Contributions on the Multi-Tier Architecture of Electronic Health Record Systems

—Thesis Summary—

Stud. Daniel-Alexandru Jurcău

Healthcare is a domain which generates a lot of data every day, however, traditionally, in paper form. Automating healthcare processes by using computer software systems opens the way to a lot of improvements, like recording information with less room for error, allowing for much easier/faster information retrieval and providing easier ways of sharing information between systems.

The first obstacle encountered when setting of to build a software system for dealing with healthcare data is that the domain in question is both complex and huge. Depending on variables such as the medical specialty, the clinical setting or the country in which one operates, each medic might encounter a completely different slice of the healthcare information domain on a day to day basis. Considering that some type of information is useful from a clinical perspective, while a different piece is only used for administrative purposes like health insurance further complicates matters.

Because of this diversity, creating simple Create-Read-Update-Delete (CRUD) applications that merely store a few bits and pieces of information in a database is not an option as it provides no more utility than simply using a word processor for recording information instead of pen and paper. The challenges involved in creating a flexible system have inspired many research projects.

This thesis brings contributions to finding methods of designing software systems that manage electronic health records. It investigates a way to structure the data based on types of inputs identified and then goes on to improve the flexibility by finding ways to integrate existing specifications, standards and terminologies into the solution.

As medical professionals need to interact with such systems on a daily basis, the thesis also investigates and evaluates ways of creating a modern user interface for storing electronic health records.

The aim of this thesis is to bring valid knowledge which is both useful and usable in practice for the creation and improvement of software solutions dealing with electronic health records, in all the areas of interest concerning their development: storage, processing and the presentation of data; all in a structured manner. As such, the author brings contributions with the purpose of obtaining medical applications that are both efficient and easy to use by medics.

To this end, the thesis set out to accomplish the following goals:

- Identify the type of data stored in electronic health records
- Research into a way of structuring the required data in a manner that is both easy to implement and flexible enough to change
- Identify a way of implementing domain and service logic for dealing with the structured electronic health record data
- Find ways to include standard medical terminologies in an existing solution
- Research ways of building an appropriate user-interface and how to evaluate the usability.

This thesis is divided into four major chapters, accompanied by an introduction and conclusions:

1. Introduction
   1.1. Thesis Goals
   1.2. Thesis Structure
2. Medical Standards And Terminologies
   2.1. Health Level Seven Clinical Document Architecture
   2.2. Medical Code Systems
   2.3. The openEHR Specifications
3. Designing a Solution for the Structured Collection of Medical Data
   3.1. Identifying the Business Domain
   3.2. Rapid Prototyping Using XML
   3.3. Designing Business Domain Classes
4. Integrating Medical Standards
   4.1. Translating Logical Observation Identifiers Names and Codes
   4.2. Matching ICD-10 Codes Using Full-Text Search Engines
   4.3. Achieving Interoperability
5. Developing a Web Front-End for Electronic Health Records
   5.1. Developing the Web Application
   5.2. Evaluating the User Interface
   5.3. Developing a Framework for Automated User Activity Tracking
   5.4. Analyzing the Users' Activity
   5.5. Improving the User Interface
6. Conclusions

Chapter 2, *Medical Standards and Terminologies,* starts by providing an extensive literature study on current methods and standards created to support developing health related applications.

The chapter starts by presenting the Health Level 7 (HL7) Clinical Document Architecture (CDA) standard which provides a way to format (usually as XML) documents which exchange medical information. After pointing out the key points of HL7 CDA, the study also cites various opinions from literature on disadvantages and short-comings of using this standard.

In an attempt to also identify other approaches, the literature study continues with analyzing the methodology of the openEHR specification. This approach favors reusable content created by medical domain experts. From the technical point of view, it presents how software should interact and make use of such content (archetypes, templates) but doesn't introduce constraints into how the implementation should be created (frameworks, persistence methods, etc.).

Besides standards and methodologies for managing electronic health records, chapter 2 also looks at medical coding systems which are freely available: the Logical Observation Identifiers Names and Codes (LOINC), mostly used for coding laboratory results, and International Classification for Diseases (ICD), both used in multiple countries. The information presented in this chapter is later used in chapter 4 which analyses ways of integrating such standards into an existing application.

Chapter 3, *Designing a Solution for the Structured Collection of Medical Data,* analyses the requirements for building an application that manages electronic health records (EHR) and offers an original contribution consisting in a new architecture for the business and data layer of an electronic health record application.

The architecture is based on placing the main focus on modeling the types of inputs that can occur in a medical input form and building the solution around them. Such an approach allows focus on the present requirements and still manages to provide a good separation of what parts of the application require a developer to work on the

code and what parts allow a non-technical domain expert, with medical training, to perform tweaks and customizations on the business data.

This approach is designed to easily provide a solution to current requirements and does not take specifications such as openEHR into account. The prototype created is used in the next chapter as a starting point for linking openEHR artifacts and also finding ways of using HL7 standardized documents for exchanging medical data.

When it comes to persistence solutions, the chapter presents a classical relational database example, coupled with an object-relational mapper to ease the link between relational data and object-oriented entities. Besides that, another example was given which uses a more modern, NoSQL, approach for storing electronic health records in document databases.

Chapter 4, *Integrating Medical Standards*, brings contributions on how an existing electronic health record system can be integrated with openEHR archetypes for extracting predefined medical rules and terminology links.

This chapter also emphasizes the role of medical coding and terminology standards. In the case of Logical Observation Identifiers Names and Codes, a system which has only been translated into few languages, the chapter brings an important original contribution by means of a study into how these codes can be more easily translated. The study shows a method which results in a decrease of 65% in the average length of each value that requires translation, together with a reduction by about 27% of the total number of unique values that need to be translated.

The same chapter also brings an original contribution in regards to International Classification for Diseases diagnostic codes. In this case it analyzes how ICD-10 codes could be matched by name using four full-text search engines (two dedicated ones and two part of relational databases). The results show the viability of using open-source full-text search engines for finding precise matches when searching such codes, indicating the possibility of using about 2.6 words, on average, for a query that uniquely identifies a code in 86% of cases (best-case scenario).

Lastly, the chapter proposes a solution for combining the elements above in an architecture which provides interoperability by generating specific connectors which consume or produce health data in the form of Clinical Document Architecture (CDA) documents.

Chapter 5, *Developing a Web Front-End for Electronic Health Records,* deals with the aspects of building a front-end for an application that deals with electronic health records.

The chapter compares multiple approaches and then proposes and implements a prototype of a flexible solution based on passing the information about the structure of each input form and its constituent input elements from the server to the client side code. The client side code, a single-page rich-client application, would then use that information to generate the required input elements on the fly and also link them with the required behavior.

As the front-end is the gateway to the application, seen from the eyes of the users, its usability is very important. The chapter thus contains contributions that analyze multiple documented ways of user interface evaluation, from heuristics, to aesthetics calculated from the geometry of a web page, all applied on a prototype application.

Another original contribution is presented in the form of describing how to build a simple yet powerful framework that tracks the users' actions. This framework is put to work in a user evaluation involving multiple medical doctors and residents, an evaluation which reveals important ways in which the web application is used and which are the areas for improvement.

Chapter 6, *Conclusions*, shows that the goals set out in the first chapter have been accomplished, the thesis providing important and original contributions to methods regarding the way electronic health record systems can be designed at both the business and data layers as well as concerning the user interface.

With the help of rapid prototyping, types of inputs have been stored in XML documents as soon as they have been identified, following the analysis of medical forms. The thesis proposes a flexible architecture for storing electronic health record inputs and also presents an implementation of the business and data layer. Another important goal that the thesis deals with is identifying ways of including medical standards, terminologies and specifications into an existing solution. When it comes to user interaction, an entire chapter is dedicated to building and evaluating the usability of a web application aimed at electronic health records.

On the basis of those presented, the list of claimed original contributions is as follows:

- A new architecture for the business layer of an electronic health record application based on placing the main focus on modeling the types of inputs
- Considerations on persisting electronic health records based on the above architecture using both relational databases and NoSQL
- Contributions on how an existing electronic health record system can be integrated with openEHR archetypes for extracting predefined medical rules and terminology links
- An architecture for generating code that makes use of clinical documents to transfer medical information between electronic health record systems
- A study on how to make use of patterns in order to reduce the amount of work required for translating LOINC codes
- An analysis on how ICD-10 codes can be matched by name using full-text search engines
- A prototype web application that uses metadata transferred from the server to generate EHR forms on-the-fly
- A heuristic and aesthetic analysis of a prototype EHR web application
- Ways of building a simple yet powerful framework for tracking user activity for the purpose of usability analysis
- Usability evaluation on a EHR web application combined with improvement suggestions

As with any research, various directions have been identified which warrant further studies. Among the first is a deeper dive into ways of automating the analysis of electronic health data/documents with the goal of being able to automatically generate connectors that import or export specific information. When it comes to medical codes, studies into optimal ways of matching them can be extended and compared for multiple languages besides English. In the case of usability evaluations of health record applications, recording the activity of users using a production system can reveal important new insights.