Gamification To Promote Guideline Training In Health Care

Ben-Richard Sletten Ebbesvik

Master thesis in Software Engineering at

Department of Computing, Mathematics and Physics,
Western Norway University of Applied Sciences

Department of Informatics,
University of Bergen

Supervisors: Yngve Lamo and Svein Ivar Lillehaug

June 2019

Abstract

Clinical practice guidelines are recipes for how clinician can identify a specific medical condition in a patient, how to manage and provide treatment for such a patient. These are evidence based statements, which includes recommendations to optimize patient care. Well defined guidelines has shown the effect of improving the quality of health care at a lower cost, as well as reducing practice variability.

Despite the positive effects of clinical practice guideline, they have shown a limited effect on changing the clinicians practice methods. Some clinician don't know that such guidelines exists, they are not familiar with the guideline content, they lack the self-confidence to execute the recommended treatment, the previous practice methods make it difficult to adapt to the new recommendations, or the guideline format itself is to cumbersome to read and use.

The purpose of this master thesis is to address some of the reasons why the clinical practice guidelines haven't been put more into use. We propose a serious game which will contribute to awareness of the guidelines and training in the guideline content itself. We do present four models. A guideline model, which describes the workflow of the clinical practice guideline. A domain or entity model, which describes the patient, his symptoms, diagnosis and how the clinicians have managed his medical condition. A student learning model which keeps track of the student's performance at different quizzes. A game model which holds information about game elements and the ordering of the game/learning material. By using information from the game model and the student learning model, we can make the game adaptable to the knowledge and progression of the student, as well as flexible such that the student can choose different paths through the learning material.

Acknowledgement

This thesis wouldn't been possible without the effort and contribution of some of the people I have had the pleasure to work with the last 18 months.

My supervisors Yngve Lamo and Svein Ivar Lillehaug which have contributed with valuable help, feedback and ideas in periodic meetings, arranging workshop and introducing me for people which have helped me under way.

Job Nyangena has been my closest collaborator, and have contributed a lot with his medical knowledge, ideas and feedback on my work. Our many discussions have had a great impact on the direction of the project.

Fazzle Rabbi has been of great help with his genuine interest for the project. He has always been available when I've needed help with challenges in model driven engineering.

A big thanks to Rosaline Barendregt which contributed with her knowledge and experience in interaction design and gamification.

Idar Syslak which helped me comparing similar products, as well as coming up with ideas and helping me design alternative prototypes for guideline training.

August Hoel and Fredrik Hoel graduated as medical doctors and started a career in research during my masters degree project. They spent many hours helping me discussing the role of clinical practice guidelines in medical training for students, and how the guidelines are used in the daily practices of health workers. They also spent hours evaluating and discussing the final version of the serious game for guideline training.

Thomas Berge for using his time to evaluate the final version of the serious game for guideline training. He introduces me to some of his colleagues at polyclinic for pulmonary diseases at Haukeland University, which also contributed to the evaluation.

A great thanks to Nikolai Grieg and Gard Engen for moral support, their contribution to a great daily work environment and subject-related input and discussions.

To Malin, my parents, my brother and his family for their patience, support and understanding during this period. I wish I had more time to spend with you all.

To Malin
and my parents

Contents

1 Background **1**
- 1.1 Clinical Practice Guidelines . . . 1
  - 1.1.1 Discussion . . . 3
- 1.2 Serious games . . . 3
- 1.3 Motivation . . . 4
  - 1.3.1 Asthma . . . 5
- 1.4 Related work . . . 5
- 1.5 Summary . . . 5

2 Introduction **6**
- 2.1 Paper publication . . . 6
- 2.2 Research questions . . . 6
- 2.3 Structure of the thesis . . . 7
- 2.4 Summary . . . 8

3 Method **9**
- 3.1 Design study . . . 9
- 3.2 Iterations . . . 11
  - 3.2.1 Testing mobile app technology, conceptual model . . . 11
  - 3.2.2 Focus group . . . 11
  - 3.2.3 Studying similar products . . . 14
  - 3.2.4 Technology used to represent CPGs . . . 14
  - 3.2.5 Designing alternatives . . . 18
  - 3.2.6 Entity and workflow models . . . 20
  - 3.2.7 User experience . . . 23
  - 3.2.8 Workshop . . . 25
  - 3.2.9 Game Engine and Content Flow . . . 31
- 3.3 Summary . . . 32

4 Architecture Overview **34**
4.1 Architecture of the whole system 34
4.1.1 Presentation Layer 34
4.1.2 Game Engine 35
4.2 Summary 36

5 Datamodels **37**
5.1 Extracting knowledge from the clinical practice guidelines . . . 38
5.2 MDE and meta modelling 38
5.3 DPF 38
5.4 Entity model 39
5.4.1 Generic entity model 45
5.5 Workflow model 46
5.6 Metamodeling 47
5.7 Game model 48
5.8 Student learning model 50
5.9 Summary 50

6 Game Elements **52**
6.1 Question Flow Manager 52
6.1.1 Dynamic Content Management 53
6.1.2 Types of questions 56
6.1.3 Unlocking harder levels at a certain category 57
6.2 Conversation Manager 58
6.2.1 Constructing scenarios and answer keys 58
6.2.2 Multiple-try feedback 62
6.2.3 Reward system 65
6.3 User manager 66
6.3.1 Visualization of game statistics 66
6.4 Conceptual model of how all the classes are connected in Game Engine 67
6.5 Summary 68

7 Application Walkthrough **70**
7.1 The mobile application 70
7.1.1 User interface and flow of the user interaction 70
7.2 Summary 75

8 Evaluation **78**
8.1 Research questions 78
8.2 Evaluations 78

8.2.1 Evaluation of the models 78
8.2.2 Evaluation of the application 83
8.2.3 Results 88
8.3 Findings 89
8.3.1 Limitations of the entity model 91
8.4 Summary 92

9 Discussions and conclusion 93
9.1 Future work 93
9.1.1 Automatically generating new questions 93
9.1.2 Dissemination of guidelines 95
9.1.3 Scalability 95
9.1.4 Multimedia as part of the quiz questions 95
9.2 Conclusion 96

Appendices 104

Comparison of CPG applications 104

A Model Driven Approach to the Development of Gamified Interactive Clinical Practice Guidelines 112

List of Figures

3.1 The project followed a similar process to the life cycle of interaction design. Figure derived from Preece, Sharp, and Rogers (2015) 10
3.2 A very simple conceptual model to test one of the proposed technologies, as well as acting as a starting point for discussions 12
3.3 The paediatric possible asthma guideline (Republic of Kenya 2016) modelled using elements from GLIF 16
3.4 An attempt at modelling the paediatric possible asthma guideline (Republic of Kenya 2016), using Asbru editor tool AsbruView (The Aasgard Project 2006) 17
3.5 A design alternative, displaying the CPGs as interactive flowcharts. The edges represents the flow, while clicking vertices will display more information or redirect to subguideline. A suggestion was to fill in with patient and examination data, to get a customized CPG for a patient 19
3.6 The first design alternative proposed as a game. The student tries to pick the right tests in the right order for that patient. As well as choosing the correct treatment and give the right advises to the patient. A high score for choosing the right treatment and advises, as well as correct tests in the right order. Medium score for right tests in the wrong order 21
3.7 Screenshots of the prototype used to simulate a clinical encounter with a domain expert 22
3.8 Pictures of the whiteboard during a meeting with domain expert in MDE and DPF. To the left there are some discussion around workflow models. To the right is entity models with vertices which holds a textual representation of the parent vertex 23
3.9 User experience copied from Hassenzahl and Tractinsky (2006) 24

3.10 The participants on the workshop. The master student is absent as he is the photographer . . . 26
3.11 The student needs to acquire basic knowledge about assessment, diagnosis, management, before he can follow-up the treatment. This is a prerequisite for follow-up . . . 28
3.12 Assessment and diagnosis are components in the treatment plan. In the learning map they are themes. Under each theme there are difficulty levels. Questions for each level are written on post-it notes . . . 29
3.13 Management and follow-up are components in the treatment plan. In the learning map they are themes. Under each theme there are difficulty levels. Questions for each level are written on post-it notes . . . 29
3.14 What type of questions the student will get at each level . . . 30
3.15 Suggestions for questions were written on post-it notes and attached to a difficulty level under a theme on the black board 31

4.1 Overview of the system architecture . . . 35

5.1 Metamodeling from a meta-metamodel to a real world object. Derived from (Brambilla, Cabot, and Wimmer 2017) . . . 39
5.2 An excerpt of the entity graph. Entity graph represents a patient at a certain point in the clinical encounter . . . 40
5.3 Showing the details of the Patient and Diagnosis vertices of the entity graph . . . 41
5.4 Showing the implementation of History in the entity graph. What the patient or caregiver tell about the patient's condition 42
5.5 Showing the implementation of Examination in the entity graph. What symptoms the clinician can observe the patient has . . . 42
5.6 Showing the implementation of Medication in the entity graph. How to administer a medication to patient . . . 44
5.7 To the left we don't know which medication has been inhaled using the nebulizer or inhaler. To the right we have specified this in the instantiation, by using two Inhaled-vertices and uniquely identifying them by giving them different names . . . 45
5.8 A generic entity graph contains abstractions. An entity graph for a specific guideline, contains specifications for those abstractions. Yellow boxes indicate specifications we have shown for paediatric guideline of possible asthma (Republic of Kenya 2016) . . . 46
5.9 The workflow models is a model of the clinical encounter . . . 47

5.10 A model and an instance of the entity model. For a valid instance, every vertex and edge in the instance has a corresponding vertex and edge in the model 48
5.11 An instance of the workflow model at the bottom, working together with an instance of the entity model at the top . . . 49

6.1 Knowledge map 54
6.2 Learning map 55
6.3 By looking at the workflow- and entity models we can identify the type of questions for each content unit in the knowledge map. The content units are the parent nodes, while the leaves are the type of questions 56
6.4 The patient object used with the template 59
6.5 Making graph variables fit a story format 61
6.6 Making graph variables fit a story format 61
6.7 The entity- workflow- and game models working together to produce a scenario, taking the patient through the steps of a clinical encounter. Note that these are two entity graphs E1 and E2, where E2 gets updated with an additional red vertex Ipratopium bromide. This illustrates a patient which doesn't respond to the initial treatment, and then gets treated with ipratopium bromide 63
6.8 Conceptual model for the game elements 68
6.9 Façade pattern hides the complexity of the game engine and makes it easier for other parts of the system to use it 68

7.1 The entry screen where the student can choose between playing different quiz categories 71
7.2 The student is shown his position in the learning map. Completed, active and locked levels have all different colours 72
7.3 The first question is an assessment question. Here answered correctly and an answer key explanation is given 73
7.4 The second question is for diagnosis 73
7.5 Third question is for management 74
7.6 The fourth question is follow up. We answer incorrectly the first time. The answer key explanation is not shown, and we can choose to revise the question by clicking "try again". If we press "learn more", the answer key explanation is instead shown, and we continue to the next question 75

7.7 After completing the quiz, we are shown the student's updated position in the learning map. The brown background shows levels which has been completed. Grey is the levels which are still unlocked. Green is that we have advanced to the next level. A red box would have indicated that we had performed poorly, the level would have been locked and we have to complete an easier level before we can play it again 76
7.8 The bars show our scores for the played game in the different parts of the clinical encounter. The red line shows the scores needed to complete this level 77

8.1 Only an inheritance on the Diagnosis vertex makes the excerpt of pneumonia guideline (Republic of Kenya 2016) different from the original excerpt in figure 5.2 79
8.2 The history part of the entity model of paediatric pneumonia guideline (Republic of Kenya 2016) is identical to figure 5.4 . . 80
8.3 The symptoms need to be adapted to the paediatric pneumonia guideline (Republic of Kenya 2016) when modelling the examination 80
8.4 When modelling the Diagnosis of the paediatric pneumonia guideline (Republic of Kenya 2016), we need to use inheritance to support differential diagnoses and different treatments when the patient has combinations of diagnoses 81
8.5 The paediatric pneumonia guideline (Republic of Kenya 2016) uses oxygen and antibiotic treatment, which can be modelled in the same way as figure 5.6 82
8.6 We are showing out entity model working together with the workflow model for paediatric pneumonia guideline (Republic of Kenya 2016) 83

9.1 An entity instance of the wheezing sound 96

List of Tables

3.1 As the student progress to higher difficulty levels, the questions will ask for more details. Here we show the detail levels of Management 27

6.1 The content unit Assessment 54
6.2 Student Map T1:Assessment 55
6.3 The type of questions at each difficulty level 57
6.4 Overview over different quiz concepts 64

8.1 89

Chapter 1

Background

1.1 Clinical Practice Guidelines

Fervers, Carretier, and Bataillard (2010) claims that for clinicians, increased medical knowledge is associated with an exponential growth of scientific data and published material. It is impossible to keep up, as well as integrating all the new information into daily practice to give patients the best possible care. Masic, Miokovic, and Muhamedagic (2008) gives an example where a general practitioner should read 19 articles per day to keep up with the new medical information, while only having time for reading one hour per week. Reading 19 articles per day, would acquire more time than the clinician has available for treating patients. This problem is known as academic isolation (Masic, Miokovic, and Muhamedagic 2008).

Evidence Based Medicine (EBM) suggests that instead of routinely reading dozens of articles, the clinicians should target their reading to specific patient problems. Developing clinical questions and then searching for the answer (problem based approach) may be a more productive way to keep up with the new medical knowledge (Masic, Miokovic, and Muhamedagic 2008). The EBM definition further puts an emphasize on integrating the best evidence in decision making with the clinicians expertise and the patients values and expectations (Masic, Miokovic, and Muhamedagic 2008).

The concept of EBM is about transferring knowledge from clinical research into clinical practice, and Clinical Practice Guidelines (CPG) can play an instrumental role in this process (Fervers, Carretier, and Bataillard 2010).

The Institute of Medicine (IOM) has given the following definition of clinical practice guidelines: "CPGs are statements that include recommendations intended to optimize patient care. These statements are informed by a systematic review of evidence and an assessment of the benefits and costs of alternative care options" (Institute of Medicine (US) et al. 2011)

The definition given by IOM covers the goals in EBM, and also takes the cost into account. In fact, Clayton and Hripcsak (1995) have shown that in some situations good use of appropriate guidelines and protocols can reduce as much as 25% of the cost of healthcare.

Even though the CPGs have proven to improve the quality of health care while reducing practice variability and the cost of patient care (De Clercq, Kaiser, and Hasman 2008), it is well recognized that CPGs have had a limited effect on changing the clinicians practice methods. Cabana et al. (1999) lists the following reasons:

- **Lack of awareness:** the clinician is not aware of the guideline's existence.
- **Lack of familiarity:** the clinician is not familiar with the content of the guideline.
- **Lack of agreement:** the clinician had various reasons to disagree with the guideline, such as they are oversimplified, disagree with the evidence or not worth the patient risk, discomfort or cost.
- **Lack of self-efficacy:** is the lack of self-confidence in that the clinician can execute the recommendations of the guideline correctly.
- **Lack of outcome expectancy:** the clinician doesn't believe the outcome of the recommended treatment will meet the outcome expectancy.
- **Inertia of previous practice:** the custom, habit or previous training can hinder the adaptation of clinical practice.
- **External Barriers:** the guidelines are not easy to use, not convenient, cumbersome and confusing.

One example of external barrier is the Guidelines for the Diagnosis and Management of Asthma (National Heart Lung and Blood Institute and U.S. Department of Health & Human Services 2007), which consists of 440 pages. Such a large document is not convenient to use at the point of care. According to Shortliffe (1998), CPGs in monographs and journal articles tend to sit on book shelves at the time their knowledge could prove the most valuable to the clinicians.

1.1.1 Discussion

According to Woolf et al. (1999), clinicians sometimes have good reasons to disagree with some of the content of a guideline. Woolf et al. (1999) points out three reasons:

1. The scientific evidence of the recommendation can be lacking, misleading or misinterpreted.

2. The recommendations may be influenced by the authors. What the authors believe, may be inferior to other options, ineffective or harmful.

3. As the guideline may be written to control cost, serve societal needs or protect special interest, the recommendations may be suboptimal for the patient.

There exists grading systems which grade the quality of evidence and strength of recommendations. GRADE is such a grading system (Guyatt et al. 2008). When displaying guidelines to clinicians, it is a strong point to display the grade of evidence, as the clinician has to choose between several treatment options.

1.2 Serious games

When searching the literature for the definition of serious games, there seem to be many different understandings of what serious games really is. However, these definitions seem to have the common understanding that serious games are games which are used for other purposes than just pure entertainment (Susi, Johannesson, and Backlund 2015). This is actually a very broad category, where we can find games which are used to test job applicants or to improve our health by encouraging us stay more active.

Michael and Chen (2006) defines serious games as "a serious game in which education (in its various forms) is the primary goal, rather than entertainment". Michael and Chen (2006) emphasizes that education and entertainment should not be in conflict, but that they can overlap. The feeling of learning something new or getting better at something, can be quite satisfying and can serve as an entertainment factor.

Serious games also have the advantage over educational books and movies that the student can demonstrate and apply what he has learnt, through tasks in the game (Michael and Chen 2006). Serious games seem more effective than training with conventional instruction methods, as the knowledge gains persists in the long term memory, and the learner can build on this well-structured prior knowledge through his learning career (Wouters et al. 2013). However, serious games seems to be most effective when they are supplemented with instructional learning methods. Not only gets the student to learn by doing, but he also gets the opportunity to reflect over what he has learnt and to verbalize the new knowledge, making it easier to integrate it into his knowledge base (Wouters et al. 2013).

1.3 Motivation

By making a serious game for clinical practice guideline training, we can address some of the reasons why the CPGs haven't had a greater impact on clinicians practice methods (Cabana et al. 1999):

- **Lack of awareness:** The more projects around CPGs, the more focus will they get and more people will be aware of their existence. By making a serious game, we may be able to target some user groups which where hard to reach in traditional ways.

- **Lack of familiarity:** By playing the game, the student will learn more about the content and will become familiar with the CPGs. The student may also be encouraged to study the CPGs in the traditional ways after having played the game.

- **Lack of self-efficacy:** By repeatedly solving practical tasks in the game, the student may become confident in that they are capable of executing the treatment recommended by the CPG.

- **External barriers:** convenient, cumbersome and confusing CPGs will by approach be converted to a game format. Even though a game isn't a good encyclopaedia at the point of care, for some user groups a game might be a better format for studying. Especially a combination of instructional learning methods and serious games have shown positive learning results (Wouters et al. 2013). Having built well-structured prior knowledge may also help at the point of care.

Another motivational reason for making a serious game is the scalability. How can we best train 10, 100 or 1000 clinicians in the best practices of medical guidelines? There are logistics problems with instructional courses and training sessions, such as cost of money, time and there's a practical limit for how many attendees can attend a course at the same time. A mobile game scales much better as downloading an mobile application is much cheaper, can be played almost anywhere at any time. There's no limitation on how many playing participants.

1.3.1 Asthma

Asthma is a repository disease, which in the recent years have had an almost exponential growth rate among children in Oslo. From 0.4% in the first Norwegian report, to 8% in 1993 and 20.2% in 2006. Similar results were found in the rest of Norway in the early 90s. (Carlsen et al. 2006).

Asthma growth amongst children is not only an issue in Norway. According to Odhiambo et al. (1998), 3% of children in rural areas in Kenya had asthma in 1998 and 9.5% of the children in urban areas. Before this study it was a claim that asthma among African children was rare, which is no longer true (Odhiambo et al. 1998).

It is urgent to find answers to prevent further increase of asthma amongst children in the years to come(Carlsen et al. 2006).

As our contribution to put focus on the dramatically growth of asthma in children, our work will centre around the paediatric possible asthma guideline (Republic of Kenya 2016) when developing a serious game to promote guideline training in health care.

1.4 Related work

1.5 Summary

Clinical practice guidelines are evidence based statements, which includes recommendations to optimize patient care. Well defined guidelines improve the quality of health care at a lower cost, as well as reduce practice variability.

However, clinical practice guidelines have had an limited effect in changing clinician practice methods. In this thesis we will develop a serious game to address some of the reasons why the clinical practice guidelines haven't been put into more use.

Our work will be based around paediatric asthma as a contribution to put a focus on the the dramatically increase of asthma in children.

Chapter 2

Introduction

2.1 Paper publication

In December 2018, we submitted a paper "A Model Driven Approach to the Development of Gamified Interactive Clinical Practice Guidelines", which was a summary of our work so far in this project and related projects. The paper was accepted for publication by ENASE 2019 – 14th International Conference on Evaluation of Novel Approaches to Software Engineering, where we held a presentation on their conference in Heraklion, Greece.

The paper can be found in appendix 9.2 in this thesis.

2.2 Research questions

- **RQ1:** Based on clinical guidelines, how can we define and represent a generic data structure that can be used to implement applications such as online guidelines or training games for such guidelines, and where applications can adapt to the level of their users?
- **RQ2:** Can the generic data structure in RQ1 be used to generate a specific data model for another domain such as paediatric asthma?
- **RQ3:** How can we use the data model in RQ2 to implement a game for guideline training that can adapt to the level and progression of users?
- **RQ4:** Is the guideline meta model at an abstraction level such that it can be used for other guidelines?
- **RQ5:**

2.3 Structure of the thesis

1. **Background:** in this chapter we give an explanation of what the clinical practice guidelines are and the purpose they serve. We propose a serious game which will address some of the reasons why clinical practice guidelines haven't been put more into use.

2. **Introduction:** we presents some research questions which are related to making abstract and specific data structures of the guideline content, as well as representing the guideline content at the knowledge level of the student, adapt it during the student's progression and make it flexible such that the student can pick his own path through the learning material.

3. **Method:** in this chapter we describe our adoption to design science and go through the seven guidelines. The concern that we must deliver value to the medical community, as well as a contribution to the science of health informatics. We describe our work as iterations with specific goals and evaluations.

4. textArchitecture: we describe the overall architecture of the game. We have a presentation layer, where we discuss all the technologies in that layer. We have a separate game engine, which uses four managers and three conceptual managers. The managers describes the responsibilities of the game engine. This is a brief introduction, and more detail about the game engine will be discussed in the following chapters.

5. **Data models:** We give a short explanation of model driven engineering and DPF.We proposed four data models, an entity model, a workflow model, a game model and student learning model. We will cover the models in detail in this chapter.

6. **Game elements:** We will discuss the conceptual question flow manager, conversation manager and user manager. We will use Dynamic Content Management to make the learning material adaptable to the knowledge level and progression of the student, as well as flexigle such that the student can choose his own way through the learning content. We have to expand the entity model with presentation vertices, such that we can produce textual questions. We discuss the multiple-try feedback and other quiz game approaches, as well as a rewarding system. We show a conceptual model of the game engine.

7. **Application walktrough:** we play through level 2, presents screenshots and discuss features, ideas, user interface and design choices.

8. **Evaluation:** we evaluate the data models by modelling the paediatric pneumonia guideline (Republic of Kenya 2016). We evaluate the game itself with two nurses and two medical doctors. We discuss our findings in relation to the research questions.

9. **Discussion and conclusion:** we briefly discuss topics which may be done in relation to this project as future work. We present some related work and then concludes this thesis.

2.4 Summary

Here we have defined a set of research questions, which is related to the development of serious games for clinical guidelines. Making games which are adaptable to the knowledge level, progression of the user, and making guideline models which are at an abstraction level where they can be used to represent other CPGs, are the main focus points.

We have also given a short presentation of each chapter in the thesis.

Chapter 3

Method

3.1 Design study

The research methodology we used for this project was an adaptation of design science (Hevner et al. 2004). In short, design science requires an artefact that either solves a an organizational problem. The artefact itself and the construction of it goes through evaluations in iterations, to make sure its relevance to the problem domain. We must ensure that the research contributes value to both the health care domain as well as computer science.

Hevner et al. (2004) proposed some guidelines design science in information system research. We will go through them and connect them to our research project.

- **Guideline 1: Design as an Artefact.** The artefact will be a serious mobile game with the associated data models.
- **Guideline 2: Problem Relevance.** Clinical Practice Guidelines are improving the quality of health care, reduce practice variability and the cost of health care (De Clercq, Kaiser, and Hasman 2008). But they have had a limited effect on changing the clinicians practice methods (Cabana et al. 1999). Some of the reasons Cabana et al. (1999) lists are that some clinicians are not aware of the guidelines, they are not familiar with the guideline content, as well as the guidelines are difficult to use, cumbersome, inconvenient and confusing. These are issues we can address with a game.
- **Guideline 3: Design Evaluation.** The design process is described in the next section, where each iteration has clear goals and evaluations. Cognitive walkthroughs and usability tests with users are some of the

evaluation methods used. In the end of the thesis, we will demonstrate that the datamodels can be used to represent other clinical practice guidelines, as well as evaluating the game with medical experts.

- **Guideline 4: Research Contributions** are our data models, which are the backbone of the game. The data models should be generic enough to represent and make games for other clinical practice guidelines. Adaptable to the user's knowledge level and with flexible learning paths.

- **Guideline 5: Research Rigor** and **Guideline 6: Design as a Search Process.** The construction of the artefact followed a similar process described in figure 3.1. It's an iteratively process where requirements were established through focus group, workshop, studying similar products, talking with domain experts and informal literature studies. Then designing alternatives, which also can be functionality suggested to the artefact. Prototyping was making prototypes or constructing the artefact itself. An evaluation decides whether the new functionality/prototype can be used as it is or update the requirements and make a new version.

  Problems are uncovered and solved in the evaluation - construction iterative phases. Here the research questions appear from the design.

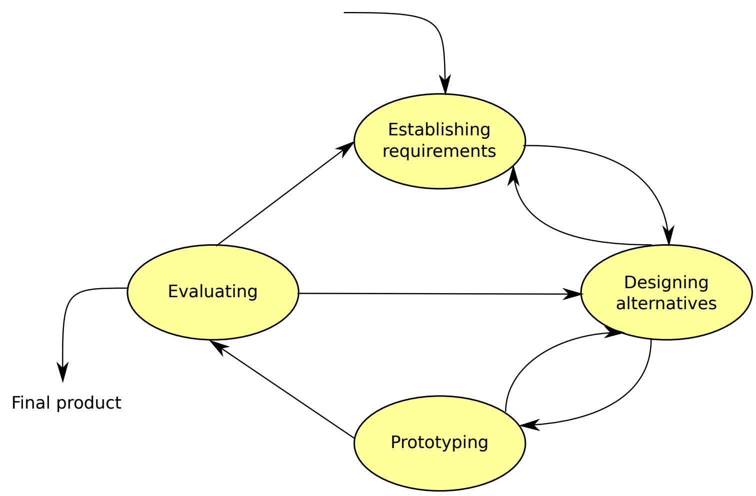

Figure 3.1: The project followed a similar process to the life cycle of interaction design. Figure derived from Preece, Sharp, and Rogers (2015)

- **Guideline 7: Communication of Research,** which is the purpose of this masters thesis. We have also published a paper, which can be found in the appendices.

3.2 Iterations

3.2.1 Testing mobile app technology, conceptual model

The goal of this iteration was to make a prototype using the proposed technology. The prototype will tell us if it is sensible to use the technology for the project, as well as having a conceptual model to use when discussing initial ideas around the aim of the project, functionality, as well as interaction design.

The technology to be tested was React Native. The reason for this to be the first choice, was to support multiple platforms and not having to rewrite the application for every platform. The same application can be used for Android and iPhone with little modification to the code. React Native is build upon the React front end web development framework, which means some code can be reused for web as well. But all the the views need to be modified, as they use specific React Native components for mobile units. React and React Native are JavaScript frameworks.

We have also used the React web development framework in previous projects. It s one of the more popular and mature front end web development frameworks in the market.

The prototype itself and the development process showed that the technology could be used for displaying and organize CPGs. Lessons learned about content flow, databases, app navigation, displaying information and dialogues. JavaScript in the view gives a lot of flexibility compared to just tags.

The interface was evaluated with the supervisors, using cognitive walkthroughs. It also worked as a conceptual model, using the prototype as a base for discussing ideas. Both for discussing the purpose of the project, what the user should be able to do, how to organize content and what functionality to add.

3.2.2 Focus group

The 25th of February 2018, we arranged a focus group at a kitchen in Alrek student home. The participants were:

- Two 6th year medical doctor students.
- Two master degree students in computer science.

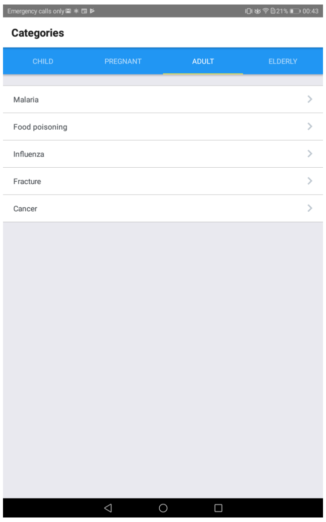

Figure 3.2: A very simple conceptual model to test one of the proposed technologies, as well as acting as a starting point for discussions

As the project was in a very early stage, the purpose of the meeting was exploring how do medical students use CPGs in their training. How the CPGs are used today. How the students work in their practice periods at the hospital. What format are the CPGs in now. What challenges limits the use of CPGs among health workers and medical students at the point of care.

By using a an unstructured interview form, we managed to collect broad and general information, as well as going into detail on interesting topics. As the focus group was small, the contestants could discuss between themselves, highlighting consensus and conflicts (Preece, Sharp, and Rogers 2015). A very free and exploitative approach was very successful, as the master degree students in computer science had a very limited knowledge of the medical domain, compared to the medical students. The discussion was documented using an audio recorder.

Topics explored:

- The students are learning new medical routines by studying typical cases. Drilling the routines.

- Red and yellow flags, which are alarm systems they need to be aware of. A red flag is when the patient's condition is quite critical. What triggers the flags and how to act upon them is something the clinicians need to know by heart, as time is critical and the action needed might be advanced like surgery. No time to use the guidelines. Orange, yellow and green are further degrees of how urgent the clinician's intervention is. Orange is less urgent than red, but more urgent than yellow. Green is the least urgent one.

- Clinicians and medical students use a collection of short guidelines in a pocket book format for references. The guidelines are written mostly in text format and sometimes takes use of tables for presentation.

- Mobile devices can't be used at the point of care because of condemnation risks. There are also techniques of consultant with a patient. A mobile device will get in the way for important non-verbal communication.

- For departments they need access to specific cases where there isn't much written material. They need access to scientific articles.

- They brought up the case where doctors in developing countries have so many patients, and the time to each patient is very limited. To be able to look up information in the guidelines, they need to have a format which makes it very fast to extract that kind of information. Flowcharts is more suitable for developing countries where acting quickly is more often important than in developed countries where they can have a focus on being more thorough in green flag situations.

- In addition to guidelines, hospitals can also have their own protocols. The protocols are for situations where the treatment is more specific for this area or hospital. An example is a patient with a blood clot. The medical personnel in Finnmark will start removing the blood clot immediately, because of the long distance to the nearest hospital. They want to reduce the risk for complications. While in Bergen, they will wait with such a treatment. In developing countries you might have to put into consideration what kind of equipment and staff is available at that specific hospital or health care station. The likelihood of different diseases is different from each geographically position, the patients background or the season in the year. Social and economic status also matters, even in Norway.

- Discussion about presentation of learning material in an application. Medical cases are often too obvious, too simple or too complicated in existing applications. Flashcards. Show image of an ECG or a picture of a symptom.

- Notifying the student about guidelines would be useful. But only for the most common and dangerous conditions, relevant for the students medical field to avoid unnecessary notifications which will only be ignored.

- The diagnostic process. The doctor have several conditions in mind, but tries to eliminate the statistically most common and dangerous first. Trying to narrow down the alternatives until the doctor is quite certain about the medical condition of the patient.

3.2.3 Studying similar products

The purpose of the iteration is to see what exists in the market. What have others done. What can we improve, where can we add value to both the medical community and to computer science.

The results from this iteration is presented in appendix 9.2, and was done together with a fellow master of science student in software engineering. As a conclusion, none of these application have a data model which can represent CPGs, nor a patient in a clinical encounter. The representations of CPGs are mostly text based, flow charts or flow charts which expand when you click on decision vertices. LIFE: Neonatal Resuscitation Training is a pretty advanced 3D game, but there is no data model representing the content in the quizzes and tasks.

3.2.4 Technology used to represent CPGs

In this iteration we evaluated technologies which could be used to model CPGs.

- **GLIF** and **PROforma** was compared during a literature study as a semester assignment in health informatics. We will here give a brief summary of GLIF.

  GLIF or the Guideline Interchange Format, was developed by the InterMed Collaboratory. The intention was to make a guideline representation language, which can be viewed with a various of software application as well as adapting them and making them valid for different local uses. The representation should be precise, ambiguous, readable by humans and interpretable by computers, adaptable to different clinical information standards and facilitating guideline sharing (Peleg et al. 2000).

  To make the guidelines readable for humans, interpretable by computers and adaptable by different institutions, GLIF defines the guidelines at three different abstraction levels: conceptual, computable and implementable levels (De Clercq, Kaiser, and Hasman 2008)

The conceptual level is the highest abstraction level. It consists of flow-charts which can be viewed by humans, using guideline viewing programs. At this level the guidelines can not be used for computations in decision support (Peleg et al. 2000).

At the computable level expressions, patient data elements, clinical actions and guideline flow are specified at this level. The guidelines can also be verified for logical consistency and completeness (Peleg et al. 2000).

The implementable level contains the information to incorporate the guidelines into the particular institutions knowledge or information system such as EPR (Peleg et al. 2000).

In figure 3.3 we present a model of the paediatric possible asthma guideline (Republic of Kenya 2016) at the conceptual level. We will now explain the guideline steps the UML-model in figure 3.3 consists of.

- **Action step** a representation of a recommended tasks or action. There are three types of actions steps: Medically oriented (such as a recommendation for a treatment), programming oriented (such as retrieving information from EHR), control-oriented that invokes nested control structures (subguidelines or macros to support recursive specification) (De Clercq, Kaiser, and Hasman 2008). Action steps are green squares in figure 3.3.
- **Decision step** represents a decision point in the model. There are two types of decision steps: Case and Choice. In a Case step the decision will be made up of a number of logical expressions, a deterministic decision. On the other hand, the Choice step displays a various suggestions and the agent (clinician e.g.) needs to choose between them (De Clercq, Kaiser, and Hasman 2008). Decision steps are turquoise hexagons in figure 3.3.
- **Patient state step** which characterizes the specific state or condition a patient is in. It can be used as an entry point into the flow-chart, or as an summation to describe the clinical state a patient is in (Boxwala et al. 2004). Each patient state step includes attributes which describes the condition a patient is in (e.g. iron level in the blood has been reduced since last blood sample). When this condition occurs in practice, the guideline corresponding to

the Patient state step is executed (De Clercq, Kaiser, and Hasman 2008). Decision steps are yellow diamonds in figure 3.3.

– **Synchronization step** working together with the **branch step** to provide multiple concurrency paths (Boxwala et al. 2004). Multiple guidelines that follows a branch step, eventually comes together in a synchronization step. A continuation attribute specifies whether all or some conditions need to be fulfilled before we can continue to the next step (De Clercq, Kaiser, and Hasman 2008). Synchronization and branch steps are not used in figure 3.3.

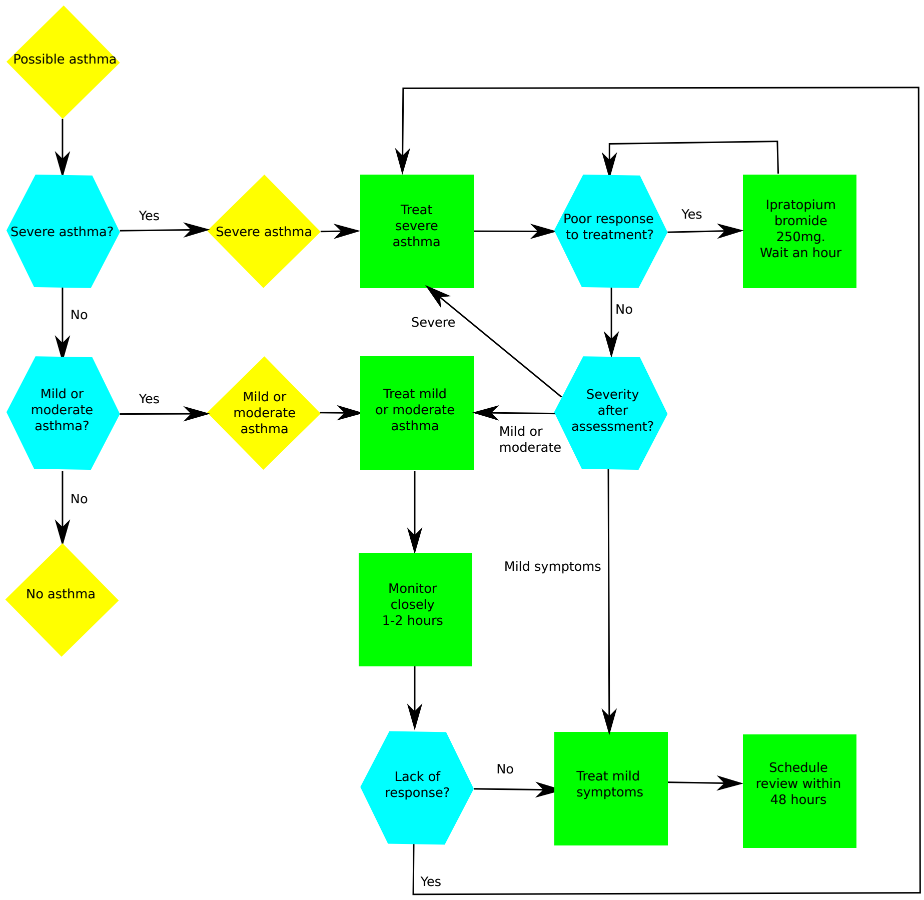

Figure 3.3: The paediatric possible asthma guideline (Republic of Kenya 2016) modelled using elements from GLIF

- **Asbru** is another approach for computer interpretable guidelines, and sort of a competitor to GLIF and PROforma. The project was developed at the Stanford University, Vienna University og Technology and Ben-Gurion University. It uses time-oriented skeletal plans. These skeletal plans are schemata at different detail levels. To manage skeletal plans, Asbru introduces some key functionality: the representation of high-level goals (intentions), temporal patterns, time annotations, as well as graphical user interfaces to view the skeletal plans (De Clercq, Kaiser, and Hasman 2008).

In Asbru, skeletal plans are used to represent guidelines. Each plan's functionality is described by attributes such as: preferences, intentions, conditions, effects and plan body (De Clercq, Kaiser, and Hasman 2008).

In figure 3.4, we have modelled the paediatric possible asthma guideline (Republic of Kenya 2016) using a graphical interface tool AsbruView (The Aasgard Project 2006). The sequential plans are organized from a start point to an end poing along a time x-axis. Plans can be executed in a parallel, so the y-axis is lanes. Subplans are stacked in z-axis, making the presentation of a guideline 3 dimensionally. In addition to execute plans in sequential or parallel order, we also have the option to execute the in an arbitrary order or without any fixed order (unordered). There are also sequential plans and control structures with if-then-else.

For the skeletal plans shown in figure 3.4, we can also set the attributes conditions, intentions, effects and preferences for each of them. These describes the functionality of each plan (De Clercq, Kaiser, and Hasman 2008).

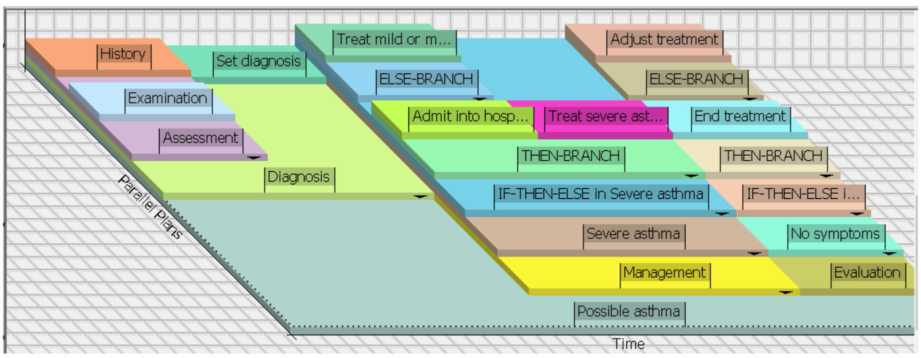

Figure 3.4: An attempt at modelling the paediatric possible asthma guideline (Republic of Kenya 2016), using Asbru editor tool AsbruView (The Aasgard Project 2006)

- See the description of **DPF** in other parts of this thesis.

Studying PROforma, GLIF and Asbru we learn that there are several ways to approach the task of making computer interpretable guidelines. We can learn about challenges and see how they solved them. Dissemination, where one needs to regularly update the guidelines and make sure that the different computer systems use the updated versions, is such a topic. Dissemination is relevant for this project, as you want to make sure that every student trains for the most current guideline. You don't want them to memorize old and outdated content.

Our conclusion is that Asbru, GLIF and PROforma are rather large systems, putting a lot of emphasise on working in a hospital setting, communicating with other computer systems such as decision support on the electronic health record. With large computer systems, it is much more difficult to customize.

They also work on a higher abstraction level than the symptoms the clinician needs to look for when doing an examination for asthma. When training clinicians, we are very interested in modeling the details a clinician has to do during a clinical encounter. Asbru, GLIF and PROforma are more concerned with customizing the workflow for one or a sequence of treatments for a medical conditions. We want to use more general models which can be reused to represent several guidelines. With DPF we can create custom domain specific modelling languages.

3.2.5 Designing alternatives

The goal of the iteration is to decide what kind of application to build. What type of application can we best promote the CPGs, and encourage health personnel to use and learn them.

For this iteration it was quite critical to get background information on CPGs, how they are used in the hospital, how they are used by medical students today and how they work in different areas. For this we needed the results from the iterations about the focus group and studying similar products. We also had a medical domain expert in our project. Informal literature studies was also necessary.

A lot of different prototypes at lower and higher fidelities were made. Some were made initially at higher fidelity levels as a part of testing technology. Figure 3.2 was such a prototype to test technology, as well as exploring how to organize and categorize CPGs. Figure 3.7 was a prototype of a simulation tool, but also had the purpose of uncovering details of the paediatric possible asthma guideline (Republic of Kenya 2016) when communicating with a medical domain expert.

Figure 3.5 was also a prototype initially at higher fidelity as we tested SVG generation with JavaScript. The idea of the prototype was to display CPGs as interactive flow-chart. The vertices have a topic which can be clicked. By clicking, the vertex will either expand to show more detailed information or will redirect to a subguideline. The edges represent decisions and workflow. An idea was also to enter patient specific details such as gender, age and weight and possibly symptoms, and get the CPG customized for that patient.

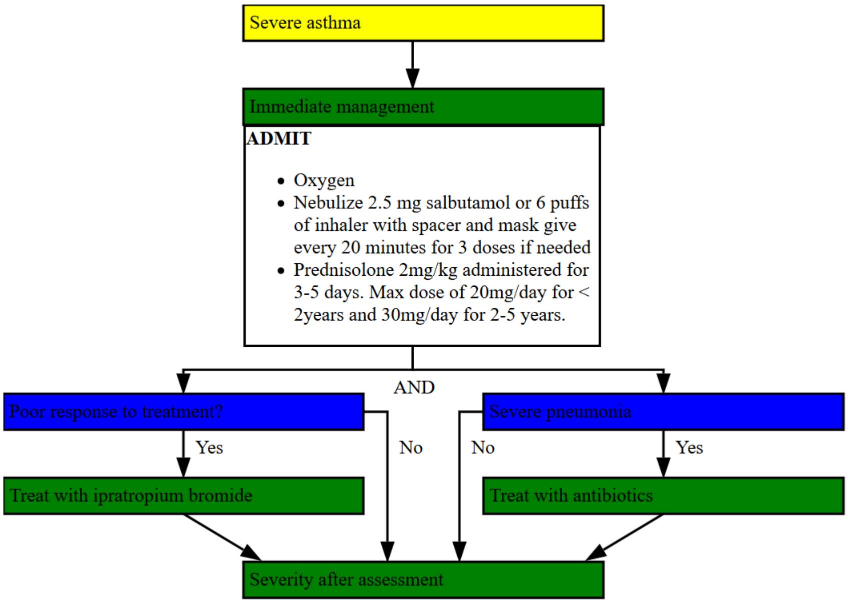

Figure 3.5: A design alternative, displaying the CPGs as interactive flow-charts. The edges represents the flow, while clicking vertices will display more information or redirect to subguideline. A suggestion was to fill in with patient and examination data, to get a customized CPG for a patient

The prototype in figure 3.6, started as sketches with pen and paper. By going through several iterations of redesign, it reach the fidelity of figure 3.6. This was our first prototype where we presented the CPGs as a game. We got help from an external masters degree student in software engineering to make this prototype. The concept of the game is that you initially get presented with a list of tests you can do on the patient. By picking a test, you will immediately get presented with a test result. You can do more tests or proceed. In the next screen you will be presented with a list of treatment and advises, and you should pick the correct ones with the knowledge you acquired in the previous screen. In the last two screens, you will be presented with the answer keys. You will get points for choosing the correct treatments and advises. You will get points for picking the right tests, and you will get higher scores if you picked the tests in an optimal order.

The design alternatives were evaluated with the project's supervisors and their master degree students. There was also separate evaluations with medical domain expert. The evaluation method was cognitive walkthroughs as well as discussions what is the best approach for promoting CPGs and encourage medical students to learn and use CPGs. The conclusion was to continue with game development.

3.2.6 Entity and workflow models

The goal of the iteration was to model patients at different stages of the clinical encounter. The entity model should be able to represent patients in scenarios in quizzes with answer keys.

We based our models on the paediatric possible asthma guideline (Republic of Kenya 2016). The first step was to understand every symptom, medication, equipment used in the treatment and keywords such as "admit". Resources like Disease et al. (2011) and Johansen et al. (2018) were some of the resources used, but most importantly was an expert of domain.

The developer made a prototype of his understanding of the CPG, which simulated a clinical encounter with a patient. See figure 3.7. The GLIF chart presented in figure 3.3 was used as a base when making the simulation tool. The simulation would start with listing symptoms of asthma and the clinician would choose among the symptoms, redirecting to the treatment for severe, mild, moderate and no asthma. The simulation would continue with cycles of evaluating the given treatment, and the clinician needs to act accordingly to the evaluation for each cycle until the treatment can be ended. An expert of domain got the task of going through the entire simulations of use cases such as "a patient with severe asthma responds poorly to salbutamol treatment". The stimulation gave a thorough understanding of the guideline, emphasized details which would have been missed or details of the treatment which is left out of the guideline itself, such as differential diagnoses.

The guideline is also unclear at some points, where a discussion with a domain expert was needed. For example: wheeze + history of cough or difficulty breathing. When parentheses are not used, it is difficult to see if wheeze always needs to be present, or if it is enough that cough alone is. It is also difficult to understand the effect and model the situation where the presence of a symptom at a certain age gives "increased likelihood" for asthma. In computer science the terms need to be very clear.

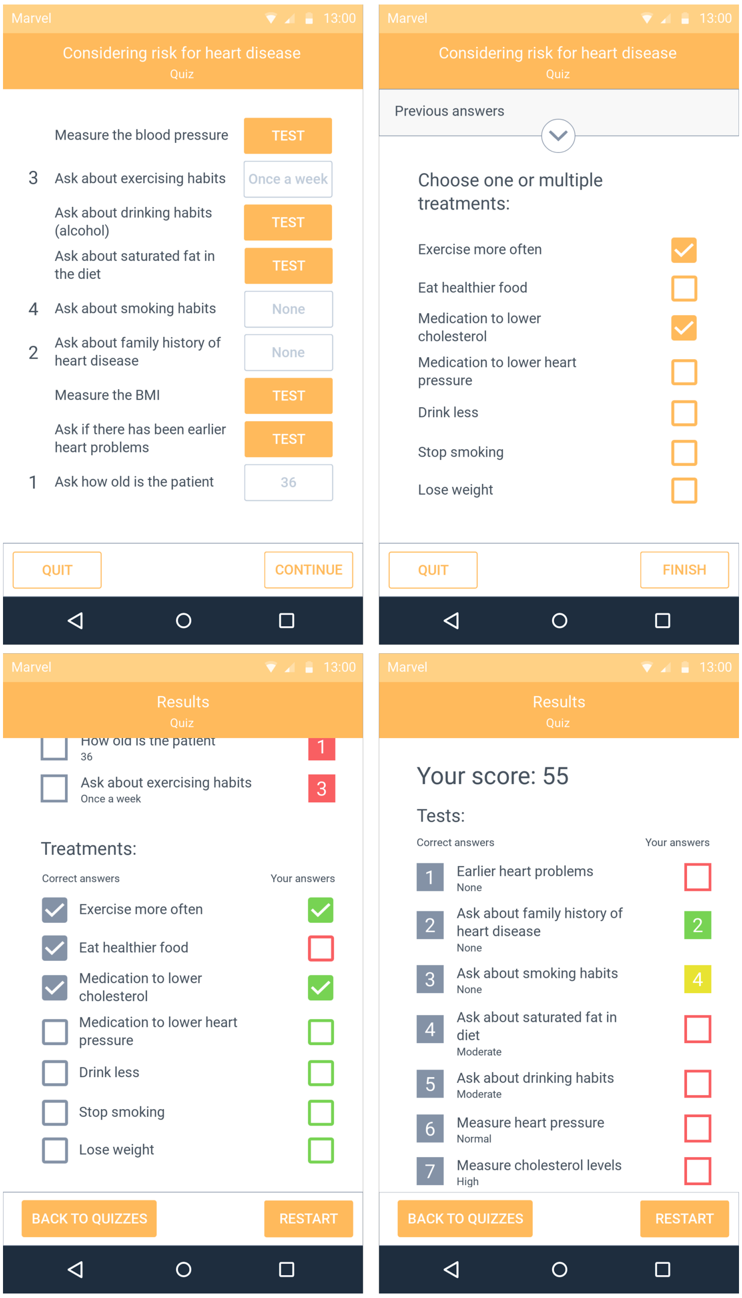

Figure 3.6: The first design alternative proposed as a game. The student tries to pick the right tests in the right order for that patient. As well as choosing the correct treatment and give the right advises to the patient. A high score for choosing the right treatment and advises, as well as correct tests in the right order. Medium score for right tests in the wrong order

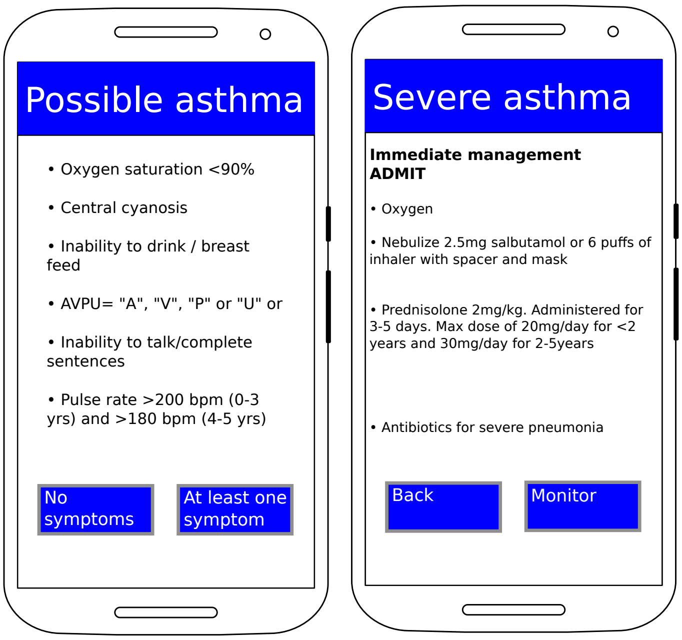

Figure 3.7: Screenshots of the prototype used to simulate a clinical encounter with a domain expert

A domain expert in Model Driven Engineer and DPF was consulted for discussing problems such as inheritance, how to create good sentences from vertex values in the graph, as well as metamodelling and the use of constraints in DPF. In figure 3.8 there are two images from the whiteboard during the meeting. On the left we see a workflow and an instance below. The XOR is a constraint, which tells that we can choose one of the two paths only. On te right we see the entity model, and a suggestion with vertices which make good textual presentation of graph values which can be used in texts.

For evaluating the entity model, a domain expert in medicine made quizzes with factual questions as well as scenario based. By replacing patient related variables in the questions and scenarios, with tags pointing to variables in the entity model, we could see whether the model fulfilled the requirements of displaying good and valid sentences. The sentences were equally good and valid when using different instances of the entity model.

The workflow model got evaluated by making quiz scenarios, and by confirming that these scenarios covered the entire guideline.

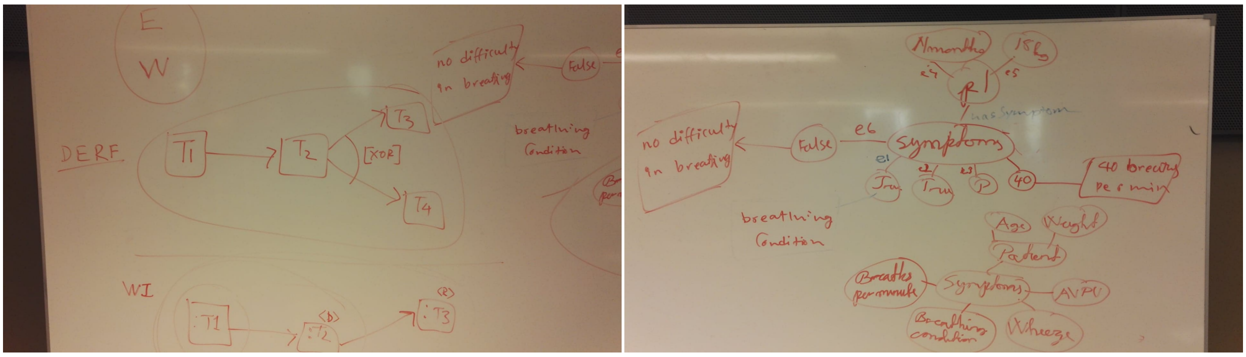

Figure 3.8: Pictures of the whiteboard during a meeting with domain expert in MDE and DPF. To the left there are some discussion around workflow models. To the right is entity models with vertices which holds a textual representation of the parent vertex

3.2.7 User experience

The goal of the iteration is to facilitate the application for good user experiences.

First of all we need to clarify what a good user experience is. We are not going to dive in detail into figure 3.9, but shortly explain the components he Hassenzahl and Tractinsky (2006) defines.

- Beyond instrumental, has to do with other aspects than reaching a work related goal or solving a work related task. It has to do with attraction and feelings, such as attraction to a beautiful design, the feeling of trust, quality and reliability or the user feels motivated through personal growth and increase of knowledge and skills (Hassenzahl and Tractinsky 2006).

- Emotion and effect, has to do with how the users feel when interacting with the application. Typically we want the user to feel joy and excitement, and damper or prevent feelings such as frustration, anger and disappointment (Hassenzahl and Tractinsky 2006). Typically the user feels satisfaction when completed a work task, or frustrated when the work task takes a long time to complete.

- The experiential emphasizes that the experience is strongly related to the situation it is used in and at which time. (Hassenzahl and Tractinsky 2006). As an example the user experience of listening to your favourite music artist in the car on the way to work, is different from being at a concert in the weekend with the same music artist.

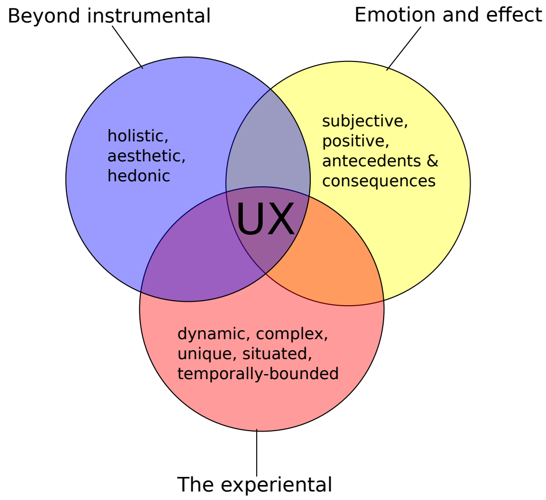

Figure 3.9: User experience copied from Hassenzahl and Tractinsky (2006)

For the beyond instrumental, we used several iterations to make a pretty user interface to make the application desirable. As this is an educational tool, it is important that the user feels his skills and knowledge grow, so the visualization of progress was quite important. Both that the user advances in levels, but also that we visualize the progress during a level, where the user stretches to slowly meet the requirement of the next level.

For emotion and effect, we should trigger joyful emotions, such as emphasizing when the user answers correctly and progresses. We should use kind and encouraging wording when the user makes an attempt and doesn't get the right answer or doesn't progress to the next level. The interactions should be clear such that the user easily can fulfil his goals and doesn't make mistakes. Keep system errors to a minimal. In our evaluations we could probably more encouraging to the user when he gives a wrong answer.

The experiential use is a bit hard to adapt to. We have been thinking that this is a mobile app, often played when the user is bored and can be situated anywhere. Perhaps on the bus to the university, or waiting for his friends in the cafeteria. The buttons should be large, large font size and small amounts of text on the screen, clear instructions and that the user doesn't have to remember information from previous screens are good guidelines in an environment which can be distracting. This is also according to Jakob Nielsen and Rolf Molich heuristics about user design. "Minimize the User's Memory Load" and "Simple and Natural Dialogue" (Molich and Nielsen 1990).

The user experience was evaluated using cognitive walkthroughs with the supervisors and their master degree students. We had a cognitive walkthrough with a domain expert in interaction design and gamification, followed by discussion and tips for further improvement. Some of the feedbacks were "wrong" in red colour may be too harsh, as well as a penalty of negative points even though the penalty is not displayed. A user should be able to get a higher score than just the requirement for the next level.

We used usability tests in controlled environment, where friends, family, fellow students and domain expert in medicine were asked to start the application and play a full quiz. Notes were taken at points where the user would do mistakes, get confused or do a mistake. Typically the user would become very confused when seeing the screen with the learning map, displaying which levels have been completed, which we are currently playing and the locked levels. The user would typically try to click on completed or current levels. We also noticed that the students wouldn't revise a wrong answer, just continue to the next question.

3.2.8 Workshop

The 22nd of February 2019 we had a workshop. The purpose of the workshop was to

- Identify components in the treatment plan of asthma patients.
- Identify difficulty levels, and how the questions will be more detailed for every difficulty level.
- Make a map of the learning content. Where the content is categorized in components and difficulty levels. Identify paths the student can take through the learning content.

The antecedences for the meeting was

- Professor in computer science. Background in model driven engineering and health informatics.
- Assistant professor in computer science. Background as a researcher in health informatics.
- Postdoctoral fellow. Background in model driven engineering.
- Medical doctor and PhD student in health informatics.
- PhD research fellow in interaction design. Has written a master thesis in gamification.

- PhD candidate in computer science and health informatics.
- Master degree student in computer science.

Figure 3.10: The participants on the workshop. The master student is absent as he is the photographer

The meeting started with the master student informing the status of the project by doing a cognitive walk-through and a demonstration of the application.

The professor presented ideas for further development of the application. The important thing, was the concept of splitting up the questions in themes which relates to components in the treatment plan. The medical doctor helped identify these themes as assessment, diagnosis, management and treatment. Further we identified what type of questions we wanted to ask, and how they fits into different difficulty levels, based on the details of the questions. We ask factual questions for level 1. We use scenarios in level 2, where we apply facts and the detail level is categories. I.e. what class of medication should be administered to the patient. In level 3 we continue with scenarios, but here we ask for much more details, like the dosage of a medication or how often it should be administered. See figure 3.1 to for an example of how the detail level gets higher as the student progress.

When playing level 1 the student should get questions from all themes in level 1. When the student completes level 1, the student should no longer get questions from that theme. This is to avoid boring the user by repeating the questions the student already knows the answer of. He should only get questions from themes he struggles with. I.e. on the first run level 1, the

Level	Question	Answer key	Distractions	Explanation
1	Which of the following class of drugs is NOT indicated in the immediate management of severe asthma?	Inhaled corticosteroids	Short acting beta agonists, Oral corticosteroids, Anticholinergics	Inhaled corticosteroids are not indicated for acute management of asthma, they're indicated for long term control of symptoms
2	Karen is found to be experiencing severe asthma and is given salbutamol. What other medications should she be given?	Prednisolone and oxygen	Ipratropium bromide, Formoterol	When treating a patient with severe asthma, always start with oxygen, salbutamol and prednisolone. The evaluation of how the patient responds, will determine the next step of the treatment
3	You diagnos Malin with severe asthma and initiate a treatment plan. What dose of salbutamol will you administer with the nebulizer?	2.5mg	3.5mg, 4.5mg, 5.5mg	In the scenario of severe asthma, nebulize 2.5mg of salbutamol

Table 3.1: As the student progress to higher difficulty levels, the questions will ask for more details. Here we show the detail levels of Management

user gets every question in assessment right, but have some mistakes with diagnosis and management. Then on the next run, he only gets questions from level 1 diagnosis and management. This continues until he has reached the passing condition on every theme in level 1.

We further identified a dependency in the treatment plan. To be able to do a follow-up, the students first needs to know something about assessment, diagnosis and management of the patient. The follow-up is actually an evaluation of the treatment which have been given, based on the suggested diagnosis. The evaluation will tell how the patient responded to the treatment, and the student needs to take actions whether the patient responded or his condition became better or worse. When we have such a dependency, the student needs to complete assessment, diagnosis and management before follow-up gets unlocked. Since follow-up is only relevant in a situation where there has already been set a diagnosis and given a treatment, the follow-up is only part of level 2 and 3, where the questions are given as scenarios.

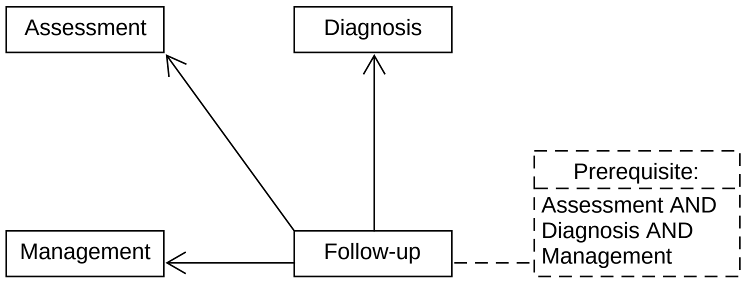

Figure 3.11: The student needs to acquire basic knowledge about assessment, diagnosis, management, before he can follow-up the treatment. This is a prerequisite for follow-up

To complete a level, all passing conditions at that level in each theme have to be met. When the student qualifies for a new level, he only gets questions from the level he plays. The same concept of only getting questions from the themes at that level you haven't met the passing condition, continues for level 2 and 3.

We planned to have a visualization of the passing condition in the application. The passing condition will be shown in a chart in the summary section after each game. The passing condition will be marked as a line over every theme for the level the student plays. The students scores for each theme at that level be shown as bars. When a bar reaches the line, a passing condition is met.

The medical doctor continued the meeting by talking about the guidelines. The paediatric guideline of asthma is called "possible asthma". That

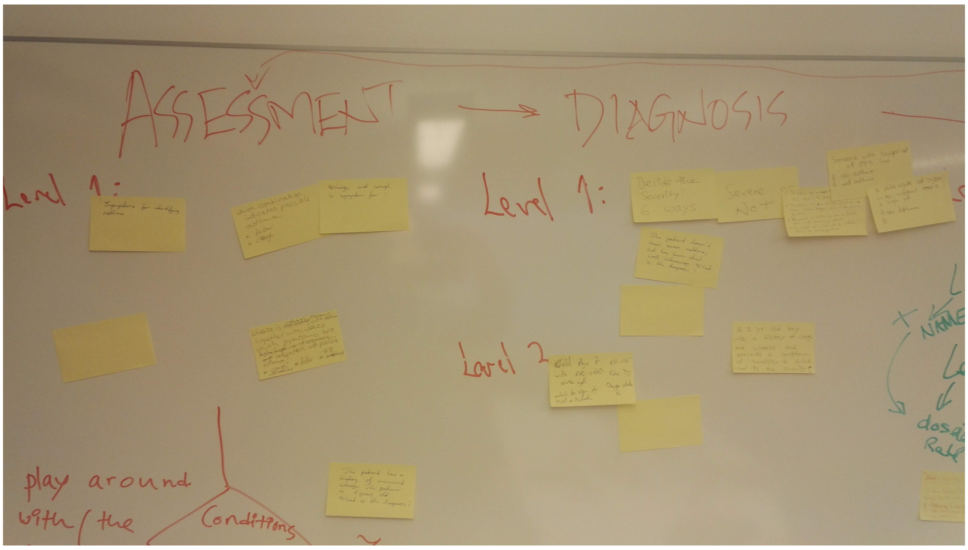

Figure 3.12: Assessment and diagnosis are components in the treatment plan. In the learning map they are themes. Under each theme there are difficulty levels. Questions for each level are written on post-it notes

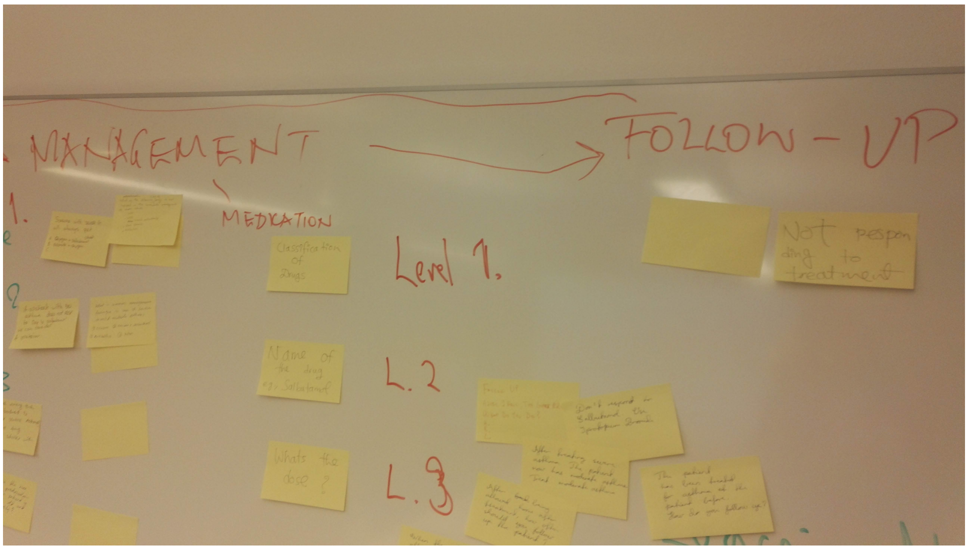

Figure 3.13: Management and follow-up are components in the treatment plan. In the learning map they are themes. Under each theme there are difficulty levels. Questions for each level are written on post-it notes

is because in an emergency situation asthma is the most dangerous airway condition and can be lethal. If the patient shows signs of asthma, he will be treated for asthma to reduce risks of an unwanted scenario.

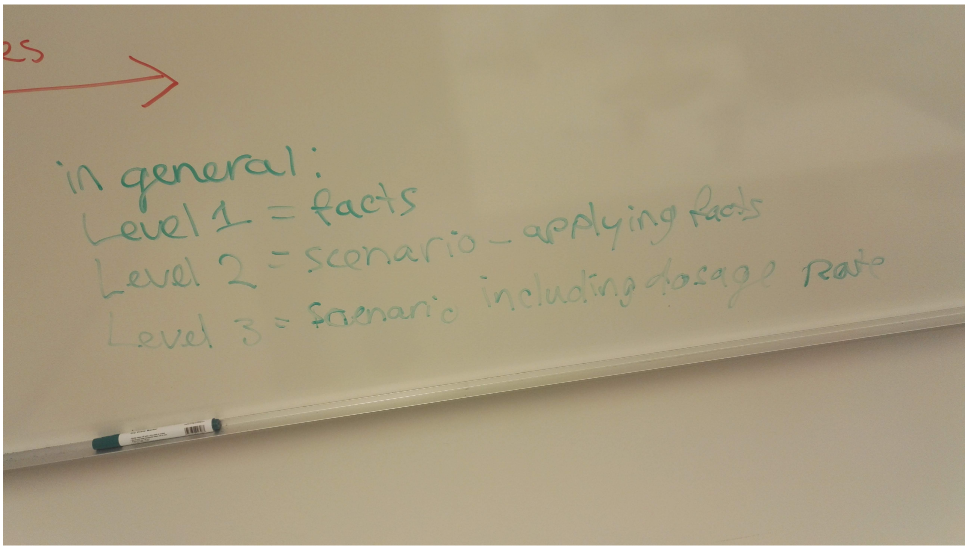

Figure 3.14: What type of questions the student will get at each level

We identified users of the application:

- Formal training, where last year medical students are reading for their exams.
- Anyone can learn, so it can be used to inform and educate the public.
- In countries such as Kenya, where there are a large deficit in doctors and nurses, sometimes nurses has to work as doctors. Or community workers need to take the role as doctor or nurse. The application will help educating nurses and community workers for such scenarios.

There was also talked about how the situation in medical training is for the student. When a patient comes to the emergency room with severe asthma, the medical doctors will have all their focus on that particular patient. The medical student will typically not take part in the assessment, diagnosis or the initial treatment of the patient. The medical student will typically only take part in the monitoring, evaluation and follow-up of the patient, when the situation is less critical. The application will give the medical student an alternative way to train in assessment, diagnosis and initial treatment of a made-up patient with severe asthma.

The medical doctor continued the workshop by going through the Kenyan paediatric guideline of possible asthma (Republic of Kenya 2016). This is the guideline we will base our quiz on. The medical doctor answered questions from the group about details of the guideline. It is important to understand the general flow as well as the details to be able to make good questions for the quiz. The guidelines is poorly written in terms of wrong use of sentential operators. These mistakes needed to be clarified.

The rest of the workshop was for the participants brainstorming around the questions which will be used in the quiz. Each participant wrote questions on post-it notes, and placed them at a suiting level and theme on the blackboard. At the end, The medical doctor went through the questions and we had a small discussion around the suggested questions. We managed to produce question templates to be used in asthma quiz of the application.

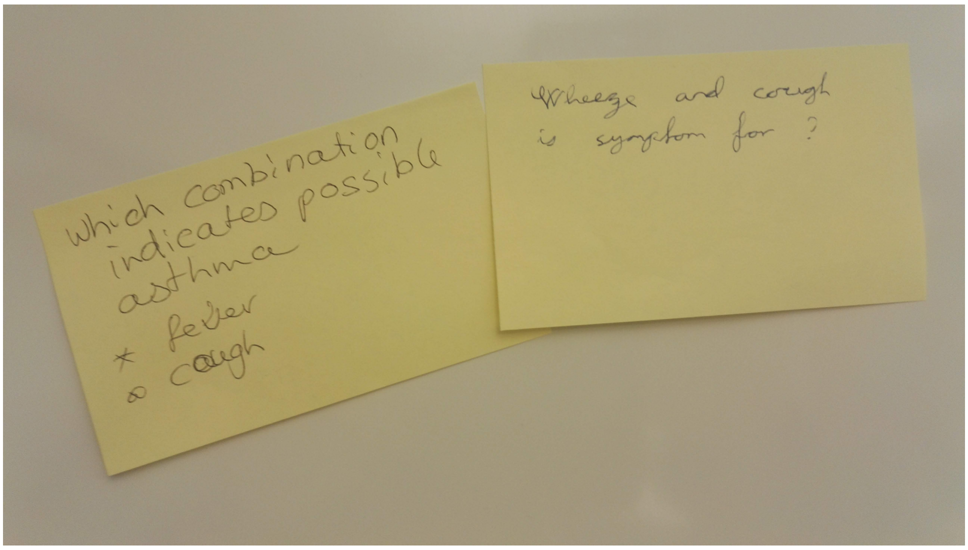

Figure 3.15: Suggestions for questions were written on post-it notes and attached to a difficulty level under a theme on the black board

3.2.9 Game Engine and Content Flow

The goal of the iteration is to have a playable game, which produces questions, answering elements, rewards, several difficulty levels and the flow of the game content.

This iteration draws upon the work which have been produced in other iterations. Producing the content quiz questions was done together with a domain expert in medicine. As part of the Game Engine, we developed a template system, where we replace variables in the question content with tags which refers to vertices in the entity graphs. We can evaluate to see that the game engine produces sensible questions and answer elements.

The ideas of difficulty levels and the flow of the game content were discussed in the workshop section. This iteration was about implementing and testing these parts of the Game Engine.

The scoringsystem was developed, where we give scores per question element in the workflow model. A penalty system was introduced, such that a student with many attempts on each question doesn't get the same total score as a student which answered every question correctly on the first try.

The evaluations of the iteration were done as part of the user experience iteration. Mainly cognitive walkthroughs with domain experts, but also usability testing in controlled environment with non-experts.

3.3 Summary

We have proposed an adaption of design science, where we have a short discussion of the seven guidelines proposed by Hevner et al. (2004). An important part of design science is that we ensures we produce value for the medical community, as well as a scientific contribution to health informatics.

We describe iterations with clear goals and evaluations for our work.

- We have established requirements by doing a focus group of last year medical students, studying similar products such as mobile applications, GLIF, PROforma and Asbru, doing informal literature studies and making conceptual models to discuss with domain experts and team members.
- We have designed prototypes at different fidelity levels, to find our approach to how we can contribute to more use of clinical practice guidelines in health care.
- We discuss the process of how we developed the entity and workflow models, which represents the patient at a given point in the clinical encounter, which symptoms and diagnosis he may have and the treatment given. The workflow model is a model of the workflow of the clinical encounter itself.
- An iteration for optimizing the user experience of the application.
- A workshop with different domain experts where we
    1. Identified components in the treatment plan of asthma patients.
    2. Identified difficulty levels, and how the questions would be more detailed for every difficulty level.

3. Made a map of the learning content. Where the content was categorized in components and difficulty levels. Identified paths the student can take through the learning content.

- Testing and evaluating a fully playable game. Demonstrating the game engine, scoring system and flow of the content.

Chapter 4

Architecture Overview

4.1 Architecture of the whole system

The architecture of the system is shown in figure 4.1. The system in it's whole is running on the student's phone. The main components are the presentation layer and the game engine, where the purpose has been to make the game engine very generic and loosely coupled from the presentation layer. By replacing the presentation layer, the Game Engine can be implemented in a web application or using Google Assistant which uses voice to interact.

We'll briefly dicuss the the architecture of the system in the following subsections, before we'll give a more thorough expalantion.

4.1.1 Presentation Layer

The presentation layer is what the user sees and interacts with when using the application. React Native (*React Native · A framework for building native apps using React*) is a JavaScript framework, used to build cross platform mobile applications for Android, iPhone and UWP. It is based on React (*React – A JavaScript library for building user interfaces*), where it uses React components to build user interfaces for mobile applications.

For managing the state of the application, we use another JavaScript framework, Redux (*Redux · A Predictable State Container for JS Apps*). It is sort of a repository of functions and variables. When the student clicks on a button in the application, it will trigger a function in the Redux repository. The function can send a request to the Game Engine or do some calculations in its own. Then update a variable in the repository which is connected to a variable in the React Component, and the result is shown on the student's phone.

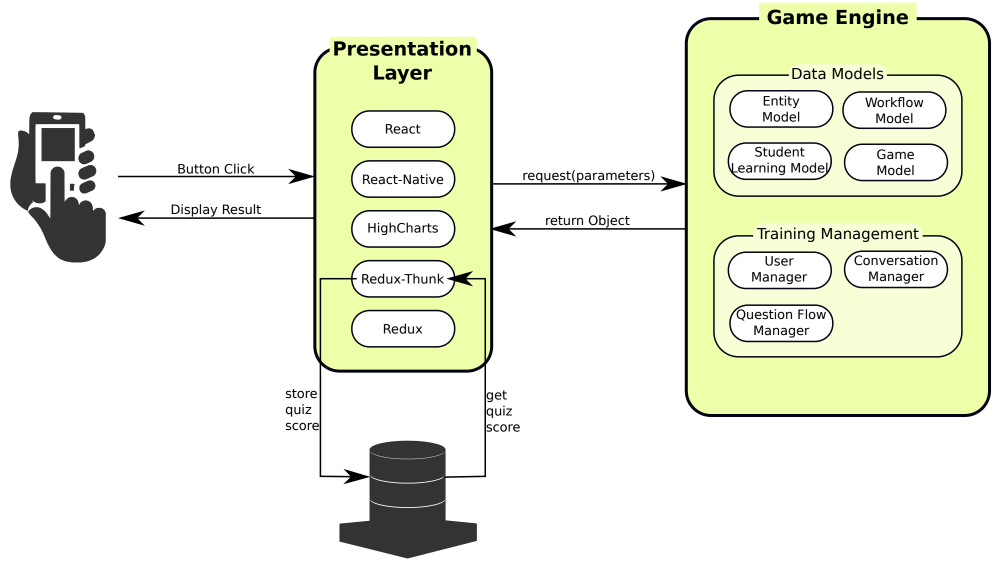

Figure 4.1: Overview of the system architecture

As the Redux repository is synchronous, we need the framework Redux-Thunk (*reduxjs/redux-thunk: Thunk middleware for Redux*) to make asynchronous calls. A student's scores for a quiz is stored in the database on the student's phone. The game engine uses the scores to find questions at the right difficulty level for the student. As database calls are asynchronous, we need Redux-Thunk to make functions which can do asynchronous communication between Redux and the database.

HighCharts (*Interactive JavaScript charts for your webpage — Highcharts*) is a JavaScript framework used for making interactive charts. We use it to visualizing how well the user performed, and how far he is from advancing to more difficult questions.

4.1.2 Game Engine

The game engine initializes the quiz, controls the training flow, produces questions with answering elements and keeps track of the user's performance.

The data models are used to hold information and represent a domain of concepts. The entity model represents a patient at different stages under a clinical encounter. The workflow model describes different processes at a clinical encounter. The game model holds relevant game information such as questions at different categories and difficulty levels, distractions, points and penalties for each distraction, answer keys and answer key explanations. The student learning model keeps track of the student's scores at different quizzes.

The managers are conceptual, used to better describe the responsibilities of the game engine in this section.

The question flow manager adapts the questions to the knowledge level of the student, and is flexible such that the student can choose his own path through the completion of each difficulty level.

The conversation manager puts the questions with belonging answer elements in a sequence such that they fit to a scenario, using the workflow model. The conversation manager produces questions and answer keys from a template, in which the entity model is used to fill in data for the place holders in the template.

The user manager keeps track of the user's skill, based on the performance on previous quizzes. The scores on the current quiz, and it measures the user's progression.

4.2 Summary

We have proposed an architecture of the systems. The mobile application logic is in a separate loosely coupled presentation layer, sending requests to a platform independent game engine.

For the game engine, we have separated the concerns. We have four data models and three conceptual training managers, and given a brief description on all of them. The game models and the conceptual training managers will be thoroughly discussed in the two following chapters.

Chapter 5

Datamodels

For this project and according to the research questions, we will be using four models.

1. An **entity model**, which will be a model of the domain concepts. The patient, the different methods for discovering and measuring symptoms. The patient's medical condition and the procedures done to manage the patient's medical condition.

2. A **workflow model**, which describes the different processes in a clinical encounter. This will be the flow of the guideline itself at an abstract level.

3. A **game model** which will contain the game elements, as well as the order in which the student will learn parts of the guideline. By game elements we mean the questions, distractions, answer keys, answer key explanations, rewards and penalties for the different answer alternatives, as well ass passing conditions for the different difficulty levels.

4. A **student's learning model** for monitoring the student's learning progression.

Models are good for sketches, where we can easily discuss with team members, domain experts and possibly stakeholders. The models can hold a detailed specification of the system, as well as we can use models instead of code to develop the system (Brambilla, Cabot, and Wimmer 2017).

For this project communication is very important, as the developer needs to understand and be able to model the clinical practice guideline and the clinical encounter at a very high detail. The domain expert in medicine needs to be able to verify that the model is correct.

5.1 Extracting knowledge from the clinical practice guidelines

5.2 MDE and meta modelling

We will be using a Model Driven Engineering (MDE) approach, mainly for the entity and workflow models. In MDE models serves as the key artefacts that drive the development process (Rodrigues da Silva 2015). A model in this case is an abstraction of a system under study. It is usually a partly and simplified version of the system, where we often need multiple models to better represent and understand it (Rodrigues da Silva 2015). A domain model is the conceptual model of a field or expertise that we are examining to solve a problem (Brambilla, Cabot, and Wimmer 2017).

A metamodel is a model where one define the modelling concepts that will be used to represent the system in.

A metamodel is a further abstraction of a model, which highlights properties of the model (Brambilla, Cabot, and Wimmer 2017). In figure 5.1 we see that moving up through the levels is an abstraction of the previous model. While moving downward, we are making instantiations of the previous metamodels. With metamodels we can define new languages, as well as defining new properties or features of the available information (Brambilla, Cabot, and Wimmer 2017). This allows us to make Domain Specific Languages (DSL), which is designes specifically for a specific domain and context (Brambilla, Cabot, and Wimmer 2017)(Rodrigues da Silva 2015). By using DSLs, we can develop more expressive models, which can enhance communication with or be used by domain experts.

5.3 DPF

For the enitity- and workflow models, we use a formal diagrammatic approach to model driven engineering, which is called Diagram Predicate Framework (PDF). In software engineering, diagrams are data structures based on graphs. A graph is a collection of vertices, which may be connected by edges.

A definition for DPF is given by Rutle (2010): "in DPF, models are represented by a diagrammatic specification $\mathfrak{S} = (\mathcal{S}, C^{\mathfrak{S}} : \Sigma)$. It consists of a graph $\mathcal{S}$ and set of constraints $C^{\mathfrak{S}}$ specified by a signature constraint Σ ". To simplify the definition to how it is used in this thesis, the models and initiations are represented and implemented using graphs and a set of constraints. The following sections will discuss these graphs in detail.

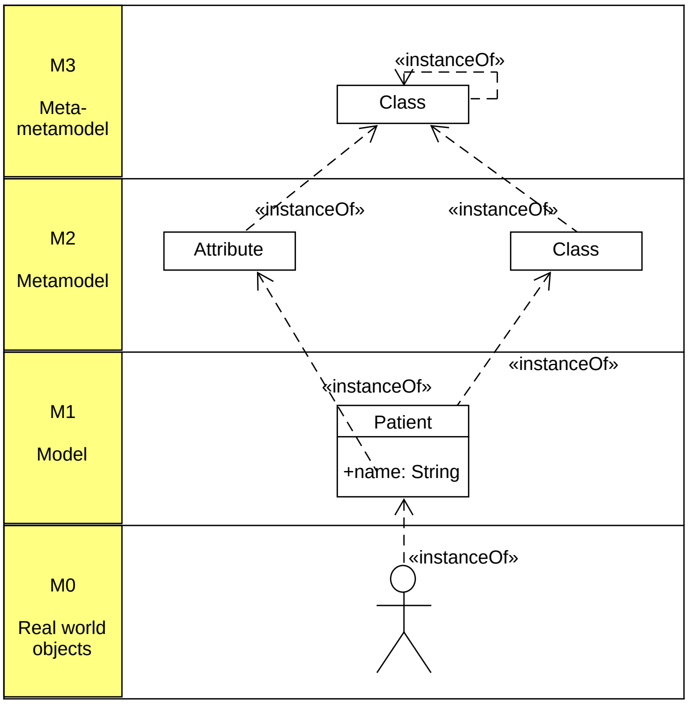

Figure 5.1: Metamodeling from a meta-metamodel to a real world object. Derived from (Brambilla, Cabot, and Wimmer 2017)

Why do we use DPF and not a system such as fHIR? DPF is an abstract modelling formalism that could be used to model different aspects of the system. HL7 fHIR is a domain specific language to represent (and share) resources (i.e. domain concepts) in health care.

5.4 Entity model

We will now present the entity model by showing an excerpt, hiding away the details. This will let us easier focus on the concepts and the flow, as the model itself is quite large and complex. See figure 5.2

The entity graph stores information about a specific patient at a certain point of time in the clinical encounter. The patient comes to the emergency clinic. He has some symptoms which the clinician needs to uncover, by doing examinations and asking questions about the patients conditions. What the patient or caregiver tells is modelled as history, while quick examinations such as listening to the chest, looking at the skin, count the number of breaths per minute are modelled in the examination vertex. In some cases the clinician wants to run medical tests which require more time and resources, such as MRI scan, spirometry or blood tests. These tests are modelled as investigations.

Based on the the symptoms collected in history, examination and investigation, the clinician will set a diagnosis. The procedures for what to do with a patient with a given diagnosis is modelled under management. Hospitalization is to change the patients status to outpatient, or inpatient if he is admitted into the hospital. He might need some medication or be given advise for how he should deal with his condition the in every day life. Here the model can be expanded with routines found in other guidelines, we have identified surgery as an example.

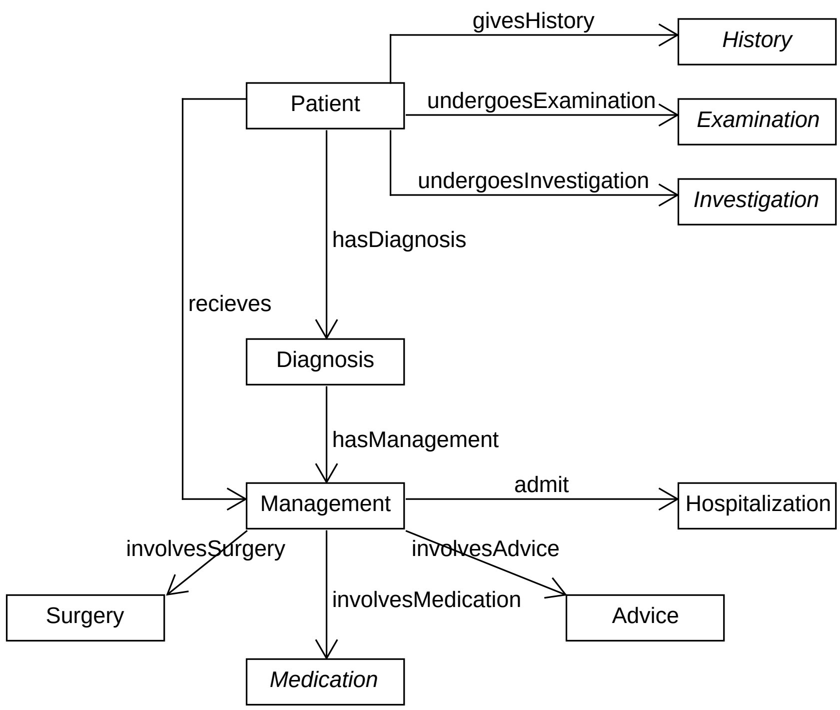

Figure 5.2: An excerpt of the entity graph. Entity graph represents a patient at a certain point in the clinical encounter

In figure 5.3 we have expanded the Patient and Diagnosis vertices to reveal more details. PatientName and Gender, identifies the patient with a name and gender. These attributes are important when presenting a patient and his condition in a narrative or scenario. By using a name, it is easier for the reader to see that this is the same patient in different stages of the clinical encounter.

A diagnosis has a name. In the paediatric possible asthma guideline (Republic of Kenya 2016), the diagnosis has a severity. A lot of medical conditions doesn't have a severity, or they are classified in another way. Here we add multiplicity to the edge which specifies this requirement.

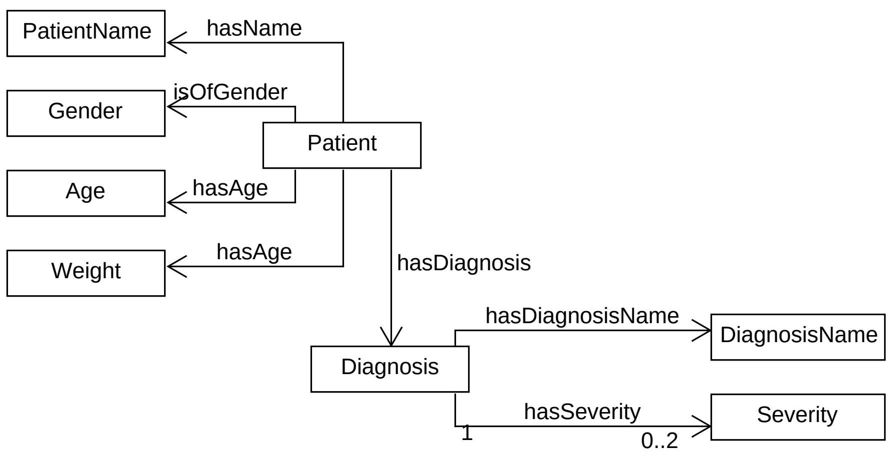

Figure 5.3: Showing the details of the Patient and Diagnosis vertices of the entity graph

In figure 5.4 we have shown our implementation of the History vertex from figure 5.2. History is what the patient or the caregiver tells about the patient's condition. In the paediatric possible asthma guideline (Republic of Kenya 2016) we have identified three symptoms which the clinician can ask the patient or the caregiver about. Here we introduce inheritance, where the specific symptoms inherits the examination vertex. Each symptom the patient or caregiver tell about, will have a measurement. In this specific case, all the history symptoms are boolean. Either they have the symptom or they don't. The symptom values inherits from a Measurement vertex.

For Examination we follow the same principles as History. We implement it by letting each symptom inherit an Examination vertex. Each symptom has value which inherits from a Measurement vertex. In figure 5.5 we have shown the inheritances for each vertex with one arrow and a box. This is of practical reasons when drawing, as there are so many symptoms and it will be confusing to draw an arrow for each of them. Here the values which are stored are a bit more mixed than for History. Consciousness is measured using an AVPU scale, where A is the patient is Alert, V is Verbal, P is responding to

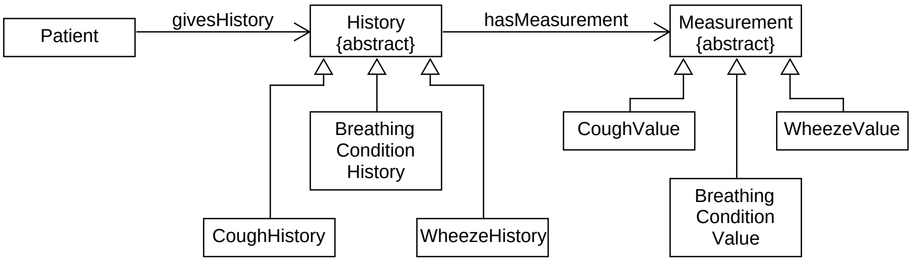

Figure 5.4: Showing the implementation of History in the entity graph. What the patient or caregiver tell about the patient's condition

pain and U unconscious. These are enumerates, where we store either A, V, P or U. Pulse Rate, Respiratory Rate, Oxygen Saturation are all numerical values. The other symptoms are registered as boolean values. Keep in mind that Jaundice is really not a part of the paediatric possible asthma guideline (Republic of Kenya 2016). It is included as part of the antibiotic treatment.

The implementation of Investigation would be just as we did with History and Examination. For asthma they use a lab test called spirometry, but it is not included in the paediatric possible asthma guideline (Republic of Kenya 2016), so we don't include it in our model

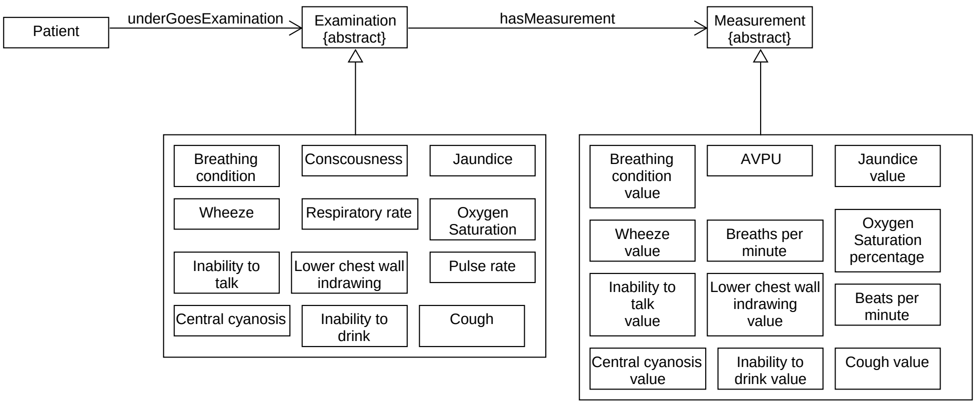

Figure 5.5: Showing the implementation of Examination in the entity graph. What symptoms the clinician can observe the patient has

In the paediatric possible asthma guideline (Republic of Kenya 2016), there are used five medications to treat the patient, as well as antibiotics which is a class of medications. We will talk about each medicine in the model as there are some details which needs to be clarified. See figure 5.6.

- **Oxygen** is a medication, which is given to patient which doesn't get enough oxygen by breathing. In asthma this happens because of the airways are tightened. The paediatric possible asthma guideline (Republic of Kenya 2016) doesn't specify how the oxygen should be administered, but the paediatric guidelines contain a guideline for prescribing oxygen(Republic of Kenya 2016). We have decided to model a part of the prescribing oxygen guideline, to be able to ask simple control questions to verify that the student still remembers how to administer it. Oxygen has a route, which is inhaled. The method is oxygen face mask with reservoir bag. The rate will be given in litres per minute. The duration of the treatment will be until the oxygen saturation is at a high enough level.

- For **antibiotics**, the CPG says the antibiotic should be given according to the paediatric pneumonia guideline (Republic of Kenya 2016). We have decided to include the antibiotic treatment in detail in the model, but we have also kept Antibiotic as a medication, such that we later can ask a question where we don't go in more detail than just antibiotic, which is the detail level of paediatric possible asthma guideline (Republic of Kenya 2016). The antibiotic treatment consists of two medications gentamicin and penicillin. Gentamicin and penicillin are given as an injection either intramuscularly or intravenously, which is modelled as Route and Method. The rate is given by iu per kg, mg per kg, per 6hours or per 24 hours. If the patient has jaundice, he shouldn't be given penicillin. How much penicillin and gentamicin will be given according to the patient's weight.

- **Prednisolone** is a steroid used to calm and prevent inflammation in the airways. The clinician will administer a dosage calculated by the patient's weight. There is a max dosage per day, and the age will determine how high that max dosage is. The rate is given in mg per day. The duration is 3-5 days. The route is oral and the form is tablet. In situations where prednisolone needs to be given more than 5 days, it will be administered with the route inhaled and form inhaler.

- **Corticosteroid** is another steroid, which is given to in scenarios of recurrence asthma symptoms. The CPG specifies that corticosteroid should be inhaled. This is represented in the model by the Route vertex. The method the which will be used to inhale corticosteroid is represented by the Form-vertex. The Form is MDI with spacer, preferably with spacer with face mask.

- **Salbutamol** is inhaled to open the airways of an asthma patient. The Rate vertex tells at which rate the patient should be taking salbutamol. Asthma patient is given salbutamol at a rate of 2.5mg per 20 minutes if nebulized, or 6 puffs per 20 minute if the inhaler is used. The Duration is up to one hour or three doses if needed. The method is inhaled and the form is either nebulizer or inhaler.

- **Ipratropium bromide** is modelled much like salbutamol with a Rate and a Duration. The rate is given in mcg every 20 minutes for a duration of one hour if needed. The route is inhaled and the form is inhaler.

We have kept the Dosage vertex, while it is not in use. It can be used to represent the total amount of a specific medication given to the patient during this treatment.

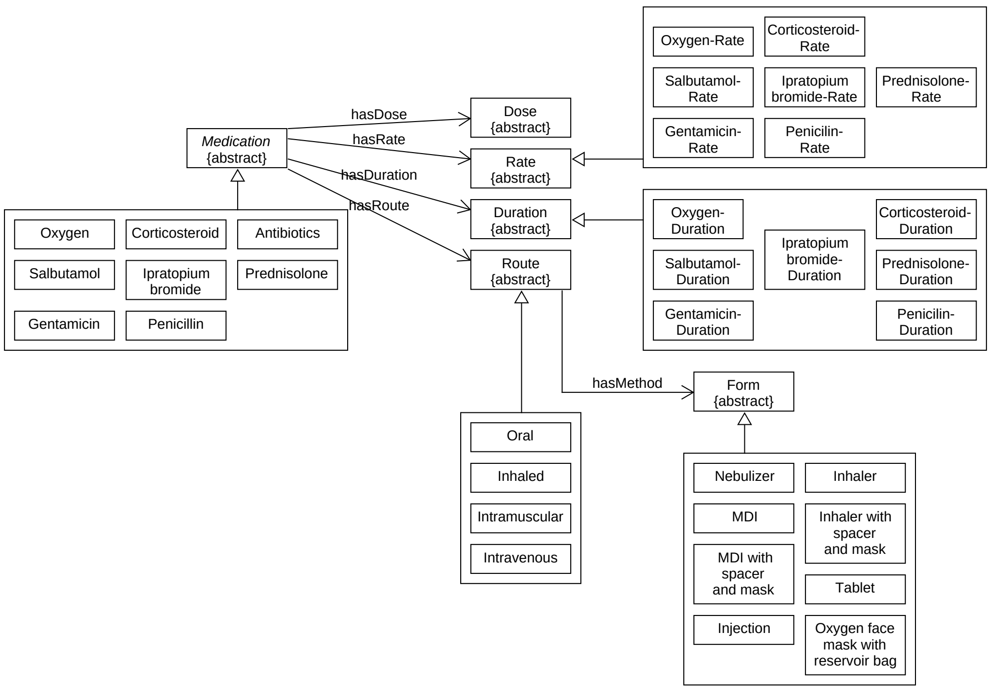

Figure 5.6: Showing the implementation of Medication in the entity graph. How to administer a medication to patient

Another detail must be clarified when implementing instances of the medications. The route must be given names which can be uniquely identified for each medication. An example is Ipratopium Bromide and Salbutamol which both uses the Route Inhaled. Ipratopium Bromide uses only the inhaler, while Salbutamol can be given using the nebulizer. If Ipratopium Bromide is sharing the same Inhaled vertex as Salbutamol during instantiation, it will look like Ipratopum Bromide can be nebulized. By creating a new instantiation of Route per medication, and uniquely identify them by name Inhaled1, Inhaled2 or InhaledSalbutamol, InhaledIpratopium, we avoid this problem. Now we can connect Salbutamol with Inhaled1 and have edges to Form Inhaler and Nebulizer. Ipratopium Bromide has an edge to Inhaled2, which has an edge to Inhaler. Then we have an instantiation where Ipratopium Bromide can only be inhaled using the Inhaler. See figure 5.7.

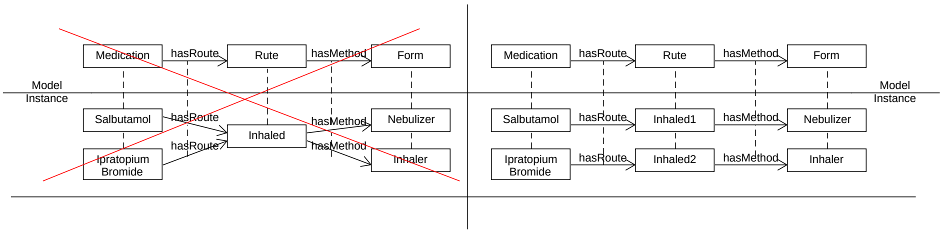

Figure 5.7: To the left we don't know which medication has been inhaled using the nebulizer or inhaler. To the right we have specified this in the instantiation, by using two Inhaled-vertices and uniquely identifying them by giving them different names

5.4.1 Generic entity model

In the previous section, we had the focus on developing a specific entity model for the paediatric guideline of possible asthma (Republic of Kenya 2016). In fact by introducing inheritance, we actually made a specific and a generic model. In the model, we use the keyword {abstract} to note an abstracted vertex. The inheriting vertices become the specifications of the abstract vertex.

In figure 5.8 we have showed the the generic model. It only shows the abstracted vertices and not the specifications. A specific entity model for a specific guideline, will show both the abstractions and the specifications of them. The specifications need to be adapted for that specific guideline, and can not use the same as for paediatric guideline of possible asthma (Republic of Kenya 2016).

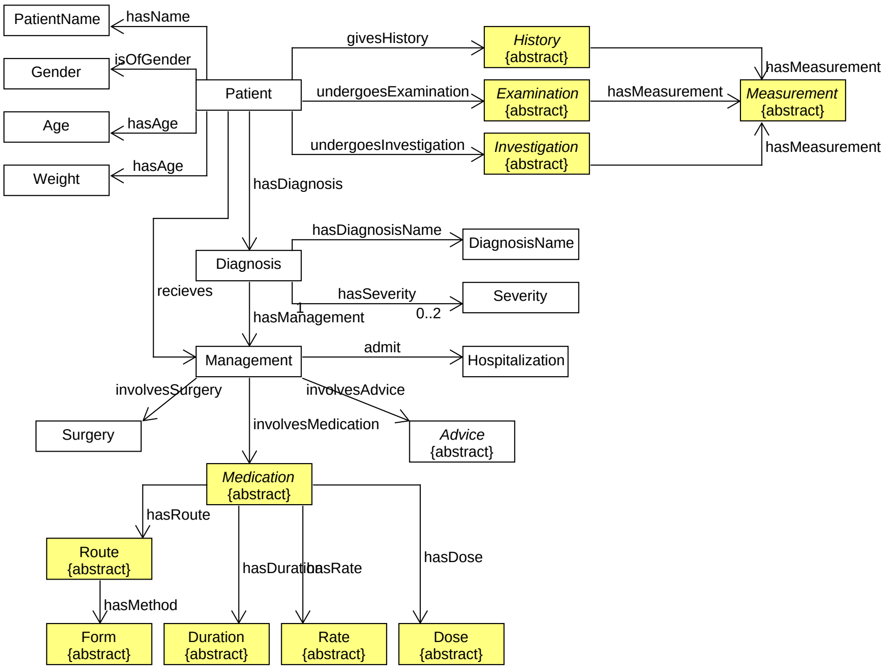

Figure 5.8: A generic entity graph contains abstractions. An entity graph for a specific guideline, contains specifications for those abstractions. Yellow boxes indicate specifications we have shown for paediatric guideline of possible asthma (Republic of Kenya 2016)

5.5 Workflow model

We model the dynamics of clinical encounters with a workflow model. The clinician starts with the assessment, where he examine the patient and listen to what the patient and the caregiver has to say about the the patient's condition. The clinician starts to get an idea of what condition the patient may suffer from. The clinician continues with the diagnostic part, where he asks more targeted questions to the patient and caregivers about the condition, do more of the examination and perhaps order lab tests as part of the investigation. This process can strengthen the clinician's assumptions about the condition, and he may be able to set a specific diagnosis.

The next step is the management and treatment. This can be changing the patient's status from outpatient to inpatient, do surgery, medication, physiotherapy, cognitive behavioural therapy or other forms of treatment. The treatment may be done in iterations or repeated.

The final step is to evaluate. The treatment may have to be adjusted to get the right effect. The diagnosis has changed. For example the severity of asthma may have changed from severe to mild or moderate after the treatment. Or we have initial set the wrong diagnosis, for example we have treated a patient for possible asthma, but in fact an object was stuck in the airways of the patient.

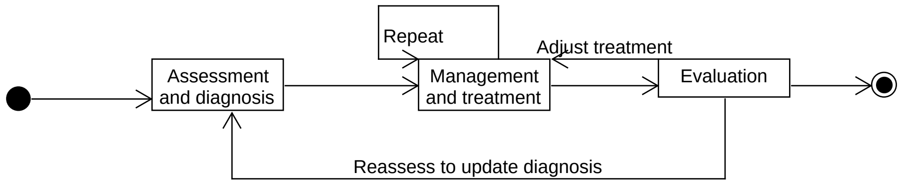

Figure 5.9: The workflow models is a model of the clinical encounter

The idea of the workflow model is to describe the process of a clinical encounter. When making scenarios for the game, we know in which order the scenarios should come. The entity model is also connected to the workflow model. When doing a an assessment and diagnosis, you are looking at the examination, history and vertices of the entity model, where the diagnosis vertex answer to what the specific diagnosis is. For doing management and treatment, you look at the vertices under management in the entity model. An evaluation will be done by looking at the examination and investigation vertices to see if the patient has become better. The treatment needs to be adjusted accordingly to the evaluation. If the evaluation says we can't do more for the patient and there's no need for a follow-up, we exit the workflow model.

5.6 Metamodeling

In figure 5.10, we make an instance of the entity model. An instance of the entity model describes an actual patient at one point in the clinical encounter.

For an instance to be valid, the vertices and edges have to correspond to a part of the model. We demonstrate this by adding dotted arrows in figure 5.10.

In figure 5.10 a patient tells the clinician that he struggles with a wheeze and a cough. Cough and Wheeze inherit from History in the model. Difficulty breathing is part of 5.4, but is not represented here as the patient hasn't brought up this issue or been asked about it. We see how two inheritances og History translated in the instance from the model. The Measurement vertex holds the measurements of the History vertices. A patient with a wheeze and a cough is diagnosed with asthma, which is shown in the instance.

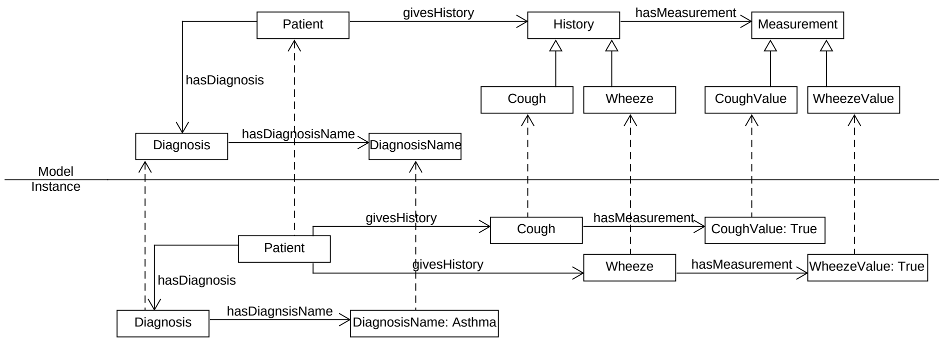

Figure 5.10: A model and an instance of the entity model. For a valid instance, every vertex and edge in the instance has a corresponding vertex and edge in the model

In figure 5.11 we show a entity instance working together with the workflow model. For the assessment, we look at the History and Examination vertices. For Diagnosis, the DiagnosisName and Severity. Keep in mind that under Diagnosis, the clinician may do further examinations and questions to the patient to confirm his assumption, or which may cause him to think about other diagnosis. Management, the asthma is severe so we change the patient's status to inpatient by updating the Hospitalization vertex. We also look at the Medication vertex under Management. We only care about the medications for now in this example, and not how the medications should be administered. The Evaluation holds a reference to a new entity instance, which holds the updated information about the patient's symptoms. The clinician needs to act accordingly and adjust the treatment.

5.7 Game model

The game model is not implemented using DPF and MDE like the entity model and workflow model. However, a MDE game model could be relevant in future work.

Here we will give a brief explanation of the content of the game model. It can work as a reference point for the next chapter, where we will discuss the game elements more thoroughly.

- **Category:** a CPG is written for a specific medical condition. The quiz category will be the medical condition, or the name of the CPG the

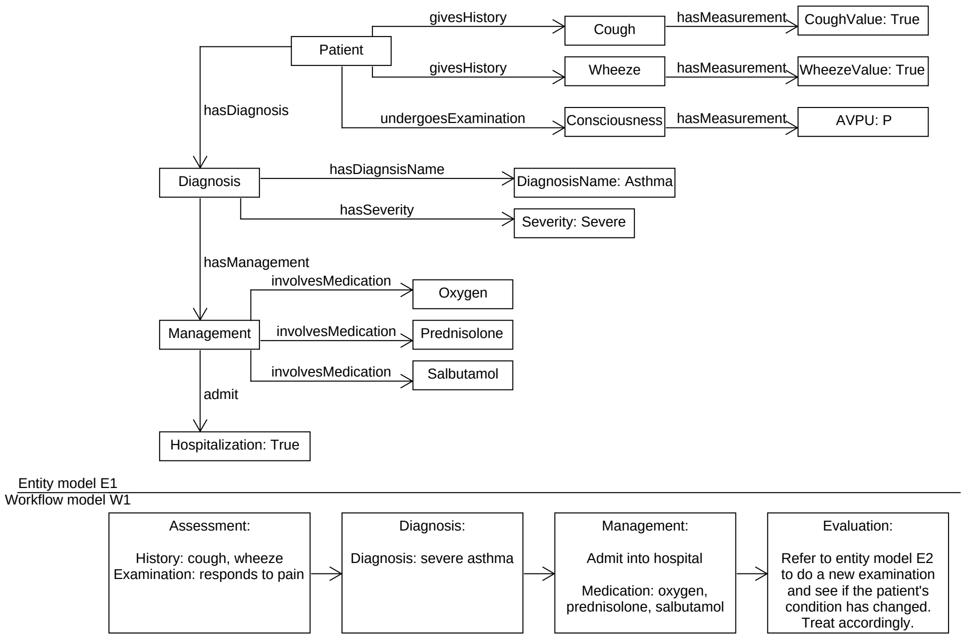

Figure 5.11: An instance of the workflow model at the bottom, working together with an instance of the entity model at the top

quiz is written for.

- **Discipline:** a question is written for a specific step in the clinical encounter, which the workflow model is an abstraction of. Discipline is a step in the clinical encounter.
- **Level**: Is the difficulty level of the question.
- **Passing condition:** a condition the student has to fulfil to pass a specific difficulty level.
- **Required Minimum Skill:** a condition the student has to fulfil to be able to play questions at a specific difficulty level.
- **Entity Instance**: a pointer to an instance of an entity graph. It is used together with a template to generate text.
- **Question:** a pointer to a template model. By using a template model, we can reuse templates on many entity instances.
- **Alternative:** or distraction. It is one of the answer alternatives for a question.

- **Reward:** a reward or penalty is given based on the correctness of the distraction. A distraction can be degrees of right or wrong, which is reflected by the reward.

The game model is paired with a template model. By separating the template into its own model, we can reuse the same template for many questions.

- **Narrative:** template which represents a question. The template contains tags which points to vertices in an entity graph. When the template is paired with an instance of such a graph, it will produce a textual presentation of a question. The same template can be used with different instances of entity graphs to make many different questions.

- **Answer key:** is a tag which points to a vertex in the entity graph. When paired with an instance of the entity graph, it will produce a textual presentation of the answer key. The answer key can be used with many different entity graph instances.

- **Explanation:** this is the answer key explanation to a narrative/question. It gives an description of how to solve the problem and what is the correct answer.

- **Evidence and Guideline:** these are not implemented yet, but are place-holders for future functionality. The evidence is the strength of the evidence the recommendations in the guideline. The guideline is a pointer to the guideline itself. Such that that the student can read the guideline itself when he is stuck on a question or need further explanation. This will hopefully enforce learning.

5.8 Student learning model

The student learning model keeps track of the student's scores at different quizzes. It is based on the principles of learning map and student map, which can be read about in the next chapter about game elements. By using such a model, we can adapt the questions to the right difficulty level for the student. We can also keep track of his progression

5.9 Summary

In this chapter we have discussed the entity-, workflow-, game- and student learning models. We have been especially thorough with the two first models, which are developed using MDE and DPF. In MDE it is the models which drives the development process, and DPF is an approach to MDE where the models are represented as a graph with a set of constraints.

In the entity model, we modelled the patient, the symptoms the clinician might find under examination and by the history the patient or parents give. We modelled the diagnosis, and how the clinicians manage thee medical condition of the patient. The clinician may admit the patient into the hospital, treat with medications and give some advise on how the patient or parents should manage the medical condition of the patient.

A workflow model which is the workflow of the clinical encounter.

We do an example where we make instances of the entity model my metamodelling. Then we instantiates the entity model together with the workflow model, to see how they work together. Describing the patient, the condition, symptoms and the management through the whole clinical encounter. Through assessment and diagnosis to management and treatment to evaluation.

We gave a brief explanation of the elements in the game model, and the concept of the student learning model.

Chapter 6

Game Elements

6.1 Question Flow Manager

As each question in a quiz are related to a certain component in the treatment plan or theme in the learning map, the student will be measured how well he performs on each of these themes. For the asthma guideline (Republic of Kenya 2016), we have identified four themes. Assessment where the student will be tested in the initial examination. Diagnosis, where the student will determine a diagnosis as well as the severity. Management, where the student will determine which actions should be done to treat and best give the best care to the patient. The last discipline is the follow-up, where the student will be tested in evaluating the treatment, give advise to and educate patient and caregivers, provide the right medication and regular follow-up.

By splitting up the score in themes, the student can easily see which areas he is strong and where he needs more training.

We can also adapt the questions in each discipline to the student's level. If the student has proven to be very good in providing the right amount of medicine to asthma patient, we can provide more difficult questions to challenge the student some more. If he struggles at setting the right diagnose, we can provide more basic questions to strengthen the students basic knowledge.

The student will also be provided with a total score, which will be the aggregated score of each of the disciplines. The student can compare the total score of e.g. the asthma quiz and the jaundice quiz, and see which medical condition he needs to train more on.

6.1.1 Dynamic Content Management

The game engine is based on some of the concepts presented in the articles of Eide, Kristensen, and Lamo (2008), Kristensen (2011) and Kristensen, Bech, and Dyngeland (2013). The motivation for using DCM, is to support the principles of adaptive learning as well as flexibility in the learning process. Adaptive learning means that the student can solve problems which are suited to his knowledge level. Flexibility in the sense that the student can go through the learning content in many different ways.

Eide, Kristensen, and Lamo (2008) presents a dynamic content management system (DCM) made for e-learning. In DCM the focus is on removing the tight coupling between the learning material and the teaching course. By analysing the learning material and course, they can define conceptual atomic units of knowledge which they put into a knowledge repository. From this repository they may draw knowledge elements and organize into the hierarchy of a course. To model a course they use concept maps, which are directed graphs, where the vertices are concept labels and edges indicates the relationships between vertices. DCM operates with three concept maps: knowledge map, learning map and student map. The knowledge map is used to model the entire content of the knowledge repository and the hierarchy structure of a course. A learning map is used to model a specific course and is a representation of the learning process. The content units (vertices) in the knowledge map gets expanded, and becomes evaluation and resource vertices in the learning map. Content units from the knowledge map can be omitted if they are not needed in the specific course. Detailed prerequisites can be specified for the content units. The student map represents the progress of a specific student taking a specific course. The edges shows which resources he has used, the evaluations of the student and in which order.

When we apply this to our project, we will first decouple the content from the flow-chart of the paediatric possible asthma guideline (Republic of Kenya 2016). To decouple the flow-chart, we use the workflow model as a helping tool, ordering on how they should learn the clinical guideline. In figure 6.1 we have identified knowledge elements and hierarchically structured the knowledge repository into a knowledge map. The edges shows the dependencies in the learning process. To learn how to evaluate a treatment and act accordingly to the evaluation, the student needs to know something about how to set a diagnosis and how to initially treat it. Hence there is an edge between follow-up and the other knowledge units.

In table 6.1 we have chosen a knowledge element from the knowledge map, and defined a content unit. The content unit contains a theme "Assessment". Resources is the learning material, which are relevant sections of

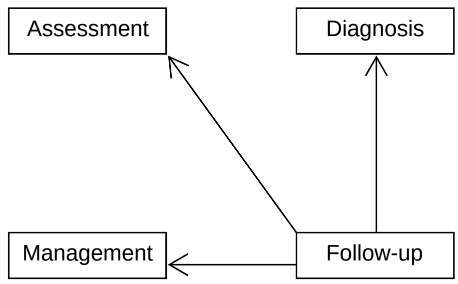

Figure 6.1: Knowledge map

the guideline. Evaluations are the tests to see if the student has reached the learning goals. At level 1 we try to learn facts about the guideline. In level 2 and 3 we give the student scenarios to work with.

T1: Assessment

Resources	Evaluations	Aspects
R1: History and examination sections in the possible asthma guideline (Republic of Kenya 2016)	E1: Quiz Level 1	Facts
	E2: Quiz Level 2	Scenario
	E3: Quiz Level 3	Scenario

Table 6.1: The content unit Assessment

In figure 6.2 we have identified four content units T1: Assessment, T2: Diagnosis, T3: Management and T4: Follow-up from the knowledge map. The importance of these content units in the asthma guideline (Republic of Kenya 2016) was the reason these four got selected. The hierarchically structure of the knowledge map, makes the child nodes of the content units become learning material for their parent nodes.

Inside the content units in the learning map, we see the relationships between resources and evaluation. There is also dependencies between the content units. To be able to do a follow-up, a student needs to learn the assessment, diagnosis and management first, because follow-up is an evaluation and reaction to how the patient responded to the previous steps. We have also specified the prerequisites for each evaluation. The prerequisites are written as logical expressions, as seven edges and operators per content unit would be confusing to read. What the prerequisites says is that all level 1 evaluations need to be completed before any level 2 evaluation can be taken. All level 2 evaluations need to be completed before any level 3 evaluation can be taken.

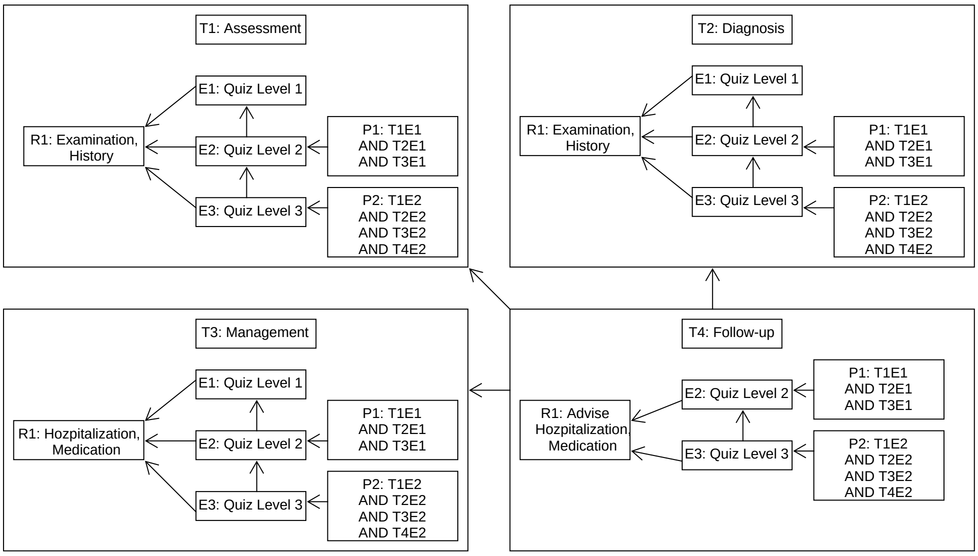

Figure 6.2: Learning map

A student map would show the progressions for one specific student, and the path he has taken and the scores for each evaluation. Table 6.2 shows a student's student map for the Assessment part of the course. He got the score 34 on the first evaluation. 34 matches the passing condition for assessment level 1, so he got a passing grade for that test. However, he scored 43 points on evaluation 2 and didn't meet the passing condition for that test. He has no attempts for evaluation 3, as he doesn't meet the prerequisites for that test. The tests which have been completed are stored in the database on the students phone. In that sense, the database shows the student's current position in the learning map.

T1: Assessment		
Resources	Evaluations	Passed
R1: History and examination guideline	E1: 35 E2: 43 E3:	True False

Table 6.2: Student Map T1:Assessment

6.1.2 Types of questions

We can further identify which kind of questions we are going to ask, by further examining the entity model and the workflow model combined. When we are doing an assessment in the workflow model, we are looking at the examination and history vertices of the entity graph. History is what the patient or the patient's dependent can tell around the patient's condition. Such that he has been coughing a lot the last days. Examination and history will provide the clinician with an idea of a diagnosis.

Diagnosis in the workflow model is connected to the history, examination and investigation. The clinician will continue asking questions, do examinations and probably order some tests to strengthen his assumption of the diagnosis. The clinician can also set a more specific diagnosis, such as in the asthma guideline (Republic of Kenya 2016), where we categorize in severe, mild and moderate asthma.

Management is how we manage the patient with the given diagnosis. Hospitalization is if we are going to set patient status to inpatient or outpatient. Medication is given to treat the patient or relieve the symptoms. Surgery is not part of the asthma guideline (Republic of Kenya 2016), but it is important to be aware that management procedures needs to be added when making quizzes for other conditions than asthma.

Follow-up will contain questions related how to evaluate the treatment and how to act upon the evaluation. The symptoms of the patient may have worsen, getting better or are unchanged after the treatment, and the clinician needs to act accordingly. The treatment may have been given at the hospital or it may have been something the patient have had to do at home. The patient may also get some instructions from the clinician when there is something he should do on his own.

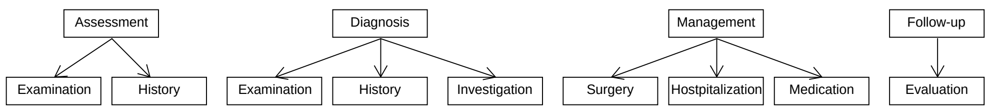

Figure 6.3: By looking at the workflow- and entity models we can identify the type of questions for each content unit in the knowledge map. The content units are the parent nodes, while the leaves are the type of questions

As the student learns, the questions need to adapt to the student's new level of knowledge. How we do this is by defining levels, where the questions becomes more detailed at higher levels. At level 1 the questions are all about stating facts. In level 2, we create scenarios such that we follow one patient through all of the steps assessment, diagnosis, management and follow-up. The same for level 3, but the detail level will be higher. Typically the student will only be asked for categories of medication in level 2, but in level 3 the students needs to be specific about the names of the medication as well as measurements for both dosages and symptoms. See table 6.3.

Level	Assessment	Diagnosis	Management	Follow-up
1	Factual	Factual	Factual	-
2	Scenario	Scenario	Scenario	Scenario
3	Detailed scenario	Detailed scenario	Detailed scenario	Detailed scenario

Table 6.3: The type of questions at each difficulty level

6.1.3 Unlocking harder levels at a certain category

One of the strengths with Dynamic Content Management is the focus on adaptive learning and flexible learning (Eide, Kristensen, and Lamo 2008). By adaptive learning, we mean that the student can solve problems which are suited for his knowledge level. While flexible learning means that the student can go through the learning material in the way he prefers, as long as the knowledge dependencies are met.

For the adaptability we have already covered how we progress from factual statements to scenarios and the scenarios with a higher ability to make the questions more difficult. For the flexibility we have divided the content into knowledge units: assessment, diagnosis, management and follow-up. It doesn't matter which of assessment, diagnosis and management the student finishes first.

The evaluations in each content unit has passing conditions. As an example; to complete assessment level 1, you need a score of 30 in assessment. These passing conditions are provided by the quiz author. We also recall that to unlock questions, such as management level 2, there are certain prerequisites that need to be fulfilled. In this example, all evaluations in level 1 need to be completed to unlock any level 2 questions. The flexibility is that it doesn't matter the order of level 1 evaluations the student finishes first, as long as all of them are finished.

To avoid that the student gets bored, he will not have to redo an evaluation once the passing condition is met. That means that if the student meets the passing condition of level 1 assessment and diagnosis, but not management, the student will only get questions from management level 1 the next time he takes the quiz. This makes a challenge for the scenarios. All scenario questions must be formed in a way, such that the student doesn't have to remember information from one question to another. All the necessary information should be listed in every scenario question. In that way the student won't miss important information if he completes diagnosis in a run before management and follow-up.

Each evaluation also has a minimum required skill value. If the student gets a lower score than the required minimum skill, the evaluation gets locked and the student needs to complete the evaluation at lower level. An example is that the student plays management level 2. He completes the evaluation with a lower score than the required minimum skill. The student will have to redo management level 1 evaluation to learn the basic skills necessary to play level 2. When this situation happens, the student no longer meets the prerequisite for the other level 2 evaluations. Level 1 management needs to be completed before any level 2 evaluations can be taken.

6.2 Conversation Manager

6.2.1 Constructing scenarios and answer keys

A quiz consists of several questions, where each question has answer keys and distractions. At level 1 these questions will be factual, and in level 2 and 3 we will work with scenarios. We write the questions in the format of a template, where we use tags to refer to variables in the entity model. The tag is a path in the entity graph. The game engine will traverse through the graph and return the value of the vertex specified by the path. The entity graph represents a patient at a certain point in the assessment or treatment. By replacing the entity graph, we can reuse the same template with different patients, generating many different questions with the same template.

The same goes for the answer key. The answer key can be a text, one or several tags referring to variables in the entity model, or a combination of both. The application always uses the same entity graph for the scenario and the answer key. When the application traverses entity graph, the answer key and the question given always matches as they are the same patient at the same given time.

One of the problems we encountered with this method, was how to present the variables returned by the graph in a text. As an example, we can look at some of the symptoms of asthma and their type.

- Wheeze is a whistling sound when you breath. In the model it is represented as a boolean. True or false. Either you have it or you don't.

- Age is relevant several places in the guideline. In the model it is stored as an integer.

- AVPU is a scale system, which clinician use to measure a patient's level of conciousness. A is alert. V, the patient is verbal which means he can somehow respond to questions. P, the patient responds to pain. The patient will react if you pinch him. U is unresponsive or unconscious. He doesn't respond to either voice or pain. The AVPU is represented in the model as an enumerate.

- Severity classifies the asthma severity to be either severe, model or mild. The severity is represented as a string in the model.

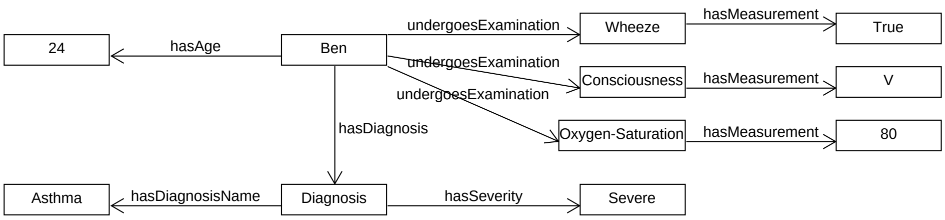

Figure 6.4: The patient object used with the template

```
Question:
A <%Patient.hasAge.Age%> months old patient arrives at
the  emergency department.
The patient has a
<%Patient.underGoesExamination.Wheeze%>,
has conciousness level
<%Patient.undergoesExamination.Consciousness%>
and has an oxygen saturation of
<%Patient.undergoesExamination.Oxygen-Saturation%>%.
What is the asthma severity?
```

Listing 6.1: Question template

```
Answer key:
<%Patient.hasDiagnosis.Diagnosis.hasSeverity.Severity%>
```

Listing 6.2: Answer key template

This translates to

```
Question:
A 24 months old patient arrives at the emergency
department.
The patient has a true,
has conciousness level V
and has an oxygen saturation of 80%.
What is the asthma severity?
```

Listing 6.3: Question instantiation

```
Answer key:
Severe
```

Listing 6.4: Answer key template

We see that the template author needs to be aware of how the variables will be printed. Here he knows that the model will just return an integer for the oxygen saturation. He writes a descriptive text of the value first, and then adds a percentage after the variable. The severity gets nicely printed as answer key.

The problem is the boolean for wheeze, which prints a "true". It really should have printed "wheeze". We solved the presentation of consciousness in the same way as we did with oxygen saturation. However, it could be nicer to write "responds to pain", rather than just "V". When the child 12 months or older, it is often easier to read if we can present the age in years.

Another problem arrives when we replace the entity graph with another, where some of the examinations haven't been done. In traditional model driven engineering, we use something called the closed world assumption (Sadowska and Huzar 2019). If a node doesn't exist, we say the value is false. But how can we say that patient doesn't have wheeze when we haven't examined? In open world assumption a none-existing vertex is "unknown" (Patel-Schneider and Horrocks 2006) (Bergman 2018), and this strategy seems more correct for our scenario. If the vertex doesn't exist, we simply return an empty string. This further motivates us to remove the variable specific text from the template, as we don't want text representation for a variable we don't present. Example when consciousness and oxygen saturation haven't been examined:

```
Question:
A 24 months old patient arrives at the emergency
department.
```

```
The patient has a true, has conciousness level
and has an oxygen saturation of %.
What is the asthma severity?
```

Listing 6.5: Question instantiation

How we solved the problem was to add a textual presentation vertex to each of the vertices in the graph referred to by the template. If there exist a presentation for the vertex, return the presentation. If there doesn't exist a presentation, simply return the variable.

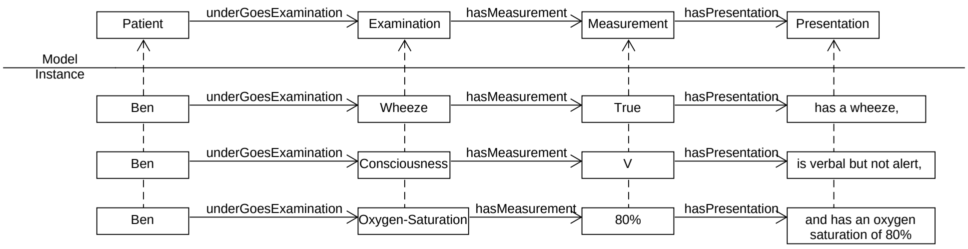

Figure 6.5: Making graph variables fit a story format

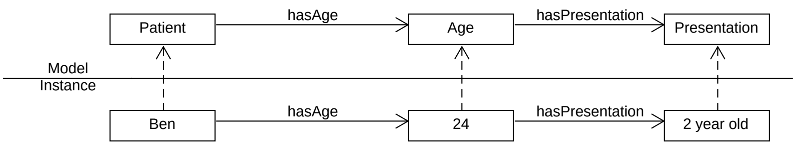

Figure 6.6: Making graph variables fit a story format

```
A <%Patient.hasAge.Age%> old patient arrives at
the emergency department.
The patient <%Patient.underGoesExamination.Wheeze%>
<%Patient.undergoesExamination.Consciousness%>
<%Patient.undergoesExamination.Oxygen−Saturation%>.
What is the asthma severity?
```

Listing 6.6: Question template

Which translates to

```
A 2 year old patient arrives at the
emergency department.
The patient has a wheeze, is verbal but not alert
```

```
and has an oxygen saturation of 80%.
What is the asthma severity?
```

Listing 6.7: Question instantiation

This also works with the open world assumption, as a patient which haven't undergone consciousness and oxygen saturation examinations, would result in the following text:

```
A 2 year old patient arrives at the emergency
department.
The patient has a wheeze.
What is the asthma severity?
```

Listing 6.8: Question instantiation

For future work, the commas and "and" should not be in the presentation vertex. This becomes a limitation where the variable can only be used in a list and has to be in a specific place in the list. The solution would be to have a list tag in the template, and have all the paths inside that tag. Then the game engine can see how many of the list items are in the graph, and can set the commas and "and" at the appropriate places.

In figure 6.7 we show how the entity-, workflow and game models work together to make a scenario. Because of limited space, we don't show the presentation vertices in the entity graph. We also combined entity graph E1 and E2 in the presentation, where E2 gets and additional medication vertex marked with red.

6.2.2 Multiple-try feedback

The quiz uses a concept which is called multiple-try feedback (MTC). That means for every question the student gets more than one attempt to get the answer right. A feedback will be given immediately after each answer is submitted. The feedback consists of a message which tells whether the answer is correct or wrong. If the answer is correct, the user will receive "correct" and an explanation of the answer. If the answer is wrong, there will be no hints or explanations than just "incorrect".

The point of doing MTC, is to make the student think over what was wrong with his first answer. Did the student misinterpreter the question? Was there a detail he missed? Does the student lack the knowledge or was he just sloppy in his first attempt?

Clariana and Koul (2006) did a study where they divided 82 students into five groups. DF-, KCR, MTC and two control groups. The first control

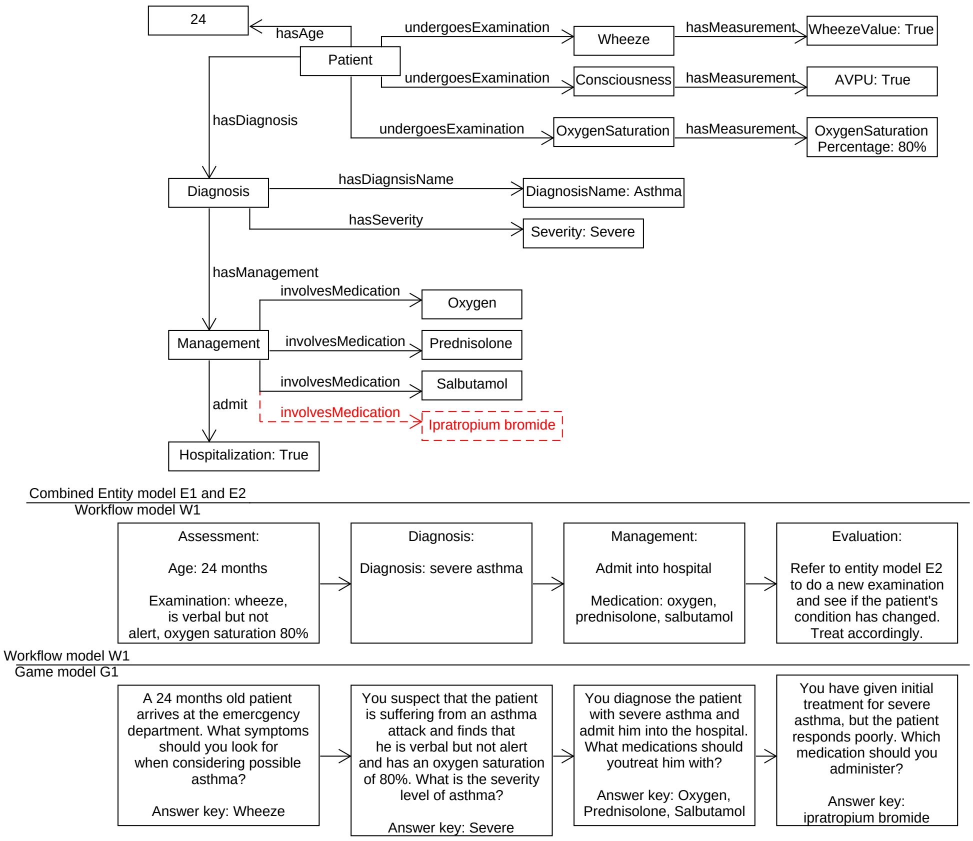

Figure 6.7: The entity- workflow- and game models working together to produce a scenario, taking the patient through the steps of a clinical encounter. Note that these are two entity graphs E1 and E2, where E2 gets updated with an additional red vertex Ipratopium bromide. This illustrates a patient which doesn't respond to the initial treatment, and then gets treated with ipratopium bromide

group got a text and a question at the end. The second control group got a text, but there were no question given. After 5 days,post-test was held to see what the students had learned and remembered. The post-test questions were either identical to the questions in the learning material, transposed where the order of the stem of the question and the correct-response gets reversed, paraphrased where post-test questions had the identical content as the learning material, but the phrasing was different and used different words, and a combination of transpose and paraphrasing. The results showed that DF and KCR groups performed better on identical, transposed and paraphrased-transposed questions. MTC performed better on paraphrased

Concept	Abbreviation	Feedback after each question	Multiple attempts at each question	Hints on wrong answer
No or delayed Feedback	NF or DF	No	No	No
Knowledge of Correct Response	KCR	Yes	No	No
Multiple-Try feedback with knowledge of Correct response	MTC	Yes	Yes	No
Multiple-Try feedback with Hints	MTH	Yes	Yes	Yes

Table 6.4: Overview over different quiz concepts

questions. The conclusion was that DF and KCR was much better methods for remembering the learning material word for word, but MTC was better when you have to think and reason about what you have learned.

Attali (2015) further did a did a study on NF, KCR, MTC and MTH using open ended and multiple choice questions on mathematical problems. They showed that solving an open ended question rather than multiple choice was a more efficient way to learn. The learning outcome was the same for the students using NF and KCR. However the learning transfer was greater when using multiple-try (MTC), and even more so when getting a hint on incorrect answer (MTH). They explained the results effortfull and mindful problem solving. In a multiple-try feedback, the user will have to reflect on their errors, re-evaluate the problem and understand the initial error. An open ended question will also require more effort of the student, as they have to generate a an answer rather than selecting from alternatives. On the combination of multiple-try and multiple-choice, it was suggested that some users might be less likely to review their incorrect answer and mindlessly clicking on another alternative.

According to Morrison et al. (1995), students which perform badly on answer until correct questions, will often become frustrated, loose interest for reviewing the material and probably depress learning.

As thinking and reasoning about a diagnosis, treatment plan, evaluation and follow-up of a treatment is part of a medical procedure, we believe that multiple-try feedback is the right approach. Because of the nature of a mobile app, where gestures are more convenient than typing sentences, multiple-choice seems to be the right choice even, though open ended questions has proven better results in. There's also a technical problem with evaluating free typed sentences.

Some of the questions in the app are too simple for a hint to be meaningful.Example: "the symptoms for asthma is" and the answer can be "cough and wheeze". Where hinting "cough", would be giving away the answer, especially in a multiple-choice format. However, the data model supports hints as links to external learning material. E.g. the student could look for the answer in the guideline itself.

We solved the "answer until correct"-problem described by Morrison et al. (1995), by having a "read more" button displayed upon incorrect answer. The "read more"-button will display the correct answer, an explanation and continue to the next question. Avoiding the user becoming frustrated and discouraged by having to brute-force the answer keys to progress.

6.2.3 Reward system

By having multiple-try feedback, another problem rises, and that is the reward system. If there is no penalty for incorrect answers, a student which needs ten attempts per questions, will get the same score as a student which answers all the questions correctly on the first attempt.

Attali (2015) solved the problem by giving 1 point for answering correctly on the first attempt. 2/3 points for the second attempt, 1/3 for the third and 0 points if the third attempt was incorrect. A limitation with this method is that it makes no sense for the student to make more than three attempts. Morrison et al. (1995) had another strategy where they adjust the scores by dividing the total score by the total number of attempts during the quiz. A consequence is that attempt number two will have a huge penalty which is halving the students total score. While attempt number twenty will give a very small penalty from attempt nineteen. A method to dampen this effect could be dividing the total score by the sum of reviews and number of questions. Another method could be taken the square root of the question reward for every attempt the student has on that question.

The solution we used was having a fixed value for every answer alternative. The quiz author chooses the penalty for each distraction and reward for each answer key. The idea is that the distractions can have some sort of degree of wrong or right, and this can be reflected in the scoring. On the question "what are the symptoms of asthma?", "difficulty breathing" is a more correct answer than "fever", as "difficulty breathing" is a symptom of asthma in combination with wheeze. Fever is not an asthma symptom at all. In future work, the penalties can be automated as you can see from the entity model whether the symptom belongs to the asthma guideline or not. A distraction from respiratory disorders may give a larger penalty than a distraction from the asthma guideline, but smaller penalty than symptoms not belonging to respiratory diseases.

Both Attali (2015) and Morrison et al. (1995) avoids the scenario where the user gets a total minus score. This may be a strength of these methods, as a negative total score seems like a very harsh feedback and might demotivate the student. In our solution we use negative numbers as penalties on distractions, such that a negative total score may happen. We try to limit the likelihood of a negative score by providing a very high reward for a correct answer and a very small penalty for a distraction. Typically the reward is 10 points and the penalty -1 og -2 points. The intention is to encourage the student to review the incorrect answer and try again. As the format is multiple-choice and the penalty-reward ratio, there is a little risk involved trying multiple times. But giving up by clicking "learn more", the student will not get an additional penalty, but will miss out on the reward. By clicking answer alternatives mindlessly and consequently clicking "learn more" will probably not end up in a negative score, but is more likely to end up in a negative score than mindlessly click answer alternatives until correct.

6.3 User manager

6.3.1 Visualization of game statistics

The student needs some feedback, where he is in the learning map, how close he is to pass an evaluation, how close he is to progress to the next level and in the case where he gets relegated to an easier difficulty level.

The scores for the evaluations can be shown as bars in a graph, and the passing conditions as a line. When the bars reaches or passes the line, the student knows that the he has met the passing condition for the evaluation at that level. This can also work as a motivation for the student, as he sees that he gets closer and closer to the passing condition as he learns he more and performs better.

To visualizing where the student is in the learning map, has been solved by showing the learning map in almost a table format. Colour combinations shows which evaluations have been unlocked, which are locked and which are the current ones the student plays. Red and green indicates whether a student gets relegated from a level or whether he progresses to the next level.

6.4 Conceptual model of how all the classes are connected in Game Engine

The conceptual model of the game engine is shown in figure 6.8. Category is a quiz game for a certain CPG, such as the paediatric possible asthma guideline (Republic of Kenya 2016). Each of these quiz categories are divided into several disciplines. Disciplines are themes in knowledge units, which we identified using DCM (dynamic content management). Each of these knowledge units contain evaluations at different difficulty levels. These evaluations are collections of questions, which contain a collection of multiple choice questions and answer elements.

The quiz data are read from JSON-files, which are produced by a content author. The quiz author defines a quiz for a specific CPG. He identifies themes for content units, which contains evaluations with varying difficulty levels. The questions are written in a template format, which we have already covered. The templates contain tags, which refer to vertices in the entity graph, representing a given patient. The answer key is also such a tag, referring to one or more vertices in the entity graph. When the quiz is initialized, the tags will be replaced with values from the entity graph. The content author will write answer alternatives for each question, where he specifies a reward (or penalty) for each of the answer alternatives. An answer alternative which matches the answer key value in the entity graph will represent the correct answer. The content author needs to know what is in the entity graph to be able to provide meaningful rewards.

The student skills is the scores from the last played game, and is fetched from the database on the student's phone. The allowed- and unallowed levels are calculated from the student skills and the requiredMinSkill for each level. Unallowed levels are levels that the student can't play and are locked for the student, as the student's score is lower than the required minimum skill.

A new instance of skill will be created when the student starts a new game. This instance will hold the scores for the current game, and will be stored to the phone's database once the game has been completed. This score determines which levels the student can play in the next game of this category.

One problem with the conceptual model is that it is somewhat complex. Once we have determined which levels we should pick questions from and we

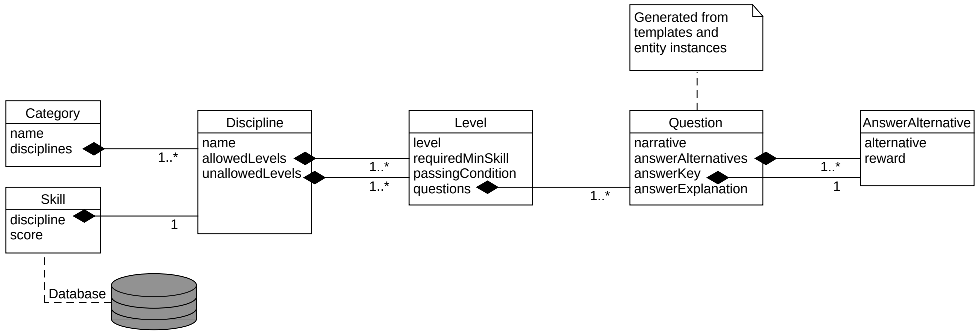

Figure 6.8: Conceptual model for the game elements

have generated the questions, we really don't need the structure. Especially when playing scenarios, it is nice to just go through an ordered array, instead of dealing with "now I've played the third question of level 2 diagnosis, the next question in this scenario is the third question in level 2 management, unless I've already completed level 2 management in the previous run of the game. Then the next question is follow-up instead". We rather deal with the problem at the initialization phase and just go through the array when playing the game. A solutions is to implement the façade pattern (Gamma et al. 1994), such that other parts of the system can use the game engine, without having to deal with the underlying complexity.

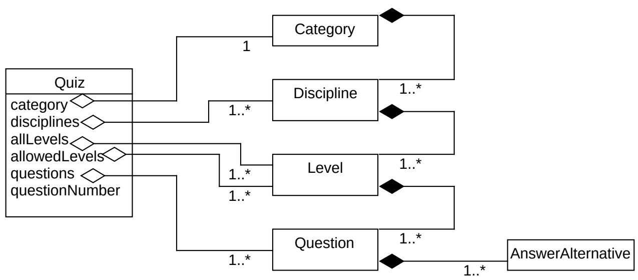

Figure 6.9: Façade pattern hides the complexity of the game engine and makes it easier for other parts of the system to use it

6.5 Summary

The chapter is structured after the conceptual training managers in the game engine. Question flow manager, conversation manager and the user manager.

The question flow manager uses the Dynamic Content Manager (DCM) to make the questions adaptable to the student's knowledge level and flexible such that the student can choose his own path through the learning content. We used the workflow model as a base, to categorize the learning material, and we identified knowledge dependencies between them, making a knowledge map. We saw how we could make three difficulty levels by adjusting the detail level of the questions, by going from factual questions, to scenarios and then detailed scenarios.

The conversation manager made textual questions by using instances of the entity model and templates. Tags in the template pointed to vertices in the entity model, such that we could produce many questions with one template, pairing it with different instances of the entity model. However, the entity model had to be expanded with presentation vertices as the data types couldn't be directly parsed into a textual presentation.

Further we argued why we wanted to use multiple-try questions with feedback. The student can try as many times as he wants to get the answer correct. The idea is that the student has to revise the question and reflect on why the question was wrong and how can he correct it. We also described the reward system, which will fit the multiple-try approach. A student with ten attempts should get a lower score than a student which answers correctly on the first attempt.

We then described a conceptual and technical model of how all the classes are connected in the game engine.

Chapter 7

Application Walkthrough

7.1 The mobile application

7.1.1 User interface and flow of the user interaction

We will now go through the application, by playing a level 2 scenario. We will discuss the game elements, the user interface, the technologies which are used. Whenever it is necessary to clarify we will also go deeper into the implementation.

The application is written in JavaScript, using the mobile application framework React Native (*React Native · A framework for building native apps using React*). The intention of using such a framework, is in the future to make iOS and UWP versions, without having to rewrite the entire application. As React Native uses the React web framework (*React – A JavaScript library for building user interfaces*), JavaScript, and that we can decouple some of the data management logic through Redux (*Redux · A Predictable State Container for JS Apps*), allows us to reuse a lot of the code if we later want to make a web version at a later point.

When the student first starts the application, he will presented with the screen in figure 7.1. The Game Engine will keep track of which quizzes we have installed, and displays them here. At a later point we can make knowledge dependencies, where a user have to complete basic courses like prescribing oxygen (Republic of Kenya 2016) and antibiotics administration (Republic of Kenya 2016), before the paediatric possible asthma guideline (Republic of Kenya 2016) can be played. This is theory from Dynamic Content Management (Eide, Kristensen, and Lamo 2008).

In 7.2 we get presented with the learning map from Dynamic Content Management (Eide, Kristensen, and Lamo 2008), and the student's position

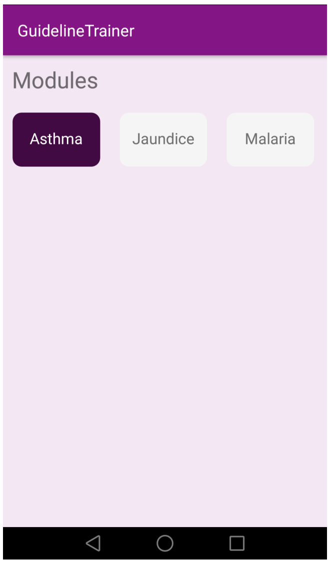

Figure 7.1: The entry screen where the student can choose between playing different quiz categories

in the learning map. The boxes with the green background indicates the student's current position and which levels he will get questions from. The other background colours indicates which levels have been completed and which levels are uncompleted and locked.

The student's position in the learning map is fetched from the database on the student's phone, and gets updated every time the student completes a quiz for this category. The database calls are asynchronously and loosely coupled using redux thunk (*reduxjs/redux-thunk: Thunk middleware for Redux*), such that the web version would replace the database related functions with fetch related functions to for example do REST calls to a REST service.

In the header we see a back arrow. This is React Native Navigation (*Home - React Native Navigation - truly native navigation for iOS and Android*) which handles the navigation in the application, which makes it possible to click the arrow to go back and choose another quiz to play instead. This fulfils "Provide Clearly Marked Exits", which is one of the usability heuristics for user interface design given by Jakob Nielsen and Rolv Molich (Molich and Nielsen 1990).

In 7.3 we are presented with a multiple choice question. The question is in the form of a scenario, where we go through the workflow model, where the first step is the assessment.

The Game Engine holds a template and a reference to the entity instance graph which is being used with this template. The template contains tags

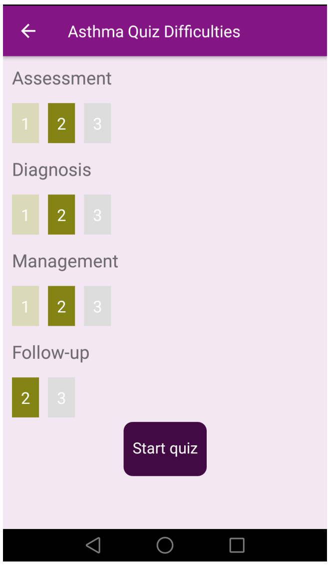

Figure 7.2: The student is shown his position in the learning map. Completed, active and locked levels have all different colours

which refers to vertices in the entity graph. The entity instance graph holds textual presentation of each vertex which is being printed into the template, which results in a neat and coherent text.

The Game Engine an answer key, which is a tag pointing to one or more vertices in the entity instances. The Game Engine also holds hard coded distractions, where one of them matches the vertex/vertices which the answer key points to. Points and penalties are specified for each of the distractions in the Game Engine.

Here the student answers correctly, is awarded points and is being presented with an answer key explanation. The Game Engine holds the answer key explanation together with the question.

The student can click on "next" any time he is ready to proceed to the next question.

In figure 7.4 we continue to the next step in the workflow model, the diagnosis. Here the student will be provided with symptoms which will determine the severity of the asthma.

In figure 7.5, the student gets a question from the management part of the workflow model. Management can be admitting the patient to the hospital, medication administration or advise.

In figure 7.6, the patient has been through the initial treatment, and the student is asked on how to follow up the patient. Here the textbox in the screenshot is a bit small for the text, but the student has the possibility to

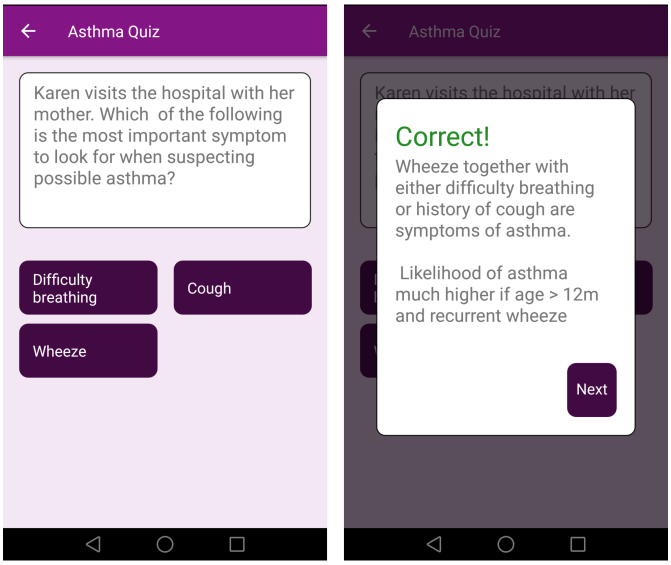

Figure 7.3: The first question is an assessment question. Here answered correctly and an answer key explanation is given

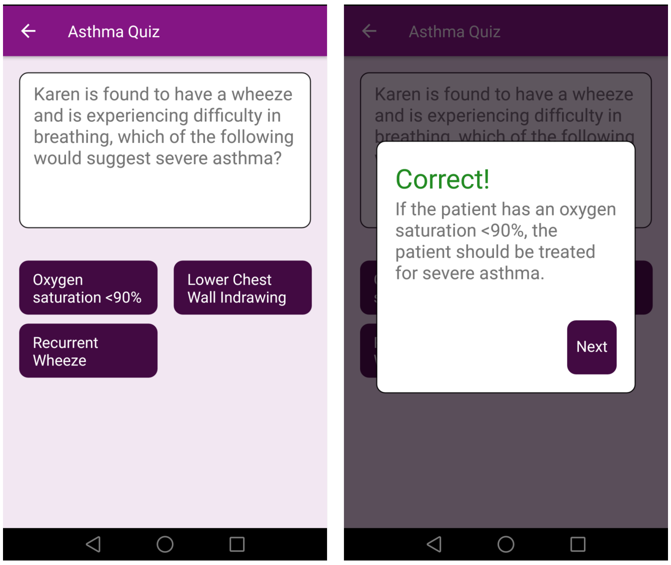

Figure 7.4: The second question is for diagnosis

scroll inside the textbox. The question is what is the maximum of days the patient can be on prednisolone.

Initially the student picks the wrong answer. The application displays a hint and a feedback, telling the student that this answer was wrong. The student is awarded with a penalty in points. As no further information is

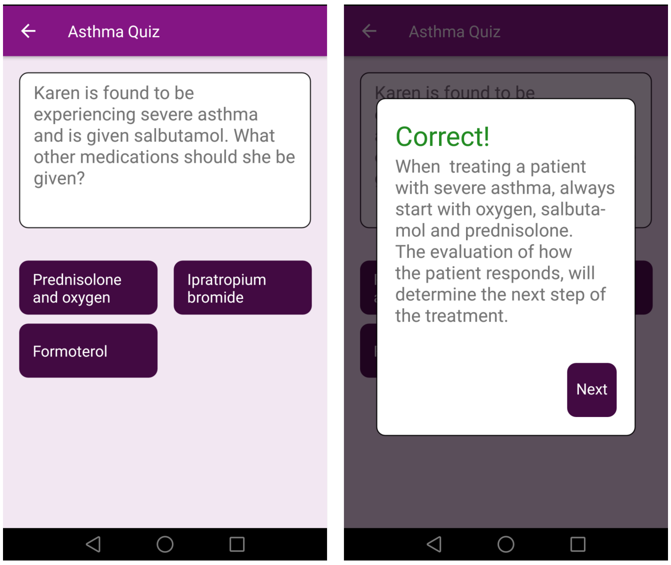

Figure 7.5: Third question is for management

given rather than that the previous answer was wrong, the student is given the possibility to try again. If the student tries again, a wrong answer will give a further penalty. We have made sure that the penalty is much smaller than the reward, typically 10-20% of the reward, trying to motivate the student to revise his answer. The student can try as many times that he would like, to get the answer correctly and collect the reward.

As we don't want to force the student to retry until correct, we have the option "learn more". When the student clicks on this button, he will be presented with the answer key explanation as well as the option to proceed to the next question. When the student decides to click on "learn more", he will miss out on the reward.

When having answered follow-up, which is the last question in the workflow model and the scenario, the student will be presented with his updated position in the student map in figure 7.7. Brown means previously unlocked levels. Grey means levels which are still locked and green means levels that have been unlocked after this game. If the student performs poorly, red boxes will appear. These boxes indicate that the current level has been locked, and that the student needs to complete quizzes at lower levels to repeat some of the more basic guideline content.

The last screen in this game sequence is shown in figure 7.8. The blue bars show our scores for the questions categorized according to the workflow model. The scores are for the current game we just played. The red lines show the scores needed to complete each category for the current level.

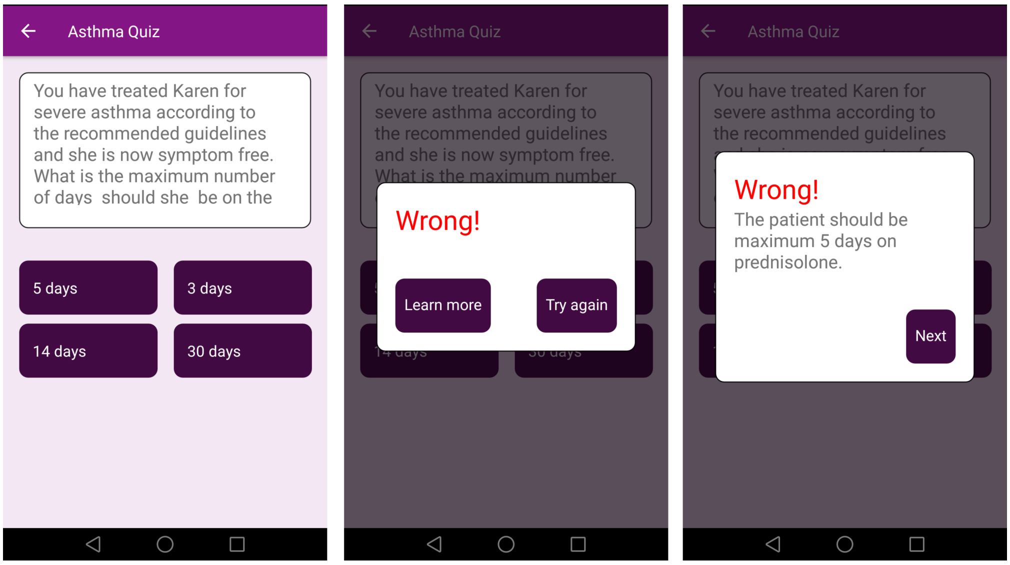

Figure 7.6: The fourth question is follow up. We answer incorrectly the first time. The answer key explanation is not shown, and we can choose to revise the question by clicking "try again". If we press "learn more", the answer key explanation is instead shown, and we continue to the next question

From the learning map and graph, we see that we completed the current level of assessment, diagnosis and management. We didn't complete follow-up, so the next time we play the asthma quiz we will only get questions from level 2 follow-up. The reason why the student won't get questions from assessment, diagnosis and management, is that we don't want to bore the student with continually asking the same questions that we know the student already knows the answers of.

The idea of the chart is to make the student motivated, when he sees his progress getting closer and closer to the red requirement line for every time he plays the asthma quiz at that level. Confirming that he learns more every time.

We used Highcharts (*Interactive JavaScript charts for your webpage — Highcharts*) to display the interactive chart of the game scores and the scores required to complete each category.

7.2 Summary

In this chapter we go through the mobile application by playing a game at level 2. We discuss the technologies which have been used, the user interface, intended user experience, the scoring system and some design choices.

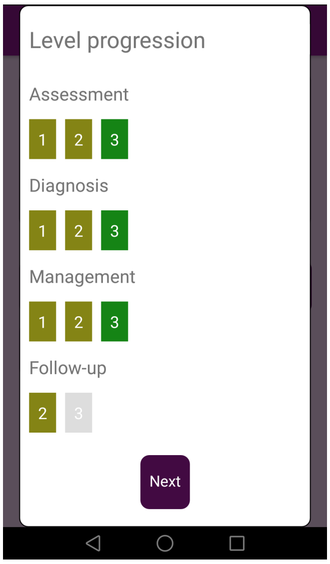

Figure 7.7: After completing the quiz, we are shown the student's updated position in the learning map. The brown background shows levels which has been completed. Grey is the levels which are still unlocked. Green is that we have advanced to the next level. A red box would have indicated that we had performed poorly, the level would have been locked and we have to complete an easier level before we can play it again

We show the student's performance in the current game with a chart. The chart represent the student's scores in different sections of the clinical encounter. A red line indicates the passing conditions at that difficulty level.

We see how the application adapts the difficulty to the knowledge progression of the student, by updating the student's position in the learning map.

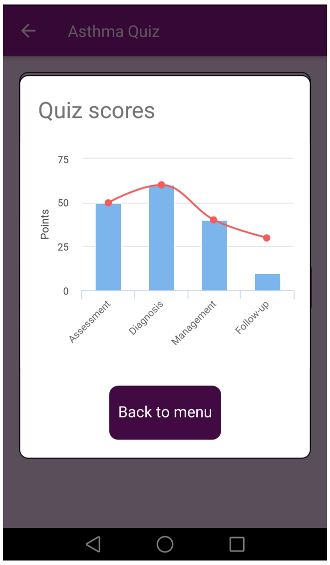

Figure 7.8: The bars show our scores for the played game in the different parts of the clinical encounter. The red line shows the scores needed to complete this level

Chapter 8

Evaluation

8.1 Research questions

- **RQ1:** Based on clinical guidelines, how can we define and represent a generic data structure that can be used to implement applications such as online guidelines or training games for such guidelines, and where applications can adapt to the level of their users?
- **RQ2:** Can the generic data structure in RQ1 be used to generate a specific data model for another domain such as paediatric asthma?
- **RQ3:** How can we use the data model in RQ2 to implement a game for guideline training that can adapt to the level and progression of users?
- **RQ4:** Is the guideline meta model at an abstraction level such that it can be used for other guidelines?
- **RQ5:**

8.2 Evaluations

8.2.1 Evaluation of the models

To evaluate our model, we have decided to model the paediatric pneumonia guideline (Republic of Kenya 2016). As we have a strict time limitation it is better for us to evaluate a respiratory condition, as we have already worked with a respiratory condition in asthma. There is quite a lot of work for software developers to learn a new clinical guideline, so we save a lot of time when the concepts are similar.

In figure 8.1 we have modelled the excerpt of the entity model of the paediatric pneumonia guideline (Republic of Kenya 2016). We have removed unnecessary vertices such as Investigation and Surgery, as neither the pneumonia guideline nor the possible asthma guideline use them. On the Diagnosis vertex we now have a new inheritance. We'll soon explain.

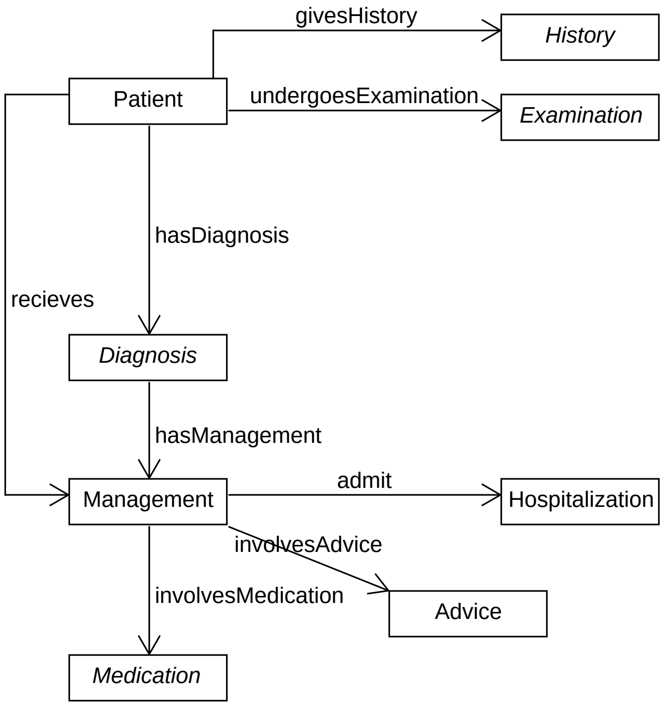

Figure 8.1: Only an inheritance on the Diagnosis vertex makes the excerpt of pneumonia guideline (Republic of Kenya 2016) different from the original excerpt in figure 5.2

We see that both pneumonia and possible asthma guidelines have a history part, figure 8.2, where the patient or dependants can tell something about the condition of the patient. Both guidelines also have an examination part, figure 8.3 where the clinician looks for several symptoms. The symptoms are a bit different from paediatric possible asthma guideline, but they both have several symptoms in common. Wheeze is something the clinician has to be aware about. If the patient is wheezing, the patient should be treated according to the paediatric possible asthma guideline instead.

The big difference is the diagnostic part, figure 8.4. For the paediatric possible asthma guideline (Republic of Kenya 2016), there is only one medical condition described. However, for the paediatric pneumonia guideline

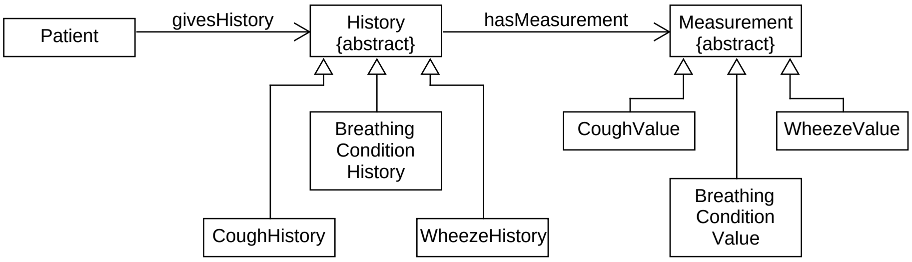

Figure 8.2: The history part of the entity model of paediatric pneumonia guideline (Republic of Kenya 2016) is identical to figure 5.4

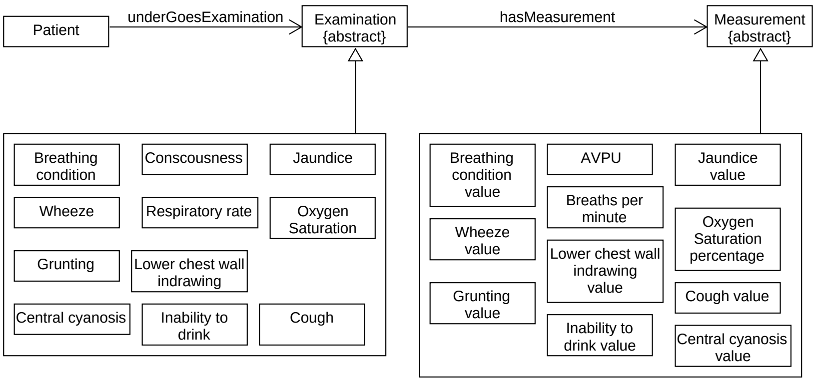

Figure 8.3: The symptoms need to be adapted to the paediatric pneumonia guideline (Republic of Kenya 2016) when modelling the examination

(Republic of Kenya 2016) patients with tuberculosis or HIV will receive a different treatment. We have not modelled the treatment for pneumonia patients with HIV or tuberculosis as they are separate guidelines. But we need to identify such patients and refer to their respective guidelines. The same goes for asthma. Wheezing patients needs to be identified for asthma treatment.

To support several conditions, we have used inheritance on the diagnosis vertex. A new problem occurs as how should we model tuberculosis, no tuberculosis and that we don't know if the patient has tuberculosis? Earlier on, we used the open world principle for the symptoms. If the vertex doesn't exist, we haven't done the examination for the symptom and we don't know if the patient has it or not. The same goes for diagnosis. If the vertex is not there, we need to clarify if the patient has tuberculosis. To model the situation where we know that the patient hasn't tuberculosis, we have introduced the diagnosis ”no tuberculosis”. An alternative solution could be to introduce an attribute ”status”. The attribute could hold information about the patient evidently has the condition, evidently don’t have the condition and if it is not clarified. A fourth status could be if the condition is recurring.

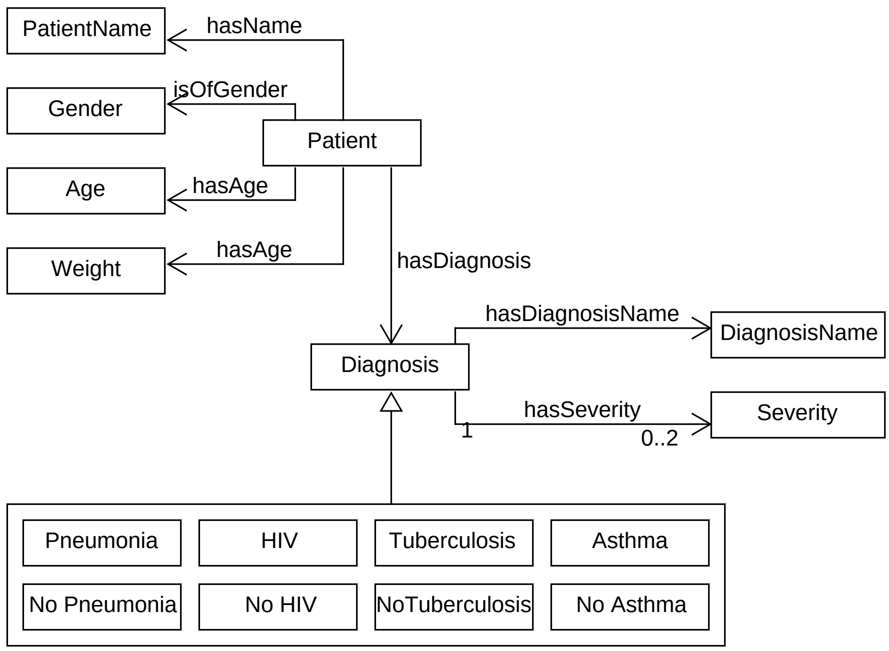

Figure 8.4: When modelling the Diagnosis of the paediatric pneumonia guideline (Republic of Kenya 2016), we need to use inheritance to support differential diagnoses and different treatments when the patient has combinations of diagnoses

The management part is also quite similar to the paediatric possible asthma guideline (Republic of Kenya 2016). The patient and dependants will be given some advise concerning the medical condition of the patient. If the pneumonia is severe, or the patient has lower chest wall indrawing and the patient cannot be reviewed within 48 hours, the patient should be admitted into the hospital. The medication, figure 8.5, is quite similar to the paediatric possible asthma guideline. For pneumonia there are fewer medications, but both guidelines have treatment with antibiotics and oxygen.

To evaluate the workflow model in figure 5.9, we do like we did for the paediatric possible asthma guideline (Republic of Kenya 2016). We see the guideline in combination with the entity entity and the workflow model. The paediatric possible asthma guideline (Republic of Kenya 2016) has an assess-

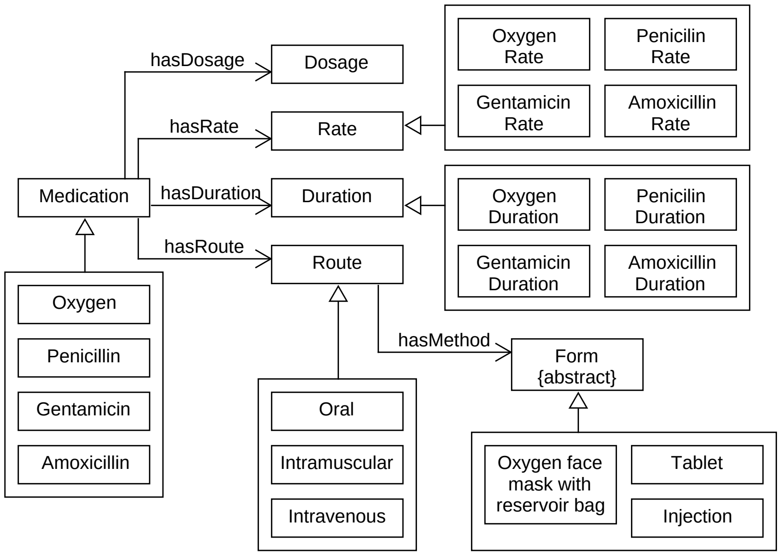

Figure 8.5: The paediatric pneumonia guideline (Republic of Kenya 2016) uses oxygen and antibiotic treatment, which can be modelled in the same way as figure 5.6

ment part, where we look for patient older than 60 days, cough or difficulty breathing. If he has those symptoms and does not wheeze, we continue looking for other pneumonia symptoms. In the diagnostic part we strengthen our assumption of pneumonia, we set the severity of the diagnosis and we further keep in mind that if the the patient is wheezing he should be given the asthma treatment. Pneumonia patients with HIV or tuberculosis should be referred to specific guidelines for those condition combinations (Republic of Kenya 2016). In the management part, some patients get admitted into the hospital. The management further has an advise for review within 48 hours. The patients condition is then evaluated either on the hospital for admitted patients, or in a review within 48 hours for other patients. We can conclude with that the workflow model also covers the paediatric pneumonia guideline (Republic of Kenya 2016).

In figure 8.6 we demonstrate an instance of the entity model working together with an instance of the workflow model. The patient goes through assessment, diagnosis, management and evaluation. As the patient has jaundice, he won't receive penicillin treatment. As the the pneumonia is severe, the treatment will be evaluated at the hospital and schedule for review is unnecessary.

After the evaluation, we see that we have modified the entity and work-

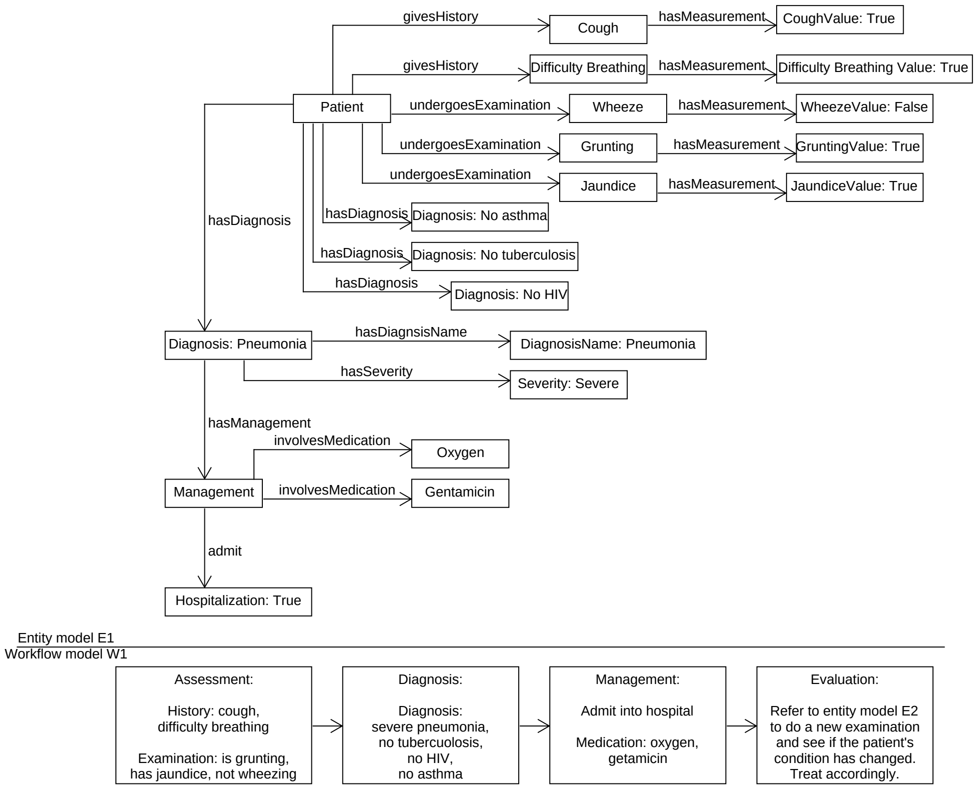

Figure 8.6: We are showing out entity model working together with the workflow model for paediatric pneumonia guideline (Republic of Kenya 2016)

flow models to support both pneumonia and asthma in paediatric medicine. It is likely that further modification and expansion of the entity model is needed when modelling other CPGs. However, we see that we have identified reusable elements of the guidelines, and our models can work as a stepping stone for guideline formalization standardisation.

8.2.2 Evaluation of the application

The evaluation of the application was done with clinicians. Two medical doctors and two specialist nurses. Both of the specialist nurses are employees at the polyclinic for pulmonary diseases at Haukeland University. One of the nurses is a specialist in sleep apnea, and his master thesis was writing clinical guidelines for sleep apnea. The other nurse is a specialist in asthma, but in adult medicine. The two doctors are educated as general practitioner, but are now working in research.

The evaluation methods were a combination of the cognitive walkthrough and usability test in controlled environment, with follow-up questions. In detail what we did was, the nurses would be asked to play the most difficult level of the game, speak what they are thinking when playing the game and manoeuvring in the application. The two medical doctors would play the entire game, from the easiest level and to completing the most difficult one. By playing all the levels, the doctors would to far greater extent evaluate the learning model.

Topics to discuss would occur when the clinicians had to think out loud. The master student would observe and take notes when problems and confusions occurred, or that the clinician expressed emotions such as joy, excitement or disappointment. After the clinicians had played through the game, the master student would go through a check-list of topics to discuss. The discussions would be in a semi-structured format, where the check-list worked as a guide. The discussion with the two nurses would be individually, while the discussion with the two doctors was done in a small focus group, consisting of the two medical doctors and the master student.

1. Can the application be a useful learning tool for medical students, nurses and doctors?

    - **Nurse1:** Very useful indeed. Would be nice to take a test after a lecture about asthma or after having read about asthma to see how much I have learnt and remember. A quiz is far more fun than a check list in paper format. The application is also good for scalability, as you can train a lot of clinicians without adding any resources. Also great if a course leader can see the progress or the level of his students.
    - **Nurse2:** Absolutely useful, and I feel I have learnt a lot by just doing this quiz. The nurse found the game to be very engaging, cheering when getting a correct answer.
    - **Doctors:** For medical doctors, the quiz will be too basic. For nurse it might be good. For medical students it will be very good, as it fits with how the students works and how they will be tested for exams.

2. How is the flow of the questions? Is the idea of scenarios where we go from assessment, diagnosis, management and follow-up a good approach?

    - **Nurse1:** Happy with the float and the use of scenarios.

- **Nurse2:** Very happy with the float, being able to follow the patient from the start to the end of the treatment.
- **Doctors:** The categories weren't very clear. The questions are floating into each other. One suggestion is to have oxygen and antibiotic administration as own categories. Then you can measure how well they perform in these categories and ask them to repeat the basics if they perform poorly.

3. Did we manage to present the important elements of the asthma guideline?

    - None of the clinicians know the paediatric possible asthma guideline (Republic of Kenya 2016) well enough to answer that question.
    - Doctors: in Norway they usually look at even more parameter. Does the patient smoke? Does he have allergies? The use of CRP to measure the inflammation levels in the body, are just some of the parameters a clinician in Norway would look at.

4. Is the detail level the element to adjust for the difficulties of questions?

    - **Nurse1:** Yes, but would like to have an even harder level with more details.
    - **Nurse2:** Yes, it seems like a right approach. The target group of users is relevant here, that this is meant for the emergency clinic.
    - Doctors: Yes, but the detail level of the questions need to be much harder. One example of going to higher detail level could be "what oxygen administration device would you initially use to a neonate?" to "administering oxygen using nasal prong to a neonate doesn't work. What do you do?".

      In Norway, the patients will visit the hospital with a lot more variation of illnesses and with a higher frequency of less severe diagnoses. Then differential diagnoses gets more important and to represent a lot more clinical conditions as quizzes. The clinicians also work a bit different in Norway. If a patient comes into the hospital with symptoms of severe asthma, they will usually just treat and stabilize the really alarming symptoms and not go through a whole list of treatments.

5. How are the answer key explanations?

    -

- **Nurse2:** I like how the measurements corresponds and are calculated with the scenario and the patient they are presented with. The answer key explanations gives relevant answers to the questions asked.
- **Doctors:** The answer key explanations are good. We like that we get an explanation when we answer correctly. We preferred to try until getting the answer correctly rather than clicking "learn more" and proceed to next question. It could be nice to get an explanation why the answer was wrong, but we are rather impatient, we want to proceed and find the correct answer quickly.

The quality of game questions:

1. The assessment, none of the distractions is a wrong answer. Even though wheeze is what we are looking for.

2. When asking for salbutamol dosage, it is relevant to know where and in which stage of the treatment it is being given. Also saying something about rate or duration can help clarify that.

3. Very nice that we in some questions change the way we ask. Sometimes we give a diagnosis and asks which symptoms identifies it. Then we give some symptoms and then ask about the diagnosis.

4. In the assessment part of the scenarios, we tell that clinical encounter happens at the emergency clinic. But we should find a way to amplify it even more, as one of the test persons missed that detail.

5. Descriptions such as "age ¿ 12 months" is hard for clinicians to read. We should avoid using logical operands.

6. "Recurrence of asthma" is unclear. Who says that it is recurrent? Is it recurrent when the patient is in the emergency clinic? Does the patient tell that it is recurrent? Is it in the journal? All the clinicians had commented on the use of recurrence.

7. We have a trick question and we should avoid those. We ask what is not the right way to administer salbutamol to a patient. The answer is oral. The clinician thought the right answer could be one of the other alternatives as an oral version of salbutamol doesn't exist.

8. We are using history i an imprecise way. History is not only what the patient or the caregiver tells. The patient can have been visited the

hospital for that symptom earlier or have had it in an earlier point of time. It may be something from their journal. More detail where and when the history comes from is needed.

User interface and game experience

1. All the clinicians were very positive to the multiple-try with hint approach. When they give a wrong answer, they get a new chance to revise the question and correct their answer.

2. When the clinicians answer a question incorrectly, they are presented with a "learn more" button. When they click on this button they get presented with an answer key explanation. When they click on "next" the nurses get very surprised and disappointed that were taken to the next question and don't get the chance to get the question right. The medical doctors were fine with that approach.

   Here the specialist in asthma had a very interesting suggestion. When she clicks on "learn more", she want an explanation of why that suggestion was wrong or in which situations that answer would have been correct. Then she want to try again to collect the reward. She was noticeably disappointed at the summary when she saw that she was penalized hard for clicking "learn more".

3. The clinicians wanted a link or a button to display the clinical guidelines in the screen they were presented with the answer key explanation. Then they can read and learn more.

4. The clinicians noticed that they didn't have to remember details from the previous question before they answered the next. That was something they appreciated.

5. The two doctors didn't think text questions were the best approach. They wanted pictures, sound and perhaps short film clips, and the text could come as a supplement. A picture with a blue face just lying in the bed, would be a severe case of asthma indeed, by just observing the picture. A child sitting and playing while having difficulty breathing could be a case of mild or moderate asthma. A sound clip of an asthma patient wheezing or the gurgling sound of a breathing pneumonia patient could also be helpful. A patient sitting in the bed, getting oxygen through the nose, would say something about the treatment given and the evaluation of the patient. Perhaps a film clip where the clinician tries to talk to the patient. The use of film, pictures and sound would

provide more realistic scenarios to what the clinicians will meet at the hospital.

6. When having met the passing condition of one of assessment, diagnosis, management and follow-up, it is a good thing that the passed category gets excluded from the quiz when replaying the level. This makes the game less tedious.

7. In the situation where the user performs very poorly and gets demoted from a difficulty level, an explanation should be shown.

8. The two doctors meant the "wrong" hint in red when answered a question incorrectly was ok. If sugar coating the feedback, the satisfaction of getting a green "correct" would have been dampened.

8.2.3 Results

Here we make a brief summary of the results from the evaluation. We want to verify that we have data models at an abstraction level such that they can be used to represent other clinical practice guidelines. The models will be our contribution to the science of health informatics. We want to ensure that our serious game is a valuable contribution to the medical community. Contribution to science and to the community are requirements from design science.

Evaluation of the models We have shown by a practical example how the entity model can be used to represent the paediatric guideline of pneumonia (Republic of Kenya 2016). A requirement was identified and met by making an abstraction of the Diagnosis in the entity model. This was necessary to support differential diagnosis, and to support treatments which are dependent on which other diagnoses the patient may have.

We showed how we could make instantiations of the entity and workflow models and how they work together to represent a full clinical encounter for patient suffering from pneumonia.

Evaluation of the application In table 8.1, we have made a brief summary, where we quantify the evaluation of the game application. The

8.3 Findings

Here we will relate our findings to the research questions.

Questions	Positives	Negatives
Can the game be a useful learning tool for		
Medical students?	2	0
Nurses?	2	0
Doctors		2
How is the flow of the questions in scenarios	2	2
Is adjusting the detail level to make questions more difficulties the right approach?	4	0
Is the quality and approach answer key explanations correct?	3	
Number of feedbacks about the quality of quiz questions	1	7
User experience		
Wants the possibility to read the guideline in the game	4	0
Didn't have to remember details from previous screen	3	0
Textual questions instead of multimedia		2
Game elements		
Multiple-try feedback	4	0
"Learn more" doesn't give a last attempt to get the question right	2	2
Learning model	2	0
Don't repeat questions from completed quiz categories	2	

Table 8.1

RQ1: Based on clinical guidelines, how can we define and represent a generic data structure that can be used to implement applications such as online guidelines or training games for such guidelines, and where applications can adapt to the level of their users? For this research question, we have proposed four data models. The first two data models are a domain (entity) model 5.8 and a guideline (workflow) model 5.9. The domain model represents a patient, his symptoms, medical condition and what the clinicians do to mange his medical condition. The guideline model represents the steps in a clinical encounter. When the domain model and the guideline model work together, we can represent the patient, symptoms, diagnosis and management of the medical condition for every step through the clinical encounter. The model allows us to make online guidelines or training games with specific patients examples through the whole guideline.

We also proposed a student learning model and a game model. By using Dynamic Content Management (Eide, Kristensen, and Lamo 2008), we showed how we can split the learning content into content units. Each of these content units contain learning resources and evaluations at different difficulty levels. In that sense, the learning content can be adapted to knowledge of the student. Presenting difficult evaluations in areas where the student is strong, and easier evaluations where the student is weak.

RQ2: Can the generic data structure in RQ1 be used to generate a specific data model for another domain such as paediatric asthma? In figure 5.2, 5.4, 5.5, 5.3 and 5.6, we showed a specific entity model for the paediatric guideline of possible asthma (Republic of Kenya 2016).

In figure 5.11, we show an instantiation of the specific entity model, working together with an instantiation of the workflow model. This shows that also the workflow model can be used for a specific domain.

For the student learning model, game model, entity model and workflow model, we also did a proof of concept. We developed a serious game which tested students in the paediatric guideline of possible asthma (Republic of Kenya 2016). Then the student learning model and the game model got applied for a specific domain, in this case paediatric possible asthma.

RQ3: How can we use the data model in RQ2 to implement a game for guideline training that can adapt to the level and progression of users? For the student learning model and the game model, we have applied the Dynamic Content Management (Eide, Kristensen, and Lamo 2008). We have split the learning content into content units to be able to adapt the learning content to the knowledge level of the student. Then we identify knowledge dependencies between the content units. When the student progresses, we can provide more difficult questions, given that the knowledge dependencies are followed. This ensures that the student has the basic understanding in relevant subjects to progress.

We make the difficulty level tougher by adding more details to the questions. Going from factual statements, to scenarios to detailed scenarios.

The adaption to the level and progression of users were demonstrated in the serious game.

RQ4: Is the guideline meta model at an abstraction level such that it can be used for other guidelines? In figure 8.1, 8.2, 8.3, 8.4 and 8.5, we demonstrate how the entity model can be used to to represent the paediatric guideline of pneumonia (Republic of Kenya 2016). However, the model modifications to be able to support differential diagnoses and the situation where the treatment depends on which other diagnoses the patient has.

In figure 8.6 we show an instance of the specific entity model working with an instantiation of the workflow model. This shows that the workflow model is at an abstraction level such that it can be used for other guidelines.

The student learning model and game model divides the learning content into content units based on the workflow model. When the workflow model is abstract enough to be used the paediatric guideline of pneumonia (Republic of Kenya 2016), then the student learning model and the game model would be as well.

8.3.1 Limitations of the entity model

- We can't ask questions like "what are the symptoms for severe asthma?". The entity model is a model of one specific patient at one stage in the clinical encounter. To have severe asthma he may only have one of the symptoms for severe asthma and not all of them.

- It is difficult to ask what NOT to do. If the vertex doesn't exist, only an empty string is returned. "NOT to do" can only be used were we actually have specified that the action should not be done. An example is "don't admit to the hospital" where we have specified "no" at the Hospitalization vertex.

- The inheritance makes it difficult to generalize some questions. We can't make a template which asks about the Rate a (any) medicine should be taken with. We need to specifically ask for that medicine. To be able to ask for a general medicine, one solution can be to introduce a new tag which compares the substring of the type of the vertex. Another solution is to use the meta model and not the instance model. In future work we can study GraphQL to see if it can be used as another alternative to solve this problem.

8.4 Summary

In this chapter, we have evaluated the entity and workflow models by showing that they can be used to represent other guidelines, such as the paediatric pneumonia guideline (Republic of Kenya 2016).

We have evaluated the serious game with two nurses and two medical doctors to ensure that we are delivering value to the medical community.

We have discussed the findings in relation to the research questions, and we have identified some of the limitations of the entity model.

Chapter 9

Discussions and conclusion

9.1 Future work

9.1.1 Automatically generating new questions

One of the main problems with the application is generating entity model instances used for generating questions. To make one scenario with assessment, diagnosis, management and follow-up, we need at least two instances of the entity graph. One to show the status of the patient before the treatment, and one after the treatment. As the treatment needs to be evaluated and the clinician needs to act accordingly. One single entity model can be several hundreds of lines written in JSON, as there are quite many vertices and edges to describe, as well as the complexity of the model. This is tedious work, both because of the complexity, the amount of code lines, but the author also needs to put a great emphasize on writing the code correctly. The latter can be solved with an authoring tool.

As so much effort has to be put in typing these instances of entity graphs, a better way would be to make functionality to automatically generate new instances. We already have made templates with tags to vertices in the entity graph. By generating a lot of new entity instances, one template can be reused to make just as many questions. By looking at the paediatric possible asthma guideline (Republic of Kenya 2016), we see that there are described twelve symptoms we can combine in every way we like. Some of the symptoms also have more possibilities than just true/false. They have numerical ranges like pulse rate, breaths per minute and oxygen saturation. So in fact, we can generate a quite large amount of graph instances and equally as many questions per template.

The method we suggest for automatically generating such graph instances, is by introducing constraint checks in DPF. These constraint checks will be code which ensures that the vertices of the graph follows some predefined rules. For example, we know that the graph stores a boolean measurement value for wheeze. A constraint check will ensure that the value we generate for wheeze will be either true or false. For conciousness we have the enumeration values A, V, P and U. And we have value ranges, such as oxygen saturation and pulse rate. To get reasonable constraints, a domain expert is needed to define the upper and lower limits a constraint should allow. For example a pulse rate for an old man shouldn't be set to 200 beats per minute, as his maximum pulse rate would be much lower than that.

However, some vertices are dependent on other vertices. For the paediatric guideline of possible asthma (Republic of Kenya 2016), wheeze will always have to be present to set an asthma diagnosis. For children less than 12 months, either or both of cough and difficulty breathing needs to be present in addition to wheeze. For the asthma to be severe, there are at least one additional symptom which needs to be present.

When the student will be asked for medication dosages, we know that some drugs may have standard measurements. For example, a medication we want to administer to patient, is 2.5mg per tablet. Then the dosage administered to the patient will be something which can be divided by 2.5. Here the dosages can be 2.5mg, 5mg, 7.5mg and so on.

Some of these constraints can be reused between guidelines. The same medication can be used for different diseases, as well as a person's maximum pulse rate probably will be the same whether he he has asthma or pneumonia. The AVPU-scale is a standard for measuring consciousness. However it can be tedious to get full constraint coverage, and when the guideline gets updated, we also need to update the constraints. In cases of not full constraint coverage, the entity instance graphs need to be validated by a domain expert, to see which of the generated instance graphs which can be used or not. Even though this method relies on a domain expert, it will save us from a ton of work. Instead of carefully write hundreds of JSON formatted lines for each graph instance, we can simply approve or disapprove the graph instances which are valid or invalid. The constraints will also greatly reduce the noise the domain expert needs to go through.

When we have built up a repository of entity graph instances, we can use this repository to generate distractions for quiz elements. Randomly choose distractions from other entity graph instances, which corresponds to the answer key tag given by the template. A small penalty can be set on distractions from the asthma guideline. Medium penalties for distractions from respiratory diseases. A bit larger penalties for distractions from completely unrelated guidelines. Then we have automated the quiz scores as well!

9.1.2 Dissemination of guidelines

At one point we want to have a game, where the student can choose between many different quizzes where he will be tested in different guidelines for different medical conditions. These guidelines has to be managed from somewhere. New questions arrives: should all the guidelines be packed with the game application itself? Or should the student install quizzes for just the guidelines he is interested in? When there is a change in a clinical practice guideline. How do we make sure that the clinician trains with the most recent quiz version?

Our suggestion is that the game comes with a predefined set of quizzes for the most used guidelines. Then the clinician can install quizzes or categories of quizzes which he is the most interested in. The application automatically checks for new versions when the application starts. The quizzes needs to have a version tag, such that version checking can be possible.

9.1.3 Scalability

How do we train, 10, 100, 1000 clinicians? Our solution so far is that the clinicians are responsible and in charge of their own training. Another scenario can be a teacher or a course leader having many students. The application can track the student's knowledge level in quizzes of interest and the progression of the student. The student can submit their data to a central server, where individual and collective skill and progress of the students can be visualized through chart and statistics for the teacher. In this way he can better target students which is in need for individual help, or subjects where the class is weak and should be addressed by the teacher.

9.1.4 Multimedia as part of the quiz questions

In the evaluation of the application, we saw that some of the clinicians wanted pictures and sound as part of the questions. Pictures, sound and video is far more similar the situations the clinicians meet at the hospital. Where they need to look, listen and observe the patient, rather than having the examination results handed to them in a text format.

One way to integrate multimedia into the questions, is by meta tagging the important observations done in the videos, pictures and sound. Then we can build an instance of a specific entity model for that video, picture or a sound.

For example: a sound clip of a wheezing sound would have a graph is modelled in figure 9.1. By subtracting this graph from the one used for textual questions, we don't get wheeze as a part of the textual question. The student needs to listen and analysing the sound to understand that the patient is wheezing.

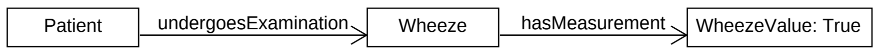

Figure 9.1: An entity instance of the wheezing sound

We can use this method in combination with other types of pictures. A patient laying in the bed with a facial mask, tells that this is a patient which has been treated with oxygen. As the patient is not alert, the patient is probably still having severe asthma. This is information which can be removed from the textual question when we have a graph which tells that this can be seen in the picture.

From a picture it is also easy to see if the patient is either a child, an adult and where he is located. A picture of a child playing with his toys could be mild or moderate asthma as he is alert. A patient which doesn't respond to the questions from the doctor, could be severe asthma as he is showing an inability to talk because of the breathing difficulties.

9.2 Conclusion

Well defined clinical practice guidelines have shown the effect of improving the quality of health care at a lower cost, as well as reducing practice variability. Despite the positive effects of clinical practice guideline, they have shown a limited effect on changing the clinicians practice methods. Our contribution to the medical community is a serious game which addresses some of the reasons why clinical practice guidelines haven't been put into more use. By playing the game, clinicians will be aware of the paediatric guidelines of possible asthma (Republic of Kenya 2016), they will get familiar with the guideline content, they may gain the self-confidence to execute the recommended practices of the guideline, well defined scenarios may help the clinicians overcome inertia of previous practices, and a serious game may overcome some of the external barriers such as cumbersome format for some of the clinicians. Our evaluation verifies that our contribution is valuable to the medical community.

We also proposed four data models. A guideline (workflow) model which describes the workflow of a clinical encounter. A domain (entity) model which describes a patient, the symptoms, the diagnosis and how the clinicians manages the patient for his medical condition. A student learning model which keeps track of the student's performance at different quizzes. A game model which holds information about game elements and the ordering of the game/learning material. By using information from the game model and the student learning model, we can make the game adaptable to the knowledge and progression of the student. It can also be made flexible such that the student can choose different paths through the learning material.

Through evaluation we saw that the our domain (entity) model needed to be adapted to support differential diagnoses, and multiple diagnoses as the treatment can depend on which other diagnoses the patient may have. This indicate that our work may be a stepping stone. That the domain (entity) model need further adoptions the more we learn about the requirements from other clinical practice guidelines.

Bibliography

Attali, Yigal (2015). “Effects of multiple-try feedback and question type during mathematics problem solving on performance in similar problems”. In: *Computers & Education* 86, pp. 260–267. ISSN: 0360-1315. DOI: 10.1016/J.COMPEDU.2015.08.011. URL: https://www.sciencedirect.com/science/article/pii/S0360131515300312.

Bergman, Michael K. (2018). “Keeping the Design Open”. In: *A Knowledge Representation Practionary*. Cham: Springer International Publishing, pp. 183–205. DOI: 10.1007/978-3-319-98092-8_9. URL: http://link.springer.com/10.1007/978-3-319-98092-8{_}9.

Boxwala, Aziz A et al. (2004). “GLIF3: a representation format for sharable computer-interpretable clinical practice guidelines”. In: *Journal of Biomedical Informatics* 37.3, pp. 147–161. ISSN: 1532-0464. DOI: 10.1016/J.JBI.2004.04.002. URL: https://www.sciencedirect.com/science/article/pii/S1532046404000334.

Brambilla, Marco, Jordi Cabot, and Manuel Wimmer (2017). *Model-Driven Software Engineering in Practice: Second Edition*. 2nd. Morgan & Claypool Publishers. ISBN: 1627057080, 9781627057080. URL: https://dl.acm.org/citation.cfm?id=3103551.

Cabana, Michael D. et al. (1999). “Why Don’t Physicians Follow Clinical Practice Guidelines?” In: *JAMA* 282.15, p. 1458. ISSN: 0098-7484. DOI: 10.1001/jama.282.15.1458. URL: http://jama.jamanetwork.com/article.aspx?doi=10.1001/jama.282.15.1458.

Carlsen, K. C. L. et al. (2006). “Asthma in every fifth child in Oslo, Norway: a 10-year follow up of a birth cohort study*”. In: *Allergy* 61.4, pp. 454–460. ISSN: 0105-4538. DOI: 10.1111/j.1398-9995.2005.00938.x. URL: http://doi.wiley.com/10.1111/j.1398-9995.2005.00938.x.

Clariana, Roy B. and Ravinder Koul (2006). “The effects of different forms of feedback on fuzzy and verbatim memory of science principles”. In: *British*

Journal of Educational Psychology 76.2, pp. 259–270. ISSN: 00070998. DOI: 10.1348/000709905X39134. URL: http://doi.wiley.com/10.1348/000709905X39134.

Clayton, P D and G Hripcsak (1995). "Decision support in healthcare." In: *International journal of bio-medical computing* 39.1, pp. 59–66. ISSN: 0020-7101. DOI: 10.1016/0020-7101(94)01080-K. URL: http://www.ncbi.nlm.nih.gov/pubmed/7601543.

De Clercq, Paul, Katharina Kaiser, and Arie Hasman (2008). "Computer-Interpretable Guideline formalisms." In: *Studies in health technology and informatics* 139, pp. 22–43. ISSN: 0926-9630. URL: https://www.researchgate.net/publication/23271685{_}From{_}guidelines{_}to{_}careflows{_}Modelling{_}and{_}supporting{_}complex{_}clinical{_}processes.

Disease, Division of Leprosy Tuberculosis et al. (2011). *Guidelines for Asthma Management in Kenya.* Division of Leprosy, Tuberculosis and Lung Disease. URL: http://apps.who.int/medicinedocs/documents/s21973en/s21973en.pdf.

Eide, Sigvat, Terje Kristensen, and Yngve Lamo (2008). "A model for dynamic content based e-learning systems". In: *Proceedings of the 2008 Euro American Conference on Telematics and Information Systems - EATIS '08.* New York, New York, USA: ACM Press, pp. 1–8. ISBN: 9781595939883. DOI: 10.1145/1621087.1621089. URL: http://portal.acm.org/citation.cfm?doid=1621087.1621089.

Fervers, B., J. Carretier, and A. Bataillard (2010). "Clinical practice guidelines". In: *Journal of Visceral Surgery* 147.6, e341–e349. ISSN: 1878-7886. DOI: 10.1016/J.JVISCSURG.2010.10.010. URL: https://www.sciencedirect.com/science/article/pii/S1878788610001542?via{\%}3Dihub.

Gamma, Erich et al. (1994). "Design patterns : elements of reusable object-oriented software". In: Pearson Education. Chap. Structural, p. 185. ISBN: 0321700694. URL: https://books.google.no/books/about/Design{_}Patterns.html?id=6oHuKQe3TjQC{\&}source=kp{_}cover{\&}redir{_}esc=y.

Guyatt, Gordon H et al. (2008). "GRADE: an emerging consensus on rating quality of evidence and strength of recommendations." In: *BMJ (Clinical research ed.)* 336.7650, pp. 924–6. ISSN: 1756-1833. DOI: 10.1136/bmj.39489.470347.AD. URL: http://www.ncbi.nlm.nih.gov/pubmed/18436948http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC2335261.

Hassenzahl, Marc and Noam Tractinsky (2006). "User experience - a research agenda". In: *Behaviour & Information Technology* 25.2, pp. 91–

97. ISSN: 0144-929X. DOI: 10.1080/01449290500330331. URL: http://www.tandfonline.com/doi/abs/10.1080/01449290500330331.

Hevner, Alan R. et al. (2004). "Design Science in Information System Research". In: *MIS Quarterly* 28.1, pp. 75–105. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.103.1725{\&}rep=rep1{\&}type=pdf.

Highsoft. *Interactive JavaScript charts for your webpage — Highcharts*. URL: https://www.highcharts.com/ (visited on 05/20/2019).

Institute of Medicine (US) et al. (2011). *Clinical Practice Guidelines We Can Trust*. National Academies Press (US), abstract. ISBN: 9780309164221. DOI: 10.17226/13058. URL: http://www.ncbi.nlm.nih.gov/pubmed/24983061.

Johansen, Ingrid H et al. (2018). *Astma - Obstruktiv lungesykdom - Nedre luftveier og lunger - Legevakthåndboken*. URL: https://www.lvh.no/symptomer{_}og{_}sykdommer/nedre{_}luftveier{_}og{_}lunger/obstruktiv{_}lungesykdom/astma (visited on 05/08/2019).

Kristensen, T. (2011). "The Dynamic Content Management system". In: *2011 International Conference on Information Technology Based Higher Education and Training*. IEEE, pp. 1–8. ISBN: 978-1-4577-1673-7. DOI: 10.1109/ITHET.2011.6018677. URL: http://ieeexplore.ieee.org/document/6018677/.

Kristensen, T., Ø. Bech, and M. Dyngeland (2013). "Towards a Multi-Agent E-learning Platform". In: *Journal of Computer Engineering and Informatics* 1.2, pp. 64–81. ISSN: 23070072. DOI: 10.5963/JCEI0102004. URL: http://www.academicpub.org/DownLoadPaper.aspx?paperid=5471.

Masic, Izet, Milan Miokovic, and Belma Muhamedagic (2008). "Evidence based medicine - new approaches and challenges." In: *Acta informatica medica : AIM : journal of the Society for Medical Informatics of Bosnia & Herzegovina : casopis Drustva za medicinsku informatiku BiH* 16.4, pp. 219–25. ISSN: 0353-8109. DOI: 10.5455/aim.2008.16.219-225. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3789163/.

Michael, David R and Sande Chen (2006). *Serious games : games that educate, train and inform*. Thomson Course Technology PTR. ISBN: 1-59200-622-1.

Molich, Rolf and Jakob Nielsen (1990). "Improving a human-computer dialogue". In: *Communications of the ACM* 33.3, pp. 338–348. ISSN: 00010782. DOI: 10.1145/77481.77486. URL: http://portal.acm.org/citation.cfm?doid=77481.77486.

Morrison, Gary R. et al. (1995). "The Effects of Feedback and Incentives on Achievement in Computer-Based Instruction". In: *Contemporary Educational Psychology* 20.1, pp. 32–50. ISSN: 0361-476X. DOI: 10.1006/

CEPS.1995.1002. URL: https://www.sciencedirect.com/science/article/pii/S0361476X85710028.

National Heart Lung and Blood Institute and U.S. Department of Health & Human Services (2007). "Guidelines for the Diagnosis and Management of Asthma (EPR-3) — National Heart, Lung, and Blood Institute (NHLBI)". In: URL: https://www.nhlbi.nih.gov/health-topics/guidelines-for-diagnosis-management-of-asthma.

Odhiambo, J A et al. (1998). "Urban-rural differences in questionnaire-derived markers of asthma in Kenyan school children." In: *The European respiratory journal* 12.5, pp. 1105–12. ISSN: 0903-1936. URL: http://www.ncbi.nlm.nih.gov/pubmed/9864005.

Patel-Schneider, Peter F. and Ian Horrocks (2006). "A comparison of two modelling paradigms in the Semantic Web". In: *Proceedings of the 15th international conference on World Wide Web - WWW '06*. New York, New York, USA: ACM Press, p. 3. ISBN: 1595933239. DOI: 10.1145/1135777.1135784. URL: http://portal.acm.org/citation.cfm?doid=1135777.1135784.

Peleg, M et al. (2000). "GLIF3: the evolution of a guideline representation format." In: *Proceedings. AMIA Symposium*, pp. 645–9. ISSN: 1531-605X. URL: http://www.ncbi.nlm.nih.gov/pubmed/11079963http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC2243832.

Preece, Jennifer, Helen Sharp, and Yvonne Rogers (2015). *Interaction Design: beyond human-computer interaction*. 4th. Wiley, pp. 196, 224.

React. *React – A JavaScript library for building user interfaces*. URL: https://reactjs.org/ (visited on 05/20/2019).

React-Native. *React Native · A framework for building native apps using React*. URL: https://facebook.github.io/react-native/ (visited on 05/20/2019).

Redux. *Redux · A Predictable State Container for JS Apps*. URL: https://redux.js.org/ (visited on 05/20/2019).

ReduxJS-thunk. *reduxjs/redux-thunk: Thunk middleware for Redux*. URL: https://github.com/reduxjs/redux-thunk (visited on 05/20/2019).

Republic of Kenya, Ministry of Health (2016). *Basic Paediatric Protocols for ages up to 5 years*. 4th. Republic of Kenya, Ministry of Health, p. 32.

Rodrigues da Silva, Alberto (2015). "Model-driven engineering: A survey supported by the unified conceptual model". In: *Computer Languages, Systems & Structures* 43, pp. 139–155. ISSN: 1477-8424. DOI: 10.1016/J.CL.2015.06.001. URL: https://www.sciencedirect.com/science/article/pii/S1477842415000408.

Rutle, Adrian (2010). "Diagram predicate framework: A formal approach to MDE". In: URL: http://bora.uib.no/handle/1956/4469.

Sadowska, Małgorzata and Zbigniew Huzar (2019). "Representation of UML Class Diagrams in OWL 2 on the Background of Domain Ontologies". In: *e-Informatica Software Engineering Journal* 13, p. 89. URL: `http://www.e-informatyka.pl/attach/e-Informatica{\_}-{\_}Volume{\_}13/eInformatica2019Art03.pdf`.

Shortliffe, Edward H. (1998). "The Evolution of Health-Care Records in the Era of the Internet". In: URL: `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.18.2653`.

Susi, Tarja, Mikael Johannesson, and Per Backlund (2015). "Serious Games - An Overview". In: URL: `https://www.researchgate.net/publication/220017759{\_}Serious{\_}Games{\_}-{\_}An{\_}Overview`.

The Aasgard Project (2006). *AsbruView.* URL: `http://www.asgaard.tuwien.ac.at/asbruview/` (visited on 05/09/2019).

Wix. *Home - React Native Navigation - truly native navigation for iOS and Android.* URL: `https://wix.github.io/react-native-navigation/{\#}/` (visited on 05/20/2019).

Woolf, S H et al. (1999). "Clinical guidelines: potential benefits, limitations, and harms of clinical guidelines." In: *BMJ (Clinical research ed.)* 318.7182, pp. 527–30. ISSN: 0959-8138. URL: `http://www.ncbi.nlm.nih.gov/pubmed/10024268http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC1114973`.

Wouters, Pieter et al. (2013). "A meta-analysis of the cognitive and motivational effects of serious games." In: *Journal of Educational Psychology* 105.2, pp. 249–265. ISSN: 1939-2176. DOI: `10.1037/a0031311`. URL: `http://doi.apa.org/getdoi.cfm?doi=10.1037/a0031311`.

Appendices

Comparison of CPG applications

This appendix shows two comparative tables of applications and websites that display clinical guidelines; one for international applications and one for Norwegian applications. The selection is based on a few articles concerning apps for clinical practice and a search for clinical guideline applications on Google Play. Helsebiblioteket provides access for Norwegian health workers and students to some of the applications and websites below [1], but users do not explicitly need accounts for this.

Table A1: Comparison of international guideline applications

Name	Type	Platform(s)	Account needed	Account type(s)	Content	Interaction	Remarks
Guidelines [2]	Mobile app	Android	Yes	Paid	Guidelines as text documents and flowcharts; notes	Flowcharts require separate PDF reader	Free in UK
AgileMD [3]	Mobile app	Android, iOS	Yes	Paid	Guidelines as lists and flowcharts; notes		Customizable for each organization
UpToDate [4]	Mobile app, website	Android, iOS	Yes	Paid	Guidelines as text documents; calculators	Can view outline of documents	
Medscape [5]	Mobile app	Android, iOS	Yes	Free	Guidelines as text documents; drug interaction checker	Can view outline of documents	Focus on medicine; allows resizing of text in app
PEPID [6]	Mobile app, computer app	Android, iOS, Windows Phone, Windows 10	Yes	Paid (free trial)	Protocols as text documents; drug interaction checker; notes		Focus on drugs and pills
Epocrates [7]	Mobile app	Android, iOS	Yes	Free, paid	Guidelines as text documents; drug interaction checker	Each guideline organizes its contents in an accordion	Unlocked features might offer more interactivity
Essential Medical Guidance (EMGuidance) [8]	Mobile app	Android, iOS	Yes	Free	Guidelines as text documents (some with flowcharts)	Limited through forms	Currently focused on South Africa

Table A1 (cont.)

Name	Type	Platform(s)	Account needed	Account type(s)	Content	Interaction	Remarks
MSF Medical Guidelines [9]	Mobile app	Android, iOS, Windows Phone	No	Free	Store medical guidelines as books; list of recent guidelines		Content can be accessed with low quality for weak connections
NICE Guidance [10]	Mobile app, website	Android, iOS	No	Free	Guidelines as text documents; bookmarks	Each guideline starts with an (blocking) outline containing links to its sections	
RCH Clinical Practice Guidelines [11]	Mobile app	Android, iOS	No	Free	Guidelines as text documents		
ACP Clinical Guidelines [12]	Mobile app, website	Android, iOS	No	Free	Guidelines as text documents (some with flowcharts)		Each guideline contain a link that brings you to the corresponding webpage
ALS Handbook [13]	Mobile app	Android, iOS, Windows Phone	No	Free	Guidelines as text documents		Focus on so-called checklists; dark and visually different style compared to others
Guideline Central [14]	Mobile app	Android, iOS	No	Free, paid	Guidelines as text documents		Some features require an account
BC Guidelines [15] [16]	Mobile app, website	Android, iOS	No	Free	Guidelines as text documents (some with flow charts)	Figures can be scrolled horizontally	Outline of guidelines can be accessed through menu button

Table A1 (cont.)

Name	Type	Platform(s)	Account needed	Account type(s)	Content	Interaction	Remarks
ACC Guideline Clinical App [17]	Mobile app	Android, iOS	No	Free	Guidelines as interactive and static flowcharts; calculators; bookmarks; notes	Clicking on answer buttons reveal further instructions	Uses breadcrumb elements that, instead of being on a line, are stacked on top of each other
ESC Pocket Guidelines [18]	Mobile app	Android, iOS	Yes	Free	Guidelines as interactive and static flowcharts	Clicking on answer buttons reveal further instructions	Uses breadcrumb elements that, instead of being on a line, are stacked on top of each other
ESMO Interactive Guidelines [19]	Mobile app	Android, iOS	No	Free	Guidelines as interactive and static flowcharts	Clicking on answer buttons reveal further instructions	Uses breadcrumb elements that, instead of being on a line, are stacked on top of each other
Walsall Join COPD Interactive Guidelines [20]	Mobile app	Android	No	Free	Guidelines as interactive flowcharts	Clicking on answer buttons reveal further instructions	Elements are very separated on tablets
Skyscape [21]	Mobile app	Android, iOS	Yes	Free, paid	Guidelines as text documents (some with flowcharts); instructions for nurses and a quiz with scoring system		Some content requires purchases
Doctor Diagnose Symptoms Check [22]	Mobile app	Android	No	Free	Interactive diagnosis systems for people to diagnose themselves	Clicking on answer buttons reveal further instructions	
BMJ Best Practice [23]	Mobile app, website	Android, iOS	Yes	Paid (free trial)	Guidelines as text documents	Some elements can be expanded	Uses breadcrumb elements; images are separated from the documents
Daily Anatomy [24]	Mobile app	Android, iOS	No	Free	Flashcard quizzes to learn anatomy	Clicking on answer alternatives within a time limit	Displays user statistics and uses learning strategies
LIFE: Neonatal Resuscitatotion Training (ETAT+NR) [25]	Mobile app	Android	No	Free	Visualization of clinical tasks with quizzes	First person 3D game. Quizzes and the use of game elements	Multiple try with hints. Doesn't let the user jump over questions

Table A2: Comparison of Norwegian guideline applications

Name	Type	Platform(s)	Account needed	Account type(s)	Content	Interaction	Remarks
Norsk Elektronisk Legehåndbok (NEL) [26]	Mobile app, website	Android, iOS	Yes	Paid	Guidelines as text documents	Some parts are organized in an accordion	Website has something about clinical procedures
LegeAppen [27]	Mobile app	Android	No	Free	Guidelines as text documents; links to other Norwegian health apps and websites	Uses Material elements such as cards and floating action buttons	Open source
Legevakthånd-boken [28]	Mobile app, website	Android, iOS	No	Free	Guidelines as text documents	Menus with tabs and tables	Available in Norway only
Sykepleiehånd-boka (SHB) [29]	Mobile app	Android, iOS	Yes	Paid	Guidelines as pictures, videos and text documents	Uses two tabs on each page to provide contextual information	
eHåndboken [30]	Mobile app, website	Android, iOS	No	Free	Guidelines as text documents; bookmarks	Can filter offline and bookmarked documents on the same page	Few flow charts
Tiltaksboka [31]	Mobile app, website	Android, iOS	No	Free	Guidelines as text documents (some with flowcharts)	Uses expandable menu lists	
SykepleiePluss [32]	Mobile app, website	Android, iOS	Yes	Paid	Lecture videos; quizzes; calculators	Feedback to quiz questions are given immediately after answering	Tailored towards medical students; no guidelines

Table A2 (cont.)

Name	Type	Platform(s)	Account needed	Account type(s)	Content	Interaction	Remarks
Akuttveileder i pediatri [33]	Mobile app, website	Android, iOS	No	Free	Guidelines as text documents		
Felleskatalogen [34]	Mobile app, website	Android, iOS, Windows Phone	No	Free	Mostly documents and info about medical products and pharmacies	Horizontally scrollable icon list on the main page	Not easy to see that the icon list is scrollable
Veileder i akuttpsykiatri [35]	Mobile app, website	Android, iOS, Windows Phone	No	Free	Guidelines as text documents	Guidelines are organized into an accordion	Feedback bar always visible; reduces area for content
Trygfonden Hjertestart [36]	Mobile app	Android, iOS	No	Free	Guidelines as text documents and pictures; map of heart starters		Focused on Denmark
Redd Liv [37]	Mobile app	Android, iOS	No	Free	Instructions for CPR	Clicking on answer buttons reveal further instructions	Huge buttons
Helsedirektoratet [38]	Website		No	Free	Guidelines as text documents	Uses accordion elements	Guideline header is a bit buggy

References

[1] http://www.helsebiblioteket.no/om-oss

[2] https://play.google.com/store/apps/details?id=uk.co.isai.android.mgp.guidelines

[3] https://www.agilemd.com/download-organizational-app

[4] https://www.uptodate.com

[5] https://www.medscape.com

[6] http://www.pepid.com

[7] http://www.epocrates.com

[8] https://emguidance.com

[9] https://medicalguidelines.msf.org

[10] https://www.nice.org.uk/guidance

[11] https://www.rch.org.au/clinicalguide

[12] https://www.acponline.org/clinical-information/guidelines/mobile-clinical-guidelines

[13] https://apps.appmachine.com/alshandbook

[14] https://www.guidelinecentral.com

[15] http://bcguidelinesapp.ca

[16] http://www.bcguidelines.ca

[17] http://www.acc.org/tools-and-practice-support/mobile-resources/features/guideline-clinical-app

[18] https://www.escardio.org/Guidelines/Clinical-Practice-Guidelines/Guidelines-derivative-products/ESC-Mobile-Pocket-Guidelines

[19] http://www.esmo.org/Guidelines/Pocket-Guidelines-Mobile-App

[20] https://play.google.com/store/apps/details?id=com.walsall.copd

[21] https://www.skyscape.com/sml/

[22] https://play.google.com/store/apps/details?id=com.appcolliders.doctordiagnose

[23] http://bestpractice.bmj.com/info/app/

[24] http://www.dailyanatomy.com/

[25] https://oxlifeproject.org/

[26] https://legehandboka.no

[27] https://olejon.github.io/mdapp/

[28] http://www.lvh.no

[29] https://www.shb.no/info

[30] http://ehandboken.ous-hf.no

[31] http://tiltaksboka.no

[32] https://sykepleiepluss.no

[33] http://www.helsebiblioteket.no/retningslinjer/akuttveileder-i-pediatri/forside

[34] https://www.felleskatalogen.no/medisin/smarttelefoner-og-nettbrett

[35] https://sykehuset-ostfold.no/helsefaglig/fagprosedyrer/veileder-i-akuttpsykiatri-vap

[36] https://hjertestarter.dk

[37] https://www.lhl.no/lar-a-redde-liv-med-gratis-app/

[38] https://helsedirektoratet.no

A Model Driven Approach to the Development of Gamified Interactive Clinical Practice Guidelines

A Model Driven Approach to the Development of Gamified Interactive Clinical Practice Guidelines

Job N. Nyameino[1,3], Fazle Rabbi[2], Ben-Richard Ebbesvik[2], Martin C. Were[3,4,5], Yngve Lamo[2]
[1]*Department of Informatics, University of Bergen, Bergen, Norway*
[2]*Department of Computing, Mathematics, and Physics, Western Norway University of Applied Sciences, Bergen Norway*
[3]*Institute of Biomedical Informatics, Moi University, Eldoret, Kenya*
[4]*Department of Biomedical Informatics, Vanderbilt University Medical Center, Nashville, TN, USA*
[5]*Vanderbilt Institute for Global Health, Vanderbilt University Medical Center, Nashville, TN, USA*
jbngena@gmail.com, Fazle.Rabbi@hvl.no, brebbesvik@gmail.com, martin.c.were@vanderbilt.edu, Yngve.Lamo@hvl.no

Keywords: Clinical Practice Guidelines; Model Driven Engineering; Gamification

Abstract: Clinical practice guidelines (CPGs) play a fundamental role in modern medical practice since they summarize the vast medical literature and provide distilled recommendations on care based on the current best evidence. However, there are barriers to CPG utilization such as lack of awareness and lack of familiarity of the CPGs by clinicians due to ineffective CPG dissemination and implementation. This calls for research into effective and scalable CPG dissemination strategies that will improve CPG awareness and familiarity. We describe a formal model-driven approach to design and implement a gamified e-learning system for clinical guidelines. We employ gamification to increase user motivation and engagement in the training of guideline content. Our approach involves the use of models for different aspects of the system, an entity model for the clinical domain, a workflow model for the clinical processes and a game model to manage the training sessions. A game engine instantiates a training session by coupling the workflow and entity models to automatically generate questions based on the data in the model instances. Our proposed approach is flexible and adaptive as it allows for easy updates of the guidelines, integration with different device interfaces and representation of any guideline.

1 Introduction

Medical knowledge is increasing at an exponential rate and it is difficult for clinicians to keep up with this quantity of knowledge production (Fervers et al., 2010). The development and use of clinical practice guidelines (CPGs) is a promising solution to this problem. CPGs are systematically developed statements that assist practitioners and patients to make decisions about appropriate health care for specific circumstances (Lohr et al., 1992). Guidelines are a comprehensive summary of the available evidence about medical conditions and provide recommendations for the management of those conditions (Goud et al., 2009). A well-developed guideline reduces variations in care, improves diagnostic accuracy, promotes effective therapy and discourages ineffective therapies all which contribute to improved quality of care (Shiffman et al., 2004). The mere availability of guidelines does not necessarily mean that the recommendations will be used in actual care. Indeed, there has been a reported gap between recommended care according to the evidence base and actual practice leading to preventable errors in practice (Donaldson et al., 2000; Baker, 2001). This gap can be attributed to several barriers to guideline dissemination and implementation which include: internal barriers (lack of awareness, lack of familiarity, lack of agreement with the guideline content, and the inability to overcome the inertia of previous practice) and external barriers (i.e., patient, environmental, and guideline related factors such as ease of use and complexity of the guideline) (Cabana et al., 1999).

The nature of guideline development means that published guidelines are well-researched, comprehensive documents that can be prohibitively voluminous. For example, the National Heart, Lung, and Blood Institute (NHLBI) 2007 Guidelines for the Diagnosis and Management of Asthma full report is 440 pages long (NHLBI, 2007) while the National Institute for Health and Care Excellence (NICE) guidelines for the diagnosis monitoring and management of chronic asthma (2017) report is 39 pages long (NICE, 2017). Such large texts are impractical for use at the point of care. Additionally, poor guideline presentation has been identified as a factor in the lack of physician familiarity as some of the guidelines have been described as being tedious, repetitive, confusing, and unclear (Cabana et al., 2000).

To mitigate some of the barriers to knowledge acquisition of guideline content, new dissemination strategies aimed at improving awareness and familiarity of guideline content are required. Active guideline dissemination strategies have been found to be more effective than passive strategies at improving the application of evidence based recommendations in patient care (Grimshaw et al., 2012). In particular, educational interventions (e.g. distribution of printed guidelines, educational meetings and outreaches) strengthen the effect of clinical educational material. Further, the more intensely the information is provided through these interventions, the greater its effect on the recipients (Marriott et al., 2000). Research into active strategies for clinical guideline dissemination are timely and relevant as they will potentially help to plug the gap between recommended and actual clinical practice.

One potentially useful active educational intervention is in the distribution of gamified guidelines. Gamification is the use of game design elements in non-game contexts (Deterding et al., 2011b; Deterding et al., 2011a). It uses game based mechanics, aesthetics and thinking to engage people, motivate action, promote learning and solve problems (Kapp, 2012). The concept of Gamification is relatively new and has been used to describe the use of game-based concepts and techniques, with the goal of increasing the motivation and engagement of the participants and improving the results.

The implementation of guideline summaries as interactive, gamified flowcharts on a mobile platform will potentially mitigate the problems of guideline complexity and presentation that plague the effective dissemination of guideline content. In this paper we present a formal model driven approach to gamification of clinical practical guidelines. To illustrate the approach, we present three models, an entity model of the clinical encounter domain, a workflow model for the clinical processes and a game model all of which will be integrated to create our gamified system. We also describe a prototype mobile-based guideline app that incorporates these models to present a gamified interactive guideline training tool.

The rest of the paper is organized as follows: In section 2 we give an introduction to the Diagram Predicate Framework (DPF) and show how it can be used for modelling CPGs workflows and entity models for the clinical domain. Moreover, we illustrate how the guideline workflow information is synchronized with the domain information. In section 3 we introduce our approach to gamification of workflows and discuss in further detail the use of the different models in our design. In section 4 we describe the implementation of our approach in the development of the prototype system. Finally in section 5 we compare our approach to other works before we conclude the paper and envision further work in section 6.

2 Background

In this work we use a formal diagrammatic approach to model driven software engineering (MDE), called Diagram Predicate Framework (DPF). MDE is a system development paradigm that promotes the use of models as the primary artefacts that drives the whole development process. In MDE models are specified using a modelling language whose syntax and semantics are defined by a metamodel (Rodrigues da Silva, 2015). This allows for the development of domain-specific modelling languages (DSLs) using notations and abstractions that are unique to a given domain. The use of DSLs allows for the development of more expressive models and ease of use by domain experts.

In this section we provide an overview of how diagrammatic models can be created using DPF. We chose to use DPF as it can be used to create custom domain specific modelling languages. Furthermore, we'll present a metamodel for representing a CPG workflow, a simple entity model for the medical domain, a model for the game engine and finally an integrated multi-metamodel that incorporates the entity and CPG workflow models. UML is another alternative approach for the modeling of various kinds such as UML entity model, sequence model. However, DPF allows us to do multilevel metamodeling and also visualizes constraints in the models.

2.1 Diagram Predicate Framework (DPF)

DPF formalizes software development activities such as metamodelling (Rutle et al., 2009) and model transformations (Rutle et al., 2012) based on category theory (Barr and Wells, 1990) and graph transformations (Löwe, 1993). By applying DPF we can formalize clinical guidelines and clinical domain models at different abstraction levels in form of diagrammatic specifications. The diagrammatic nature of DPF also

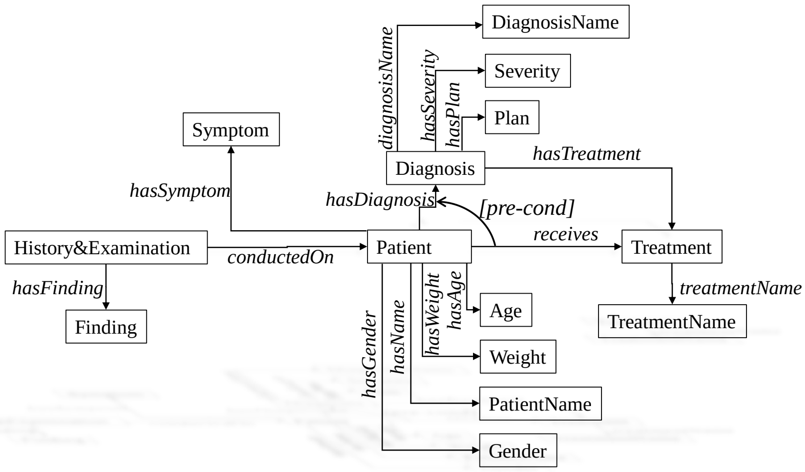

Figure 1: A simplified entity model of the clinical encounter domain

facilitates visual representations of guidelines that can be presented at different level of abstraction. A model in DPF is represented by a diagrammatic specification $\mathfrak{S} = (\mathcal{S}, C^{\mathfrak{S}} : \Sigma)$ which consists of a graph $\mathcal{S}$ and a set of constraints $C^{\mathfrak{S}}$ specified by a predicate signature Σ.

The predicate signature is composed of a collection of predicates, each having a name and an arity (shape graph). A constraint consists of a predicate from the signature together with a binding to the subgraph of the models underlying graph which is affected by the constraint. In order to apply DPF for the modeling of a game that operates over clinical practice guideline we need to formalize the concepts of a guideline using DPF and also model the gamification concepts with DPF. In the following subsection we present how DPF can be used to model different aspects of guidelines and representing the concepts for gamification.

2.2 Entity modelling

We will now present an entity model of the clinical domain. To do this we use a metamodel containing Concepts, Attributes and References. An actual model typed over this metamodel is shown in Figure 1. We have concepts and corresponding relations for domain entities such as Patients, Diagnosis, Treatments etc (see Figure 1). The model consists of a constraint modeled with a predicate named 'pre-condition' and visualized with symbol $[pre - cond]$. The constraint specifies that all the treatment instances must have a reference to a diagnosis instance.

2.3 Workflow modelling

Clinical practice guidelines often consist of a flow of information. Workflow models may be used to represent the flow of a guideline. In Figure 2 below we see an example metamodel (M_2) for behavioural models, where Tasks can be connected by Flow edges. On the next abstraction level (M_1) we see a generic treatment model that is typed by the flow model. The treatment model has three tasks Assessment and Diagnosis, Treatment and Evaluation. Finally, at (M_0) we see an instance of the treatment workflow of a severe asthma diagnosis.

2.4 Game modelling

At their core, games are goal-oriented activities with reward and progress tracking mechanisms. The design of gamified e-learning systems should be undertaken in the view of these core concepts. In our system, the training will be done through a series of questions based on the guideline content. The game engine in our model automatically generates questions from the entity and workflow models to instantiate a training module. The questions are categorized according to the learners skill level (beginner, intermediate, advanced) and each question has a reward in the form of points. A game model should also specify a learner profile that tracks the learners activities.

2.5 Integrating Models

The training model is built by the integration of the entity and workflow models based on the principles introduced by Rabbi et al (Rabbi et al., 2014a). The states of the training module TM are defined by a set

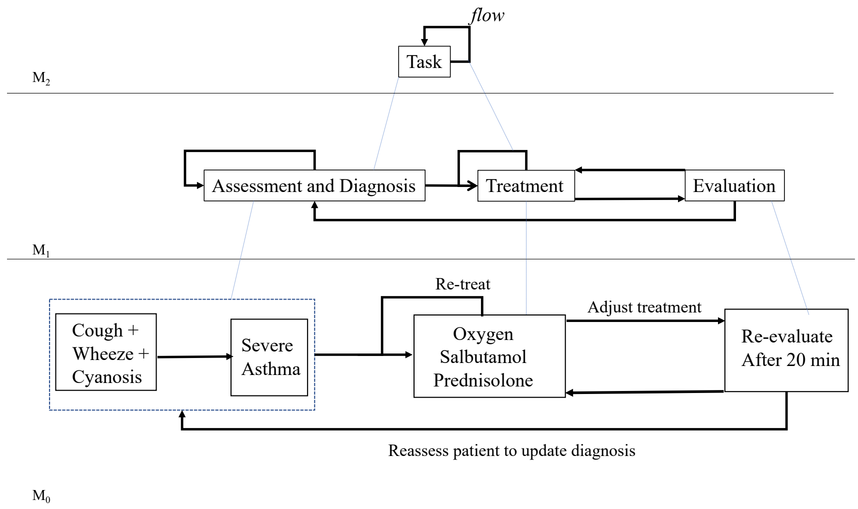

Figure 2: The workflow model with its metamodel

of elements that include a pair of workflow instance WI and an entity instance EI: $TM_i =< EI_i, WI_i >$ where i is a natural number. This integration of models is shown in Figure 3 and the concept is discussed in more details in section 3. In Figure 3, we show a section of the entity model with values from a given scenario where based on the *History & Examination findings*, a *Diagnosis* of Severe Asthma is made and its *Treatment* specified. The flow of how this process should happen is shown in the workflow model.

3 Proposed Method

3.1 Gamification elements

The core concepts of games that should inform the design of gamified e-learning systems are goal oriented activities with reward mechanisms and progress tracking (Strmečki et al., 2015). In the training of guideline content, the main goal is for the trainees to learn how to treat different aspects of a disease as described in the guideline. The reward mechanisms and progress tracking aid in increasing the users engagement and motivation (Bernik et al., 2018).

3.2 CPG modelling

In our approach, we separate two aspects from a clinical practice guideline (CPG). Medical conditions and clinical encounters of patients is one aspect which we model in an entity model; recommended clinical processes is another aspect which we model in a workflow diagram. A flow in a guideline often consist of medical conditions such as 'start giving oxygen if a child is convulsing for more than 5 minutes'. Typically a modeling approach specifies all the recommended processes in a workflow diagram. An instance of the diagram would then specify a concrete scenario representing the care processes executed for a patient.

In our approach, we skip modeling the general recommendations in a workflow diagram and directly specify concrete scenarios. For example, in our approach we model a scenario where a 2 year old boy who is convulsing for 7 minutes is admitted to a hospital and we start treatment by giving oxygen. We follow this approach as it allows us to model concrete scenario with less effort and we do not need to spend time on encoding the whole guideline. Since the purpose of modeling the guideline scenario is to generate questions using our approach, it is sufficient for us to model the scenario representing the recommended clinical processes according to the guidelines. To represent such scenarios we integrate a CPG workflow model with an entity model which is encoded as a DPF model. Typically, a CPG consist of a large number of pages with information from the clinical domain. There exists some approaches that allows us to design a visual model of a CPG.

In (Rabbi et al., 2014a; Rabbi et al., 2014b), the authors presented an approach where different aspects of a system were coordinated by means of multiple metamodels. The approach is based on the foundation of DPF.

In the multi-metamodeling approach, a workflow model is integrated with an entity model by means of metamodel coordination. A workflow metamodel is

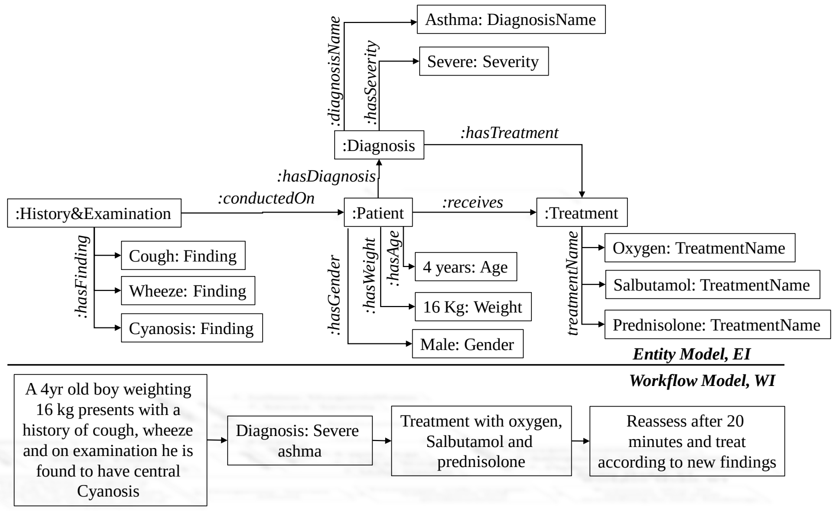

Figure 3: Integrated entity and workflow models

used to design the flow of a system and an entity metamodel used to design the entities and relationship of a domain. A workflow model can be used to represent an abstraction of a CPG but we need to incorporate the detailed domain knowledge in our modelling. In this paper we exploit the use of the multi-metamodeling approach to represent the domain knowledge of a clinical guideline and the clinical process and apply them to execute a training session. The idea of using the CPG workflow model is to control the flow of the game such that the user is interacting with the right gaming element at the right time.

In this section we explain a training module which consist of one or more CPG models and one or more entity models represented as DPF specifications. The states of the training module TM are defined by a set of elements that include a pair of CPG workflow instance and a DPF entity instance that represents the entities within a domain and relationships between them. Figure 4 illustrates an example of two states TM_1 and TM_2 of a training module. The state TM_1 consists of a set of elements that include a pair of workflow instances and DPF instances: $\{< WI_0, EI_0 >, < WI_1, EI_1 >, .. < WI_n, EI_n\}$ where $WI_1, WI_2, ..WI_n$ are workflow instances and $EI_0, EI_1, ..EI_n$ are DPF entity instances. Figure 4 shows a training session flow which consists of a sequence of states of training module i.e., $Training_{Flow1} :=< TM_1, TM_2, ...TM_k >$. Figure 3 shows an instance of a training session. In Figure 3 the game engine instantiates a training session by generating questions based on the DPF entity model and CPG workflow model. For example, it could initially generate a scenario based on the patient details and history and examination findings and ask what the diagnosis is. If answered correctly, it will move on to the next task and ask about the treatment. A training session is composed of a sequence of training modules and is evolved from the initial state of a training flow and progresses based on the answer provided by the user.

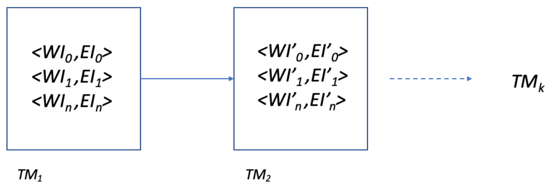

Figure 4: States of training module

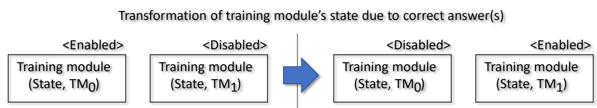

Figure 5: Progression of the states of training module

In our approach a training session is evolved from the initial state of a training flow and progresses based on the answer provided by the user. Figure 5 illustrates the idea of the progression of the states of training session. Depending on the answer given by the user, a game engine consults with the training flow and evolves the state of the training session. We use two DPF predicates $< Enabled >, < Disabled >$ to represent the current status of the training modules. A training module TM_0 when annotated with the $< Enabled >$ predicate indicates that the training

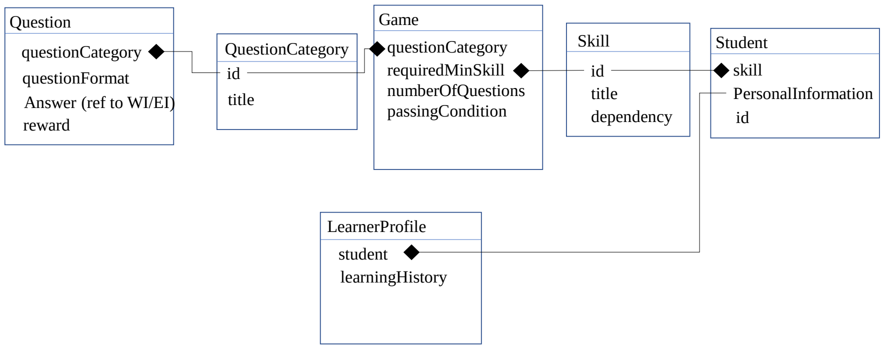

Figure 6: A conceptual model for the game elements

module is currently active and is being considered for training.

The answers are collected from the user in two different ways. We can ask the user to answer some questions about the domain ontology i.e., entities and their relationships; also the question can be based on the workflow instance. We utilize another DPF model to formulate the questions that can be asked to the user. Figure 6 shows the DPF model representing the concepts of an e-learning game. The model represents the game elements which include information about how the game engine should control the game. In our approach a game instance is associated with a training module. Using this DPF model we can specify the number of questions to be asked, passing condition and can include questions under some question category. We use references to the CPG instances and entity instances for specifying correct and wrong answers.

A general concern about this approach is the validation of the training flow. While constructing a training flow one might make mistake in two ways: (a) wrong composition of instances of CPG model and entity model; (b) wrong flow of CPG instances. To reduce the number of errors we apply inconsistency checking as described in the next section.

4 System Description

We propose to use a generic system based on the idea of multilevel-metamodeling and their coordination. Figure 7 shows an overview of the system. The responsibility of the 'Game Engine' is to control the training flow, maintain the status of the trainee, produce dialogues or control the visualization of the screen. The user should be able to interact with the game engine via 'Google Assistant' or 'Mobile application'.

We plan to support different types of devices for the training to facilitate training considering various learning style of the trainee.

Question Flow manager: The question flow manager selects the questions to be asked depending on the level of difficulty of a training session. It maintains the order of questions to be shown to the user. For example, user-A has skill-1 and chose to go through the beginning session. While randomly selecting questions that falls under the difficulty of 'Beginner', it also looks into the questions that has been used before for user-A. It puts more emphasize on the questions that the user has been struggling with.

Conversation manager: The conversation manager keeps track of the conversation and manages the context of the conversation. For example, if there are three questions to be asked that is related to a child who is 2 years old, then the conversation manager produces a context for three questions and starts the conversation saying "A 2 year old child comes to the emergency department with <some condition>, answer to the following questions:". Afterwards it asks the first question, followed by the 2nd and 3rd questions.

User management: The user management module keeps track of the trainees skill, progress and effort. The user management module is also used to produce visualization showing the performance of a population. If a group of trainee is particularly struggling with a set of questions or question category then the user management module will produce a report and the trainer will be able to monitor it.

4.1 Adaptiveness

There are two ways our system facilitates adaptiveness:

- Customizing the gamification process by means of model-driven-engineering approach.

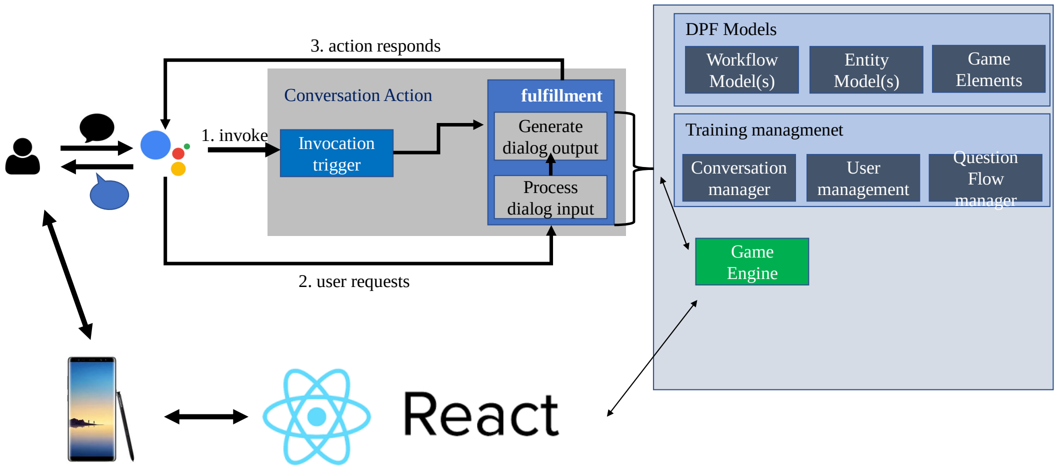

Figure 7: Overview of the proposed system architecture

- Capturing the learning behaviour of trainee and changing the model accordingly.

4.2 Methods

Our approach consists of modeling the entity which represents the concepts related to clinical information; the workflow which represents the flow of a guideline; and the structure of a game. We propose to use WebDPF tool for the modeling of these elements. The multilevel metamodeling supported by the WebDPF tool allows us to design domain specific modeling languages. The tool is also facilitated with a constraint checker which allows us to check if the models being constructed are valid or not. We persist our models using JSON format. Figure 8 shows an overview of the proposed system where the game engine reads the models using a model parser. The game engine executes the game by asking questions to the trainee and also stores the answer in a relational database. We have developed a report engine which retrieves data from the relational database and visualizes learning analytics to the trainer. The trainer can also interact with the report interface and can visualize the data from different perspective. The trainer can get an overview of the training modules and can select individual trainees usage statistics. In our current implementation we have not incorporated any machine learning algorithms. Therefore, adapting a training module according to the requirement of the trainee are done manually by the knowledge engineer.

We used this modeling architecture to develop a proof-of-concept game for the asthma guideline training. Figure 9 shows a sample conversation from the asthma guideline training. While the participant is using google assistant we use the google account for registering the participant to our system. It is planned to use OAuth 2.0 protocol for authenticating the user from the mobile application to the participants Google account. It will allow the user to switch from one device to another. While the participant is using the mobile application they get more feature such as browsing the guideline.

4.2.1 Implementation of the mobile application

The application is developed using React-Native and JavaScript. React-Native is based on the React framework, and is used to build mobile applications for Android and iPhone. The motivation for using such a framework is reuse of code when supporting both mobile platforms as well as the web.

The game consists of a collection of quizzes, where each quiz contains several questions. These questions are based around a scenario, where the student is presented with answer alternatives. Picking an answer alternative will give the student points for how close he was to the right action. The student is presented with the answer key, an explanation, as well as pointers to the evidence and the relevant guideline for further study.

The quiz will conclude with a summary, giving feedback and statistics on students performance. The quiz should have a passing condition to unlock quizzes at a higher difficulty level. This is illustrated in Figure 10.

4.2.2 Generating the scenarios

To generate questions, we will write small scenarios in the form of narrative templates where we use tags to refer to variables in the entity model. The tag refers

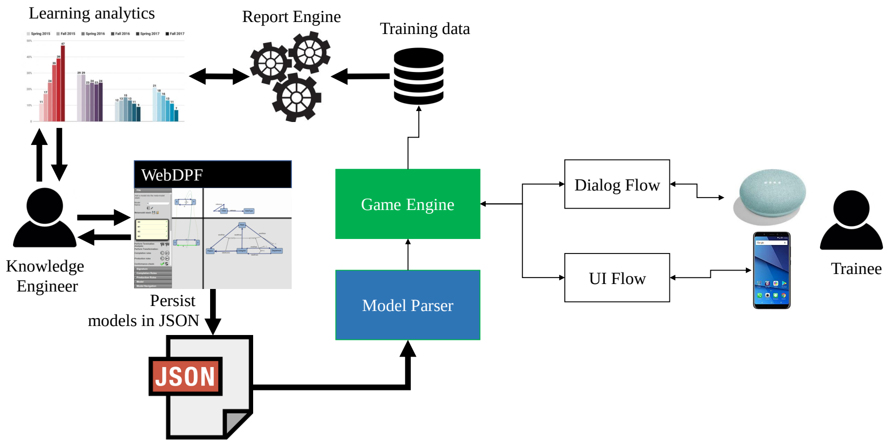

Figure 8: Proposed method of the system

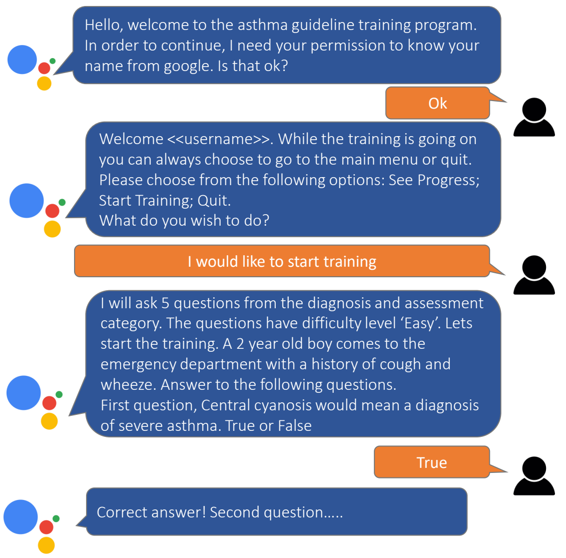

Figure 9: Sample flow of conversation from the asthma guideline training

to a path in the entity graph. The application will traverse through the graph and return the value of the given vertex.

A challenge with this method, is how to present the data returned by the graph in a text. The value from a measurement of the pulse-rate is just an integer. An observation that the patient has a breathing condition is a boolean, and an observation of the patients level of consciousness is an enumerate of the AVPU (Alert, Verbal, Pain, Unresponsive) scale. These values will have to be presented differently to make a good sentence in the scenario. How we solved this issue was by letting the vertex hold a string representation of its value. This is illustrated in Figure 11.

```
<%Ben.name%> arrives at the emergency
department.
He <%Ben.hasConsciousness.value.name%>.
```

translates to

```
Ben arrives at the emergency
department.
He is not alert and not verbal,
but responds to pain.
```

5 Related Work

In (Farkash et al., 2013) Farkash et al. presented a model-driven approach to formalize clinical practice guideline using natural rule language (NRL). Specifying the constraints of a guideline with English-like rule language reduces the gap of the representation and processing of guidelines. The authors presented a set of software components that support the representation, interpretation of CPGs using NRL and also can be applied directly to a patient's EHR data for analysis. Their approach is supported by a proof-of-concept implementation for a simple essential hypertension guideline directive. Our approach is different with their approach as we use a graph based modeling technique and the main contribution of our approach is to support the training of a guideline by means of gamification.

In (Kristensen et al., 2009) Kristensen et al. presented a conceptual model for e-learning where the learning materials are divided into atomic units and organized in several graph based models such as 'Knowledge map', 'Learning map' and 'Student map'. These conceptual models provide a better structure for representing an e-learning environment

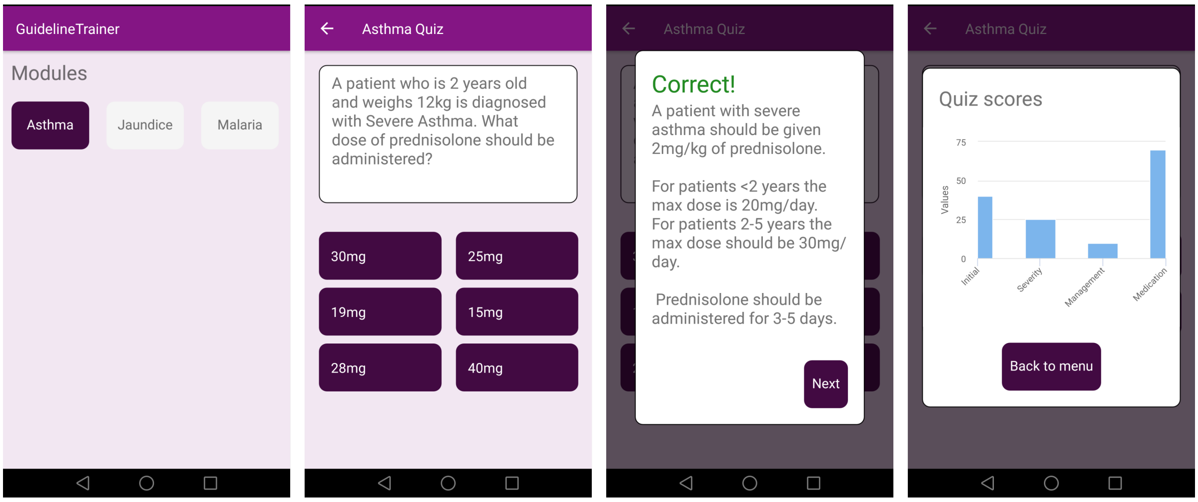

Figure 10: Flow of the mobile application

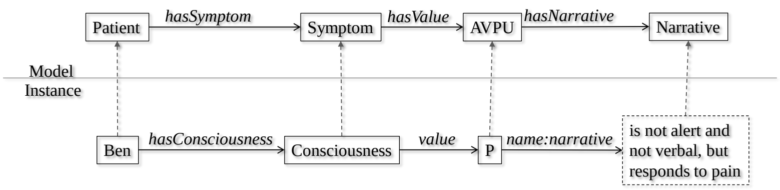

Figure 11: Importing variables from graph into scenario

and an easy-to-use navigation interface for existing learning materials. We borrowed concepts from this paper and adapted them for representing CPGs and game elements by means of Diagram Predicate Framework and multi-metamodelling approach.

A gamification approach was presented in (Akl et al., 2008) where the authors followed the format of TV game shows in which two teams of residents compete in answering questions that are based on the recommendations of guidelines. However, their approach is lacking formalization and does not support model based analysis. In our approach we emphasize on the modeling part and apply model driven engineering technique to interface with various platforms.

In (Del Cura-Gonzlez et al., 2016) the authors conducted a study to assess the effectiveness of a teaching strategy for the implementation of CPGs using educational games. They demonstrated the results for an e-learning game EDUCAGUIA to improve knowledge and skills related to clinical decision-making by residents in family medicine. The game consists of educational games with hypothetical clinical scenarios in a virtual environment. To identify the effectiveness of teaching strategies through e-learning, they proposed an average score comparison of hypothetical scenario questionnaires between the EDUCAGUIA intervention group and the control group. Such evaluation is very important and it reflects the usefulness of utilizing games in teaching guidelines. We plan to conduct similar evaluation of our gamification approach with healthcare professionals in future.

(Aouadi et al., 2016) uses Technology-Enhanced Learning standards to develop serious games which can be used in technological/professional/academic fields for learning. Their goal was to make a scenario-building approach, built upon a model driven architecture. The game includes a health course with demonstrative videos and evaluation quizzes with each course having a passing condition. The game is also demonstrated as a 3D game in a context of medical training. In their approach Aouadi et al., used a platform independent model for the development of game components which was transformed into a platform specific model by means of ATL transformation. While their approach is very close to our proposed method, they lack modularization and separation of concerns. In our approach we do not only apply multilevel metamodelling but also the integration of different modeling hierarchies which allows us to conveniently articulate various aspects of an e-learning system.

(Wyatt et al., 2013) presents OKWA (Okay with Asthma) which is a game targeted on children. They aim at educating children with asthma in self-management skill. This includes information about medications, how to avoid triggering the asthma, monitoring, when and how to get help from others. The game is an interactive animated movie-style narrative, where the actions the child chooses will have an effect on how the story develops. Our project differs from this one as our target group is adult health care workers. We will also focus more on evaluation of medical knowledge through tasks and quizzes rather than just story telling.

(Shegog et al., 2006) is similar to the OKWA project, as it targets self-management skills for children with asthma during a role playing game. The game uses the child's asthma profile, so the child can see the responses to his/hers health information in the game. To complete a scenario, the character in the game needs to be symptom free.

(Zolfo et al., 2010) describes an approach where they use mobile phones as a personal learning environment for health care workers in resource limited environments. They put an emphasize on the importance of avoiding health care workers being absent from the health station for training programs. They use didactic learning material (3d animations, video, presentations, sound) and evaluates learning through multiple choice questions. They also use Skype and Facebook to have clinical case discussions with a network of experts. The project differs to ours, as they have an emphasize on didactic distance-learning while we are learning through actively solving problems and tasks through gaming elements.

(Bartel et al., 2017) aims for a generic gaming platform for implementing gamified learning arrangements in engineering education. Their approach to implementation is based on the concept of domain-specific modeling, which is descirbed as an instance of model-driven software design. However both theirs and our projects are in the field of model driven development, Bartels work aims at engineering education, while we aim at education of health care workers. Their project is a work in progress, and have limited results to analyze and compare to.

(Pesare et al., 2016) presents both Edugame and Simulation of Clinical Cases. Edugame is aimed at patients and caregivers, to manage the disease and promote a healthy lifestyle to avoid critical situations and hospitalization. The game is a role-playing game, and the users mission is to answer correct on the problems posed on the character in the game. Simulation of Clinical Cases is a single player simulation game. The goal is to save the character in the game, by the user suggesting the right therapy, action and/or examination to solve the condition the patient in the game has. The game adapts to which role the clinician has, so the game will be different for a nurse and a physician. The game provides scores according to if the answer was correct, partially correct or wrong. Our project has a larger focus on model driven development and a data model to easier add new content and other types of games.

Septris (Evans et al., 2015) is an online training tool to help emergency clinicians to identify and differentiate between the different forms of the sepsis syndrome. Pick the right diagnostic tests and provide optimal management of the syndrome. Diagnose and treatment is a big part of our project as well, but it will be made general enough to make games for several different medical conditions.

6 Conclusion

In this work, we have presented a model-driven approach to the design and development of a gamified system for learning clinical guideline content. We also present a prototype mobile e-learning system that utilized our design approach in its development. In the near future we aim to test our system with clinicians to evaluate its usability, acceptability and effectiveness.

ACKNOWLEDGEMENTS

This work was supported in part by the NORHED program (Norad: Project QZA-0484). The content is solely the responsibility of the authors and does not necessarily represent the official views of the Norwegian Agency for Development Cooperation.

REFERENCES

Akl, E. A., Mustafa, R., Slomka, T., Alawneh, A., Abhishek, V., and Schnemann, H. J. (2008). An educational game for teaching clinical practice guidelines to internal medicine residents: development, feasibility and acceptability. *BMC Med Education*, 8.

Aouadi, N., Pernelle, P., Ben Amar, C., Carron, T., and Talbot, S. (2016). Models and mechanisms for implementing playful scenarios. In *2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, pages 1–8. IEEE.

Baker, A. (2001). Crossing the quality chasm: a new health system for the 21st century. *BMJ: British Medical Journal*, 323(7322):1192.

Barr, M. and Wells, C. (1990). *Category theory for computing science*, volume 49. Prentice Hall New York.

Bartel, A., Hagel, G., and Wolff, C. (2017). Work in progress: Towards a generic platform for implementing gamified learning arrangements in engineering education. In *2017 IEEE Global Engineering Education Conference (EDUCON)*, pages 1502–1505. IEEE.

Bernik, A., Bubaš, G., and Radošević, D. (2018). Measurement of the effects of e-learning courses gamification on motivation and satisfaction of students. In *41th International Convention-Mipro 2018*.

Cabana, M. D., Ebel, B. E., Cooper-Patrick, L., Powe, N. R., Rubin, H. R., and Rand, C. S. (2000). Barriers pediatricians face when using asthma practice guidelines. *Archives of Pediatrics & Adolescent Medicine*, 154(7):685–693.

Cabana, M. D., Rand, C. S., Powe, N. R., Wu, A. W., Wilson, M. H., Abboud, P.-A. C., and Rubin, H. R. (1999). Why don't physicians follow clinical practice guidelines?: A framework for improvement. *Jama*, 282(15):1458–1465.

Del Cura-Gonzlez, I., Lpez-Rodrguez, J. A., Sanz-Cuesta, T., Rodrguez-Barrientos, R., Martn-Fernndez, J., A.-C. G., Polentinos-Castro, E., R.-C. B., Escortell-Mayor, E., Rico-Blzquez, M., Hernndez-Santiago, V., Azcoaga-Lorenzo, A., Ojeda-Ruiz, E., Gonzlez-Gonzlez, A. I., vila Tomas, J. F., B.-C. J., Molero-Garca, J. M., Ferrer-Pea, R.and Tello-Bernab, M. E., and Trujillo-Martn, M. (2016). Effectiveness of a strategy that uses educational games to implement clinical practice guidelines among spanish residents of family and community medicine (e-EDUCAGUIA project): a clinical trial by clusters. *Implementation Sci.*, 11:71.

Deterding, S., Dixon, D., Khaled, R., and Nacke, L. (2011a). From game design elements to gamefulness: defining gamification. In *Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments*, pages 9–15. ACM.

Deterding, S., Sicart, M., Nacke, L., O'Hara, K., and Dixon, D. (2011b). Gamification. using game-design elements in non-gaming contexts. In *CHI'11 extended abstracts on human factors in computing systems*, pages 2425–2428. ACM.

Donaldson, M. S., Corrigan, J. M., Kohn, L. T., et al. (2000). *To err is human: building a safer health system*, volume 6. National Academies Press.

Evans, K. H., Daines, W., Tsui, J., Strehlow, M., Maggio, P., and Shieh, L. (2015). Septris: a novel, mobile, online, simulation game that improves sepsis recognition and management. *Academic medicine : journal of the Association of American Medical Colleges*, 90(2):180–4.

Farkash, A., Timm, J. T. E., and Waks, Z. (2013). A model-driven approach to clinical practice guidelines representation and evaluation using standards. *Studies in health technology and informatics*, 192:200204.

Fervers, B., Carretier, J., and Bataillard, A. (2010). Clinical practice guidelines. *Journal of visceral surgery*, 147(6):e341–e349.

Goud, R., de Keizer, N. F., ter Riet, G., Wyatt, J. C., Hasman, A., Hellemans, I. M., and Peek, N. (2009). Effect of guideline based computerised decision support on decision making of multidisciplinary teams: cluster randomised trial in cardiac rehabilitation. *Bmj*, 338:b1440.

Grimshaw, J. M., Schnemann, H. J., Burgers, J., Cruz, A. A., Heffner, J., Metersky, M., and Cook, D. (2012). Disseminating and Implementing Guidelines. *Proceedings of the American Thoracic Society*, 9(5):298–303.

Kapp, K. M. (2012). *The gamification of learning and instruction: game-based methods and strategies for training and education*. John Wiley & Sons.

Kristensen, T., Lamo, Y., Hinna, K. R. C., and Hole, G. O. (2009). Dynamic content manager - A new conceptual model for e-learning. In Liu, W., Luo, X., Wang, F. L., and Lei, J., editors, *Web Information Systems and Mining, International Conference, WISM 2009, Shanghai, China, November 7-8, 2009. Proceedings*, volume 5854 of *Lecture Notes in Computer Science*, pages 499–507. Springer.

Lohr, K. N., Field, M. J., et al. (1992). *Guidelines for clinical practice: from development to use*. National Academies Press.

Löwe, M. (1993). Algebraic approach to single-pushout graph transformation. *Theoretical Computer Science*, 109(1-2):181–224.

Marriott, S., Palmer, C., and Lelliott, P. (2000). Disseminating healthcare information: getting the message across. *BMJ Quality & Safety*, 9(1):58–62.

NHLBI (2007). *Expert panel report 3: guidelines for the diagnosis and management of asthma*. Number 97. DIANE Publishing.

NICE (2017). Asthma: diagnosis, monitoring and chronic asthma management. *Nice Guideline 80*.

Pesare, E., Roselli, T., Corriero, N., and Rossano, V. (2016). Game-based learning and Gamification to promote engagement and motivation in medical learning contexts. *Smart Learning Environments*, 3(1):5.

Rabbi, F., Lamo, Y., and MacCaull, W. (2014a). Coordination of multiple metamodels, with application to healthcare systems. In *The 5th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN-2014)/ The 4th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH 2014)/ Affiliated Workshops, September 22-25, 2014, Halifax, Nova Scotia, Canada*, volume 37 of *Procedia Computer Science*, pages 473–480. Elsevier.

Rabbi, F., Lamo, Y., and MacCaull, W. (2014b). A flexible metamodelling approach for healthcare systems. In Jaatun, E. A. A., Brooks, E., Berntsen, K. E., Gilstad, H., and Jaatun, M. G., editors, *Proceedings of the 2nd European Workshop on Practical Aspects of Health Informatics, Trondheim, Norway, May 19-20 , 2014.*, volume 1251 of *CEUR Workshop Proceedings*, pages 115–128. CEUR-WS.org.

Rodrigues da Silva, A. (2015). Model-driven engineering. *Comput. Lang. Syst. Struct.*, 43(C):139–155.

Rutle, A., Rossini, A., Lamo, Y., and Wolter, U. (2009). A diagrammatic formalisation of mof-based modelling languages. In Oriol, M. and Meyer, B., editors, *Objects, Components, Models and Patterns*, pages 37–56, Berlin, Heidelberg. Springer Berlin Heidelberg.

Rutle, A., Rossini, A., Lamo, Y., and Wolter, U. (2012). A formal approach to the specification and transformation of constraints in mde. *The Journal of Logic and Algebraic Programming*, 81(4):422–457.

Shegog, R., Bartholomew, L. K., Gold, R. S., Pierrel, E., Parcel, G. S., Sockrider, M. M., Czyzewski, D. I., Fernandez, M. E., Berlin, N. J., and Abramson, S. (2006). Asthma Management Simulation for Children: Translating Theory, Methods, and Strategies to Effect Behavior Change. *Simulation in Healthcare: The Journal of the Society for Simulation in Healthcare*, 1(3):151–159.

Shiffman, R. N., Michel, G., Essaihi, A., and Thornquist, E. (2004). Bridging the Guideline Implementation Gap: A Systematic, Document-Centered Approach to Guideline Implementation. *J Am Med Inform Assoc*, 11(5):418–426.

Strmečki, D., Bernik, A., and Radošević, D. (2015). Gamification in e-learning: introducing gamified design elements into e-learning systems. *Journal of computer science and technology*, 11(12):1108–1117.

Wyatt, T. H., Li, X., Huang, Y., Farmer, R., Reed, D., and Burkhart, P. V. (2013). Developing an interactive story for children with asthma. *The Nursing clinics of North America*, 48(2):271–85.

Zolfo, M., Iglesias, D., Kiyan, C., Echevarria, J., Fucay, L., Llacsahuanga, E., de Waard, I., Suàrez, V., Llaque, W., and Lynen, L. (2010). Mobile learning for HIV/AIDS healthcare worker training in resource-limited settings. *AIDS Research and Therapy*, 7(1):35.