with the same semantic meaning that can be used to make the search a little wider. The query used in this SLR was:

("digital nudge" OR "persuasive system") AND "user profile".

Phase 2: The query from phase 1 is to be used to search on relevant online digital libraries. The libraries chosen in this SLR are IEEE Xplore[1], SpringerLink[2] and ACM digital library[3]. IEEE Xplore and ACM digital library did not have any additional filtering options that would narrow the search down to make the result more accurate. However, there was an option to include Preview-content on SpringerLink. This was checked off to make the result only contain accessible articles.

The query gave a different number of results on all three libraries. The total number of results from all three libraries was 17 articles.

- IEEE Xplore gave 0 article
- ACM digital library gave 11 articles
- SpringerLink gave 6 articles

After running the search query, the first step is to read the title and abstract of all the articles in the search results and pick any article that looks relevant to the research. From the result of 17 articles, 7 was chosen for the next step after reading the title and abstract.

The next step is to read the introduction and conclusion to filter out articles that were not as relevant as it seemed from the title and abstract. After reading the introduction and conclusion of the 7 articles, 1 of the articles was relevant to this research.

Phase 3: After phase 2, there was only 1 relevant article left. The relevant article was written by Anagnostopoulou et al.[2] and contains research on behavior change in the topic of mobility and transportation. Their research aims to leverage travel behavior and personality profiles to nudge towards sustainable transport. To create a profile on the users, they used sensor data from smartphones to see the behavior of the different users. They detect points of interest, meaning coordinates where the user spends time. They monitor the activity of the user to detect the activities a user performs, like cycling, running, walking, or using motorized transportation(car, train, bus). This information is stored in the user profile, but they do not go into detail about how they store or represent the user profile. The approach to nudge the users was to provide the users with several route options on how to get to a location, and then provide a message trying to nudge the user to pick the desired option. So each persuasive message is a specific message, and the system calculates each specific persuasive message's effectiveness by looking at the user's previous

1. https://ieeexplore.ieee.org/
2. https://link.springer.com/
3. https://dl.acm.org/

response to that same message. They define a set of persuasive messages, monitor the effectiveness of each message have on a user, and store this in the profile to represent what messages work on the different users. To detect if a message works on a user, they have implemented a short popup message asking the user if the nudge affected the user's choice.

The article written by Anagnostopoulou et al. [2] is similar to the research in this thesis because both pieces of research are making a user profile that should reflect how to influence the choices of the users, what persuasive messages influence a specific user and what do not influence the user. It also tries to find the user's behavior by using smartphone sensors to detect the mobility pattern and travel methods of the users. Anagnostopoulou et al. write about persuasive messages which are the same as nudges in this research. The work in this thesis differs from the work in the article in that the user profile should support dynamic nudging. The smart nudge system should be able to build tailored nudges for each user based on user preferences stored in the user profile that reflects the different nudge component's impact on a user. The smart nudge system differs from the work in the article that is using a set of predefined nudge messages and calculates user preferences on each nudge.

The method they use to calculate preferences is relevant to use in this research. They calculate the effectiveness of each nudge, and this can be transferable to each nudge component in this thesis.

They have implemented a popup message to ask the user if the nudge influenced their choice. This would be a good thing to adopt to the smart nudge system in an early phase before an automatic detection is looked into and implemented.

4

Methods

The research method used in this research is applied research. Applied research is a methodology often used to solve specific problems, and it can be used for all kinds of research and investigation. Applied research often applies information from prior research and data from real work to solve the problem or develop applications[13].

The research done in this thesis builds on prior research done on the topics included in this thesis. Prior research is used to gather relevant information and specifications that relate smart nudging and user profiles to help answer the research questions. The topics included in this research are nudging[23], digital nudging[24], smart nudging[17], designing dynamic smart nudges[6], user profile[10, 9] and privacy [25]. Privacy rules and regulations must be accounted for to make any necessary design and architecture choices to satisfy the regulations[11]. All of the mentioned topics are covered and presented in chapter 2.

Information collected about smart nudging, user profiles, and designing dynamic smart nudges is used to determine what information to include in the user profile. A user profile design is generated based on the information it should contain, and the nudge design. When the user profile is designed, a suitable representation format has to be chosen for storing the user profile in a database.

The design of the user profile and the data to store in it revealed the functionality of the user profile in the smart nudge system. This made it necessary to redesign the smart nudge architecture slightly. To make it clear where the different smart nudge components can fetch the required information from, and what information the different components should provide.

The privacy regulations from General Data Protection Regulation (GDPR) are used to discuss any measures to satisfy the regulations. Any measures used to satisfy the regulations, also preserve the privacy of the users, which is the goal of GDPR. The regulations in GDPR may have an impact on design and architecture choices for the smart nudge user profile because some of the regulations require privacy and protection by design.

When the user profile design is created, and the representation format is determined, a demonstrator is implemented to demonstrate a user profile representation in the database, and how the user profile should be updated based on results from prior nudges. Endpoints to extract the user profile are implemented to demonstrate how to extract data from the user profile. The data received by request are represented in a commonly used format.

After the implementation of the demonstrator, the results from running the demonstration are presented and discussed. Any problems discovered or improvements that can be made are enlightened and discussed.

5

Design

This chapter contains the designing part of the thesis. Information from the background chapter is used to create a design and representation of the user profile. First, Dynamic smart nudges are discussed. Then the content to store in the user profile is defined. The user profile design and structure are created based on the information to store in the user profile. The user profile has to stay updated for the smart nudge system to continuously provide accurate nudges, a proposal on how to update the user profile are presented. Design choices and system requirements to satisfy the General Data Protection Regulation (GDPR) rules are discussed. The design of the user profile called for an evaluation of the smart nudge system architecture and some changes done to the system architecture are presented. Ethics conserning this research and the smart nudge project are discussed. Lastly, how the smart nudge system benefits from the user profile design are discussed.

5.1 Dynamic smart nudges

As Dalecke proposed in his master thesis, a smart nudge can be a set of components put together to a short text with information and motivation for following a nudge[6]. By building the smart nudges with different components based on the user preferences, makes it dynamic. The smart nudges are also dynamic in the media used to present them to the users, this thesis does not include other representations than text, but other media can also be used and should be researched in the future. Other media presentations for nudges can be sound, pictures or short video clips. In order to support other media types, the user profile have to be designed to support them. And proper techniques to evaluate what is it in the different nudges that influence the users have to be looked into in order to learn how to use the different medias to influence the users.

An example on how a nudge can be represented using a map is presented below.

A nudge could display a travel route on a map, by pre defining a route, the nudge is using the simplification effect. If a user receives a nudge trying to influence the user to walk to an appointment, the travel route between the locations can be displayed on a map. The nudge in Figure 5.1, displays the walking path on a map. The nudge should include the distance to walk, and the approximate time it takes to walk to the appointment. The nudge should be presented as a map with the route. However, it should also have some text providing the weather conditions and some motivational information and clearly state the desired outcome of the nudge.

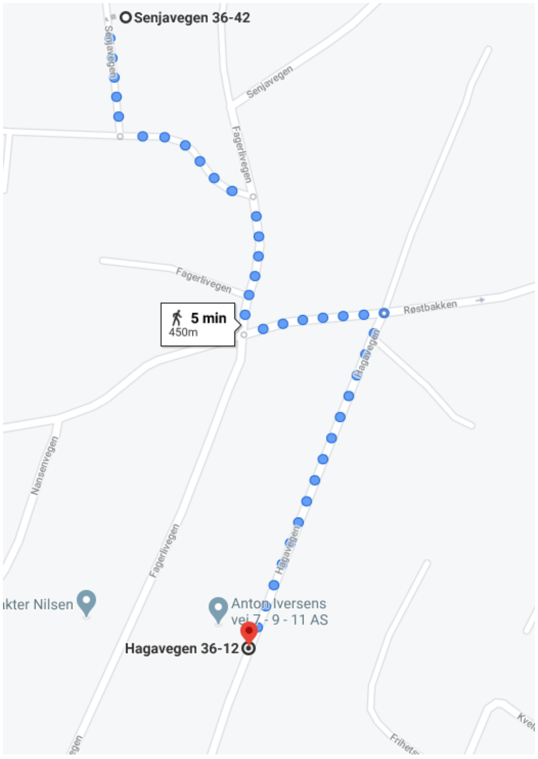

Figure 5.1: An example of a nudge that proposes a walk path to an appointment.

5.2 User profile content

In my capstone project, I evaluated the dimensions proposed by Andersen et al. [3, 17] from the physical activity aspect. In the evaluation of the dimensions, the main focus was to include only the relevant dimensions to represent what activities a user can participate in, what activities are relevant, behavior, and what type of nudges work best on the user. The evaluation resulted in a new table that are represented in Table 5.1[19]. There are added two dimensions to the table, as two more relevant dimensions came to mind. These dimensions are restrictions and prior nudges.

Table 5.1: Information to store in a user profile for smart nudging towards physical activity[19]

Dimension	**Description/examples**
Personal information	Age Gender Home address Work/school address
Equipment	What activity equipment do the user own
Restrictions	What are the distance restrictions a user have on different activities
Prior nudges	Store metadata about prior nudges to update user preferences
Nudge	Store the relevance of different approaches to the different categories of a nudge
Behavior	Store activities performed and some metadata for progress tracking and motivation
Disabilities	A user can have a disability that may exclude some activities
Activities	Store how relevant different activities are for the user

The first dimension to store in the user profile is the *Personal information*. This is data the user has to provide to the system. The data stored in this dimension are personal data about the user. Age and gender are information that can help to group users, where a nudge that worked on several users in a group may work on others in that same group. Home and work address is useful to know when nudging a user to perform an activity on a specific route or recommending an activity as a transport choice.

Equipment is a dimension in the user profile the user has to provide manual, this information is vital to the system so it can know what activities a user can perform based on equipment owned. If a user does not own skis, the user should not receive a nudge to go skiing.

Restrictions are information the user has to provide manually. Restrictions are useful to know because it reflects the distance boundaries of a user on different activities. Distance restrictions matter when proposing a predefined route for an activity to a user. If a user has some disabilities, restrictions can be essential to know when proposing a predefined route so the user can complete the activity proposed.

Prior nudges are metadata about the nudges a user has received in the past. The prior nudges provide the data needed to analyze the preferences of the user.

Nudge are the preferences a user has on different nudge components. Nudge preferences are extracted from the information stored in *Prior nudges*.

Behavior are the activities a user has performed in the past. The system should automatically detect when a user performs an activity. This information is used to calculate the preference a user has on different activities. The activity preferences are crucial when deciding what activity to propose in a nudge.

Disabilities of a user are essential because there might be some restrictions on what activities a user can perform. If a user gets an injury, the user might not be able to perform certain activities, and there is no point in nudging a user to perform such an activity.

Activities should represent the user's preferences for different activities. The activity preferences are extracted from the information in *behavior*.

5.3 Nudge requirements

When storing smart nudge information to represent how different smart nudges impact the users, the smart nudge has to be analyzed and converted to metadata. Based on Dalecke's[6] research, a smart nudge can be built by using different components. These components are useful when designing a user profile that should support smart nudges. The different components a smart nudge can be built by can be divided into groups, these groups are *activity*, *content*, *effect* and *presentation*. The user profile should reflect the different components and the user preferences on these components. Each of the nudge groups mentioned above is groups of components. For example, a psychological effect can be loss aversion, social norm, or framing. The name of the different nudge components and what group they belong in are useful when creating metadata to represent a nudge. To easily know what nudge components are used to build a nudge, metadata should be added to the nudge object so the profile learner can use the metadata to calculate preferences. Metadata should be added to the nudge object by the nudge designer, so the profile learner does not have to analyze the nudges to know what components are used to build the different nudges. The metadata should contain the name of each component group. These names should have a value that is the name of the component used from that group. In addition, the metadata should contain the time and date of when the nudge was given, and wether the nudge was followed or not.

When building a nudge, some of the nudge components are chosen based on other components used in the nudge. An example is that content can depend on the activity. The content should be different based on what activity the nudge proposes to the user. If a user receives a nudge to go skiing, the content presented should contain information about the ski trail conditions, snow conditions, and weather. While if a user receives a nudge to go for a run, the temperature and sky conditions might be enough content. The fact that content is tied to activity makes it less relevant to represent the content with relevance to the user because it depends on the chosen activity.

5.4 User profile design

When the information to store are determined, the data can be divided into different components to give a structure to the user profile. The structure should be chosen based on the data to store. The data can be categorized based on thematic, relational, privacy reasons, or it could be structured based on the design of the system that should access the stored data. The following sections present the user profile structure and examples on how to represent the different components in a suitable format.

5.4.1 User profile structure

When designing the user profile for a smart nudge system, the baseline for the design is the data that should be stored, the architecture around the user profile, what the nudge itself should contain, and how to build the nudge.

Section 2.6 presents three different structures to represent a user profile for personalizing systems. Those representations are used to personalize the content on the web by using weighted keywords to detect what topics the different users are interested in. In the smart nudge system, the goal is to detect the user's activity preferences and what nudge components can be used to best influence each user. To learn the preferences of the user, the smart nudge system has to look at the nudge and activity history of the user to find the preferences of each user. The different nudge components and activities can be represented in the user profile using the name of each nudge component and activity as keywords, the keywords should be tied to a weight that represents how relevant it is to the user. The described approach is similar to a keyword profile because every keyword has a weight representing how relevant they are to the user. However, the user profile for smart nudging will not be used as a keyword profile. The smart nudge system will choose a set of nudge components that have successfully influenced the user in the past and generate a nudge using these components. In distinction, a keyword profile is used as a keyword vector that is compared to other keyword vectors generated from different web pages and documents to find relevant content for each user. The keyword profile representation can be used to represent the user profile in the smart nudge system but will be used differently compared to personalizing systems on the web. It will choose smart nudge components based on how relevant they are to the user.

The data to be stored in the user profile should be separated based on security and thematic differences to preserve the user's privacy and make sure the user profile is structured. Highly private sections should have necessary security measures, by being stored in the edge, meaning the users device. Data that consume more storage space and are not highly private can be stored on system servers so that the storage on user's devices do not get fully consumed by the smart nudge user profile. The design created in this thesis are further developed from the design proposed in my capstone project[19]. The proposed design from my capstone project are presented in Figure 5.2.

The design in Figure 5.2 are divided in four sections, these sections are *general data*, *activities*, *behavior* and *nudge*. *General data* should contain general information about the user and activity equipment owned by the user. *Activities* should contain the preferences a user have on different activities. *Behavior* should contain the history of activities the user has performed in the past and metadata about these activities, which is the details about duration, distance travelled, time and date. The last sections are the *nudge* section, and this section should contain the user preferences on all the different components that can be used to build a nudge[19].

After evaluating the design proposed in my capstone project, a new design is developed. There are added a new section called *prior nudges* that contains metadata about nudges received in the past including the nudge messages. This

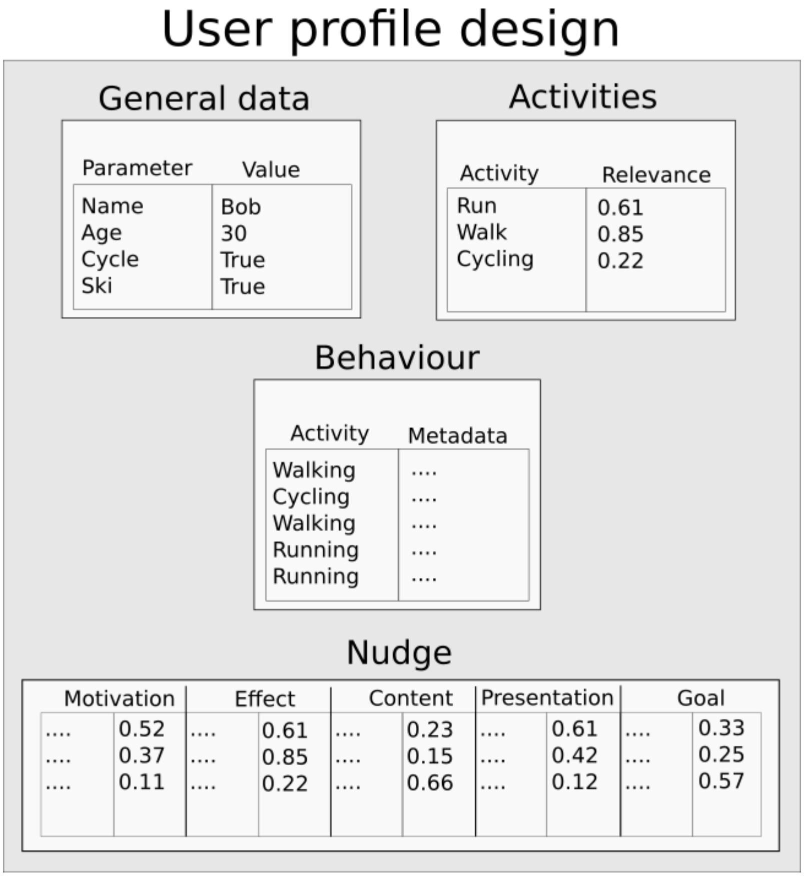

Figure 5.2: Illustration of the user profile design from my capstone project[19]

is to store the prior nudges and use them to calculate preferences on different nudge components. The *activities* section are moved to another section called *preferences*, this is because the *activities* section contained activity preferences and it makes sense thematically to store it with the other preferences. The *nudge* section is renamed to preferences because it contains the preferences a user have on different nudge components. Preferences tied to *motivation*, *content* and *goal* are removed. The *motivation* preferences are removed because the motivational factor in a nudge is a part of the psychological effect. It is therefore covered by the *effect* section of the preferences. *Content* are removed because the content that is applied to a nudge is very much depending on the activity proposed. The *goal* of the nudge, is the desired outcome of the nudge, and that is the desired activity. Therefore it is not necessary to include a goal in the preference section since activities are already represented under*activities* in the preferences section.

The new user profile design containing the changes presented above are illustrated in Figure 5.3

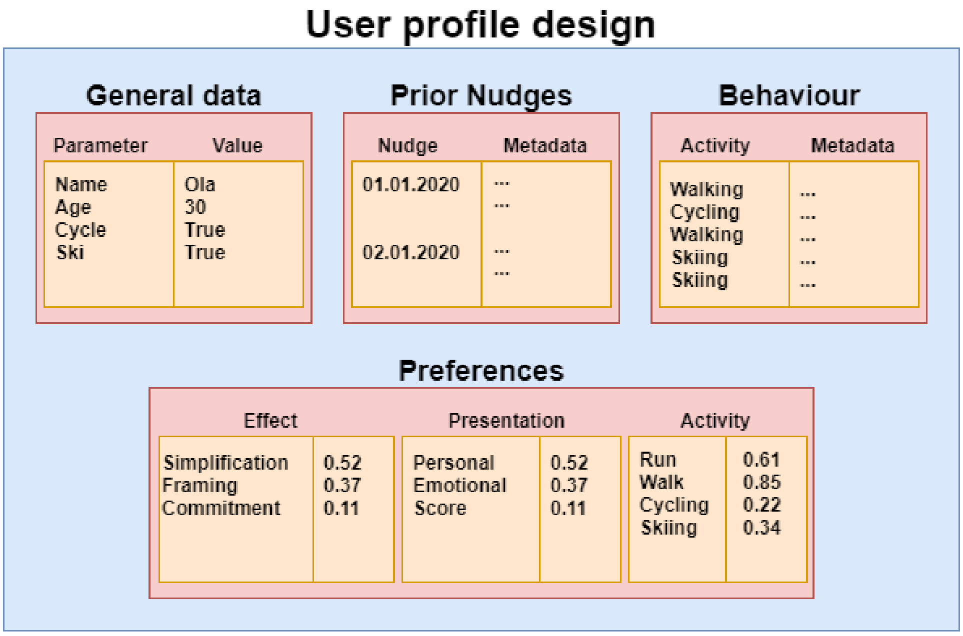

Figure 5.3: Illustration of the user profile design for smart nudging

In the user profile design in Figure 5.3, the data are separated into four sections, and these sections are explained further below.

The first section is *General data*. This section contains all the practical user information, which is the information the user has to provide. This information should contain:

- Name of the user
- Home address
- Age
- Disabilities
- Equipment owned
- Restrictions on max distance traveling for different activities

Some of the information stored in the general data section is private and should be stored in the edge on the user's device. The data in this section could be separated to isolate the private data and store non-private data on an external server. However, the size of the data stored in this section should be fairly small, so there should not be any storage space issues by storing these data together.

The second section in the user profile is the *Prior nudges*, which contains metadata about the nudges received in the past. The metadata should contain the greeting used, suggested activity, the provided content, what effect used in the nudge, whether or not the user followed the nudge, and the time and date for when the nudge was given. The metadata is used to calculate preferences the user has on the different nudge components. The data in this section will, over time, grow and consume some storage space. In the case of using old nudges to generate statistics, all nudges should be stored. Over the years, storing every nudge might consume some storage space. To not fill the storage on user devices, data used for statistics should be stored on external servers.

The third section contains the *behavior* of the user. The metadata stored in this section contains activities the user has performed and details about the activity. The metadata from the activity should be time and date for the activity, what activity was performed, travel distance, and duration. Additional information that could be stored is pulse data from a pulse watch/belt or Global Positioning System (GPS) data about the traveled route. This information could be used for statistics about user performance improvement. The information in the behavior section is used to calculate preferences the user has on different activities and determine if it is time to nudge a user. If a user already has worked out one day, there is no need to nudge the user to be more active. The data stored about activities could consume a lot of storage space over time if a user is very active. If the smart nudge system is implemented to store the routes, a user runs or cycles. There will be a lot of GPS points to store, this generates lots of data and takes up storage capacity on the user's devices, this information should, therefore, be stored on an external server.

The fourth section in the user profile presents the *preferences* of the user on the different nudge components. This section should represent what psychological effect works on the user and what presentation and activities the user prefers. The different effects, presentations, and activities should be tied to a weight that represents how well an effect or presentation works for a user. The weight for activities represents how often the user performs different activities compared to the other activities. The weights are calculated from metadata stored in *Prior nudges* and *behavior*. The data stored about user preferences is highly private. This information reflects how to influence a user and takes a minimal amount of storage space since it is only storing a keyword and a number. To store this information on the user's own devices makes the data more secure. There is little to gain attacking an individual device for information compared to attack servers that contains information about the whole user base.

The weight in *Effect* section should be calculated based on information from *prior nudges* to represent how well different psychological effects work on the user. The weights should be calculated from the number of nudges a user followed containing an effect compared to the total number of nudges given containing the specific effect. How the weights can be calculated are presented below in section 5.5.3.

Presentation is the section in the user profile that represents what presentation format that works on the user. Each presentation method is tied to a weight that represents how well each method influences the user. The weight is calculated from information stored in *Prior nudges*. How the weight is calculated is presented below in section 5.5.3.

5.4.2 User profile representation

The user profile design presented above shows how the different information in the user profile should be stored in different sections to maintain a good structure to the data. To structure the data, Extensible Markup Language (XML) or JavaScript Object Notation (JSON) can be used to format the data. Both formats are a set of rules to encode documents. They are both human and machine-readable. For the examples of how the sections of the user profile can be represented, JSON are chosen because the experiment conducted in this research is using MongoDB[1] to store the user profile. MongoDB is a document-based NoSQL database that stores documents of JSON objects. Examples of how the different sections of the user profile can be represented in JSON are presented below.

The first section is *General data*. This section contains a list of information describing the user and what equipment the user owns. This section can be represented like this in JSON:

```
{
"general_data": {
      "Name": "Ola",
      "Age": 30,
      "Address": "Olasveg 1",
      "Disabilities": null,
      "Cycle": true,
      "Cross_country_ski": true,
      "Alpine_ski": false,
      "Snowboard": true
    }
}
```

The second section of the user profile is *Prior nudges*. This section contains a

1. https://www.mongodb.com/what-is-mongodb

list of the received nudges. Each entry that is a nudge should contain a list with metadata from the nudge. The section can be represented like this:

```
{
    "prior-nudges": {
        "01.01.2020": {
            "Presentation": "Personal",
            "Proposed-activity": "Running",
            "Content": {
              "Weather": "Sunny",
              "Temperature": 17
            },
            "Effect": "Loss-aversion",
            "Nudge-followed": true,
            "Chosen-activity": {
                "Activity": "Walking",
                "Distance": 5,
                "Duration": 60
            },
            "Nudge-message": "Nudge message goes here"
        },
        "02.01.2020": {
            "Presentation": "Emotional",
            "Activity": "Cycling",
            "Content": {
              "Weather": "Cloudy",
              "Temperature": 13,
              "Distance": 15,
              "Estimated-duration": 60
            },
            "Effect": "Simplification",
            "Nudge-followed": true,
            "Chosen-activity": {
                "Activity": "Cycling",
                "Distance": 15,
                "Duration": 70
            },
            "Nudge-message": "Nudge message goes here"
        }
    }
}
```

The first nudge is an example of a simple nudge, it used a personal presentation, the proposed activity is running, content are provided to inform the user about the conditions, the psychological effect is loss-aversion, and the user did not follow this nudge. The second nudge use an emotional presentation. The proposed activity is cycling, and content about the weather are provided. This nudge has some more content on distance and estimated duration of the workout because the effect used is simplification, where a route was proposed on a map with estimated time to travel the route.

The third section in the user profile is *behavior*, which is a list of all activities performed by the user. Each activity should contain metadata from the performed activity, and the JSON representation can be formatted like this:

```
{
    "behavior": {
        "Walking": {
            "Distance": 5,
            "Duration": 60,
            "Date": "01.01.2020"
        },
        "Cycling": {
            "Distance": 15,
            "Duration": 60,
            "Date": "02.01.2020"
        }
    }
}
```

The fourth section of the user profile is the *Preferences*, which contains all user preferences. The preferences represent what psychological effects work on the user, what presentation fits best for the user, and what activities the user prefers. The effect, presentation, and activitiey preferences should be split into three lists representing each category's preferences. The preferences can be formatted to JSON like this:

```
{
    "Preferences": {
        "Effect": {
          "Simplification": 0.52,
          "Framing": 0.37,
          "Commitment": 0.11
        },
        "Presentation": {
            "Personal": 0.52,
            "Emotional": 0.37,
            "Score": 0.11
```

```
            },
            "Activities": {
              "Running": 0.61,
              "Walking": 0.85,
              "Cycling": 0.22,
              "Cross_country_skiing": 0.34
            }
        }
    }
```

5.5 Update the user profile

To keep the user preferences updated are important in a system that rely on the current preferences of a user. This section will present the process of updating the user profile, followed by how to deal with seasonal activities. Lastly, a proposal on how to calculate the different preferences are presented.

5.5.1 The process of updating the user profile

The user profile should be updated based on the users' responses to prior nudges. The preferences can be calculated from all nudges received in the past, a number of prior nudges, or the nudges given in a defined time frame.

The user profile component should not be the component to calculate new preferences and update the information, this should be done in the profile learner. Figure 5.4 gives an illustration of how the user profile update process can be implemented.

The process to update the user profile contain several processes to update the different components in the user profile. The processes illustrated in 5.4 are further explained below.

The process of updating the *prior nudge* list starts with a *User reaction* from a given *nudge* being sent to the *Profile learner* along with the *nudge*. The profile learner will request information from the *User profile* component to fetch the old profile. The profile learner combines information from the given nudge and the user reaction to the nudge to create a prior nudge. The newly created prior nudge are added to the list of prior nudges in the user profile. A prior nudge typically contains metadata about the nudge, the nudge message itself, and the user's reaction to the nudge. If the user followed the nudge, the prior nudge would be registered as followed, and the reaction will contain the activity

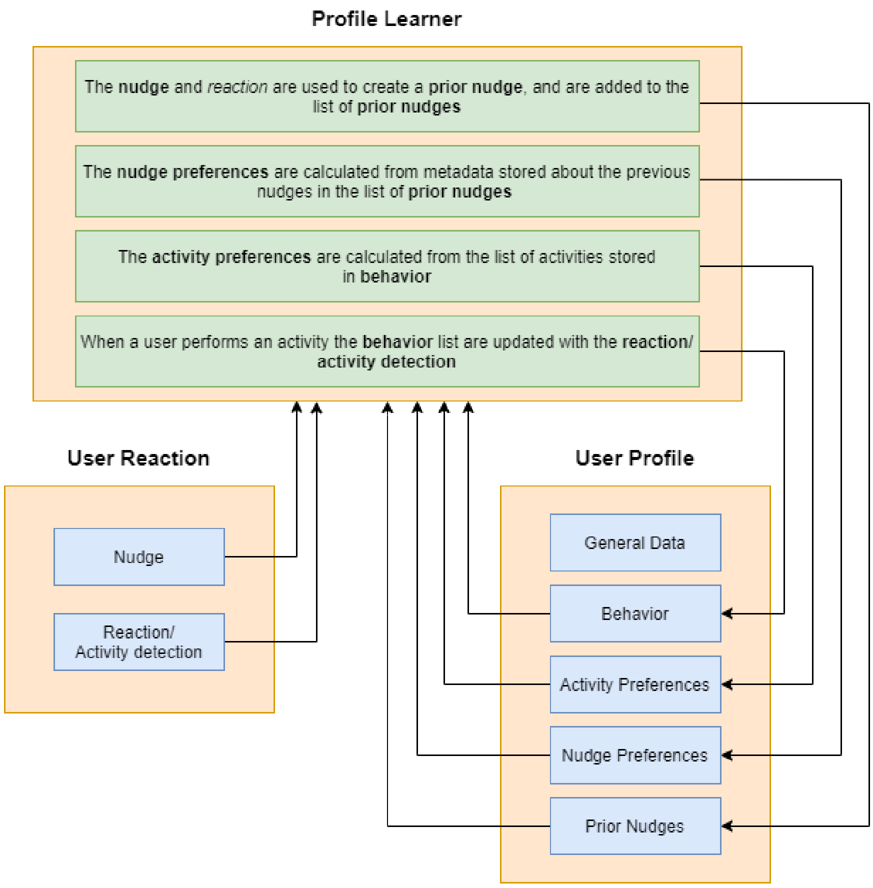

Figure 5.4: Illustration of the process to update the user profile

metadata. If the user did not follow the nudge, the prior nudge would be registered as not followed.

Updating *nudge preferences* can either be triggered each time a user has received a nudge, or it can have a set time interval that determines how often the preferences should be updated. To update the *nudge preferences*, the *profile learner* use metadata from nudges contained in the *prior nudge* list that is stored in the user profile. Metadata from prior nudges are used to calculate preferences on different effects and presentations. How to calculate these preferences are discussed in section 5.5.3.

Activity preferences can be updated after a user performs an activity, or it can have a set time interval that determines how often the preferences should be updated. Activity preferences are calculated from *behavior*, which contains a list of all activities performed by a user, the idea is to rate activity preferences based on occurrences of different activities in the behavior list. More details about the calculation are presented in section 5.5.3.

The last process of updating the user profile is to update the *behavior* list, which is a list of all activities performed by the user. The user reaction component detects an activity performed by the user and sends this to the profile learner. The profile learner will update the behavior list in the user profile.

5.5.2 Seasonal activities

Seasonal activities have to be accounted for. Especially in countries where there are significant seasonal differences. The smart nudge system should keep track of different seasonal activities, and the preferences for seasonal activities should not be fading out as the season changes. If a user likes to go skiing in the winter, the preferences on skiing should not be fading out as the snow disappears, and there are no possibilities to go skiing. The nudge system should know that winter is fading into spring and put the preferences on winter activities on hold to next winter. When the possibilities for winter activities are back again, the system should use the preferences from last year and start adjusting them again based on user behavior. Seasonal activities are taken care of in the user profile desgin, but are further discussed in section 7.4.

5.5.3 Calculate preferences

Updating preferences based on all prior nudges would not be very representative of the user preferences. The more nudges that are included in the calculation, the less impact each nudge will have on the preferences, and the behavior change will not be very clear as the number of nudges increases. Having too few nudges to calculate the preferences are also inaccurate. When using too few nudges, each nudge will have more weight and may bump the preferences back and forth between the different preference dimensions making it hard to stabilize the preferences and have a smooth transition in behavior change. When calculating preferences, a sweet spot of how many nudges to base the calculations from that gives the best representation of a user's preferences must be further researched and tested.

To calculate the preferences from a set number of nudges or a given time frame are pretty similar because a time frame from different periods should have approximately the same amount of nudges. One of these methods should be used to calculate preferences. As a user change behavior, the reaction a user had on nudges a year or two ago might not represent how a user reacts to a nudge in the present time. The number of nudges or the length of the time frame used to calculate preferences are not be investigated in this research and have to be researched further.

An approach to calculate preferences is presented below. However, this is not a critical detail for the research in this thesis. So it has to be tested and researched further to determine if it is the right method to use.

The influence (I) an effect have on a user can be calculated by fetching the number of nudges that worked on the user with a specific effect (E) and divide it on the number of nudges given with that effect (N), this gives the formula: $I(E) = \frac{E}{N}$.

The idea of calculating preference for different nudge presentations is the same as the effect. The influence (I) a presentation have on a user can be calculated by fetching the number of nudges that worked on the user with a specific presentation (P) and divide it on the number of nudges (N) given with that presentation. This gives the formula: $I = (P)\frac{P}{N}$.

The preference a user has on different activities is slightly different from the two above since the preferences on activities are calculated from the behavior section in the user profile. The preference (w) a user has on an activity can be calculated by fetching the number of times the activity (A) is performed and divide it on the total number of activities (N), this gives the formula: $W(A) = \frac{A}{N}$.

The calculations results are the percent of the nudges given with the different nudge parts that were followed, which is the same as Dalecke[6] proposed for representation of weights in his thesis.

5.6 Privacy measures

When dealing with personal data gathered from users through a system, using the data right and protecting the gathered data is vital to keep users' privacy safe. GDPR is a set of regulations to help the users know what data about them are collected and inform about security requirements[11]. The following section discusses measures to satisfy GDPR requirements that are presented in section 2.7.2. GDPR is a fairly new regulation to preserve the privacy of people using personalized systems. Because GDPR are so fresh, there are possibilities for new regulations being added to close any blind spots or gray areas in today's regulation. The measures listed below are proposals on how to satisfy GDPR as it is today. When implementing other parts of the smart nudge system, there is a possibility for any complications or any other regulations that have to be taken into account.

The first point is about transparency, to make sure the user knows what data are collected, and why this data is collected. This can be satisfied by informing about what information a feature will collect and why it is needed. Transparency can be achieved by displaying a short, accurate text when the feature is about to be used.

GDPR states that the user has to give consent for an application to process data concerning the user lawfully. The consent from a user can be asked for together with the text informing why data are collected. When the users have read the terms of a feature, they can either give their consent or reject it.

The user has the right to get confirmation whether or not personal data about the user are being processed. If personal data about a user are being processed, the user has the right to access the data, know the purpose for the data being collected, know what data are collected, and the time period for which the data are stored. This can be solved by being fully transparent by accurately providing this information to the user in, for example, an information section of the application.

If there are any inaccurate data about a user, the user has the right to obtain rectification of the inaccuracy. Some data should be editable by the user, like personal information. Name, email, address, etc. this is the user's responsibility to keep up to date. If any data is collected and processed by the system that is inaccurate, there should be implemented endpoints in the system that can be used to rectify any inaccurate data.

Users have the right to be forgotten by deleting all the information about a user in a system. If a user withdraws the consent of processing his data or the data is no longer necessary concerning the purpose of which it was collected, the data shall be deleted. This can be handled by ensuring the data are connected to a user id which makes it possible to find all data connected to a user, and delete it if the user wants to be forgotten. As far as possible, every location where data concerning a user is stored, there should be implemented endpoints to delete this data. As discussed in section 5.5.3, the relevance of different nudge components can be calculated from several nudges or all nudges in a given period, if data are outside the time frame and not used for any processing it should be deleted.

The user has the right to receive the data stored about them, and the data should be in a common machine-readable format. This requires that the data are stored in a way that makes it easy to extract it from the system and provide it to the user. Data about a user stored in the smart nudge system might be stored in different locations. Each of these locations should be implemented with endpoints to extract the data in case of a user asking to receive the data stored about them.

Lastly, the system should be implemented with the necessary security measures by design. The user profile contains a lot of personal data. Some of the data stored about a user are more private than others, the preferences on how to best influence a user is clearly more private than the nudges a user have received in the past. For privacy and protection, the most private data should be stored in the edge meaning on the user's device, while the lesser private data that have the potential to stack up and take a lot of storage space, can be stored on an external server where the storage capacity is much higher. By storing the most private data on edge, any attacks on servers will not expose any of the highly private data. If a user device is attacked, only the data stored on the attacked device can be exposed, meaning that all other users have their data safe on their device. This helps to protect the highly private data in case of an attack.

5.7 Evaluating smart nudge architecture

The smart nudge system is a set of microservices working together to create a complete system. Each component in the smart nudge system provides different information and functionality. The original architecture designed by Karlsen and Andersen[17] are presented below in Figure 5.5 and the redesigned architecture are presented below in Figure 5.6.

The changes done in the smart nudge system architecture are based on the storage location of data that has been revealed through designing the user profile. The location of data enlightens which components have to talk to each other to get the information they need. The components touched by the redesign are the *Profile learner*, *Relevant activities*, *User reaction/ activity chosen* and *User profile*.

The changes done for the *Profile learner* is that the *Personal data*, *Interests*, *Behavior* and *Capabilities* are information that creates the initial profile in the original architecture. When creating the initial profile, the only information provided by the user is the *general data*, with empty behavior, preferences, and prior nudges. If the user has a pulse watch that keeps a log of the past activities, the smart nudge system could import the behavior log from this system, and fill in the behavior to get a more complete initial profile. In the old architecture, the *Relevant activities* component provides the previous activities which are renamed to behavior. The behavior is stored in the user profile, and the relevant activities component does not have this information. The behavior should, therefore, be provided by the user profile and not the relevant activities

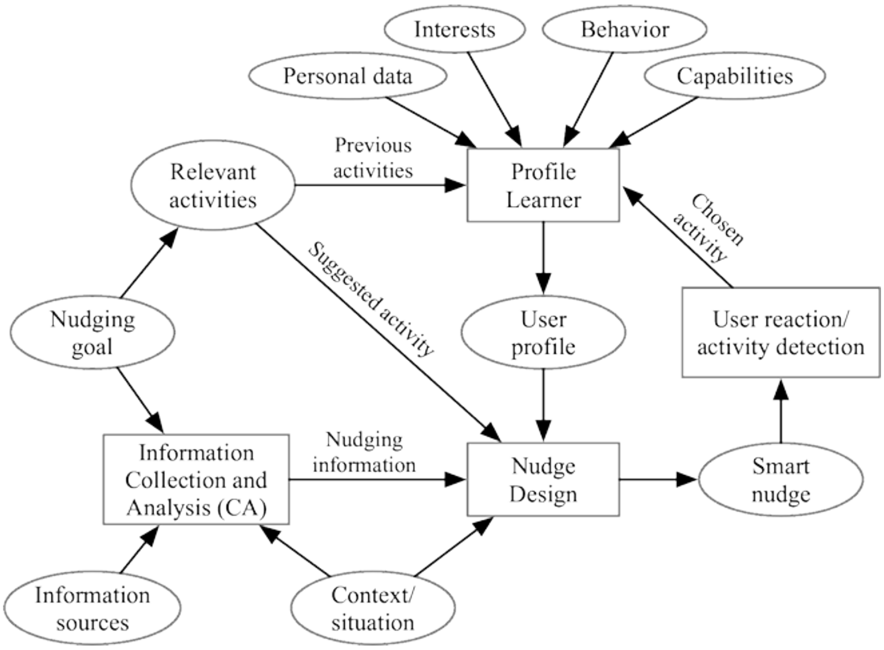

Figure 5.5: The smart nudge architecture designed by Karlsen and Andersen[17]

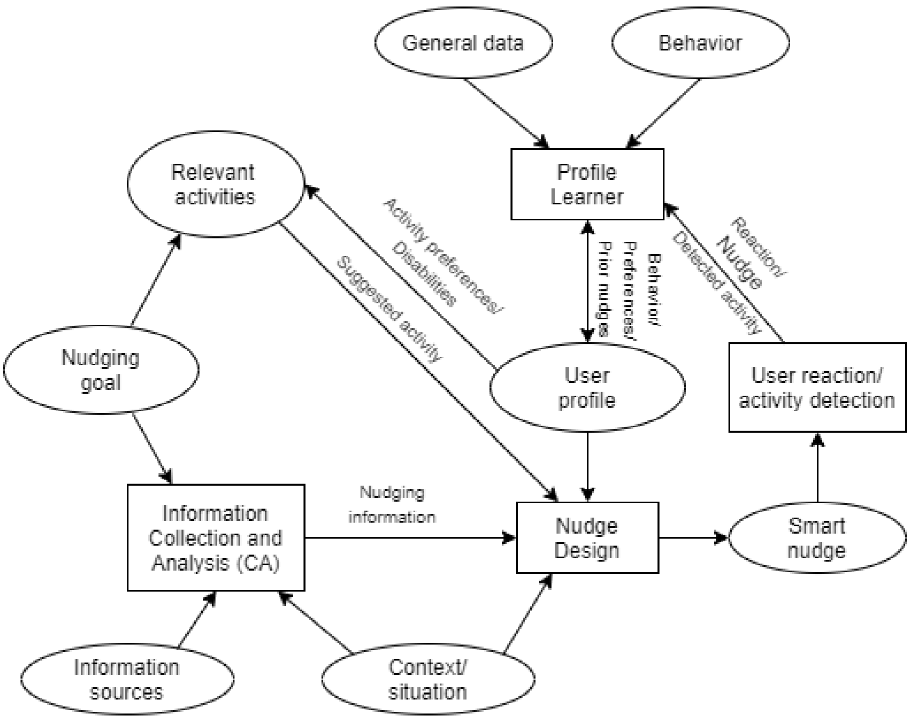

Figure 5.6: Smart nudge system slightly redesigned

component.

The *Relevant activities* component should propose a relevant activity based on the *Nudging goal*. This component use *Activity preferences* and *Disabilities* to find a suitable activity to propose. Activity preferences are relevant to finding an activity that suits the user's preferences and the nudging goal. Disabilities are relevant to rule out any activities a user cannot perform because of a disability. The relevant activities component finds a suitable activity and provides it to the *Nudge design* component.

The *User profile* component are changed to not only provide data to the *Nudge design* component, it now deliver data to the *Profile learner* and *Relevant activities* components.

In the original nudge architecture the *User reaction/ activity detection* component delivered the chosen activity. In the redesign, this component should still provide the chosen activity. However, it should also provide the nudge itself, if the nudge was followed or not, and information about the performed activity. Information about the activity will typically be what activity, duration, and distance traveled. This component should detect any activity the user performs on his own initiative and inform the profile learner that and activity was performed.

5.8 Ethics

The ethics of nudging are presented in section 2.2, which states that a nudge has to be fully transparent to be ethical. The same goes for the user profile in the smart nudge system. The system must be transparent to make the users aware of what data is collected and stored in the user profile. The user profile itself does not have the functionality to be transparent to the users, which means that the system using the user profile has to provide this information to the users. The smart nudge system must be aware of all the data it collects and what it uses the data for to be fully transparent. When a user uses a new functionality that collects personal data. The system should give the user a short message that describes what data will be collected and why it is needed, so the user can choose to agree or abort.

5.9 Benefits of the design

The smart nudge system benefits from the user profile design presented in this chapter because it stores metadata from prior nudges and activities which makes it easily available for the profile learner to calculate user preferences on nudge components and activities. The profile learner should feed the user profile with updated preferences, which makes it possible for the nudge design component to generate accurate nudges to the users based on their preferences.

6

Implementation

The main goal of this experiment's implementation is to create a demonstrator that shows how smart nudges and user preferences can be represented in a user profile and stored in a database. The demonstrator generates a user profile, convert it to the right format, and insert it to the database.

When implementing an experiment, the first step is to create the experiment's architecture to define what components are needed for the experiment to have the wanted functionality and provide the desired results. When the experiment is designed, it is time to find frameworks and technology that can satisfy the design requirements. When the framework and technology are chosen, the experiment can be implemented and tested.

6.1 Experiment architecture

Before implementing the demonstrator, the architecture has to be defined. Figure 6.1 gives an illustration of the architecture of the demonstrator.

To store the user profile, a database is needed. The database must be connected to a backend that takes care of the schema structure and communicates with the database. When the smart nudge system is implemented for testing or production, it should be implemented with different storage locations for different data, based on privacy requirements and data size. An Application

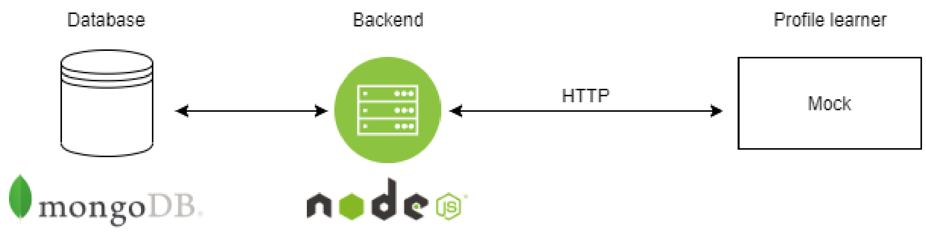

Figure 6.1: An illustration of the experiment architecture

Programming Interface (API) is used in the demonstrator to handle requests to the database. The implemented demonstrator stores all the data at the same location for simplicity. The backend hosts an API that handles incoming requests for inserting, updating, fetching, or deleting data in the database. The last component in the demonstrator is the profile learner mock, which is the component in the demonstrator that should simulate the profile learner component from the smart nudge system. Because the profile learner has not been researched yet, it has to be mocked. Mocking the profile learner means that a profile learner is implemented to statically deliver data in the format expected from the real profile learner, without the expected functionality of the profile learner. The profile learner mock generates the user profile and sends the profile to the backend.

6.2 Framework

The backend in this implementation is using an API setup with nodejs[1] that uses the module express as Hypertext Transfer Protocol (HTTP) server with mongodb[2] as database. To manage representation and structure of data in the database, the object data modeling library mongoose[3] are used. MongoDB and Nodejs are good to combine since both use JavaScript and JavaScript Object Notation (JSON) format.

Nodejs is an open-source runtime environment for javascript. It is an asynchronous and event-driven runtime used to build scalable applications that execute JavaScript code outside the browser.

Express is a tool to make small robust HTTP servers. Express are commonly used for single page applications, websites or HTTP APIs.

1. https://nodejs.org/en/about/
2. https://www.mongodb.com/what-is-mongodb
3. https://mongoosejs.com/

Mongodb is a schema-less NoSQL document-based database that stores data in JSON-like documents. MongoDB uses a rich query language to support create, read, update, and delete operations. It has high availability by using replication, and it can store data in memory for fast access[14]. By storing data in memory, it requires a lot of memory. The document size in MongoDB is restricted to 16 megabytes[7].

Mongoose is a schema-based tool for modeling application data in a mongoDB. It includes type casting, validation, and query building. Mongoose is used to define a schema that is the structure of the data to store in the database.

The reason this setup was chosen is that JSON is one of the standard representation formats for user profiles[9], and mongoDB stores the data in documents of JSON. Also, i have experience with using MongoDB in combination with express in the nodeJS framework. It is pretty straight forward to set up a simple API using these tools. Because this experiment is only a demonstration based on data structure, the performance and storage space are out of the scope, and mongoDB with data in JSON format is a suitable setup for the demonstrator.

The profile learner component is the component in the smart nudge system that should insert information to the user profile. Since this component has not been implemented and researched yet, it was necessary to create a mock that generates the user profile in the desired format and sent it to the backend. The mock of the profile learner is implemented in python and sends the user profile to the backend by communicating with the API through HTTP using the requests python library. Python is chosen for creating the profile learner mock because I have a lot of experience with the language and the libraries used.

6.3 Implementation details

The backend is implemented in the nodeJS framework using typescript with mongoDB as the database. Mongoose is used to manage the database. The user profile schema is defined with mongoose, which means that it defines how the data should be structured in the database, and what structure the backend expects the incoming data to have. The profile schema specified in mongoose should include all the parameters to store in the user profile and should have the desired structure to the data. In this case, the user profile is divided into four components. These components are the *General data*, *Prior nudges*, *Behavior* and *Preferences*. Each of these components contains the information to store, meaning the parameters and values relevant to the specific components.

ExpressJS are used as a HTTP server, which has a set of routes. The routes trigger controllers for inserting a new profile with a POST request, update an existing profile with a PUT or PATCH request, view all existing profiles with a GET request, view a specific profile with a GET request and delete a specific profile with a DELETE request. The profile controller imports the profile schema from mongoose. Hence, the controller knows the expected structure of the profile and can insert the incoming request body to the schema before it is saved in the database.

The profile learner mock is implemented in python using the json library to convert the generated user profile to JSON. The profile learner mock generates each profile component as dictionaries according to the specifications expected by the backend. The user profile components are added together to make a complete user profile that can be sent to the backend. All the integer values for the different parameters in the user profile are generated randomly, and the parameters requiring a string are randomly picked from lists of valid values. When the user profile is generated, it is converted to JSON using the python json library and then sent to the backend through HTTP using the requests library for python.

6.4 The demonstrator

The purpose of the demonstrator in this thesis is to demonstrate how a user profile can support smart nudges and how the user profile can be represented and stored in a database. The demonstrator generates a user profile with parameters that reflects the information presented in section 5.2. The values to the different parameters do not necessarily make sense in relation to other parameters that should make sense in a real smart nudge user profile. This is because the values are generated randomly, and the goal is to demonstrate the representation and structure of the information, not the information correctness.

The profile learner mock implemented in the demonstrator, generates a user profile with random numbers and convert it to JSON format. One of the generated user profiles looks like this after converting it to JSON format ready to be sent to the backend:


```json
{
    "general_data": {
        "name": "Karl",
        "age": 30,
        "email": "Karl@test.com",
        "address": "Karls veg 43",
```

```
        "disabilities": "",
        "cycle": false,
        "alpine_ski": true,
        "cross_country_ski": false,
        "snowboard": false,
        "gender": "male",
        "phone": "54682048",
        "create_date": "2020-04-28T12:49:29.383Z"
    },
    "prior_nudges": [{
        "date": "21.5.2025",
        "presentation": "quality score",
        "proposed_activity": "snowboarding",
        "content": {
            "weather": "cloudy",
            "temperature": 1.8
        },
        "effect": "framing",
        "nudge_followed": true,
        "chosen_activity": {
            "activity": "snowboarding",
            "distance": 17,
            "duration": 151
        },
        "nudge_message": "The given nudge message goes here"
    }],
    "behavior": [{
        "activity": "walking",
        "distance": 13,
        "duration": 177,
        "date": "10.9.2064"
        }, {
        "activity": "snowboarding",
        "distance": 5,
        "duration": 197,
        "date": "19.10.2051"
        }],

    "preferences": {
        "effect": {
            "simplification": 0.07,
            "framing": 0.98,
            "commitment": 0.88,
            "social_norm": 0
```

```
            },
            "presentation": {
                "personal": 0.68,
                "emotional": 0.47,
                "score": 0.2
            },
            "activities": {
                "running": 0.2,
                "walking": 0.98,
                "cycling": 0.06,
                "cross_country_skiing": 0.21
            }
        }
    }
```


The user profile are sent to the backend using a HTTP POST request. The response contains the response code and the response text. The response code is 200 which means that everything is OK, and the response text is the data object that gets stored in the database. The response from the backend looks like this:

< Response[200] >


```
{
    "messages": "New profile created",
    "data": {
        "preferences": {
            "effect": {
                "simplification": 0.07,
                "framing": 0.98,
                "commitment": 0.88,
                "social_norm": 0
            },
            "presentation": {
                "personal": 0.68,
                "emotional": 0.47,
                "score": 0.2
            },
            "activities": {
                "running": 0.2,
                "walking": 0.98,
                "cycling": 0.06,
                "cross_country_skiing": 0.21
```

```
        }
    },
    "_id": "5ea82659f360187bb23dfadd",
    "general_data": {
        "name": "Karl",
        "age": 30,
        "email": "Karl@test.com",
        "address": "Karls veg 43",
        "disabilities": "",
        "cycle": false,
        "alpine_ski": true,
        "cross_country_ski": false,
        "snowboard": false,
        "gender": "male",
        "phone": "54682048",
        "create_date": "2020-04-28T12:49:29.383Z"
    },
    "prior_nudges": [{
        "date": "21.5.2025",
        "presentation": "quality score",
        "proposed_activity": "snowboarding",
        "content": {
            "weather": "cloudy",
            "temperature": 1.8
        },
        "effect": "framing",
        "nudge_followed": true,
        "chosen_activity": {
            "activity": "snowboarding",
            "distance": 17,
            "duration": 151
        },
        "nudge_message": "The given nudge message goes here"
    }],
    "behavior": [{
        "activity": "walking",
        "distance": 13,
        "duration": 177,
        "date": "10.9.2064"
    }, {
        "activity": "snowboarding",
        "distance": 5,
        "duration": 197,
        "date": "19.10.2051"
```

```
        }],
        "__v": 0
    }
}
```

The JSON object above is the user profile as it gets stored in the database. If the response is compared to the generated profile, the back end adds a "_id:" parameter, which is the unique identity used to fetch that specific user profile. The back end has a route to fetch all stored profiles, but there is also a route to fetch one specific profile. By adding the profile id to the request Uniform Resource Locator (URL), a specific profile can be fetched alone from the database.

The back end also has an update route to update any existing profile. To update an existing profile, either the whole profile must be sent to the back end containing the updates, or just the attributes to change can be sent. To update the whole profile, a PUT request is used, and to update a few attributes, a PATCH request is used.

There are also implemented a route for deleting profiles. To do so, a DELETE HTTP request containing the id of the user profile desired to delete.

For this experiment, the interesting part is to look at the user profile's representation when stored. This means that the only request that is implemented with the profile learner mock is the insert request. All the other requests to view profiles, update profiles, and delete profiles must be sent manually. The following section will further explain the usage of the back end server.

6.5 Back end usage

This section covers how to use the back end to be able to use it for further research and development in the smart nudge project.

The API expects to receive a JSON object that contains the fields general_data, prior_nudges, behavior, and preferences, which are the four sections of the user profile. For each of these sections, some parameters are expected in a specific format.

The *general_data* section expects:

```
general_data:{
    name: String,
    age: Number,
    gender: String,
    email: String,
    address: String,
    disabilities: String,
    cycle: Boolean,
    alpine_ski: Boolean,
    cross_country_ski: Boolean,
    snowboard: Boolean,
    phone: String,
    create_date: {
        type: Date,
        default: Date.now
    }
}
```

In this section, the different activity equipment are listed, there are listed a few examples on activity equipment, more equipment have to be added to make the user profile support more activities.

The prior_nudges section expects:

```
prior_nudges: [{
    date: String,
    presentation: String,
    proposed_activity: String,
    content: {
        weather: String,
        temperature: Number
    },
    effect: String,
    nudge_followed: Boolean,
    chosen_activity: {
        activity: String,
        distance: Number,
        duration: Number
    },
    nudge_message: String
}]
```

This section contains a list of prior nudges where each item in the list contains all the parameters listed above.

The behavior section expects:

```
behavior: [{
    activity: String,
    distance: Number,
    duration: Number,
    date: String
}]
```

this section contains a list of activities performed in the past. Each item in the list contains the parameters listed above.

The preferences section expects:

```
preferences:{
    effect: {
        simplification: Number,
        framing: Number,
        commitment: Number,
        social_norm: Number
    },
    presentation: {
        personal: Number,
        emotional: Number,
        score: Number
    },
    activities: {
        running: Number,
        walking: Number,
        cycling: Number,
        cross_country_skiing: Number
    }
}
```

This section contains the preferences the user have on different nudge components. The list above have some effects, presentations and activities for demonstration purposes. More effects, presentations and activities have to be added for the user profile to support it.

By providing the backend with a JSON object containing the sections listed above, it generates a unique id and stores it in the database.

To add a new profile to the database, send the JSON object with an HTTP POST request to `http://{host}:{port}/profiles`.

One can fetch either all profiles or one specific profile from the database. To fetch all stored profiles, send a HTTP GET request to
`http://{host}:{port}/profiles`.
To fetch one specific profile send a HTTP GET request to
`http://{host}:{port}/profiles/{profile id}`
where profile id is the unique identifier for that specific profile. The response for these requests are user profiles in JSON format as it is presented in section 6.4

An already existing user profile can be updated or patched. When updating the user profile, the whole object has to be sent to the back end. This is done with a HTTP PUT request. The request has to be sent to the back end using the route: `http://{host}:{port}/profiles/{profile id}`.
When patching an existing profile, only the parameters to update have to be sent to the back end, this is done by sending a HTTP PATCH request to
`http://{host}:{port}/profiles/{profile id}`

The last route in the API are to delete profiles. To delete a profile a HTTP DELETE request are sent to
`http://{host}:{port}/profiles/{profile id}`.
Where the profile id is the unique identifier of the profile to delete.

7 Evaluation

The research in this thesis has investigated, created a design, and implemented a solution to represent smart nudges in a user profile and how the user profile itself can be represented in a database. In the search for answers on the research questions, details and requirements regarding the user profile was discovered. The research also revealed details and requirements about other components in the smart nudge system. These details and requirements are presented and discussed below, along with other solutions and observations made through the research.

7.1 User profile representation

The information in prior nudges contains metadata about the nudge, including the nudge message. The nudge design component in the smart nudge system is the component that generates the nudge. Therefore it is natural to expect that the nudge design adds this metadata to the nudge object, so the profile learner can extract it and store it in the user profile.

The demonstrator is representing the user profile as one JavaScript Object Notation (JSON) document that contains the complete user profile. This is only for demonstration purposes to show how the user profile can be represented and how the smart nudges can be stored in the user profile. In a smart nudge system, the user profile should be divided into the different sections, and stored on different locations based on storage space consumption, and privacy reasons. The chosen format for the user profile is JSON. JSON would be usable in a real system, because it is human readable and not as verbose as Extensible Markup Language (XML). By choosing a human readable format, it is easier to satisfy the requirements of General Data Protection Regulation (GDPR), in case a user wants to receive the data stored about him, there is no need for extra functionality that converts the data to a readable format. When looking at performance, JSON is a lot faster than XML when compared on transmitting data between client and server[20]. JSON also has better performance than XML when serializing a data object from memory to a presentation more suitable for storage or transmission [31]. Zunke and D'Souza also looked into storage consumption of JSON versus XML when serializing objects, and found that JSON used almost half of the space that XML used[31].

The user profile used in the demonstrator is represented with parameters and weights to these parameters. The preferences on different nudge components and activities are represented as a keyword profile, where each keyword is tied to a weight. The keyword profile structure shows the relevance of the different smart nudge component to the user. When the smart nudge system is developed and tested on actual users, it might reveal that the user profile should not only contain the preference a user has on different nudge components. It could be relevant to know the user preferences on different nudge components used together with other nudge components. A semantic network profile would be a better structure to use if it is relevant to know what nudge components should be combined with other nudge components. The user profile can reflect how relevant different nudge components are when used in pairs with other components by using a semantic network profile. This would give each nudge component a weight representing how relevant they are alone, in addition to a weight to all the other components representing how relevant they are when used as a pair in a nudge.

7.2 User profile design

The user profile are designed as one whole user profile containing four sections. For privacy and protection reasons, it is discussed to split the storage locations of different sections of the user profile. When looking at it in the end of this thesis, the user profile design could have been illustrated to reflect this split of information, to make it easier to see what information to store on the different locations.

7.3 Nudge reaction

When a user receives a nudge, the user either follows the nudge or not. But there is one more option, when the user may not feel like performing the proposed activity. For example, when a user receives a nudge to go for a run. If the activity detection detects that the user went for a walk instead of running, in this case, the nudge was not followed because the desired outcome did not happen. But one can speculate that the nudge affected the user. This scenario requires further research to determine how to process the reaction if any of the nudge preferences should be updated as if the nudge was successful or not. This has to be taken into account when developing the profile learner. The component in the smart nudge system that has to detect this is the user reaction/ activity detection component. The profile learner should receive the nudge given as not followed, but it should also receive the activity performed.

7.4 Seasonal activities

In the representation of the user profile in the smart nudge system, the preferences on activities are simply represented with each activity connected to weights. The user profile in the demonstrator does not represent how to take care of preferences on seasonal activities. Questions relating to this is, if the preferences on seasonal activities should be stored in another section, and if they should contain a start and end date of the season for which they are relevant. How can the system know the start and end date of a season on different geographical locations, the start, and the end of a season vary significantly in different locations.

Preferences on seasonal activities can either be stored in the user profile as a list of season dependent activities stored in the activity preferences section along with the other activities that are not season dependent. Another solution is to store the activity preferences that are not dependent on any seasons in the activity preferences and add another section in the preferences section of the user profile that contains seasonal activities. The activities that are season dependent have to not only contain a weight that represents the relevance to the user. It also has to contain the start and end date of the season the activity is relevant for.

The smart nudge system must know when seasons start and end to freeze the user preferences for activities that are dependent on different seasons. One solution is to have the users register when a season starts and ends based on user observations. The users know when facilities for different activities are ready as the seasons change. Another method would be to make the user provide the system with approximate dates for start and end of seasons in the area they live in. It should be possible to find information on the web about seasons, at least in Norway, there is a service that provides information about the ski trails[1] and when they are prepared. This service can be used to see when they start and stop preparing ski trails in the different areas, and one can assume that there is a seasonal change when they start or stop preparing the trails. There might be other useful services on the web that can help to determine what season it is.

7.5 Prior nudges

The user profile is presented to store prior nudges. Nudges given to users in the past are used to calculate user preferences to the different nudge components, and how to best influence the different users. When updating the user preferences on different nudge components, some approaches to determine how many nudges to include in these calculations are discussed in section 5.5.3. The nudges to use in the preference calculations can be stored in the user profile for simplicity, to make the information easy to access for the profile learner. The older nudges that are not included in the calculation of user preferences do not have to be stored in the user profile. They can be stored on an external server. The older nudges should be stored for statistical purposes. The statistical information can be used to present historical data to the user to provide a log of activities, or it can be analyzed and used to improve the smart nudge system to tailor more accurate nudges.

7.6 The cold start problem

The initial user profile contains the general data, which is information about the user, like addresses, contact information, equipment owned, boundaries, and equipment owned. This information is not sufficient to determine the user's preferences, which introduces the cold start problem. The cold start problem is a problem in recommender systems where the system does not have sufficient information to provide tailored recommendations to different users[30].

One method to fill in some information on activity preferences is to fetch an activity log from a pulse watch and calculate activity preferences from this information. This will help to add some preferences for activities. The system can also ask the users to fill in preferred activities. Or the user profile can

1. `https://skisporet.no`

be initialized with default values that are adjusted over time based on user behavior.

When it comes to preferences on nudges, and how to influence the user to perform an activity. One solution could be to create a survey that tries to gather information about what influences a user and use it to generate initial preferences on nudge components. Another solution is to have default values on the preferences and change the preferences through nudging the user over time and monitor user reactions to different nudges.

7.7 Top preferences

Different nudge components will have different relevance to the users. The smart nudge system should not only use the nudge components with the highest relevance when building a nudge, but it should also try different compositions of components to vary the nudges and keep updated preferences on the various nudge components. The user profile designed and implemented in this thesis is weighting the different nudge components, which means that if the system should use components that does not have top relevance to the user, the logic has to be implemented to the nudge design component. The nudge design component can look at the list of prior nudges to find components that have not been used in a while and choose these components. Another approach is to add an overview of when the different nudge components were used in the user profile for the nudge design component to easily find the nudge components that have not been used in a while.

7.8 The implementation

The implementation in this thesis stores a generated user profile. Each user profile sent to the back end are stored as one JSON document. When querying the back end to fetch a user profile, the whole user profile are delivered. If there was more time to work on this thesis I would implement the backed to be able to return single sections from the user profile, so the smart nudge system components can query only the sections they need from the user profile. With more time I would also implement a more comprehensive demonstrator that not only generates and inserts a user profile to the back end server, it would also demonstrate how to fetch different sections of the user profile.

This thesis discuss that different information in the user profile should be stored on the user device or an external server because of privacy reasons and storage consumption. With a more comprehensive demonstrator, this would have been a part of the implementation to demonstrate how the system would handle to keep the information from one user profile in different locations.

7.9 Research questions

A few research questions have been formed to help steer the research. The main question for this research is "*How can a user profile support smart nudging?*". To answer the main research question, a few sub questions have to be answered.

The first sub-question is "*What smart nudge information should be included in the user profile?*" To determine the information to include in the user profile, different information sources were used to determine the user profile content. Based on information usually stored in user profile[9], the information Andersen et al. [17] presented as relevant information to the smart nudge system, and information about dynamic nudges presented by Dalecke[6], the content of the user profile are chosen. The information to store in the user profile is personal information about the user, and what activity equipment the user owns. Metadata about prior nudges should be stored to help the system learn user preferences. The metadata stored about nudges should contain what nudge components used to build the nudge, date and time for when the nudge was given, what content was used and the nudge message itself. The behavior of the user, or what activities the user performs should be stored to learn what activities the users prefer. Lastly, user preferences should be stored. The preferences of a user should contain how well different smart nudge components and activities appeals to the user, so that the smart nudge system can accurately build nudges to influence the user, and propose activities the user likes. When the smart nudge system is developed, and put in a testing phase, it may reveal that the user profile should contain more information.

The second sub-question is "*How should the smart nudge user profile be represented?*" To choose the user profile representation, three typical user profile representations where evaluated. The three different representations are keyword profile, semantic network profile, and concept-based profile[10]. The presented representations are used for personalizing content for users on the web. They are used as keyword vectors to evaluate the words of web pages or documents to find relevant content to the users. The user profile representation for the smart nudge system is decided to be similar to the keyword profile because the user preferences are connected to a weight that shows how relevant they are to the user, which is similar to the keyword profile that stores weighted keywords. However, it is discussed in 7.1 that a semantic network profile might be able to provide more accurate information about how the nudge components should be used together, by storing how relevant nudge components are when used together with other nudge components. The idea is to first make it work with simple solutions, then make more complex improvements. The two representations have to be tested and evaluated to determine what works best for the smart nudge system.

The third and final sub-question is "*How should previous smart nudges be represented in the user profile?*" For the user profile to support smart nudging, previous nudges have to be stored so they can be used to calculate user preferences. For the smart nudge system to easier calculate preferences on different nudge components, prior nudges should be represented with metadata in the user profile. When determining the representation of the metadata, the process of building a smart nudge was evaluated. Different components can be used to build the nudge, and the user profile should reflect these. The various components a nudge can be built by are effect, presentation, content, and activity. Each of these components represents sets of sub-components. The user profile should reflect what sub-components used to build the nudge, so the smart nudge system can calculate preferences on how well the different components work on the user. The nudge metadata should be added to the nudge object which is created by the nudge design component in the smart nudge system, because this is the component creating the nudge, and it knows what components it used to build the nudge. By adding the metadata to the nudge object, the profile learner component do not have to analyze the nudge to detect what components used in a nudge.

8

Conclusion

In this thesis, information about user profiles and smart nudging has been gathered to design a user profile that supports smart nudging. Prior research has been used to investigate user profile representation and what information should be included in the user profile to support smart nudges. Based on the information to store, a user profile design is developed. The user profile representation specifies how different data should be stored in the user profile, such as general information about the user, previous nudges given to the user, previous user behavior, and user preferences. A demonstrator is implemented to demonstrate the storage of the user profile in an actual database. The user profile design and representation are evaluated and discussed.

When designing the smart nudge system's user profile, some minor changes to the smart nudge system architecture had to be proposed. Changes regarding what information is provided by different components, and what information the different components should receive. By defining the user profile, we could remove a few links in the communication chain between components by fetching information directly from the information source.

Based on the information stored in the user profile and the information we saw necessary to update the user profile properly, the process of updating the user profile was defined to determine how the user profile could represent user preferences. The process of updating the user profile is overall explained to substantiate the design of the user profile.

During this thesis, requirement details about other smart nudge system components have been discovered, which I hope will come in handy as the research of the smart nudge system continues.

Future work for the smart nudge user profile is to add support for seasonal activities and represent the period which they are relevant. The user profile should be able to return single sections for smart nudge system components that only needs one or two sections from the user profile, so they can fetch only what they need. An overview of how many times the different nudge components have been used in a time period can be added so the system can try different compositions of nudge components, and not only use the most preferred components. A suitable technique to take care of the cold start problem have to be further examined.

9

References

[1] Giuseppe Amato and Umberto Straccia. "User Profile Modeling and Applications to Digital Libraries." In: *Research and Advanced Technology for Digital Libraries*. Ed. by Serge Abiteboul and Anne-Marie Vercoustre. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 184–197. ISBN: 978-3-540-48155-3.

[2] Evangelia Anagnostopoulou et al. "From mobility patterns to behavioural change: leveraging travel behaviour and personality profiles to nudge for sustainable transportation." In: *Journal of Intelligent Information Systems* (2018), pp. 1–22. URL: https://doi.org/10.1007/s10844-018-0528-1.

[3] Andersen Anders, Karlsen Randi, and Yu Weihai. "Green Transportation Choices with IoT and Smart Nudging." In: *Handbook of Smart Cities: Software Services and Cyber Infrastructure*. Cham: Springer International Publishing, 2018, pp. 331–354. ISBN: 978-3-319-97271-8. DOI: 10.1007/978-3-319-97271-8_13.

[4] Luc Bovens. "The Ethics of Nudge." In: *Preference Change: Approaches from Philosophy, Economics and Psychology*. Ed. by Till Grune-Yanoff and Sven Ove Hansson. Dordrecht: Springer Netherlands, 2009, pp. 207–219. ISBN: 978-90-481-2593-7. DOI: 10.1007/978-90-481-2593-7_10. URL: https://doi.org/10.1007/978-90-481-2593-7_10.

[5] Hendrik Bruns et al. "Can nudges be transparent and yet effective?" In: *Journal of Economic Psychology* 65 (2018), pp. 41 –59. ISSN: 0167-4870. DOI: https://doi.org/10.1016/j.joep.2018.02.002. URL: http://www.sciencedirect.com/science/article/pii/S0167487017307845.

[6] Sandor Dalecke. “Designing Dynamic and Personalized Nudges - A Model.” MA thesis. University of Kaiserslautern, July 2019.

[7] *Documents*. https://docs.mongodb.com/manual/core/document/. Accessed: 2020-05-13.

[8] Adrienne Porter Felt et al. “Android Permissions: User Attention, Comprehension, and Behavior.” In: *Proceedings of the Eighth Symposium on Usable Privacy and Security*. SOUPS ’12. New York, NY, USA: ACM, 2012, 3:1–3:14. ISBN: 978-1-4503-1532-6. DOI: 10.1145/2335356.2335360. URL: http://doi.acm.org/10.1145/2335356.2335360.

[9] Min Gao, Kecheng Liu, and Zhongfu Wu. “Personalisation in web computing and informatics: Theories, techniques, applications, and future research.” In: *Information Systems Frontiers* 12.5 (2010), pp. 607–629. ISSN: 1572-9419. DOI: 10.1007/s10796-009-9199-3. URL: https://doi.org/10.1007/s10796-009-9199-3.

[10] Susan Gauch et al. “User Profiles for Personalized Information Access.” In: *The Adaptive Web: Methods and Strategies of Web Personalization*. Ed. by Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 54–89. ISBN: 978-3-540-72079-9. DOI: 10.1007/978-3-540-72079-9_2. URL: https://doi.org/10.1007/978-3-540-72079-9_2.

[11] *General Data Protection Regulation*. European Commission. May 25, 2018. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679 (visited on 06/17/2020).

[12] Pelle Guldborg Hansen and Andreas Maaløe Jespersen. “Nudge and the Manipulation of Choice: A Framework for the Responsible Use of the Nudge Approach to Behaviour Change in Public Policy.” In: *European Journal of Risk Regulation* 4.1 (2013), 3–28. DOI: 10.1017/S1867299X00002762.

[13] Anne Håkansson. “Portal of Research Methods and Methodologies for Research Projects and Degree Projects.” In: *Proceedings of the International Conference on Frontiers in Education: Computer Science and Computer Engineering FECS’13*. [ed] Hamid R. Arabnia Azita Bahrami Victor A. Clincy Leonidas Deligiannidis George Jandieri. CSREA Press U.S.A, 2013, 67–73. ISBN: 1-60132-243-7. URL: http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-136960.

[14] *Introduction to MongoDB*. https://docs.mongodb.com/manual/introduction/. Accessed: 2020-05-13.

[15] Daniel Kahneman. *Thinking, fast and slow*. Farrar, Straus and Giroux New York, 2011.

[16] Renaud Karen and Zimmermann Verena. “Ethical guidelines for nudging in information security & privacy.” In: *International Journal of Human-Computer Studies* 120 (2018), pp. 22 –35. ISSN: 1071-5819. DOI: https://doi.org/10.1016/j.ijhcs.2018.05.011. URL: http://www.sciencedirect.com/science/article/pii/S1071581918302787.

[17] Randi Karlsen and Anders Andersen. "Recommendations with a Nudge." eng. In: *Technologies* 7.2 (2019). ISSN: 2227-7080.

[18] Anders Kofod-Petersen. "How to do a Structured Literature Review in computer science." In: *Ver. 0.1. October* 1 (2012).

[19] Marius J. Mæland. "An approach to user profile for smart nudges." Dec. 2019.

[20] Nurzhan Nurseitov et al. "Comparison of JSON and XML data interchange formats: A case study." In: Jan. 2009, pp. 157–162.

[21] *Physical activity WHO fact sheet*. Accessed: 2020-04-26. URL: \url{https://www.who.int/news-room/fact-sheets/detail/physical-activity}.

[22] *Physical activity WHO fact sheet*. Accessed: 2020-04-26. URL: \url{https://www.who.int/health-topics/physical-activity}.

[23] H. Thaler Richard and Sunstein Cass. *Nudge - Improving Decisions About Health, Wealth, and Happiness*. Yale University Press, 2008.

[24] Christoph Schneider, Markus Weinmann, and Jan vom Brocke. "Digital Nudging: Guiding Online User Choices Through Interface Design." In: *Commun. ACM* 61.7 (June 2018), pp. 67–73. ISSN: 0001-0782. DOI: 10.1145/3213765. URL: http://doi.acm.org/10.1145/3213765.

[25] Awanthika R. Senarath and Nalin Asanka Gamagedara Arachchilage. "Understanding user privacy expectations: A software developer's perspective." In: *Telematics and Informatics* 35.7 (2018), pp. 1845 –1862. ISSN: 0736-5853. DOI: https://doi.org/10.1016/j.tele.2018.05.012. URL: http://www.sciencedirect.com/science/article/pii/S073658531830296X.

[26] Cass Sunstein. "The Ethics of Nudging." eng. In: *Yale Journal on Regulation* 32.2 (2015), pp. 413–450. ISSN: 07419457. URL: http://search.proquest.com/docview/1749961471/.

[27] Cass R. Sunstein and Richard H. Thaler. "Libertarian Paternalism Is Not an Oxymoron." In: *The University of Chicago Law Review* 70.4 (2003), pp. 1159–1202. ISSN: 00419494. URL: http://www.jstor.org/stable/1600573.

[28] Kristian Torning and Harri Oinas-Kukkonen. "Persuasive System Design: State of the Art and Future Directions." In: *Proceedings of the 4th International Conference on Persuasive Technology*. Persuasive '09. Claremont, California, USA: Association for Computing Machinery, 2009. ISBN: 9781605583761. DOI: 10.1145/1541948.1541989. URL: https://doi.org/10.1145/1541948.1541989.

[29] Frâncila Weidt and Rodrigo Silva. "Systematic literature review in computer science-a practical guide." In: *Relatórios Técnicos Do DCC/UFJF* 1 (2016).

[30] Mi Zhang et al. "Addressing Cold Start in Recommender Systems: A Semi-Supervised Co-Training Algorithm." In: *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*. SIGIR '14. Gold Coast, Queensland, Australia: Associa-

tion for Computing Machinery, 2014, 73–82. ISBN: 9781450322577. DOI: 10.1145/2600428.2609599. URL: https://doi.org/10.1145/2600428.2609599.

[31] Saurabh Zunke. "JSON vs XML: A Comparative Performance Analysis of Data Exchange Formats." In: 2014.