NTNU
Norwegian University of
Science and Technology

Development and exploration of an adaptive learning system in medication calculation in nursing

Kristoffer Ravik Andresen

Master of Science in Computer Science
Submission date: June 2018
Supervisor: Trond Aalberg, IDI

Norwegian University of Science and Technology
Department of Computer Science

Abstract

Medication administration is an important task and responsibility of registered nurses. Unfortunately, medication errors are an internationally significant reason for injury and death in patients. The most common type of error is related to calculations, e.g. ”wrong dose”. The overarching aim of this master thesis was therefore to develop and explore a learning system that increases nursing students’ ability to gain knowledge and understanding in medication calculations.

Adaptive learning technology offers improvements over traditional, static learning technology by tailoring the learning environment to each individual learner instead of providing the same educational material, presentation and navigation to all learners in a one size fits all model. This thesis aimed to develop and explore how individual learner’s state of knowledge and learning needs could be modelled and used to adapt learning resources and exercises in an adaptive learning system in medication calculation.

An adaptive learning system was designed and developed where exercises were adapted to each learner’s state of knowledge, meaning that the exercises were adjusted to an appropriate level of difficulty relative to the learner’s current state of knowledge. As the learner’s skill in calculations increased so did the difficulty of the exercises. The learning system also adapted the exercises based on the learner’s current learning needs, e.g. to practice on exercises in a category the learner has low knowledge of, to repeat an exercise that the learner recently answered incorrectly, or to repeat an exercise from category that the learner already has mastered in order to maintain that knowledge.

An evaluation was performed to test how health care practitioners; nursing students, nursing teachers and registered nurses experienced the adaptivity of the learning system. They experienced that the adaptive learning system adapted the difficulty of the exercises to an appropriate level, they got support and guidance on exercises they found difficult and challenging, and the exercises were repeated at a reasonable interval so that in a short time they had the opportunity to correct any errors or that they got training on tasks they had previously mastered a while ago. Overall the users reported that the adaptive learning system in this project would be a useful learning resource when learning and maintaining medication calculation skills.

Sammendrag

Medisinadministrasjon er en viktig oppgave og ansvar for sykepleiere. Dessverre er medisineringsfeil en internasjonalt viktig årsak til skade og død hos pasienter. Den vanligste typen feil er relatert til medikamentregning, f.eks. "feil dose". Det overordnede målet med denne masteroppgaven var derfor å utvikle og utforske et læringssystem som øker sykepleiers evne til å tilegne seg kunnskap og forståelse i medikamentregning.

Adaptiv læringsteknologi tilbyr forbedringer i forhold til tradisjonell statisk læringsteknologi ved å skreddersy læringsmiljøet til hver enkelt student, i stedet for å gi alle studenter det samme pedagogisk materiale, presentasjon og navigasjon i en "one size fits all" modell. Denne oppgaven hadde til formål å utvikle og undersøke hvordan den enkelte students kunnskapstilstand og læringsbehov kunne modelleres og brukes til å tilpasse læringsressurser og oppgaver i et adaptivt læringssystem i medikamentregning.

Et adaptivt læringssystem ble designet og utviklet der oppgavene var tilpasset hver students kunnskapstilstand, noe som innebar at øvelsene ble tilpasset til et passende vanskelighetsnivå i forhold til studentens nåværende kunnskapstilstand. Etter hvert som studentens ferdigheter i medikamentregning økte, økte også vanskelighetsgraden på oppgavene. Læringssystemet tilpasset også oppgavene basert på studentens læringsbehov, f.eks. å øve på oppgaver i en kategori som studenten har lav kunnskap om, å gjenta en oppgave som studenten nylig har svart feil på, eller å gjenta en øvelse fra kategori som studenten allerede har mestret for å opprettholde denne kunnskapen.

En evaluering ble utført for å teste hvordan helsepersonell; sykepleierstudenter, sykepleierlærere og sykepleiere opplevde læringssystemets tilpasningsevne. De opplevde at det adaptive læringssystemet tilpasset treningsoppgaver til et passende vanskelighetsnivå, de fikk støtte og veiledning om øvelser de fant vanskelige og utfordrende, og øvelsene ble gjentatt med et rimelig tidsintervall, slik at de på kort tid hadde mulighet til å rette opp eventuelle feil, eller at de fikk trening på oppgaver de tidligere hadde mestret. Brukerne rapporterte at det adaptive læringssystemet i dette prosjektet vil være en nyttig læringsressurs når de skal lære og opprettholde ferdigheter innen medikamentregning.

Acknowledgements

Doing this master thesis has been both exciting and learning-rich. This work would not, however, have been possible to do without the support of the supervisor and my family. I would thank my supervisor, Trond Aalberg at NTNU, for his mentoring and guidance throughout this master work. I would also like to thank the nursing students, nursing teachers, and registered nurses who tested the adaptive learning system and participated in the survey. Thanks goes also to my family for providing me with invaluable support and advice toward completion of this master thesis, in particular I would like to thank my mother, Monika, who is a nurse and a lecturer, for suggesting the topic of this master thesis and recruiting the health care practitioners mentioned above. Thank you!

Kristoffer Ravik Andresen
Trondheim, June 2018

Table of Contents

List of Tables

List of Figures

Abbreviations

SDK	=	Software Development Kit
ZPD	=	Zone of Proximal Development

Chapter 1

Introduction

Medication administration is an important task and responsibility of registered nurses (Stolic, 2014). Unfortunately, medication errors are an internationally significant reason for injury and death in patients (Williams and Davis, 2016). Medication errors are the most prevalent nursing error recorded worldwide (Brindley, 2017). The most common type of error is related to calculations, e.g. ”wrong dose” (Björkstén et al., 2016). Medication calculation skills are a main learning objective in nursing education to prepare students for the real world of practice at registration (Coyne et al., 2013; National Curriculum Regulations for Nursing Programs, 2008). Nursing students mainly have difficulty with basic math principles (Bagnasco et al., 2016). The absence of learning systems that increase nursing students' ability to learn medication calculations is a concern. Specific learning interventions are needed to improve nursing students' basic math skills and medication calculations to ensure that they become safe practitioners in the clinical setting (Park and Kim, 2018; Sulosaari et al., 2015).

1.1 The Overarching Aim of the Master Thesis

Considering the importance of proficiency in medication calculations in nursing, the issue clearly requires development of a learning system that enhances and ensures nursing students' ability to learn calculations. Therefore, the overarching aim of the master thesis was to develop and explore a learning system that increases nursing students' and registered nurses' ability to gain knowledge and understanding in medication calculations. The contribution of the master thesis was development of an adaptive learning system, facilitating learning of calculations based on the individual's state of knowledge and learning needs.

1.2 Outline of the Master Thesis

This master thesis is organized in nine main chapters. In this introduction chapter, the theme, overarching aim, and contribution to the thesis are described. Chapter two presents the background of this thesis. Research on medication calculation skills in nursing, as well as digital learning systems developed to facilitate learning in this demanding skill are presented. In chapter three, theoretical perspectives that frame this master thesis are presented. Learning perspectives embedded in cognitivism and a sociocultural learning perspective will be presented. Chapter four presents adaptive learning technology, which is an approach to learning that adapts learning resources to the individual's state of knowledge and specific learning needs. In this chapter there will be given some examples to show how the adaptive learning technology can be related to medication calculation. Chapter five proposes a design for an adaptive learning system in medication calculation. The design includes methods for how to achieve different adaptation effects, as well as how to model the domain of medication calculation and the users' state of knowledge. Chapter six presents the implementation of the adaptive learning system developed, which includes descriptions of technology choices, technical documentation, and an overview of the learning application. Chapter seven summarizes an evaluation performed via a survey. Chapter eight is a discussion related to the feedback given from users (described in the evaluation). Chapter nine presents a conclusion including suggestions for further work, as well as limitations and strengths of the adaptive learning system developed.

Chapter 2

Background

Proficiency in medication administration is highlighted as important to ensure medical treatment, well-being for the patient, and patient safety (Simonsen et al., 2014; Stolic, 2014). Calculation skills are central parts of medication administration, meaning to know how to calculate required dosages of medication accurately (Council, 2010). For example in a hectic hospital environment, e.g. calculation of doses of tablets and solutions, and intravenous fluid rates, are demanding tasks which is performed several times per day (Pirinen et al., 2015). This requires that registered nurses are proficient in calculations to avoid adverse medication events.

Many registered nurses, as well as nursing students, however, lack proficiency in calculation skills (Fleming et al., 2014; McMullan et al., 2011; Oldridge et al., 2004; Özyazıcıoğlu et al., 2018; Simonsen et al., 2014; Wright, 2007). Many of the injuries occurring in health institutions, such as hospitals, are related to medication errors in registered nurses, and a particularly concern is dosages that are incorrectly calculated (Fleming et al., 2014; McMullan et al., 2011; Wright, 2007). Patients were potentially exposed to harm because of nurses incorrect administration of medication (Calabrese et al., 2001; Fleming et al., 2014). For example, of the 132 medication errors found in the study by Calabrese et al. (2001), 26 errors were deemed potentially life-threatening. The most common type of error was wrong infusion rate with 40.1% errors. Incorrect medication administration also has large economic consequences for health institutions, such as hospitals. Globally, the cost associated with medication errors, has been estimated to be at almost 1% of the total global health expenditure (WHO, 2017).

Improving medication calculation skills in nursing students and registered nurses is a prerequisite for ensuring patient treatment and safety (Wright, 2005). In the medication calculation course in the academic setting, both nationally and internationally, nursing students learn how to use basic math principles such as addition, subtraction, multiplication, fractions, percent and the unit system to calculate the doses of medication that should be administered to patients . Following completion of the lectures in medication calculation, testing is a requirement of nursing students (Council, 2010; National Curriculum Regulations for Nursing Programs, 2008). Both nationally and internationally the requirement for students is to achieve a 100% score on a calculation test or exam. It is documented that both nursing students and registered nurses are struggling in mastering the medication calculation test (McMullan et al., 2011; Sneck et al., 2016). It is also reported that students tend to have challenges in performing calculation skills during their clinical placement periods (Manno, 2006). Manno (2006) stated that errors were documented in many students' calculation performances even when they were under supervision from registered nurses.

Over the years, several learning tools have been developed to assist students in learning medication calculations. The most common of these are textbooks, aiming to provide students with theoretical and practical knowledge in calculation skills, e.g. Olsen (2014). Additionally, other learning tools, such as audio files, pictures, animations, text, simple tests and digital learning systems have been developed to facilitate students' ability to learn calculations (Mordt et al., 2011; McMullan et al., 2011; Sherriff et al., 2012; Stolic, 2014). Digital learning systems seem to be a popular approach to facilitate learning. Mordt et al. (2011), who in a Norwegian context developed a digital learning system for medication calculation, The Medication Game, found the game to be an important supplement to textbooks and lectures. Nursing students, evaluated The Medication Game to be fun, as well as it engaged the students in solving calculation tasks. The authors, however, concluded that The Medication Game did not significantly improve the students' test results in calculations (Foss et al., 2014). McMullan et al. (2011) who compared an e-package with traditional hand-out learning support, found that students who received the e-learning package were more able to perform calculations than those students using the handouts. The software had an instructional design of simple-to-complex sequence, meaning that the students should avoid cognitive overload by learning simple calculations before the more complex calculations. The authors, however, recommended that a random database for the medication calculation questions could be created to enhance students' ability to learn (McMullan et al., 2011). More recently, Park and Kim (2018) concluded on the effectiveness of a smartphone-based dosage calculation training app on nursing students' ability to learn medication calculations. The study results revealed that the smartphone-based calculation training app improved calculation skills, but only for nursing students with higher prior knowledge. The smartphone-based training program had inadequate effect on knowledge building on nursing students with lower prior knowledge Park and Kim (2018).

A closer reading of the articles (Mordt et al., 2011; Park and Kim, 2018; McMullan et al., 2011) showed that the digital learning systems developed did not take into consideration the individual student's state of knowledge and learning needs. Park and Kim (2018), in particular, recommended that a learning system should be developed for learners with lower prior knowledge. This demonstrates the recognized need for a learning system in medication calculations that adapts learning resources to the individual's state of knowledge and specific learning needs.

Chapter 3

Theoretical Perspectives

This chapter presents a theoretical positioning that frames learning of medication calculation, and two theoretical perspectives; cognitivism and sociocultural, are drawn upon in this master thesis. These theoretical perspectives promotes an understanding of decisions that have to be taken when an adaptive learning system in medication calculations should be developed. Cognitivism, concerned with the individual as a learner, emphasizes the importance of avoiding cognitive overload, and that new knowledge should build on previous acquired knowledge. A sociocultural perspective in learning, concerned with contextual conditions influencing learning, emphasizes that feedback and support should be given from a more proficient other, involving that the learners's state of knowledge, learning needs, as well as the expected learning outcome should be taken into consideration when feedback is given.

3.1 Learning

Learning is a complex phenomenon, where there is no general common understanding of how learning should be defined (Ertmer and Newby, 2013). Learning can be roughly defined as a change in thoughts and actions as a result of the addition of new knowledge to existing knowledge, or that existing knowledge is corrected by new or other knowledge (Gross, 2015).

3.2 A Cognitive Perspective on Learning

A cognitive learning perspective is based on the fact that learning is construction and reconstruction of knowledge. Learning takes place on the basis of the individual's curiosity and desire to learn, which can be seen as an inner motivation for learning. The acquisition of knowledge is an active process by the learner in which all new information is cognitively interpreted, assessed, and structured before it is added to previous knowledge (Piaget, 1985). To integrate new knowledge into existing knowledge is what Piaget (1985) called assimilation. He also claims that all new knowledge is acquired on the basis of existing knowledge. By acquiring knowledge, new cognitive schemas are formed, which includes the organization of information and knowledge into memory as a result of mental activity such as thinking, consciousness, organization, and understanding (Ertmer and Newby, 2013).

When new information conflicts with previously formed cognitive schema, cognitive conflicts or disequilibrium occurs. The cognitive conflict is the impetus for individual learners to obtain cognitive harmony or equilibrium, and thereby to make new cognitive schema. The disequilibrium is thus the impetus in the individual's learning process. The process of creating new cognitive schema is called accomodation (Piaget, 1985). The information acquired by the individual is processed in a short-term memory, then transferred to long-term memory so it will not be forgotten. Gradually increasing difficulty is emphasized to enhance learning, meaning that complex ideas are taught at a simplified level before they are revisited at a more complex level (Bruner, 2006). The theories of assimilation and accommodation show that the learner is an active participant in the construction of knowledge and understanding. Based on the cognitive activity that occurs during learning (Piaget, 1985), his theories are defined as cognitive constructivism.

3.3 A Sociocultural Perspective on Learning

In a sociocultural learning perspective, rather than considering knowledge as something to be gained individually, learning is understood as being socially constructed (Vygotsky, 1978). Vygotsky (1978) developed the concept of the zone of proximal development (ZPD). He argued that ZPD was the difference between what a learner could do independently and what she or he was able to do only with support, hints or guidance from a more knowledgeable ”other”. Support was developed on the basis of the notion of ZPD, and is understood as the help from a more proficient other that allows the learner to work within the ZPD, and then removed when support is no longer needed. In the ZPD, learners' should have the opportunity to participate in social interaction, and progress towards a more central participation, the next step in development (Vygotsky, 1978). Vygotsky (1978) stated that the social interaction between learners is often mediated through the use of physical and intellectual artifacts; tools for learning, and communication is a central component.

Chapter 4

Adaptive Learning Technology

Adaptive learning systems are learning systems that ”attempt to be different for different students” (Brusilovsky and Peylo, 2003). They tailor the learning environment to each individual learner instead of providing the same educational material, presentation and navigation to all learners in a one size fits all model (Abdo and Noureldien, 2016). Adaptive learning systems adapt key functions such as content presentation or workflow based on how students interact with content presented by the system (Murray and Pérez, 2015).

The design of an adaptive learning system will depend many on different elements, such as the area of the domain, what kind of information should be adapted and who the target user base is. However, there are three common components that are present in most adaptive learning system, the *domain model*, the *user model* and the *adaptation model* (Abdo and Noureldien, 2016; Murray and Pérez, 2015). The flow of data between these components is displayed in figure 4.1.

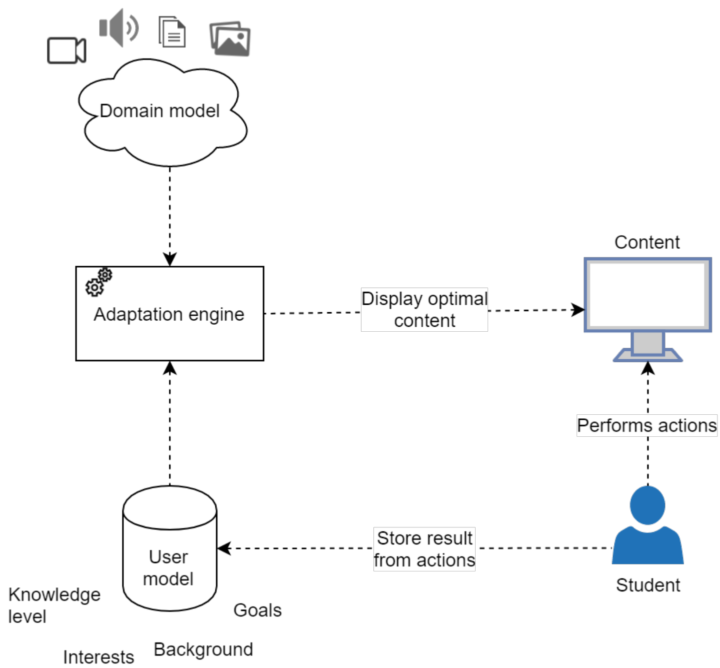

Figure 4.1: Common components in an Adaptive Learning System.

4.1 Domain Model

The domain model represents the material to be instructed. This includes domain-related elements of knowledge (often called *learning objects*), as well as the associated structure or interdependencies of those elements. In essence, the domain model is a knowledge map of what is to be taught (Shute and Torreano, 2003). Examples of learning objects can be text, images, audio, video, exercises, tests and even games.

In order for the adaptive system to decide which learning objects should be presented next, the learning objects need to be structured in some kind of semantic network. Authors can do this by providing additional metadata about each learning object, such as the relationships between objects. In medication calculation, for example, the domain can be divided into different categories (see 5.2.1), and these categories can be ordered such that the introductory categories are taught before the advanced categories.

4.2 User Model

The user model is a representation of information about an individual user. This information is essential for an adaptive system to provide the *adaptation effect*, i.e. to behave differently for different users (Brusilovsky and Millán, 2007). To create and maintain an up-to-date user model, adaptive systems collects data for the user model from various sources that may include implicitly observing user interaction and explicitly requesting direct input from the user. For example, an adaptive learning system could store results from exercises and tests that the user has completed. This process is known as user modeling (Brusilovsky and Millán, 2007).

Different users have different knowledge, goals, background, interests, and individual traits. This information is stored in the user model, and is continuously updated as the user performs different actions in the system. Data in the user model enables the adaptation model to recommend new learning objects that are personalized to each user. What kind of information that is stored in the user model largely depends on which adaptation effects the system has to deliver. For example, in an exercise drilling application in medication calculation, the results of previously solved exercises by the user can be used to determine the difficulty of the next exercise the user have to solve.

4.2.1 Knowledge Modelling

Knowledge is one of the most important features that are being modeled in educational adaptive learning systems. Most systems include knowledge as a feature in the user model, and sometimes it is the only feature that is modeled (Brusilovsky and Millán, 2007). The user's knowledge can change over time, it increases when the user is learning, and it decreases as the user is forgetting already acquired knowledge. This means that the adaptive system has to recognize these changes and update the user model accordingly. For example, in medication calculation the state of knowledge is updated whenever the user performs a calculation in order to reflect the change of knowledge.

4.2.2 The Scalar Model

The simplest form of a user knowledge model is the scalar model, which estimates the level of user domain knowledge by a single value on some scale – quantitative (for example, a number ranging from 0 to 5) or qualitative (for example, good, average, poor, none) (Brusilovsky and Millán, 2007). The knowledge value is typically produced by user self-evaluation or objective testing.

Even though knowledge modeling with a single scalar value produces a simplistic model, it can be effective in systems where the size of the domain is very small. However, the accuracy of this method decreases as the size of the domain increases. For example, in a learning application in medication calculation, a user might be an expert in calculating infusion rates, but a novice in calculating dilutions. The scalar method effectively averages the knowledge of the domain. For this reason the scalar method is not sufficient in larger domains such as medication calculation, however the scalar model is useful as a foundation in more complex models.

When the scalar method is not sufficient enough, a structural model, such as the overlay model which builds upon and expands the scalar model, can be used instead. The overlay model assumes that the body of domain knowledge can be divided into certain independent fragments. The overlay model attempts to represent different fragments of user knowledge independently. Each of these fragments, which now represents a smaller part of the domain, can be modelled using the scalar model. For example, in the context of medication calculation, the domain can be divided into different categories, e.g. measurement conversion, infusion, dilution, mixtures, etc.

4.2.3 The Overlay Model

One form of a structural knowledge model is an overlay model. The purpose of the overlay model is to represent an individual user's knowledge as a subset of the domain model, which reflects the expert-level knowledge of the subject. For each fragment of domain knowledge, an overlay model stores some estimation of the user's knowledge level of this fragment. This estimation can be either a boolean value, true if the user has sufficient knowledge of the fragment, false otherwise, or a scalar value. The scalar value can be a qualitative measure (good-average-poor) or a quantitative measure such as the probability that the user knows the concept (Brusilovsky and Millán, 2007).

The overlay model is a big improvement over the scalar model as it can more accurately model complex domains. For example, in the learning application in medication calculation, knowledge of infusion rates and knowledge of dilutions can be seen as two distinct fragments that can be modeled independently. The difference between the scalar model and the overlay model is illustrated in figure 4.2.

Scalar model

Knowledge level (0-100)
39

Overlay model

Concept	Knowledge level (0-100)
A	30
B	25
C	100
D	0

Figure 4.2: Scalar Model and Overlay Model.

Figure 4.2 shows that the scalar model only contains one measurement of knowledge for the entire domain, while the overlay model can measure knowledge in different fragments of the domain independently. Each of these fragments, which is a smaller part of the domain, can be modelled using the scalar model. Thus, an overlay model is a collection of scalar models.

4.3 Adaptation Model

The role of the adaptation model is to decide which learning objects to display next, and how to display them (Murray and Pérez, 2015). The model makes these decisions based on the current user model in a way such that the learning objects are adapted to the user's needs. For example, if a student has poor knowledge about a topic, the adaptation model may decide to show introductory level information about that topic. For a more experienced student, the model may show more details and more in-depth information about that topic.

In the context of medication calculation, it can be useful adapt presentation of learning objects based on how they skilled they are in different categories of medication calculation. Two methods that are often used in adaptive hypermedia systems, which can also be useful in an adaptive learning system in medication calculation, are *content adaptation* and *link adaptation* (Brusilovsky, 1996, 2016). Content adaptation is concerned with how learning objects are presented, and link adaptation is concerned with how the user navigates between them.

In addition to the methods for adaptive presentation of learning objects, spaced repetition is a method that improves retention of acquired knowledge (Kang, 2016), which can be useful in any learning system.

4.3.1 Content Adaptation

The content adaptation method is concerned with how learning objects are presented to the user (Brusilovsky, 2016). Various content adaptation methods can be applied to the learning objects in order to tailor them to the user's needs. Some examples of content adaptation techniques that can be useful in a learning system in medication calculation are:

Conditional Text - a technique that divides the page into several chunks (Brusilovsky, 1996). Each chunk is associated with one or several conditions which are evaluated based on data in the user model. Only the chunks where the conditions evaluates to true will be visible on the page (Kubeš, 2007). One example of conditional text, in the context of medication calculation, is to hide guidance on how to solve an exercise if the user already is at an expert level of knowledge.

Page Fragments - a technique that adaptively generates documents (Brusilovsky, 1996). Every document is an empty frame that is divided into sections. Each section can be populated by different learning objects, for example text, images, examples and links to other documents. Rules are used to decide which learning objects the different sections should contain. Which learning objects that are selected and how they are prioritized are unique to each user, thus every document is adapted to the user's needs (Kubeš, 2007).

4.3.2 Link Adaptation

In a space of many learning objects and activities, users can easily get lost or overwhelmed by all the content that exists in the system. The goal of link adaptation is to guide the user through the content in the domain model (Brusilovsky, 2016). Navigation support helps the user to choose the best path from the current learning object to the next object. Links are adapted to the user according to the current user model. Some examples of link adaptation techniques that can be useful in learning system in medication calculation are:

Link Hiding - a technique that adaptively hides links that are currently not relevant for the user (Brusilovsky, 1996; Kubeš, 2007). Hiding irrelevant links can help prevent the user from becoming overwhelmed or cognitive overloaded. Links can be hidden in different ways: *Disabling* - a disabled link looks like a normal link but it is not active (can not be clicked). Visual cues can be used to indicate that the link is disabled, for example by changing its color or add a special hover effect. *Hiding* - the link is present and active, but visually it looks like a normal word. *Removal* - the link is completely removed from the page.

Direct Guidance - a technique that provides a “next”-link that guides the user to the next learning object. The next learning object is adaptively selected according to data in the user model (Kubeš, 2007). For example, in an exercise drilling application after the user has solved an exercise, a ”next”-link can take the user to the next appropriate exercise.

Adaptive Navigation Maps - a personal overview of progress and available resources (Brusilovsky, 1996). The goal of an navigation map is to show the user where he is, where he has been and where he can go next. The map can be adapted for the user by using one or more link adaptation techniques, for example by using link hiding to disable links that the user can not visit yet (Kubeš, 2007).

4.3.3 Spaced Repetition

The timing or arrangement of practice affects learning. Practice is more effective when spaced out over time, instead of massed or grouped together, as it enhances memory, problem solving, and transfer of learning to new contexts. (Kang, 2016). An adaptive learning system can offer strategies for spacing out repetition based the learner’s earlier actions. For example, in an exercise drilling application, an exercise can be repeated at some interval, where the time between each interval is determined by how the user has previously answered that exercise. *The Leitner System* is a spaced repetition strategy that can be used for repeating learning items at increasingly larger time intervals (Edge et al., 2012).

The Leitner System

In the 1940’s, Sebastian Leitner devised a 5-step process, using index cards divided into 5 boxes. Flash cards are moved from the initial box (daily review) to the next if they are remembered, if not they are moved back to the first box. The process of moving cards is displayed in figure 4.3. Each subsequent box has a longer time lag before having its cards reviewed. Being able to remember a card in the final compartment (reviewed only after a lengthy interval) allows it to exit the system, with the assumption that is now stored in long-term memory (Godwin-Jones, 2010). The advantage of this approach is that more difficult items (i.e. items that are answered incorrectly) are reviewed more often (Edge et al., 2012).

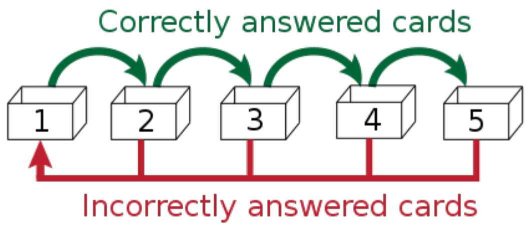

Figure 4.3: The Leitner System.

Figure 4.3 shows that correctly answered cards are advanced to the next, less frequent box, while incorrectly answered cards return to the first box.

In the context of medication calculation, the Leitner System can be used to space out repetition of different types of calculations. If the user performs a certain type of calculation correctly every time, the time interval of repetition for that type of calculation increases as the calculation moves to subsequent boxes. If the user performs a certain type of calculation incorrectly, that type of calculation will be put back in the first box, which means that it will be repeated shortly.

Chapter 5

Design of an Adaptive Learning System in Medication Calculation

In this chapter a design for an adaptive learning system in medication calculation, which aims to enhance health care providers ability to learn medication calculation skills, will be proposed. The goal of the adaptive learning system is to provide efficient training and learning of medication calculation skills by adapting learning resources to each individual learner's current state of knowledge and learning needs. This chapter will first describe how the domain model, the user model and the adaptation model are connected to each other in a system description. Then descriptions will be given on how each model can be implemented in an adaptive learning system in medication calculation.

5.1 System Description

The adaptive learning system is a platform that provides learners with exercises they can use to practice their medication calculation skills. The learning system adapts exercises to each learner's current state of knowledge, meaning that the exercises should be at an appropriate level of difficulty relative to the learner's current knowledge. As the learner's skill increases so should the difficulty of the exercises. The learning system also adapts the exercises based on the learner's current learning needs. Learning needs could be, for example, to practice on exercises in a category the learner has lower knowledge of, to repeat an exercises that the learner recently answered incorrectly, or to repeat an exercise from a category that the learner already has mastered in order to maintain that knowledge. Ultimately, the adaptive learning system's primary functions are; to decide which category should be practiced on next, select an exercise from that category, and to adjust that exercise to an appropriate level of difficulty.

Two different kinds of users exists; content managers and learners. The content manager's job is to manage learning resources. This is done through a separate content management system. The learners practice their medication calculation skills through the learning application, which is available as a web application and mobile applications for the Android and iOS operating systems. The flow of data from the content management system to the learning application through the adaptation components is displayed in figure 5.1.

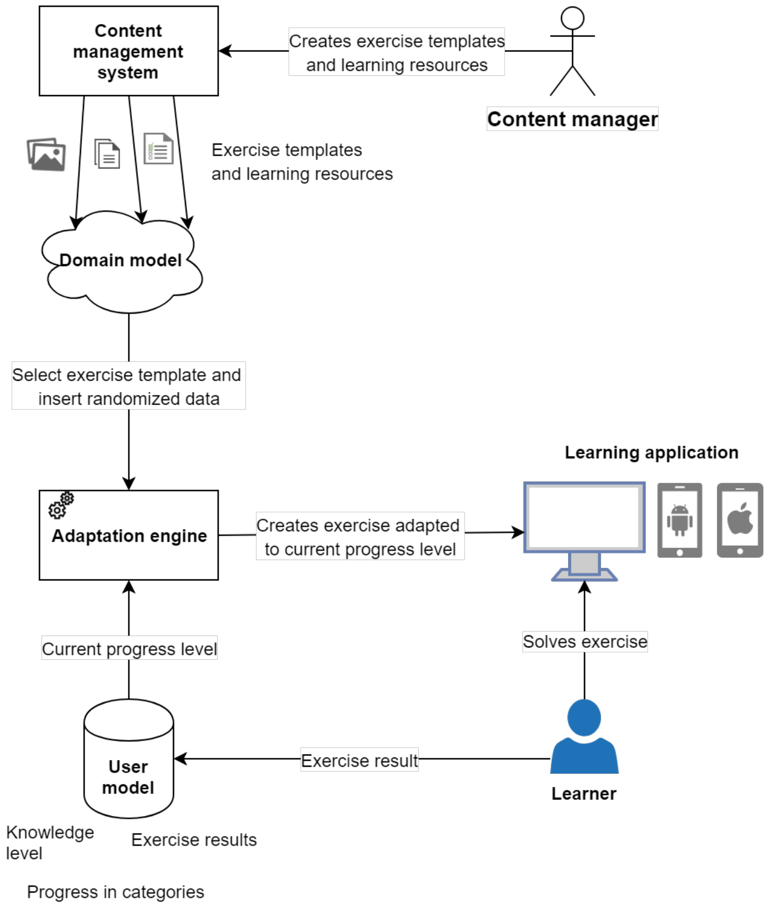

Figure 5.1: Overview of the adaptive learning system.

5.2 Domain Model

The domain model consists of different domain specific resources, such as categories, exercises templates and medication information, that are created by content managers. These resources are, together with data from the user model, used by the adaptation model to generate exercises that are adapted to the user's current learning situation.

5.2.1 Categories

The domain is divided into six different categories; measurement conversion, tablets, dilutions, infusions, mixtures and injectable medication. Every category have of a collection of exercise templates which are used to generate exercises that are specific to the given category. Each category, except measurement conversion, starts out being locked, which means that the user can not access exercises in those categories until they are unlocked. The categories require certain progress in other categories before they are unlocked. When a new category is unlocked the user gains access to exercises in that category. This is done to make sure that the user has acquired the prerequisite knowledge required to be able to perform calculations in the new category. The progression system is further explained in 5.3.

Categories have four different properties; name, order, support text, and level requirements.

Name

The name is the name of the category, for example tablets, dilutions, infusions or mixtures.

Order

The order is an integer that is used to decide the ordering of the categories when displayed in the clients. Categories with lower order are displayed before categories with higher order.

Support Text

The support text is guidance for how exercises can be solved. It can contain text, lists, tables, links to websites and images.

Level Requirements

Categories are locked until their level requirements are met. A level requirement specifies specific level in another category that needs to be achieved. For example, the category mixtures can require that the user has achieved at least level 7 in the category measurement conversion before mixtures are unlocked. A category can have multiple requirements.

5.2.2 Exercise Templates

In order to help the content managers create exercises more quickly than creating each exercise manually, an exercise template system is proposed. Instead of specifying specific values in a exercise, the manager can instead specify valid ranges for the values. When an exercise is created for a user, random values within the ranges are then selected and inserted into the exercise template. For example in an exercise about administering a medication to a child, the dose has to be calculated based on the child's weight. Instead of specifying a specific weight, e.g. 15.5 kg, a range could be used instead, e.g. [10.0, 25.0].

When a user is to solve an exercise generated by this template, a random number between 10.0 and 25.0 would be selected as the child's weight.

These exercise templates are stored in the domain model and are used by the adaptation engine to generate exercises with random values. This means that many variations of an exercises can be created from the same exercise template. The user can thus get practice on the same type of exercise many times until he understands and remembers the procedure of how to calculate the correct answer. An additional side effect of using random values is that, since hundreds (or even thousands) of different variations of the same exercises can be generated, it is not practical to just remember the correct answer, the user has to learn the correct calculation procedure.

Exercise template is the most complex type of resource in the domain model as it contains many properties and a template system.

Name

The name of the exercise template is only used by the content managers to more easily be able to distinguish between different exercise templates.

Category

Exercise templates belong to a specific category. The category can be selected from a dropdown menu of all available categories.

Medication

Exercise templates can be specified to contain a type of medication, e.g. tablets. If an exercise template is set to contain a medication, different properties for medications are made available to be used in the exercise text, exercise question, formula and multiple choice alternatives.

Exercise Text

The exercise text provides the user with all the information about the exercise that is needed to perform the calculation. For example, "Physician orders 500 mg of ibuprofen for a patient, and you have 250 mg tablets on hand.".

Exercise Question

The exercise question tells the user what they should calculate. For example, building on the example from *Exercise Text*, "How many tablets will you administer?".

Formula

The formula specifies how the correct answer to the exercise is computed. This formula is used to verify whether the user provides correct or incorrect answers.

Difficulty

Some exercises are easier to solve than others. Difficulty of an exercise template is an integer that indicates how hard the exercise is to solve. Higher difficulty means harder to solve.

Support Text

The support text is guidance for how an exercise that is generated by the template can be solved. It can contain text, lists, tables, links to websites and images. Usually the procedure of how to perform the calculation is explained in the support text.

Tablet Data

Tablet data becomes available if tablets is selected as the medication type for the exercise template. The tablet data are different properties of tablets that can be used in exercise text, exercise question, formula and multiple choice alternatives.

Custom Data

Custom data is similar to tablet data, except that the content manager can create custom data properties. Custom data consist of a name, which can be referred to in input fields that support template data, start and end of range, which creates the range in which a random value can be picked, and the number of decimals that the randomly picked value should contain.

Multiple Choice Alternatives

Multiple choice alternatives are created in the same way as the formula. One alternative should be identical to the formula (which would be the correct answer), while the other alternatives should differ from the correct answer.

Input Fields with Support for Template Data

The exercise text, exercise question, formula and multiple choice alternatives have support for template data. Template data is just a placeholder that later are replaced with real data. That real data is often a random value that is selected from a range, but it can also be a string such as name of tablet or unit of measurement. Template data is a key-value pair where the key is used as a placeholder in the input field and the value is what the placeholder will be replaced with. Placeholders are words that are surrounded with "{{}}", for example "{{Name}}". A complete example of this can also be seen in figure 6.3.

5.2.3 Medication Information

To further improve the exercise template system information about various medications are also stored in the domain model. This information, such as name, valid dose limits, unit of measurement and available medication strengths, can also be used as parameters in the exercise template in addition to the value ranges previously mentioned. Similarly to the value ranges, ranges can also be specified for the various properties of the medications, allowing the system to generate multiple variations of the same medication. Instead of specifying a specific medication and values that are valid for that medication, the content manager can instead specify that certain properties of the medication should be used in the exercise template. When an exercise is generated from the template, a random medication is then selected and values for that specific medication are randomly picked within the valid ranges and inserted into the exercise. For example if the content manager is creating an exercise about calculating the number of tablets of a medication that should be administered to a patient, he could then specify to use medication name, medication strength, and a valid dose for the medication without specifying any specific values. When the exercise is generated a random medication is selected and its name, medication strength and dose measurements are inserted into the template. The medication strength is randomly selected from a list of available strengths, and the dose measurement is randomly selected from a valid range for that medication. A more comprehensive example can be found in 6.2.1.

When adding a new medication to the domain model that medication can be used to generate new versions of all exercise templates that contain randomized medication properties. Thus, if there exist many exercises that use randomized medication properties in the domain model, then new variations of these exercises are made available every time a new medication is created in the domain model. Similarly, if there are many medications available in the domain model, then whenever a new exercise template that use randomized medication properties is created it can use all the already available medications in the domain model to generate different versions of that newly created exercise template. In addition, since information about the medications is specified as ranges of values that can be randomly selected, many variations for each exercise using the same medication can also be generated. Consequently, as content managers create resources such as exercise templates and medication information, the number of possible variation of exercise grows exponentially related to the number of resources created, which is a very efficient way of limiting the amount of time required of human resources to create a large varied database of exercises.

Different types of medications have different types of properties. In this master thesis only tablets have been implemented. However, the concept is the same for other kind of medications. Tablets have six different properties; name, unit of measurement, maximum amount in one dose, list of available strengths, maximum amount that can be administered during a day and a boolean if the tablet can be split.

When populating the database with medication information, the information should be validated using a trusted database for medication data, for example Felleskatalogen AS (s.a.). Felleskatalogen AS (s.a.) is a large database of information about medications that is used by health care professionals; doctors and nurses in Norway.

Name

The name is the name of the tablet, for example Levaxin, Paracetamol or Isoptin.

Unit of Measurement

The active ingredient is measured in a unit of measurement. Common units are μg, mg and g.

Maximum Amount in One Dose

The recommended maximum amount of the active ingredient that can be taken in one dose.

List of Available Strengths

Tablets are often produced in different variations of strengths.

Maximum Amount During a Day

The recommended maximum amount of the active ingredient that can be taken during 24 hours.

Tablet Can Be Split

Some tablets are splittable, meaning that they can be divided into two or four pieces.

5.3 User Model

The user model aims to accurately reflect how skilled the user is in the different categories of medication calculation. The user model is therefore an overlay model where the domain of medication calculation is divided into different categories (Brusilovsky and Millán, 2007). Knowledge of each category is modelled using the scalar model, where the users can progress from level 1 to level 10 (Brusilovsky and Millán, 2007).

When users solve exercises in a category their progresses are tracked using a level system. Progress in each category starts at level 1, and it increases as the user performs calculations and gains knowledge. In order to progress from one level to the next the user has to obtain a certain number of ”stars” at the current level. Stars are gained when answering exercises correctly and lost when answering exercises incorrectly, thus the user has to answer more exercises correctly than incorrectly in order to progress further. Table 5.1 shows an example of how knowledge of a user is represented in the user model.

Category	Level	Number of stars
Measurement conversion	8	1
Tablets	6	0
Dilutions	5	2
Infusions	2	2
Mixtures	1	0
Injectable medication	1	0

Table 5.1: Knowledge representation in the user model.

In the first few levels of a category one correct answer would reward you one star. However, as the user progresses further to higher levels each correct answer only rewards a fraction of a star. The higher the level the user is at, the more exercises must be answered correctly in order to ”build up” a whole star. If an exercise is answered incorrectly the user will lose all star fractions collected in the star that is currently being built up. This means that the users have to answer multiple exercises currently in a row in a category in order to build up a whole star. If an exercise is answered correctly when all stars at the current level are completely built up, the user will advance to the next level. How many stars are needed at each level, and how many correct answers that must be answered in a row in order to build up a star at each level is displayed in table 5.2.

Level	Number of stars	Correct answers in a row
1	3	1
2	3	1
3	3	1
4	3	2
5	3	2
6	3	3
7	3	4
8	4	4
9	4	5
10	5	5

Table 5.2: Number of stars and correct answers in a row needed to build a star at each level.

Table 5.2 shows that, for example, on level 5 the user has to build up 3 stars in order to progress to the next level. In order to build up a star the user has to answer 2 exercises correctly in a row. The number of required stars and the number of required correct answers in a row decides how fast the progression is at each level, therefore the progression in the earlier levels is faster than in the later levels. The progression through levels is further visualized in section 6.4.4 and section 6.4.5.

If a user answers 3 or 4 exercises correctly in a row he will receive a bonus point (star fraction) whenever he answers an exercise correctly. If he answers 5 or more exercises correctly in a row he will receive two bonus points. This will help users who are at a higher knowledge level than what is reflected in the user model to progress to a more appropriate level faster.

5.4 Adaptation Model

The adaptation model in this adaptive learning system is responsible for generating exercises for users based on their current state of knowledge and learning needs. The adaptation model does this in three steps; first select the category, then select the exercise template, lastly decide the difficulty of the exercise.

5.4.1 Category Selection

The first step in generating an exercise for a user is to select which category the user should practice on next. Which category should be selected is decided by a probability model that is based on the users current progress in the different categories. Categories where the user has progressed the furthest has the least probability of being selected. Thus, categories where the user has made the least progress is more likely to be selected. This probability model will ensure that the user spends the majority of his time practicing on categories he is least knowledgeable about, but still sometimes practice on categories he is more knowledgeable about in order to maintain that knowledge. Categories that are locked have zero probability of being selected. Unlocked categories are each given a probability factor equal to

$$\begin{aligned} x &= maxLevel - currentLevel + 2 \\ probabilityFactor &= x * log(x) \end{aligned} \tag{5.1}$$

where *maxLevel* is the maximum possible level (set to 10 in this project) and *currentLevel* is the current level the user has progressed to in that category. All probability factors from level 1 to level 10 are listed in table 5.3.

Level	$x = maxLevel - currentLevel + 2$	$probabilityFactor = x * log(x)$
1	11	26.38
2	10	23.03
3	9	19.78
4	8	16.64
5	7	13.62
6	6	10.75
7	5	8.05
8	4	5.55
9	3	3.30
10	2	1.39

Table 5.3: Probability factors for each level.

After all probability factors are calculated for all unlocked categories they are added together to create a sum of all probability factors,

$$sum = \sum probabilityFactor(category) \tag{5.2}$$

where $probabilityFactor(category)$ is the probability factor for a given category. The probability that a given category is selected is then

$$p(category) = \frac{probabilityFactor(category)}{sum} \tag{5.3}$$

For example, if a user has unlocked three categories, CatA at level 9, CatB at level 6 and CatC at level 2, the probability factor will then be 3.30 for CatA, 10.75 for CatB and 23.03 for CatC. The sum of probability factors is 37.08. The probability for each category being selected next is displayed in table 5.4.

Category	Level	Probability factor	Probability of being selected
CatA	9	3.30	8.9%
CatB	6	10.75	29.0%
CatC	2	23.03	62.1%

Table 5.4: Probability for each category to be selected next.

5.4.2 Exercise Template Selection

After the category is selected, the next step of generating an exercise is to select which exercise template of that category should be used. The selection itself is very simple. All unlocked exercise templates are ordered in a priority queue, so the process of selecting the next exercise template is just to select the template that is first in the queue. The complexity, however, lies in how the exercise templates are enqueued. All exercise templates in the queue are given a priority, the first item in the queue is the one with the lowest priority value, the last item in the queue is the one with the highest priority value. When enqueuing an exercise template to the queue a priority value must be determined, which will decide the exercise template's position in the queue. The aim is to give exercise templates that the user has proven good knowledge of a high priority value, so that there is a longer time until those exercise templates are repeated again. And on the other hand, exercise templates that the user has proven poor or no knowledge of should be given a low priority value, so that there is a shorter time until those exercise templates are repeated. As the user repeatedly answers exercises generated from the same exercise template correctly, the time between repetitions of that exercise template should increase between every interval, which is exactly how the Leitner System (Godwin-Jones, 2010) provides spaced repetition. The algorithm developed for determining the priority value is inspired by the Leitner System (Godwin-Jones, 2010), which will increase the interval between repetitions as long as the user keeps answering the same exercise correctly, or decrease the interval between repetitions if the user answers the exercise incorrectly.

Similarly to the Leitner System (Godwin-Jones, 2010) exercise templates are placed in buckets. The buckets are numbered 1, 2, 3, 4... etc. Exercise templates in buckets with lower numbers are repeated more frequently than the exercise templates in the buckets with higher numbers. The calculation of the priority value for an exercise template depends on how many exercises are solved in the given category, and which bucket that exercise template was previously in. If the user answers an exercise incorrectly, the corresponding exercise template is always put back in the first bucket. This means that the incorrectly answered exercise will be repeated soon, which gives the user a chance to correct his error. If the user answers an exercise correctly, the corresponding exercise template is put in the next bucket, i.e. the previous bucket number plus one. After the bucket number is decided the priority value is calculated using the following formula,

$$priorityValue = exerciseCount + 10 * 2^{(bucketNumber-1)} \tag{5.4}$$

where $exerciseCount$ is the number of exercises the user has previously solved in the given category and $bucketNumber$ is the number of the bucket the exercise template is put in. The priority value increases exponentially as the bucket number increases. This seems to work well for the limited size of exercise templates available in the learning system developed for this master thesis, however, this function can be tweaked if this strategy turns out to be too aggressive on a larger database of exercise templates.

The exerciseCount's function is to move all items in the queue closer to the front every time the user answers an exercise. This is done to ensure that items that are far back in the queue are gradually moved to the front so that they eventually will be selected. An alternative method would be to decrement all priority values of the exising items in the queue every time the user answers an exercise, however, this is not technically practical as every item in the queue must be updated in the database. By using the exerciseCount to increase the priority value of all new items that are enqueued the existing items in the queue do not need to be updated. Nevertheless, the effect is still the same.

5.4.3 Exercise Difficulty

After the exercise template has been selected it is time to generate an actual exercise from that template. Exercises can be generated in four different levels of difficulty. From easiest to hardest these are: with support and multiple choice, with support and no choices, no support and multiple choices, and no support and no choices. The levels of difficulty are visualized in table 5.5. Support is text and images that guides the users on how to answer the specific exercise they are currently trying to solve. On exercises with support the user can view these guidelines in order to help them solve the exercise. If support is not available on the exercise they have to solve it by themselves. On exercises with multiple choice the user are presented with multiple possible answers where one of them is correct. The user can then select one answer and submit the exercise. If there is no choices available the users have to input the answers themselves in an input field.

Difficulty (1 - easy, 4 - hard)	With support	With multiple choice
1	YES	YES
2	YES	NO
3	NO	YES
4	NO	NO

Table 5.5: Levels of difficulties.

Which level of difficulty an exercise is created with depends on the current progress the user has in the given category, and how the user answered that same corresponding exercise template last time if he has answered an exercise of that template before. If the user has answered an exercise of that template before, the level of difficulty on the next exercise will depend on the result of the last exercise. If the user answered the previous exercise correctly the next exercise will be at one level of difficulty harder, if the answer was incorrect the exercise will be at one level of difficulty easier. If the user has not answered an exercise of that template before, the difficulty will depend on his current level in the given category. The lower the level he is at, the easier the exercise will be generated. On some levels of progress, multiple different difficulty levels can be assigned to an exercise. If that is the case a random level of those difficulties will be selected. The settings for which level of difficulty are available at different levels of progression is displayed in table 5.6.

Level in category	Possible difficulties
1	support and multiple choice
2	support and multiple choice, support and no choice
3	support and no choice, no support and multiple choice
4	support and no choice, no support and multiple choice
5	support and no choice, no support and multiple choice
6	support and no choice, no support and multiple choice, no support and no choice
7	no support and multiple choice, no support and no choice
8	no support and multiple choice, no support and no choice
9	no support and multiple choice, no support and no choice
10	no support and no choice

Table 5.6: Possible levels of difficulties for each level of progress.

As displayed in table 5.6, when users first starts solving exercises in a new category they are supplied with both support on how to solve the exercise and multiple choices. As they progress further these aids are gradually removed until, at the last level, they get no support and no choices.

Chapter 6

Implementation of the Adaptive Learning System

An adaptive learning system in medication calculation has been developed where learners can practice their medication calculation skills and content managers can manage learning resources.

The learners can access the learning application through a web app, Android app or an iOS app. The web app is available through web browser, e.g. Chrome, Safari, Firefox, etc., and it is optimized for use for devices with large screens, such as computers and laptops. The Android and iOS app are available for smaller devices, i.e. smartphones and tablets.

A separate application, the content management system, has been developed for the content managers where they can manage resources in the domain model. This application is only available as a web app.

The content management system and the learning applications communicate with a common application server through a JSON HTTP API. It is on the server that the domain model and user models are stored, and it is also where the adaptation methods are executed. A high level architecture of the learning system is displayed in figure 6.1.

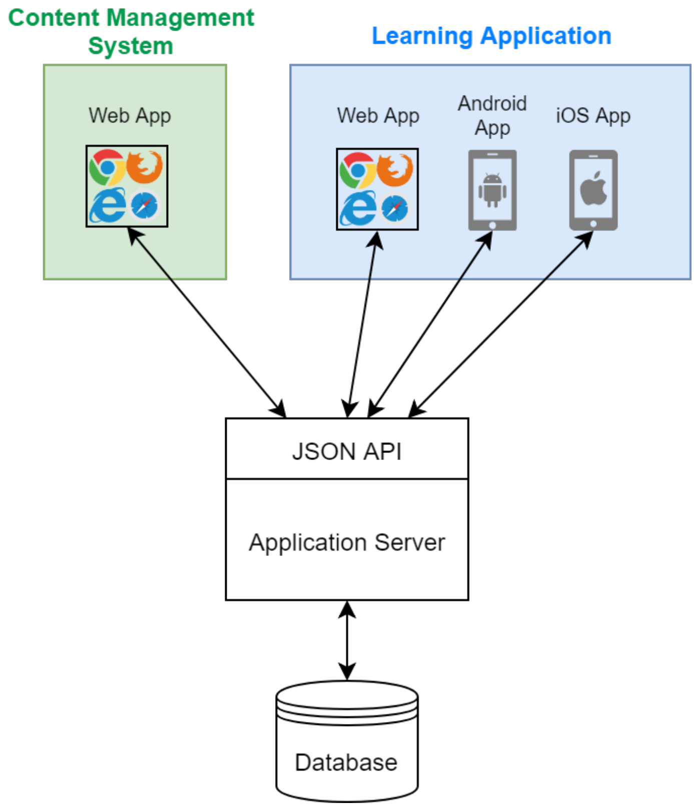

Figure 6.1: High level architecture for the adaptive learning system.

6.1 Technology Choices

Different programming languages, frameworks and libraries have been used to develop the different applications which comprise the adaptive learning system.

6.1.1 Application Server - JSON API

The C# programming language with the ASP.NET Core framework have been used to develop the application server. ASP.NET Core is a cross-platform, high-performance, open-source framework for building modern, cloud-based, Internet-connected applications (Introduction to ASP.NET Core, s.a.). This framework has built in methods that supports rapid development of HTTP APIs. For data persistence the MySQL database was chosen as it is free to use and integrates well with the ASP.NET Core framework.

6.1.2 Web App

The web apps for both the learning application and the content management system have been developed using standard web technologies (i.e. HTML, CSS and Javascript) and the Vue framework. Vue is a framework for developing user interfaces, and in particular single-page applications, i.e. a web page with a single page where instead of navigating to different pages the content of the single page is swapped out using Javascript. A single-page application is a good fit as a client for the learning applications, as the application is highly interactive and not so much concerned with browsing multiple documents.

In addition to the Vue framework the Vuetify component library was used to create the user interface. The Vuetify library contains pre-built components (e.g. buttons, form elements, animations, etc.) for the Vue framework, which follows the Material Design guidelines (Material Design, s.a.).

6.1.3 Android and iOS Apps

The Dart programming language with the Flutter framework was used to create both the Android app and the iOS app. Flutter is Google's mobile app SDK (Software Development Kit) for crafting high-quality native interfaces on iOS and Android (Flutter, s.a.). The framework is cross-platform, meaning that the same code can be used to compile both the Android app and the iOS app, which essentially cuts development time in half compared to developing a separate native app for both platforms.

The Flutter framework comes with pre-built material design components, which are visually similar to the components from the Vuetify library in the web app. Using material design components in both the web app and mobile apps makes the user interface look and feel very similar.

6.2 Application Server

The application server contains all the persisted data and all the business logic which comprise the adaptive learning system. This is where the domain model, user model and adaptation model are defined and stored. The application server exposes resources and functions through a JSON API. Clients can request or persist resources by sending HTTP requests to specified routes. For example, by sending a HTTP request to */api/exercise* the server would respond with an exercise object that belongs to the currently logged in user.

When the server receives a HTTP request it might have to execute some logic before it can generate a response. The *controller* is responsible for executing this logic. When the request is first received at the server, the *router* will route that request, based on the URI path, e.g. "/api/users", to the controller that is responsible for handling requests to that specific path. The controller can then call on different *services*, units of business logic, to execute the logic needed in order to generate a response. For example, when a user requests a new exercise on "api/exercise" the request is routed to the controller responsible for generating exercises. The controller can then call on the "Adaptation Engine" to create a new exercise that is adapted to the currently logged in user. The "Adaptation Engine" service can in turn call on other services, or fetch data from the database, in order to execute its logic and generate a new exercise. Finally the controller can respond back to the client with the newly generated exercise as a JSON object.

The data flow on the server is illustrated in figure 6.2, and it usually looks like this:

1. The server receives a HTTP request.
2. The request is routed, based on the route (path), to the corresponding controller.
3. The controller calls on different services in order to execute business logic.
4. The controller produces a JSON response which is returned back to the client.

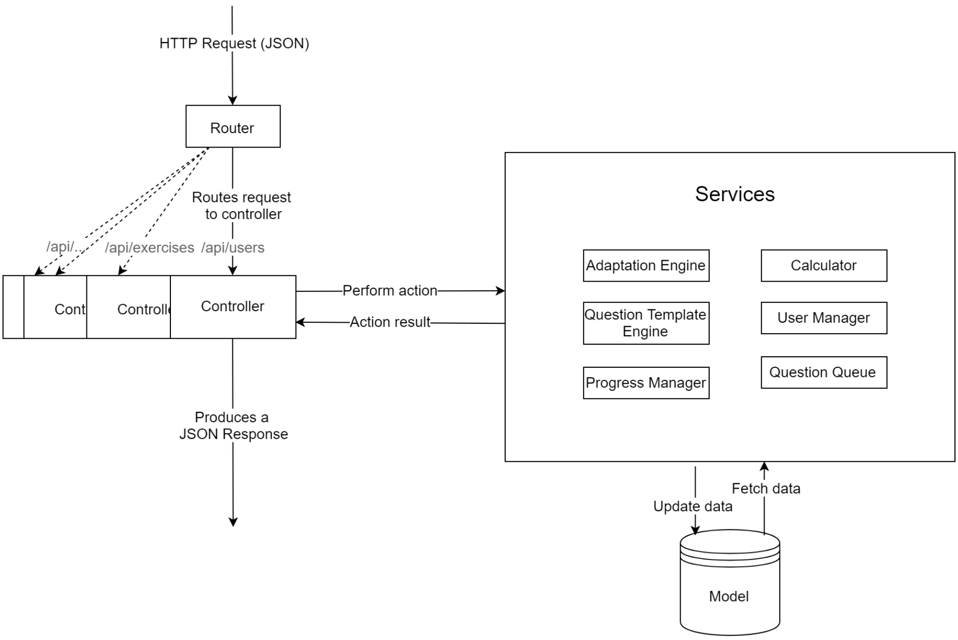

Figure 6.2: Application Server Architecture.

6.2.1 Exercise Generation

One of the primary functions of the application server is generate exercises that are adapted to the user's current state of knowledge and learning needs. Exercises are generated from exercise templates in multiple steps:

1. Category selection
2. Exercise template selection
3. Exercise difficulty selection
4. Medication randomization

5. Template rendering

A comprehensive example will be used to illustrate the entire exercise generation process. The user's state of knowledge in this example is displayed in table 6.1. The "measurement conversion", "tablets" and "dilutions" categories are unlocked while the other categories are locked. This user requests a new exercise, so the process of generating the exercise starts on the application server. The first step is selecting the category for the exercise.

Category	Level	Number of stars
Measurement conversion	8	2
Tablets	3	1
Dilutions	5	0
Infusions	1	0
Mixtures	1	0
Injectable medication	1	0

Table 6.1: Knowledge representation in the user model.

Category Selection

The probability of each category being selected is calculated by using formula 5.3. The probability of each category being selected in this example is displayed in table 6.2. In this example the *Adaptation Engine* service was used to select the next category, and it returned the tablets category. The next step is then to select an exercise template from the tablets category.

Category	Level	Probability of being selected
Measurement conversion	8	14%
Tablets	3	51%
Dilutions	5	35%
Infusions	1	0%
Mixtures	1	0%
Injectable medication	1	0%

Table 6.2: Probability of each category being selected.

Exercise Template Selection

All exercise templates in the tablets category have a priority value which is specific for the user in this example. These priority values were calculated using formula 5.4. In this example, the *Question Queue* services was used to select the exercise template with the lowest priority. The Exercise template is displayed in figure 6.3. The next step is to decide the level of difficulty for this exercise.

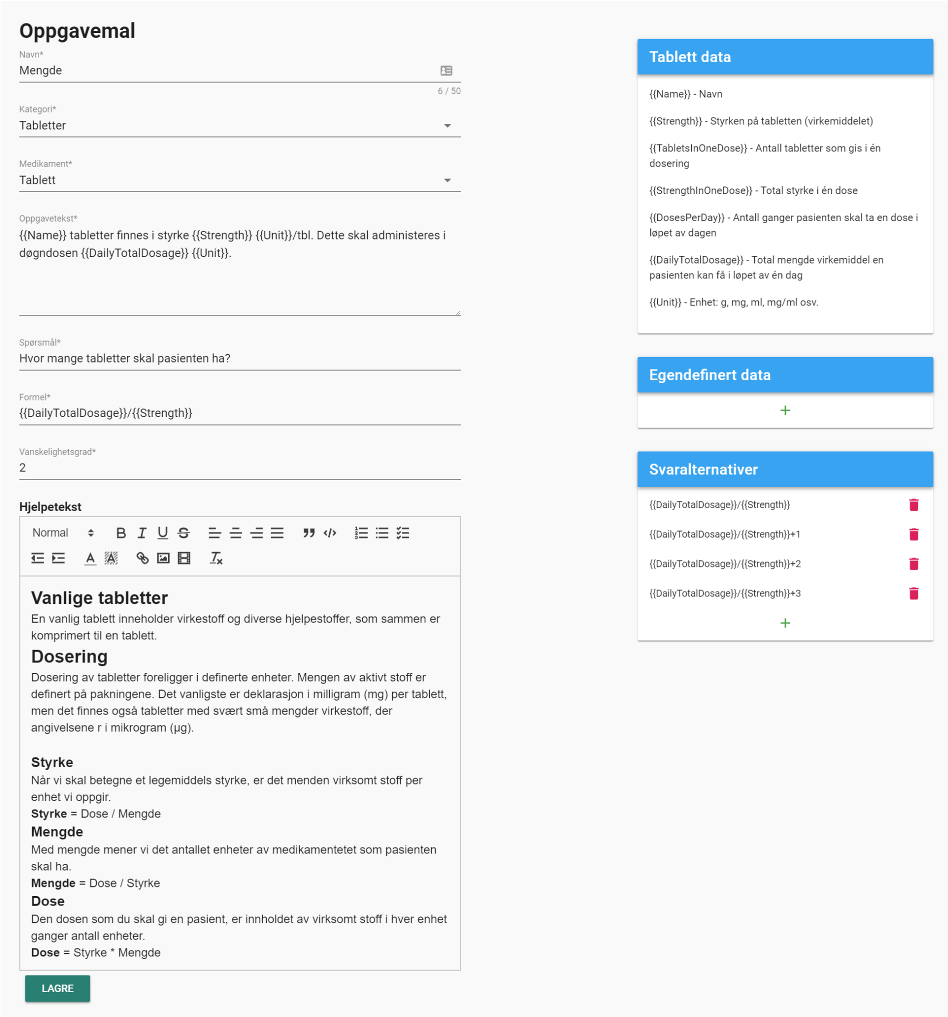

Figure 6.3: Exercise template.

Exercise Difficulty Selection

The user has not answered exercises that was generated from this template before, and therefore from table 5.6 we can see that the possible difficulties for this exercise are ”support and no choice” or ”no support and choice”. The ”no support and choice” difficulty was randomly selected by the *Adaptation Engine* service, which means that the exercise does not have any support text, but it will have multiple choice options.

Medication Randomization

The exercise template in this example uses information about medications. A specific type of medication must therefore be randomly selected among all the medications in the domain model. The *Glucophage* tablet was chosen in this example, and its properties are displayed in table 6.3.

Property	**Value(s)**
Maximum amount in one dose	1000 mg
Maximum amount during 24 hours	2550 mg
Different tablet strengths available	500mg, 850mg, 1000mg
Unit	mg

Table 6.3: Properties of the Glucophage tablet.

From the properties of the Glucophage tablet a medication object with specific values are created. The specific values will be randomly selected within the limits of the properties in table 6.3. The specific values of the medication object is displayed in table 6.4.

Property	**Value(s)**
Name	Glucophage
Unit	mg
Strength	500
DailyTotalDosage	2000

Table 6.4: Specific values of the medication object.

The *Medication Randomizer* service was used to both randomly select the medication and to randomize its properties. The specific values of the medication are now available to use in the template system.

Template Rendering

The last step of the exercise generation is to use the *Exercise Template Engine* service to replace all the placeholders in the exercise template with the specific values mentioned above.

The exercise text in the template is: ”{{Name}} tabletter finnes i styrke {{Strength}} {{Unit}}/tbl. Dette skal administeres i døgndosen {{DailyTotalDosage}} {{Unit}}.”
The result exercise text is: ”Glucophage tabletter finnes i styrke 500 mg/tbl. Dette skal administeres i døgndosen 2000 mg”.

The exercise question in the template is ”Hvor mange tabletter skal pasienten ha?”. Since it does not contain any placeholder the result exercise question is the same.

The formula in the template is: ”{{DailyTotalDosage}}/{{Strength}}”.
The result formula is: ”2000/500”.

The alternatives in the template are:

1. ”{{DailyTotalDosage}}/{{Strength}}”
2. ”{{DailyTotalDosage}}/{{Strength}}+1”
3. ”{{DailyTotalDosage}}/{{Strength}}+2”
4. ”{{DailyTotalDosage}}/{{Strength}}+3”

The result alternatives are:

1. ”2000/500”
2. ”2000/500+1”
3. ”2000/500+2”
4. ”2000/500+3”

Finally, The *Exercise Template Engine* will use the *Calculator* service to calculate the values for the formula and the alternatives. The final value for the formula will then be 4, which is the correct answer in this exercise, and the alternatives will be 4, 5, 6 and 7. The final exercise is displayed in figure 6.4.

Figure 6.4: The final rendered exercise.

6.3 Content Management System

Resources, such as categories, medication information and exercise templates, in the domain model are created and maintained by content managers. A content management system was developed where content managers can manage resources via a web application. Only users with admin rights have access the content management system.

6.3.1 Categories

Categories have four different properties; name, order, support text, and level requirements. A screenshot from the content management system on how to manage a category is displayed in figure 6.5.

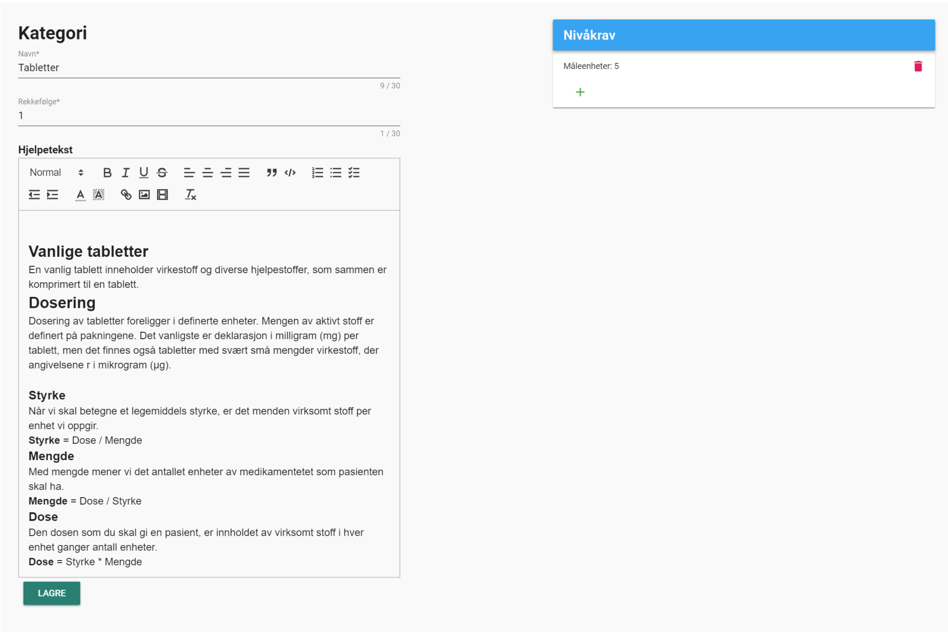

Figure 6.5: Manage category.

6.3.2 Medication Information

Different types of medications have different types of properties. In this master thesis only tablets have been implemented. However, the concept is the same for other kind of medications. Tablets have six different properties; name, unit of measurement, maximum amount in one dose, list of available strengths, maximum amount that can be administered during a day, and a boolean if the tablet can be split. A screenshot from the content management system on how to manage a tablet is displayed in figure 6.6.

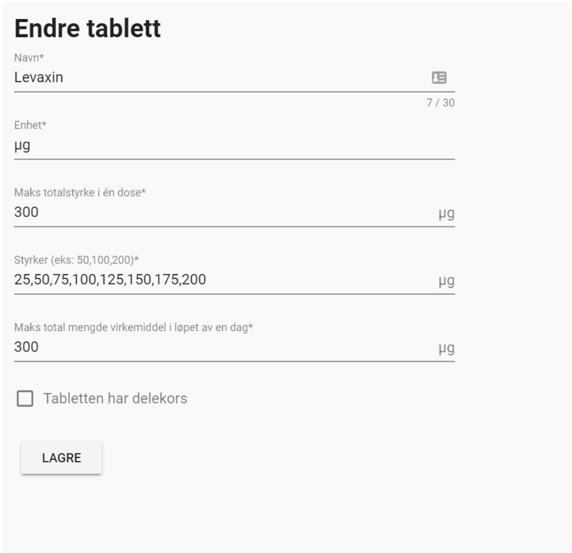

Figure 6.6: Manage medication.

6.3.3 Exercise Templates

Exercise template is the most complex type of resource in the domain model as it contains many properties and a template system. A screenshot from the content management system on how to manage an exercise template is displayed in figure 6.3.

6.4 Learning Application

The learning application is where the users log into in order to view their progresses in different categories and to solve exercises. To make the learning application convenient to use, whether they are using computers or smartphones to access the application, both a web application and mobile applications for Android and iOS have been developed.

The learning application is divided into 4 main pages; the login/registration page, the overview page, the exercise/result page, and the progress page. When users visit the learning application for the first time they are routed to the login page. Before they can access the rest of the application they have to either log in to an existing account or register a new one. The navigation flow between the different pages is displayed in figure 6.7.

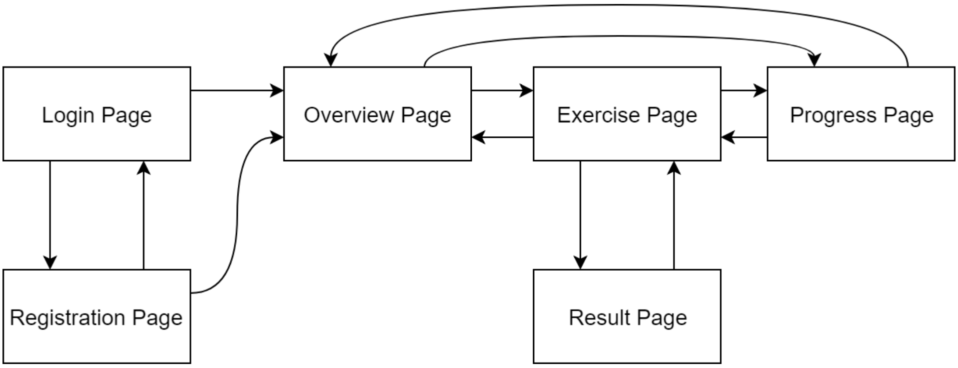

Figure 6.7: Navigation structure in the learning application.

6.4.1 Login/Registration Page

On the login/registration page the user can sign up or log in to an existing account. The user can log in on same account on both the web app on mobile apps. Screenshot of the login page is displayed in figure 6.8.

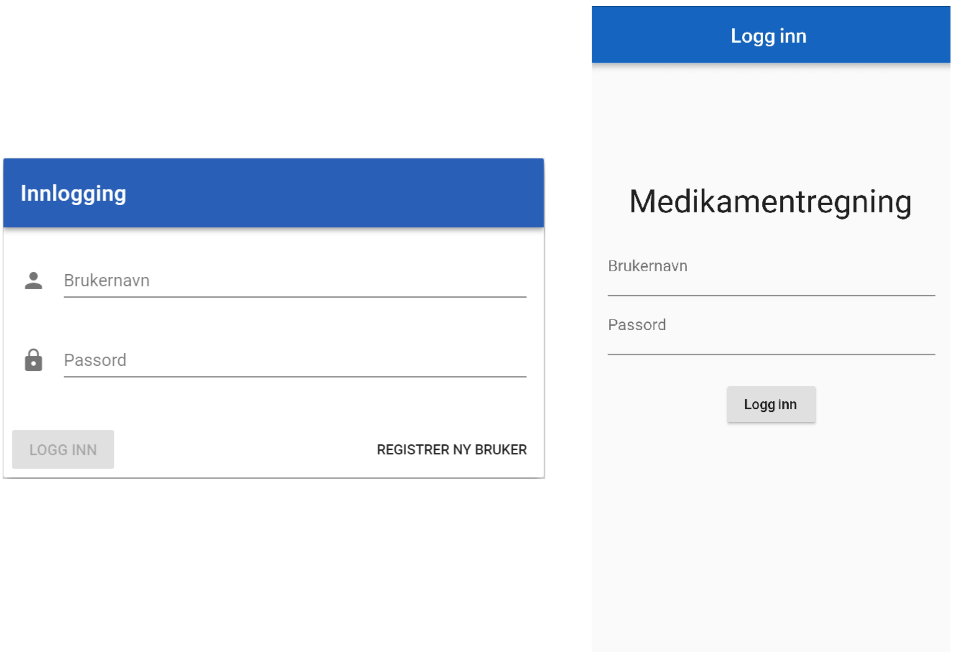

Figure 6.8: Login/registration page. Web app on the left, mobile app on the right.

6.4.2 Overview Page

On the overview page the user can see the current overall progress in all categories. The progresses are visualized using progress bars. When no exercises are solved correctly in a category the progress bar is empty. It fills up as the user progresses through the levels until the max level is reached. When the max level is reached, and all the stars are completely built up, the progress bar will be fully loaded. A screenshot of the overview page from the web app is displayed in figure 6.9, and a screenshot of the overview page from the mobile app is displayed in figure 6.10.

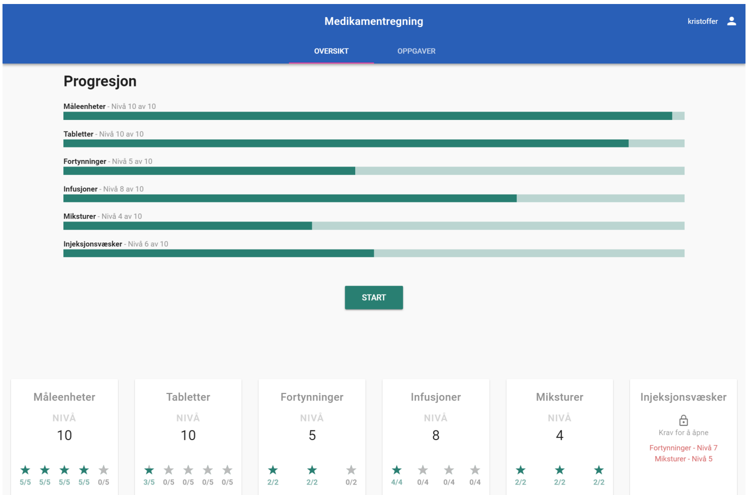

Figure 6.9: Overview page on the web app.

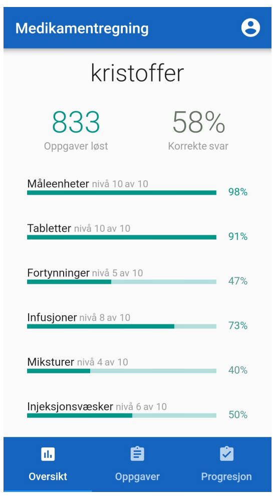

Figure 6.10: Overview page on the mobile app.

6.4.3 Exercise Page

On the exercise page the user can solve exercises. If support resources are available for the exercise it is displayed next to the exercise on the web app, and on a separate page on the mobile app. On exercises that have multiple choices the alternatives are displayed as radio buttons under the exercise question. If there is no choices an input field where the user can type in the answer is displayed instead. The exercise page can be seen as a fragmented page (Brusilovsky, 1996) where the section for support text, the section for the exercise text and question, and the section for the input/alternatives, are fragments that are subject to being adapted. In the web app, the support text section uses the *conditional text* technique where the text is visible or hidden depending on the user's current state of knowledge (Kubeš, 2007). On mobile, the *link hiding* technique is used instead to display an active or disabled link to the support text (Kubeš, 2007). In the input section adaptive rules are used to decide if the input type should be a text input field or alternatives, which is a *page fragment* technique (Brusilovsky and Millán, 2007; Kubeš, 2007). The user is guided from one exercise to the next by clicking on the "next"-button. The *direct guidance* technique is used to adaptively select the next exercise (Kubeš, 2007). A screenshot of the exercise page from the web app is displayed in figure 6.11, and a screenshot of the exercise page from the mobile app is displayed in figure 6.12.

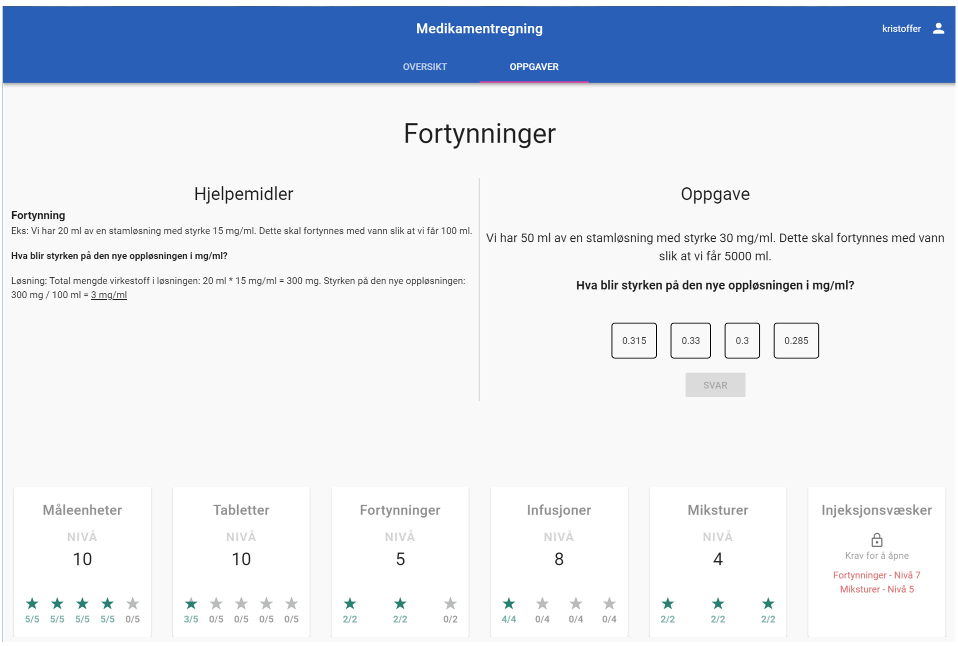

Figure 6.11: Exercise page on the web app.

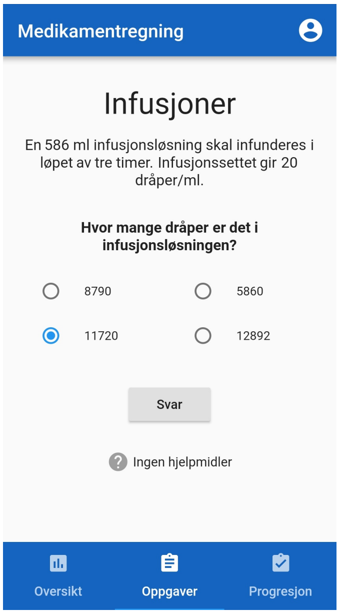

Figure 6.12: Exercise page on the mobile app.

6.4.4 Result Page

After the user submits an exercise they get redirected to the result page. The result page will display whether they answered the exercise correctly or incorrectly, and how that affected their progress. A screenshot of the result page from the web app is displayed in figure 6.13, and a screenshot of the result page from the mobile app is displayed in figure 6.14.

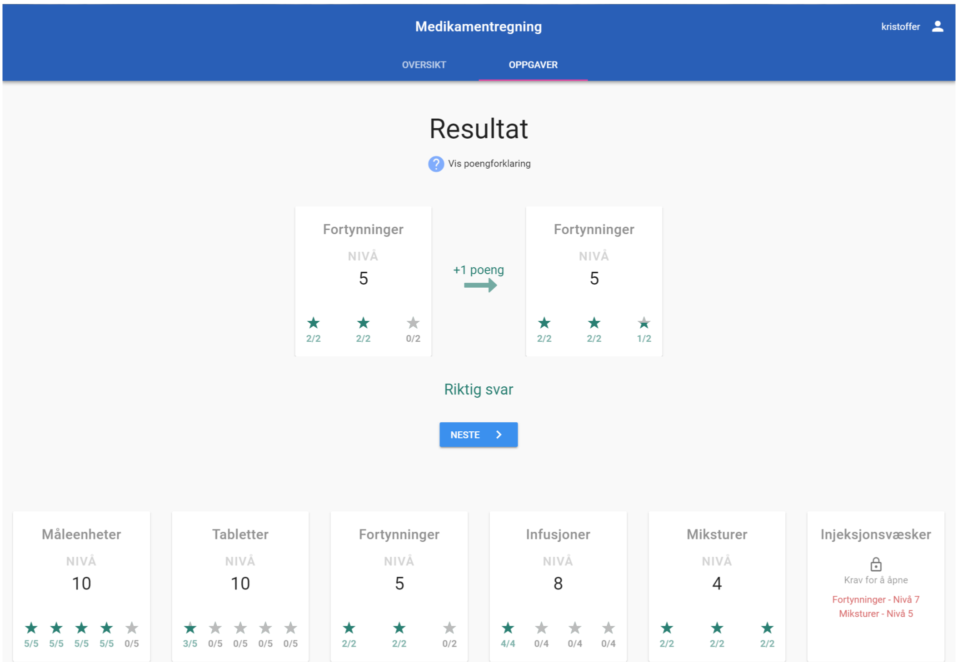

Figure 6.13: Result page on the web app.

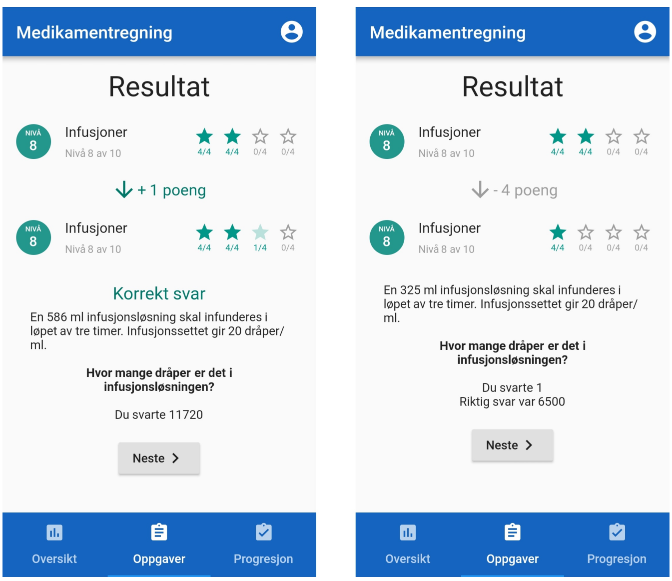

Figure 6.14: Result page on the mobile app.

6.4.5 Progress Page

In the web app, users can see their progresses in all categories at the bottom of the page, as can be seen in figure 6.9, figure 6.11 and figure 6.13. Because of the limited size of mobile screens it was not practical to display the progresses in a similar way in the mobile app. Instead, a separate page, the progress page, was created to display progresses on mobile. The progress page is a list view of all the current progress in each category. Clicking on a category will open up a more detailed page where the users can see how many exercises they have solved in that category and their rate of success. The detailed page will also show any requirements needed to unlock that category if it is currently locked. The progress page can be seen as an adaptive navigation map (Brusilovsky, 1996), it visualizes how far the user currently has progressed in each category, and it shows where the user can not navigate to yet (locked categories).

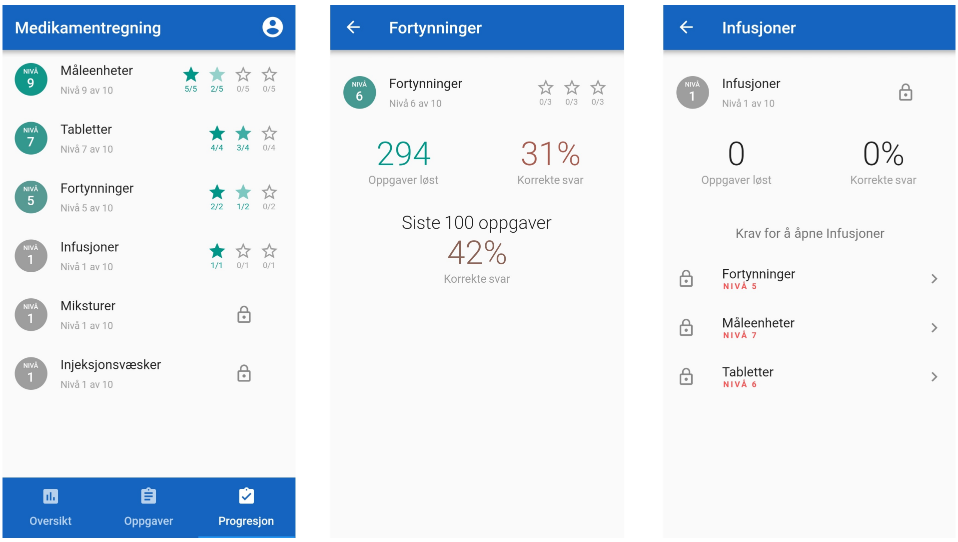

Figure 6.15: Progress page on mobile. On the left: list view of progress in each category. In the center: details page of an unlocked category. On the right: details page of a locked category.

Chapter 7

Evaluation

To test whether the developed adaptive learning systems does adapt to users' state of knowledge and learning needs or not, an evaluation has been performed. The evaluation focused on how the users experienced the adaptivity of the system. After completing 50 exercises the participants were asked to complete a survey of 8 questions as seen in table 7.1.

Id	Question
Q1	Hvor vanskelig erfarte du oppgavene?
Q2	I hvor stor grad erfarte du at oppgavene ble tilpasset ditt nivå?
Q3	I hvor stor grad erfarte du at oppgavene ble tilpasset ditt læringsbehov
Q4	I hvor stor grad erfarte du at det adaptive læringssystemet ga deg hjelp til å løse oppgaver som for deg var vanskelige?
Q5	I hvor stor grad erfarte du å repetere oppgaver du tidligere hadde løst?
Q6	I hvor stor grad hadde du ønsket hyppigere repetisjon av tidligere løste oppgaver?
Q7	I hvor stor grad ville du foretrukket å bruke dette adaptive læringssystemet fremfor tradisjonell oppgaveløsing fra lærebok?
Q8	Har du noen andre tilbakemeldinger?

Table 7.1: Evaluation Questions.

Table 7.1 show the questions the participants were asked. In questions Q1-Q7 the participants where asked to give a numerical answer on scale from 1 to 5. On Q8 they were free to write any feedback or comments they might have.

7.1 Ethical Considerations

Norwegian Social Science Data Service (NSD) (Norsk senter for forskningsdata, s.a.) has strict policies for collecting and processing personal data. NSD requires that projects that collects or processes personal data must first notify them and get approval for collecting the data. However, a project is not subject to notification if all electronic data processed through the entire research process is anonymous. For online surveys, NSD specifically require that the IT solution is completely anonymous (among other things, the respondent's email or IP address cannot at any moment be connected to the survey), and that the survey does not contain questions about identifiable information.

In this project no personal or sensitive information was collected or processed during the user registration, usage of the application, or in the survey, and therefore was not subject to notification. The survey data was collected through a custom survey system that was created specifically for this project. The participants were never asked to enter any personal data, and their IP addresses was never stored or connected to the survey in any way. The survey was also not connected to any users in the learning application. Consequently, it was not possible to track which users from the learning application that had participated in the survey. Also, since there was no way of tracking who participated in the survey, it can not be guaranteed that the same user has not participated multiple times. It is important to note that the users were informed that participation in the project was voluntary.

7.2 Evaluation Results

20 people registered a user. 18 people answered the survey. Each registered user solved 181 exercises on average. The results from Q1-Q7 are displayed in table 7.2:

Id	Average	Standard Deviation
Q1	3.3	0.7
Q2	4.5	0.8
Q3	4.3	0.7
Q4	4.4	0.7
Q5	4.3	0.7
Q6	3.2	1.0
Q7	4.3	0.7

Table 7.2: Evaluation Results.

Table 7.2 shows the average score on each numerical evaluation question (Q1-Q7), as well as the standard deviation.

7.2.1 Numerical Responses from Q1-Q7

All questions were answered with a numerical score on a scale from 1-5. In the following the average score and a description of each question will be presented.

Q1 - Exercise Difficulty

Avg. score: 3.3
A low score on this question would mean that the exercises were too easy, a high score would mean they were too difficult. An average score of 3.3 suggests that the exercises was slightly on the harder side.

Q2 - Adapting to Current Knowledge Level

Avg. score: 4.5
A low score on this question would mean that the level of difficulty of the exercises did not adjust according to the users' state of knowledge, a high score would mean that the exercises became more difficult as their skills increased, or that the exercises became easier if they had trouble solving them correctly. An average score of 4.5 suggests that the level of difficulty adjusted to meet the users current state of knowledge.

Q3 - Adapting to Learning Needs

Avg. score: 4.3
A low score on this question would mean that the user did not get exercises from the categories needed to progress further, a high score would mean that the exercises helped the user to progress further. For example, giving a user many exercises from a category that he already has mastered would not help him progress further. An average score of 4.3 suggests that the exercises helped the user to progress.

Q4 - Provided Help When Needed

Avg. score: 4.4
A low score on this question would mean that the user did not get support on exercises they found too difficult, a high score would mean that they got the support needed to solve exercises that were challenging. An average score of 4.4 suggests that the users got the support needed to solve challenging exercises.

Q5 - Repetition Frequency

Avg. score: 4.3
A low score on this question would mean that the users did not experience to repeat earlier solved exercises, a high score would mean that they experienced often to repeat earlier solved exercises. An average score of 4.3 suggests that the users often experienced to repeat exercises.

Q6 - Wants More Repetition

Avg. score: 3.2
A low score on this question would mean that the users did not want more repetition, a high score would mean that they wanted more repetition. An average score of 3.2 suggests that the users are happy with the frequency of repetition.

Q7 - Prefers Adaptive System Over Traditional

Avg. score: 4.3
A low score on this question would mean that the user prefers to solve exercises with pen and paper from a textbook, a high score would mean that the user prefers to solve exercises using the adaptive learning system. An average score of 4.3 suggests that the users preferred to use the adaptive learning system rather than solving exercises from a textbook.

7.2.2 Other Comments and Feedback

Feedback from the free text field, Q8, can be divided into two categories; challenges and issues, and improvement suggestions.

Challenges and Issues

Getting Stuck - If a user did not understand how to solve a specific exercise in a category, she would answer that exercise incorrectly every time (unless if she got lucky by selecting the correct answer in a multiple choice exercise). That exercise would then be repeated too frequently, which prevented her from progressing further in that category, even if she could answer all other exercises correctly. Two users reported there was one exercise they did not understand how to solve, even when they got support available on that exercise. That exercise required more steps to perform the calculation than other exercises, thus it can be argued that this exercise was more complex.

Larger Variety of Exercises - Users that already were comfortable at doing medication calculations had a very high accuracy at solving them correctly. Users with high accuracy will progress faster through the exercises, and thus unlocking all the exercises faster. After they had unlocked most of the exercises they spent most of their time repeating exercises they already had solved before. These users wanted a larger variety of exercises so that they would spend less time on repeating old exercises and instead spend time on new and challenging exercises.

Improvement Suggestions

Practice Specific Category - Categories are locked until their requirements are met. Some participants reported that they wanted to be able to select a specific category to practice on, even if they hadn't unlocked that category yet.

More Exercises that Use Real Medication Names - Exercises that involves medication calculation are either generic or for a specific medication. Participants reported that they liked exercises that used real medication names, and wanted more exercises of that kind.

General Feedback

Most of the participants expressed that they enjoyed using the adaptive learning system, and that they thought that it would be a useful system when learning medication calculation. They especially pointed out how it was useful to see a visualization of what they were good at and what they needed more practice on, and that the system automatically provided exercises that they needed to practice more on.

Chapter 8

Discussion

In this master thesis, an adaptive learning system has been developed and explored to enhance nursing students' and registered nurses' ability to learn medication calculations. The learning system has been tested and evaluated by some health care providers: nursing students, nursing teachers, and registered nurses. Feedback given from the users provided useful insight into how they experienced the adaptivity of the learning system, and is used as a basis for the following discussion. Discussion of feedback given from users is divided into three themes; "Knowledge acquisition based on individual state of knowledge and learning needs", "Mentoring based on individual state of knowledge and learning needs", and "Repetitive learning and correction of errors".

8.1 Knowledge Acquisition Based on Individual's State of Knowledge and Learning Needs

An overarching result of the evaluation in this master thesis was that the developed adaptive learning system did indeed adapt exercises according to the individual user's state of knowledge and learning needs. As users solved exercises, the difficulty increased or decreased to match their current state of knowledge. Feedback related to Q1, how difficult the users experienced the exercises, suggests that the difficulty of the exercises was appropriately adjusted to their current state of knowledge. Feedback related to Q2, how users experienced that exercises were adjusted to the individual's state of knowledge, suggests that the level of difficulty was continuously adjusted as their current state of knowledge changed. Feedback related to Q3, how users experienced that exercises were adjusted to the individual's learning needs, suggests that the adaptive learning system selected exercises which helped the users to progress further by gradually exploring new contents of the domain. Based on this, it could be interpreted that users at all states of knowledge were helped to fill their knowledge gaps in medication calculations. Several attempts have been made to increase nursing students' ability to learn medication calculations by using digital learning systems (McMullan et al., 2011; Mordt et al., 2011; Park and Kim, 2018). However, it seems like earlier developed digital learning systems provide learners with low level of prior state of knowledge limited ability to learn medication calculations. One explanation for this inadequate ability to learn could be that earlier developed digital learning systems probably did not help students at a lower state of knowledge to avoid cognitive overload, prolonged cognitive conflicts and confusion. The importance and process of assimilation and accommodation, the integrating of new knowledge into existing knowledge, and the creating of new cognitive schema (Piaget, 1985), were maybe not well enough taken into consideration. It is important to note that even though students are at the same level of education and attend identical learning sessions, they might come with different prior state of knowledge and experience in calculation skills. Consequently, a complex digital learning system is necessary to capture personalized learning needs (Murray and Pérez, 2015).

It seems that the adaptive learning system developed in this master thesis helped users avoid experiencing cognitive overload, long-term unresolved cognitive conflicts and confusion. By providing exercises with and without aids based on the individual's state of knowledge and learning needs, users were helped to gradually achieve learning goals. It could be interpreted that medication calculations mostly were adapted to each individual user. The difficulties that users experience due to a lack of mastering and confusion could, however, be a positive element in learning of medication calculations if the user is made aware of what is causing the confusion. Mills (2016) revealed that students who were able to work through their confusion gained knowledge and expanded their understanding. This could be interpreted according to Piaget's (1985) understanding of constructive learning. He stated that cognitive conflicts was a prerequisite for achieving cognitive equilibrium, and thereby to make new cognitive schemas. However, when cognitive conflicts remain unsolved over a prolonged period, it could be challenging for the learner to acquire new knowledge. Consequently, as revealed in the current project, when a lack of understanding occurred (as two users reported concerning one particular exercise), the user became stuck on their current state of knowledge rather than expanding it. This could illustrate that a learner who fails to work through a confusion or develop new cognitive schemas, also fail to acquire sufficient knowledge and gain a deeper understanding of medication calculation. Consequently, the learner could avoid difficulties in learning and withdraw from the learning situation (Ravik et al., 2017a).

Insights from the evaluation shows that users experienced that new knowledge was gradually building on previous knowledge by advancing exercises based on their individual current state of knowledge. In the learning system developed by McMullan et al. (2011), there was also a focus on learning that was advanced from simple to more complex. Advancing from simple to complex is a general understanding of learning. A gradual advancement in what to be learned could provide development of new cognitive schema, and furthermore provide knowledge to be transferred to long-term memory and consequently not be forgotten (Bruner, 2006). Former cognitive schemas could be reorganized and expanded based on previous knowledge and understanding (Piaget, 1985). It could be interpreted that the adaptive learning system developed in the current project equipped users to better understand and develop medication calculations, both by taking into consideration the individual's state of knowledge and learning needs, but also by advancing exercises from simple to complex depended on the individual user.

8.2 Mentoring Based on Individual's State of Knowledge and Learning Needs

The evaluation data revealed that the adaptive learning system developed in the current project guided each individual user in a way that was personalized to their current state of knowledge and learning needs. Feedback related to Q4, how the users experienced help provided from the adaptive learning system when they faced difficulties in ability to learn calculations, suggests that enough feedback and guidance were provided on exercises they otherwise could not solve without any help. The users were provided extra learning resources when they needed help to progress further. Adding individually scaffold to a learning system, seems to be distinctly different from previously developed learning systems in medication calculations. Common for previous developed digital learning systems (Mordt et al., 2011; McMullan et al., 2011; Park and Kim, 2018) is that they contain various learning resources for learning and practicing medication calculations, but also that it is up to learners to assess what they themselves need for knowledge and learning. Interestingly, earlier research (Bagnasco et al., 2016) points out that learners do not have enough self-awareness to be able to assess their actual state of knowledge, and thus their learning needs. This could may explain why students actually want feedback, and particularly one-to-one feedback, from a more proficient other to progress in their ability to learn (Mema and Harris, 2016).

The adaptive learning system could be understood as what Vygotsky (1978) described as an artifact. In this case, a technology-based artifact that aims to identify a connection between the learner and sense making consistent with medication calculation skills. The adaptive learning system thereby becomes a link between the user and what is to be learned. The personalized contribution of the learning system contributed to an individual assessment of the learner, and stimulated the user to progress in learning, to the next level of understanding and development, based on the individual's state of knowledge and learning needs. The adaptive learning system was "knowledgeable" enough to map the user's need for knowledge, and provided feedback that avoided the learning process becoming too hard or too simple, in the ZPD toward a more central participation (Vygotsky, 1978). It could be interpreted that the adaptive learning system addressed the gap in what learners could solve alone and what they could solve with help.

Based on Vygotsky (1978) taught of learning, the teacher is to be considered a key element for students' ability to learn. The teacher, as a more proficient other of medication calculations, collaborate with the learners to help them advance in learning and understanding. Rather than considering knowledge as something to be gained individually (Piaget, 1985), Vygotsky (1978) suggested that knowledge was better understood as what a learner could do with skilled help that allows the learner to work within the ZPD. This illustrated that learning of medication calculation is realized through human interaction. Based on a sociocultural learning perspective whereas a social interaction, and a more proficient human being necessary to contribute to students' ability to learn, it could be questioned whether the use of a digital learning system contributed to a learning-weak form of collaboration, and thereby restriction of users ability to learning. Ravik et al. (2017b), however, found while exploring nursing students simulation based practical skill learning, that students were inadequately mentored by the teacher, the more proficient other. Students training on a latex arm were not given feedback regarding challenging steps of the practical skill peripheral vein cannulation (PVC). Teachers need to recognize what learners actually are doing to provide adequate feedback (Hattie and Timperley, 2007). For the teacher difficulties could, however, arise when several students need feedback and support at the same time, for example to catch the individual's challenges in ability to learn. In contrary, the evaluation data in the current project showed that the adaptive learning system in medication calculations was updated as the user's state of knowledge changed, meaning that feedback was given based on the individual's need for help to understand and progress in calculations. Consequently, one could still suggest the adaptive learning system to be a learning-weak form in collaboration, but, however, a learning strategy that provided effective learning outcomes in calculations.

It is, however, important to note that information given from users in the survey revealed that two of them became stuck because they were not given ability to learn and understand how to solve a specific exercise. From the free text comment feedback in Q8, two users reported there was one particular exercise they did not understand how to solve, even when they got all the feedback and support available on that exercise. Failing to answer that exercise correctly hindered them in progressing further, even if they could answer all other exercises correctly. This effectively caused them to become stuck at their current level, or even decreased to a lower level. Inadequate feedback, consequently, did not close the gap between current and desired state of knowledge. Hattie and Timperley (2007) argued that constructive feedback is necessarily to not inhibit students' learning. It is not known what level of knowledge these two users had, but on closer inspection of the specific exercise they did not master, it is an exercise that requires a more comprehensive and complex calculation than the other exercises. This indicated that the learning resources of this exercise was not good enough. Consequently, the scaffold in the adaptive learning system should be adjusted or modified to eliminate the issue of being stuck.

8.3 Repetitive Learning and Correction of Errors

In the evaluation users gave feedback concerning repetition of exercises. Feedback related to Q5, how often they experienced to repeat previously solved exercises, and related to Q6, whether they wanted more a higher frequency of repetition, suggest that the users had different experiences with the rate of repetition that was given. Some users reported that they experienced to repeat previously solved exercises more often than solving new exercises they had not solved before. The rate of repetition could be reduced by expanding the database of exercises, making more exercises available. Nevertheless, the participants also reported that they were happy with the frequency of repetition. Repetition turns out to be important for memory, which can be seen as encoding knowledge, storing it over time, and retrieving it again (Dunlosky et al., 2013). Based on this, one could understand that exercises which are executed repetitively, could be improved from trial to trial, and consequently help the learner to become skilled and proficient. This was particularly seen in the study by Ravik et al. (2017a) who compared one student who repetitively mastered vein cannulation with one student who failed the same skill in each attempt. The student who repetitively mastered vein cannulation, became skilled and developed a deep level of understanding in this particularly invasive skill. Repetition is thereby, as Leitner also assumed, a critical element for learning to occur (Godwin-Jones, 2010). It is, however, important to note that repetition in itself does not necessarily make users become skilled in given tasks (Ravik et al., 2017a). A great importance should be given to feedback accorded to repetitive performances. Saville (2011) points out the importance of giving the learner both positive and negative feedback, where the effectiveness of repetition is increased when the learner is given feedback. She stated that positive feedback can make students aware of correct actions that improve mastery, and negative feedback that can make students aware of errors and what needs to be corrected. This could be interpreted as two approaches to feedback that intends to provide support for correcting errors, and furthermore repetitive exercises and enhanced ability in learning.

The adaptive learning system provided the learners with immediate feedback on correct and incorrect calculations, and a particular goal of the adaptive learning system was to provide support for correcting errors if a user made a mistake. Immediate error corrections is important to avoid repetitive incorrect performances (Ravik et al., 2017a). Gardner et al. (2015) argued that revealing errors could provide a deeper understanding of what to be learned, and prepare users for the development of appropriate performances. The results of the evaluations shows that when users answered an exercise incorrectly, they experienced that the system made them aware of errors and repeated that exercise shortly, with more support added. In this way, the user could deal with new knowledge through assimilation, mastering new challenges (accommodation), and consequently experience equilibrium (Piaget, 1985).

It is, however, important to note that the two users who in the free text on Q8 reported that they were stuck on a particular exercise, did not progress in their calculations even they calculated the same exercise repetitively or were made aware of errors performed, a negative feedback that Saville (2011) emphasized as important in learning. Becoming aware of errors did not mean that the users understood how to correct them. Even though they were given all the support available from the adaptive learning system, it seemed like they were operating outside their ZPD (Vygotsky, 1978). The users were not given opportunities to achieve correct calculations, and consequently they remained in an unpleasant state of disequilibrium because new knowledge not could be fitted into existing cognitive schemas (Piaget, 1985). Knowledge building in calculations suffered because the artifact, the adaptive learning system, that should mediate the connection between the users and what to be learned, did not demonstrate the complexity of calculations well enough. It could be understood that the two users did not receive enough and accurate feedback based on their current state of knowledge to make calculations correctly. This illustrated that the comprehensiveness of feedback should be considered so the learner could develop intellectually. This implies that the calculation itself as a learning resource should be designed with different options for calculation, in order to provide good feedback based on all the user's state of knowledge.

Chapter 9

Conclusion

This master thesis has highlighted that exposure to an adaptive learning system is an important step in the construction of knowledge in medication calculations, and thereby how students ability to learn calculations could be improved. The adaptive learning system offers strategies to improve students' ability to learn how to calculate medications accurately and to identify mistakes. It is, however, important to note that the adaptive learning system requires improvement to provide better learning conditions and learning outcomes.

9.1 Limitations and Strengths

Although the aim of developing and exploring an adaptive learning system in medication calculation has been reached, there were some limitations. First, because of the time limit, the exercises and learning resources created in this project does not cover the entire curriculum in medication calculation. Therefore, this learning system should not yet be used as the only platform for practicing and learning calculation skills. Second, results from the survey revealed that two users became stuck and were unable to progress further in one certain category. The learning resources did not provide adequate support for them to overcome the lack of knowledge in this specific area. Finally, the learning resources were limited to contain only text and images. Having a larger variety of learning resources, e.g. animations and video, could further improve the system's ability to better meet each individual's learning needs.

Despite the limitations, the adaptive learning system developed in this project proved to be a useful learning resource in medication calculations. It is a basis for building missing elements of knowledge such that students of all levels of knowledge have a full-fledged way to learn medication calculations. The learning system also provides insight into how users at different levels of knowledge could learn, but also what consequences it has for the learner if it lacks sufficient learning resources in order for acquisition of new knowledge to happen.

9.2 Future Work

This section includes recommendations for further development and research.

9.2.1 Expand the Variety of Learning Resources

To better meet each individual's learning needs, more different types of learning resources, e.g. animations, videos, audio tracks, should be developed to cater for more learners' preferred ways of learning. By offering more ways to learn how to solve an exercise, the learner have more options and opportunities to overcome their lack of knowledge if they get stuck on an exercise.

9.2.2 Practice on Exercises in Locked Categories

In the evaluation some users reported that they wanted the opportunity to practice in categories they had not yet unlocked. Therefore, an improvement to the system would be to allow users to select a specific category they want to practice on, regardless if they have unlocked it yet or not. If a specific category is chosen, the user will only get exercises from that category. However, it is important to note that the user could risk becoming cognitive overloaded if they choose a category outside their ZPD.

9.2.3 More Exercises with Medication Names

The evaluation also revealed that users wanted more exercises that includes real medication names. Since the exercises are not defined in the system itself, but are instead created by content managers, there is no need to implement any changes in the learning system in order to fulfill this request. Instead content managers should create more exercise templates that include real medication names.

9.2.4 Adaptive Feedback

After answering an exercise the user receives an immediate feedback on whether they answered it correctly or incorrectly. Just being made aware of an error is not sufficient for the user to be able to correct it, the feedback should additionally also include appropriate learning resources so that the learner can understand why the error occurred and how to correct it. These learning resources could be adapted to the user's current state of knowledge.

9.2.5 Evaluate the Learning Outcome

In the evaluation of this projects users where asked about how they experienced the adaptivity of the learning system. While this information was useful in evaluating the learning system in relation to the aim of this master thesis, it is not concerned with how usage of the learning system affected the learning outcome for the learners. It is recommended that a study is conducted to measure the effectiveness of the adaptive learning system in regards to learning outcome.

Bibliography

Abdo, N., Noureldien, N. A., 2016. Evolution of dynamic user models for adaptive educational hypermedia system (aehs). In: Application of Information and Communication Technologies (AICT), 2016 IEEE 10th International Conference on. IEEE, pp. 1–5.

Bagnasco, A., Galaverna, L., Aleo, G., Grugnetti, A. M., Rosa, F., Sasso, L., 2016. Mathematical calculation skills required for drug administration in undergraduate nursing students to ensure patient safety: A descriptive study: Drug calculation skills in nursing students. Nurse Education in Practice 16 (1), 33 – 39.

Björkstén, K. S., Bergqvist, M., Andersén-Karlsson, E., Benson, L., Ulfvarson, J., Aug 2016. Medication errors as malpractice-a qualitative content analysis of 585 medication errors by nurses in sweden. BMC Health Services Research 16 (1), 431.

Brindley, J., 2017. Undertaking drug calculations for oral medicines and suppositories. Nursing Standard 32 (7), 56–62.

Bruner, J. S., 2006. In Search of Pedagogy Volume I: The Selected Works of Jerome Bruner, 1957-1978. Routledge.

Brusilovsky, P., Jul 1996. Methods and techniques of adaptive hypermedia. User Modeling and User-Adapted Interaction 6 (2), 87–129.

Brusilovsky, P., 2016. Educational applications of adaptive hypermedia. Human-Computer Interaction: Interact'95, 410.

Brusilovsky, P., Millán, E., 2007. User Models for Adaptive Hypermedia and Adaptive Educational Systems. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 3–53.

Brusilovsky, P., Peylo, C., 2003. Adaptive and intelligent web-based educational systems. International Journal of Artificial Intelligence in Education (IJAIED) 13, 159–172.

Calabrese, A. D., Erstad, B. L., Brandl, K., Barletta, J. F., Kane, S. L., Sherman, D. S., 2001. Medication administration errors in adult patients in the icu. Intensive care medicine 27 (10), 1592–1598.

Council, N. M., 2010. Standards for pre-registration nursing education. NMC, London.

Coyne, E., Needham, J., Rands, H., 2013. Enhancing student nurses' medication calculation knowledge; integrating theoretical knowledge into practice. Nurse education today 33 (9), 1014–1019.

Dunlosky, J., Rawson, K. A., Marsh, E. J., Nathan, M. J., Willingham, D. T., 2013. Improving students' learning with effective learning techniques: Promising directions from cognitive and educational psychology. Psychological Science in the Public Interest 14 (1), 4–58.

Edge, D., Fitchett, S., Whitney, M., Landay, J., 2012. Memreflex: adaptive flashcards for mobile microlearning. In: Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services. ACM, pp. 431–440.

Ertmer, P. A., Newby, T. J., 2013. Behaviorism, cognitivism, constructivism: Comparing critical features from an instructional design perspective. Performance Improvement Quarterly 26 (2), 43–71.

Felleskatalogen AS, s.a. Felleskatalogen. `https://www.felleskatalogen.no/medisin/`, accessed: 2018-03-02.

Fleming, S., Brady, A.-M., Malone, A.-M., 2014. An evaluation of the drug calculation skills of registered nurses. Nurse education in practice 14 (1), 55–61.

Flutter, s.a. Build beautiful native apps in record time. `https://flutter.io/`, accessed: 2018-02-18.

Foss, B., Løkken, A., Leland, A., Stordalen, J., Mordt, P., Oftedal, B. F., 2014. Digital game-based learning: a supplement for medication calculation drills in nurse education. E-learning and Digital Media 11 (4), 342–349.

Gardner, A. K., Abdelfattah, K., Wiersch, J., Ahmed, R. A., Willis, R. E., 2015. Embracing errors in simulation-based training: the effect of error training on retention and transfer of central venous catheter skills. Journal of surgical education 72 (6), e158–e162.

Godwin-Jones, R., 2010. Emerging technologies from memory palaces to spacing algorithms: approaches to secondlanguage vocabulary learning. Language, Learning & Technology 14 (2), 4–11.

Gross, R., 2015. Psychology: The science of mind and behaviour 7th edition. Hodder Education.

Hattie, J., Timperley, H., 2007. The power of feedback. Review of educational research 77 (1), 81–112.

Introduction to ASP.NET Core, s.a. `https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-2.1`, accessed: 2018-02-06.

Kang, S. H., 2016. Spaced repetition promotes efficient and effective learning: Policy implications for instruction. Policy Insights from the Behavioral and Brain Sciences 3 (1), 12–19.

Kubeš, B. T., 2007. Application of hypermedia systems in e-learning.

Manno, M. S., 2006. Preventing adverse drug events. Nursing2017 36 (3), 56–61.

Material Design, s.a. Create intuitive and beautiful products with material design. https://material.io/design/, accessed: 2018-02-06.

McMullan, M., Jones, R., Lea, S., 2011. The effect of an interactive e-drug calculations package on nursing students' drug calculation ability and self-efficacy. International journal of medical informatics 80 (6), 421–430.

Mema, B., Harris, I., 2016. The barriers and facilitators to transfer of ultrasound-guided central venous line skills from simulation to practice: exploring perceptions of learners and supervisors. Teaching and learning in medicine 28 (2), 115–124.

Mills, S., 2016. Teaching and learning medication calculations: A grounded theory of conceptual understanding. International journal of nursing education scholarship 13 (1), 35–43.

Mordt, P., Tillerli, K., Løkken, A., Foss, B., 2011. The medication game digital game based medication calculation–development and design. THE GAMEiT HANDBOOK, 105.

Murray, M. C., Pérez, J., 2015. Informing and performing: A study comparing adaptive learning to traditional learning. Informing Science: The International Journal of an Emerging Transdicipline 18, 111.

National Curriculum Regulations for Nursing Programs, 2008. Forskrift til rammeplan for sykepleierutdanning. https://lovdata.no/dokument/SF/forskrift/2008-01-25-128, accessed: 2018-05-15.

Norsk senter for forskningsdata, s.a. Do i have to notify my project? http://www.nsd.uib.no/personvernombud/en/notify/index.html, accessed: 2018-02-09.

Oldridge, G., Gray, K., McDermott, L., Kirkpatrick, C., 2004. Pilot study to determine the ability of health-care professionals to undertake drug dose calculations. Internal Medicine Journal 34 (6), 316–319.

Olsen, L. A., 2014. Praktisk medikamentregning. Cappelen Damm Akademisk.

Özyazıcıoğlu, N., Aydın, A. İ., Sürenler, S., Çinar, H. G., Yılmaz, D., Arkan, B., Tunç, G. Ç., 2018. Evaluation of students' knowledge about paediatric dosage calculations. Nurse education in practice 28, 34–39.

Park, K. Y., Kim, M. S., 2018. Outcomes of a drug dosage calculation training smartphone app on learning achievement, metacognition, and flow state according to prior knowledge. EURASIA Journal of Mathematics, Science and Technology Education 14 (7), 2867–2876.

Piaget, J., 1985. The equilibration of cognitive structures: The central problem of intellectual development. University of Chicago Press.

Pirinen, H., Kauhanen, L., Danielsson-Ojala, R., Lilius, J., Tuominen, I., Díaz Rodríguez, N., Salanterä, S., 2015. Registered nurses' experiences with the medication administration process. Advances in Nursing 2015, 1–10.

Ravik, M., Havnes, A., Bjørk, I. T., 2017a. Conditions affecting the performance of peripheral vein cannulation during hospital placement: A case study. Nursing research and practice 2017, 1–10.

Ravik, M., Havnes, A., Bjørk, I. T., 2017b. Defining and comparing learning actions in two simulation modalities: students training on a latex arm and each other's arms. Journal of clinical nursing 26 (23-24), 4255–4266.

Saville, K., 2011. Strategies for using repetition as a powerful teaching tool. Music Educators Journal 98 (1), 69–75.

Sherriff, K., Burston, S., Wallis, M., 2012. Effectiveness of a computer based medication calculation education and testing programme for nurses. Nurse Education Today 32 (1), 46–51.

Shute, V. J., Torreano, L. A., 2003. Formative Evaluation of an Automated Knowledge Elicitation and Organization Tool. Springer Netherlands, Dordrecht, pp. 149–180.

Simonsen, B. O., Daehlin, G. K., Johansson, I., Farup, P. G., 2014. Differences in medication knowledge and risk of errors between graduating nursing students and working registered nurses: comparative study. BMC health services research 14 (1), 580.

Sneck, S., Saarnio, R., Isola, A., Boigu, R., 2016. Medication competency of nurses according to theoretical and drug calculation online exams: A descriptive correlational study. Nurse education today 36, 195–201.

Stolic, S., 2014. Educational strategies aimed at improving student nurse's medication calculation skills: A review of the research literature. Nurse education in practice 14 (5), 491–503.

Sulosaari, V., Huupponen, R., Hupli, M., Puukka, P., Torniainen, K., Leino-Kilpi, H., 2015. Factors associated with nursing students' medication competence at the beginning and end of their education. BMC medical education 15 (1), 223.

Vygotsky, L. S., 1978. Mind in society. Cambridge, MA: Harvard University Press.

WHO, 2017. Who launches global effort to halve medication-related errors in 5 years. http://www.who.int/en/news-room/detail/\29-03-2017-who-launches-global-effort-to-halve-medication-related-er accessed: 2018-04-09.

Williams, B., Davis, S., 2016. Maths anxiety and medication dosage calculation errors: A scoping review. Nurse Education in Practice 20, 139 – 146.

Wright, K., 2005. An exploration into the most effective way to teach drug calculation skills to nursing students. Nurse Education Today 25 (6), 430–436.

Wright, K., 2007. Student nurses need more than maths to improve their drug calculating skills. Nurse Education Today 27 (4), 278–285.