Software Requirements Specification (SRS)

EMR Data Analysis

Authors: James Drallos, Jordan Clare, Joseph Korolewicz, Daniel Laboy

Customer: Dr. Gary Ferenchick

Instructor: Dr. Betty Cheng

1 Introduction

This document outlines the software requirements for an electronic medical record (EMR) data analysis clinical decision support system. It will cover the overall description of the system, specific requirements as well as modeling requirements, diagrams and a description of a prototype to be built to demonstrate the system’s functionality.

1.1 Purpose

This document is intended to inform those who have a vested interest in the EMR data analysis clinical support system of its purpose and design. It should be useful to the customer(s) as well as any members of a software development team who are tasked to build and/or maintain the system itself. Users of the system need not concern themselves with the information in this document unless they desire a deeper understanding of the system’s architecture. However, this document does include a prototype, which demonstrates how the final system will look and function.

1.2 Scope

The software system of which this document concerns is an EMR data analysis-based clinical decision support system (CDSS). The system will be a software package that will be installed on a server machine and be accessible from any internet-enabled client machine in order to better assist medical personnel in the diagnosis and treatment of patients. It is intended to focus on patients exhibiting symptoms of Methicillin-Resistant Staphylococcus aureus (MRSA), a common bacteria which has become resistant to typically prescribed drugs. The system will provide access to patient data contained in an EMR such as symptoms, medical history and current prescriptions. The system will then evaluate known data about a patient and use it to suggest a case-specific treatment plan. The system will receive data from remote databases including relevant clinical trials that can be prescribed and medical best practice guidelines to be followed. Treatment will then be monitored. The system will improve the quality of medical care.
by providing a wealth of data from various sources to medical professionals, by promoting the use of best practices and tracking treatment.

1.3 Definitions, acronyms, and abbreviations

**Artifact** – A document or image containing the results of a medical treatment.

**Best Practices** – Generally accepted medical guidelines for the treatment of a condition.

**CA-MRSA** – Community-acquired MRSA, the typical case in which MRSA is contracted from the patient's community.

**CDSS** – Clinical Decision Support System. Software that generates case-specific advice based on a medical knowledge base and patient data.

**Clinical Trials** – Medical trials conducted to allow safety and efficacy data to be collected for new drugs or devices.

**EMR** – Electronic Medical Record. A digital form of a patient’s past medical information.

**EMR_DataAnalysisManager** - A class belonging to the system that manages the different subsystems.

**EMRSA** – Epidemic MRSA, a particular strain of MRSA heavily infecting a specific region.

**HA-MRSA** – Health-care-acquired MRSA, a strain of MRSA acquired from a health care facility.

**MRSA** – Methicillin-Resistant *Staphylococcus aureus*. The medical condition targeted by the system for evaluation.

**MRSA Health Surveillance System** – National surveys conducted on MRSA that provide a representative ‘snapshot’ of the overall epidemiology of MRSA.

**HL-7** – Health Level Seven standards for interoperability of health information technology.

**Patient History** – Data which includes a patient's problems, medications, allergies, family history, social history, allergies, etc.

**TreatmentManager** – A class belonging to the system that uses data to suggest a treatment plan.

**XAML** – Extensible Application Markup Language: is a declarative XML-based language which is used to initialize structured values and objects.

1.4 Organization

This document is organized in seven sections with various subsections describing various aspects of the system. Section two gives the overall description of the product, including major functions and constraints. Section three lists specific requirements, section four contains diagrams outlining the design and use of the system, section five describes the product prototype, section six lists references and section seven gives the point of contact for questions about the product.

2 Overall Description

This section contains the complete description of the project. It states the perspective of the system, how the system is just one part of a complete EMR. This section also shows the Template based on IEEE Std 830-1998 for SRS. Modifications (content and ordering of information) have been made by Betty H.C. Cheng, Michigan State University (chengb at chengb.cse.msu.edu)
constraints faced by the system, and the assumptions that were made both about the systems it will be interacting with, and the people who will be using the system.

### 2.1 Product Perspective

The system is the data analysis portion of an electronic medical record. The system will, given a diagnosis from a doctor about a certain patient, find and create a relevant treatment plan for that patient. The system will suggest these treatments using information regarding patient specific information, medical best practices and relevant clinical trials. As seen in figure 2.1, the TreatmentManager will get all of this information and then be able to design a treatment.

![Figure 2.1. TreatmentManager](image)

The system is just one part of a complete EMR system. The other pieces of an EMR system are the data collection and data dissemination pieces respectively. The data collection piece of the system would be required to collect all of the input given by medical professionals, and store all of this data into an accessible location. The project would use this information and make a treatment decision using all of this data. The other piece of a successful EMR, the data dissemination piece, would take the information, and find current medical trends. It would check whether people from a certain area were more prone to a certain disease. See Figure 2.2 for an example of a complete EMR system. Notice that the data dissemination and data analysis pieces run concurrently. This is because both systems take the same data from the data collection section, and use it in different ways.

![Figure 2.2. Complete EMR System](image)
For the system to work, there must be certain information present. There must be databases set up for patient history, medical best practices, and clinical trials. The system would then check these databases, and pull out relevant information. It would then use this information to build a treatment that would be customized for the current patient and current state of the medical field.

2.2 Product Functions

- The system shall provide customized treatment plans for patients who have been diagnosed with different diseases.

- The system shall confirm a clinical diagnosis given by a medical professional by checking the symptoms present with the diagnosis.

- The system shall provide access to relevant clinical trials for a given patient, and give a medical professional the choice of whether or not he or she would like to try and use this particular trial information for their current treatment or enter the patient into said trial.

- The system shall provide a plan for preventive care for its patients.

- The system shall show the medical best practices of a given diagnosis, and remind a physician as to what guidelines have not yet been completed.

- The system shall track previously diagnosed patients, and check to see if there are any treatments or diagnostic studies that still need to be done.

- The system shall be able to differentiate whether a patient has a certain kind of MRSA, and how to treat that patient.

- The system shall abide by HL-7 standards.

2.3 User Characteristics

This system has certain expectations of the user. The user must be able to properly input the symptoms or diagnosis, because if any incorrect input were put into the system, the treatment plan provided by the system would not be valid. Also, the user needs to have some kind of medical experience, and must have a certain level of certification as defined by the medical practice using the system. The user needs to be able to understand the treatment provided, and can be able to act upon that treatment accordingly.
2.4 Constraints

- The system must have access to databases that contain medical records of patients, relevant clinical trials, and medical best practices for different diseases.
- The system will only work for diseases that are currently in the databases.
- The input into the system must be correct, if a doctor enters a wrong diagnosis, the treatment provided will not be relevant to the current patient.

2.5 Assumptions and Dependencies

The system assumes that there are databases set up that hold patients medical records, ongoing clinical trials, and medical guidelines for specific diseases, and that the system has access to these pieces of information. Without this information, the system has no information to base its recommendations for treatment.

2.6 Approportioning of Requirements

The customer stated that it would be ideal for the project to contain voice recognition for a method of input; however, this cannot be implemented due to time constraints.

3 Specific Requirements

The following is a list of specific requirements that the system will model its behavior from.

1. Manage clinical information:

   1.1 Identify and prescribe relevant clinical trials / protocols:

   The system will query the list of clinical trials hosted on www.clinicaltrials.gov and determine if there are any ongoing trials that are relevant to the patient’s diagnosis and, if so, will display information about the trial to the physician.

   1.2 Track orders for treatment:

   The system will allow the user to view information regarding both the patient’s treatment history, as well as any treatment that has been scheduled. Information about previously administered treatments will be available in a variety of forms depending on the type of treatment, for example X-rays, lab reports, or prescriptions. Information pertaining to treatments that have been scheduled to take place sometime in the future will contain the proposed date and location, as well as any notes the physician has entered.
regarding treatment customization. Once the treatment has been completed, the relevant artifact will be available for viewing.

1.3 Referral follow-up:

The system will allow the user to view information about all treatments the patient received, including treatments that were referred to a medical facility using an identical or compatible EMR system.

1.4 Prescribe preventative care:

The system will provide the user with general guidelines for relevant preventative care for a given diagnosis.

2. Decision Support:

2.1 Confirm a clinical diagnosis:

The system will compare the signs, symptoms and risk factors the user has acquired based on a patient’s symptoms and compare them against a database of symptoms for a given diagnosis. If the system finds the diagnosis to be inconsistent with symptoms, then the system will display a message to that effect as a warning to the user.

2.2 Suggest treatment plan processes:

The system, when given a diagnosis, will compare treatments the patient has already received to a list of possible treatments for that diagnosis. The system will then suggest the relevant treatment plan and call attention to elements of the plan that have not already been administered.

2.3 Promote the use of “best practices”:

The system, when presented with a diagnosis, will present a list of generally accepted treatments for the diagnosis. The list will be retrieved from a database that contains a list of best practice guidelines and is maintained by the system.

2.4 Customize treatment according to specific conditions:

The system will allow the user to enter any notes or special instructions (i.e., allergies, recent trauma, etc.) when scheduling a treatment. Any special instructions that a user adds will be viewable by any other user observing the patient's medical history.

2.5 Distinguish between various types of MRSA:

2.5.1 Determine if MRSA is Health-Care acquired (HA-MRSA)
The system, when presented with a diagnosis of MRSA, will check its database to see if the patient has been admitted in the last 48 hours or has been in staying in the hospital. If so, the system will display a message to the user indicating the possibility of HA-MRSA.

2.5.2 Determine if MRSA is the result of an epidemic (EMRSA)

The system, when presented with a diagnosis of MRSA, will check the MRSA Health Surveillance System to see if there has been an unusually large number of MRSA cases in the area, which is consistent with a diagnosis of Epidemic-MRSA.

2.5.3 Determine if MRSA is Community acquired (CA-MRSA)

The system, when presented with a diagnosis of MRSA, will conclude that if MRSA was not contracted from a health care facility or from an epidemic in the area, that the type is CA-MRSA, acquired from the patient’s community.

3. Compliance with HL-7 standards

The system will conform to HL-7 standards in all data storage and message passing protocols.
4 Modeling Requirements

Figure 4.1. Use Case Diagram

Figure 4.1 depicts the typical use cases of the system. A user is able to view patient info, schedule treatments for patients in the system, and enter diagnoses for patients in the system. As shown, when a diagnosis is made the system confirms the diagnosis, gets a list of best practices, and relevant sensitivity charts for various medications.
Figure 4.2 (below) is a UML class diagram for the system. This diagram shows the relationships between the classes as well the attributes and methods each class contains. The PatientInfo class is a collection of all the relevant information about a patient such as height, weight, vitals, as well as a reference to the patient’s medical history which will have all the information about previous treatments, diagnoses, as well as current symptoms. The Diagnosis class contains a reference to the BestPractices class is a collection of widely approved methods for treating a given diagnosis. The EMR_DataAnalysisManager coordinates the different subsystems and lets the user move easily between various tasks such as looking at a patient’s history and scheduling a new treatment. The TreatmentManager is the core of the system. The TreatmentManager provides the vast majority of the functionality of the system and is invoked whenever a user makes a diagnosis, schedules a treatment, or wishes to view details about previous treatments. Each instance of the Treatment class, which stores information about a single treatment that a user can administer, also has zero or more Artifacts associated with it. The Artifact class has a digital copy of various results of a given treatment, such as an X-Ray, lab result, or copy of a prescription.

Figure 4.2. EMR – Data Analysis Class Diagram
Figure 4.3 is a sequence diagram displaying one example use of the system from the initial login until treatmentManager has finished initializing. Each box at the top represents an object in the system with an object lifeline protruding from the bottom. This lifeline represents the objects behavior over time. The solid arrows represent the invocations of the objects methods while the dotted arrows signify a return from said method.
Figure 4.4 is a sequence diagram illustrating a scenario in which the treatmentManager has already been launched. In this example the user views the details of a past treatment, its resulting artifact, schedules a new treatment and prescribes a medication.

Figure 4.4 Sequence Diagram 2
Figure 4.5. (below) is the state diagram for the TreatmentManager class. Each of the round-tangles represents a possible state of the object. The arrows between these round-tangles are the transitions by which the object changes states. These transitions are labeled with the action that causes the transition as well as any conditions that must be satisfied in order for the transition to occur. Initially whenever the TreatmentManager is launched it enters the Analyze state. In this state various checks and analyses are performed such as confirming the user’s diagnosis and in the event of a MRSA diagnosis determining the type of MRSA the patient may have. Once these steps are complete the TreatmentManager transitions into the View Treatment Plan/Progress state. From here the user can view the progress of treatments that have already been carried out, see a list of upcoming treatments, view a list of relevant clinical trials, as well as choose to schedule a new treatment for the patient. If the user chooses to view the progress of a treatment that has already been scheduled the TreatmentManager enters the Detailed View state and from here the user can view more detailed information about the treatment as well as any artifacts if the treatment has already been completed. If the user chooses to schedule a treatment, then the TreatmentManager enters the Schedule Treatment state and the user can then schedule a new treatment.

Figure 4.5. TreatmentManager State Diagram

Template based on IEEE Std 830-1998 for SRS. Modifications (content and ordering of information) have been made by Betty H.C. Cheng, Michigan State University (chengb at chengb.cse.msu.edu)
5 Prototype

The prototype will illustrate the interaction between a healthcare provider and the EMR – Data Analysis system to diagnose a case of a type of MRSA. This process will show how the doctor will be able to use the EMR – Data Analysis system to create, customize, and maintain a treatment plan for the specific patient relevant to the diagnosis, along with decision support provided by the software.

5.1 How to Run Prototype

The prototype will be implemented using Microsoft's Silverlight, version 3. Microsoft Silverlight is a cross-platform, cross-browser plug-in that is similar to the more widely used Adobe Flash. Silverlight was used for this application because it allows for easy creation and customization of rich application user interfaces either directly using XAML encoding or through a separate program developed by Microsoft (Microsoft's Blend) to facilitate the design and customization of applications using Silverlight. Silverlight also allows for implementation of custom interactions within the user interface elements by utilizing code behind files written in the C# programming language. Another reason for the use of Silverlight is that the default toolkit of user interface elements provided to design user interfaces is simple and has high aesthetic value.

In order to run the application, the only requirements are that the user is on a computer with an updated browser (Firefox, Safari, Internet Explorer, etc.) and that they download a small (4.3 MB) plug-in when they navigate to the application for the first time. Silverlight is not platform or browser specific, so there are no hardware, networking or operating system requirements.

The prototype can be accessed through the web via a link on the team website at http://www.cse.msu.edu/~cse435/Projects/F09/EMR-Analysis/web/.
5.2 Sample Scenario

When a doctor, or any other healthcare provider, starts the prototype he/she will first encounter the log in screen shown below in Figure 5.1. The doctor will enter his/her user credentials in the appropriate fields and then click the “Log In” button, which will authenticate on the main EMR server.

![Log In page of prototype](image)

Figure 5.1 – Log In page of prototype
Upon logging in to the system, the user will next encounter the home screen shown in Figure 5.2 below. This home screen will have a table in the upper third of the screen that contains a list of all patients accessible to the doctor. Below the patient list will be the patient detail view area. When the doctor selects a patient from the list, the detail view area will show important information about the selected patient, including: full name, measured height and weight, any recent vitals measurements, and a brief overview of the patients medical history.

![Figure 5.2 – Patient Select page of prototype](image-url)
After the user has selected a patient and has completed any review of information contained in the detail view, the user can click on the button at the bottom of the screen, which brings him/her to the diagnosis screen, shown in Figure 5.3. Considering the scope of the project, Analysis and Decision Support, the design of this page is very simple and consists of a single text box and a “Next” button. The user will enter his/her diagnosis (as shown in the diagram, MRSA) in this box and click the button to continue to the next screen.

![Diagram of Diagnosis Entry page of prototype](image)

Figure 5.3 – Diagnosis Entry page of prototype
After submitting the diagnosis, the program submits the information to the data server for cataloging and queries all relevant information on the given diagnosis and patient. The following screen, labeled the Treatment screen, shown in Figure 5.4, will display the received information from the previous query for the user to review. Next, we describe each region displayed in further detail.

Figure 5.4 – Treatment Planning page of prototype
The information contained in the table in the top left of the screen, shown in Figure 5.5, is the “Best Practices” of the corresponding diagnosis. This table lists the most widely accepted treatment plan for the diagnosis, and indicates, with green check marks, if any of those steps have been completed for the patient. Any incomplete tasks will be marked with an “X” mark and highlighted in red. Any tasks in the “Best Practices” list that have been scheduled for the patient, but have not yet been completed are highlighted in blue with a “?” symbol.

**Best Practices:**

<table>
<thead>
<tr>
<th>Status</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>✔️</td>
<td>Isolate Patient</td>
</tr>
<tr>
<td>🟡</td>
<td>Take Surveillance Culture</td>
</tr>
<tr>
<td>✗</td>
<td>Prescribe MRSA Medication</td>
</tr>
</tbody>
</table>

Figure 5.5 – Best Practices status view

In the bottom left of the screen, shown in Figure 5.6, are two tables labeled “Tasks Completed” and “Tasks Scheduled.”

**Tasks Completed:**

<table>
<thead>
<tr>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blood Test</td>
</tr>
</tbody>
</table>

**Tasks Scheduled:**

<table>
<thead>
<tr>
<th>Date Sched.</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>12/10/2009 5:50:01 PM</td>
<td>Bacteria Culture</td>
</tr>
</tbody>
</table>

Figure 5.6 – Completed and Scheduled Treatment status view

Template based on IEEE Std 830-1998 for SRS. Modifications (content and ordering of information) have been made by Betty H.C. Cheng, Michigan State University (chengb at chengb.cse.msu.edu)
The first of these tables, Tasks Completed, lists any and all work orders that have been performed on the patient. The user can select any of the items in this list, and click the “View Details” button below the table to bring up a window that will display the results and details of the work order, shown in Figure 5.7.

![Figure 5.7 – Completed Task Details view](image)

The second of these tables, Tasks Scheduled, lists all work orders that have been scheduled for the patient and are pending completion. The user can click “Add Task…” to schedule an additional task to be completed on the patient, shown in Figure 5.8. The user can also select an existing scheduled task from the list and click “Remove” to remove the task from the schedule.

![Figure 5.8 – Add New Task window](image)
In the upper right corner of the screen is a table, illustrated in Figure 5.9, indicating all, if any, medications to which the patient is currently prescribed. This list alerts the user to any combination of medications to which the patient has been prescribed that have known side effects or adverse reactions when taken in conjunction with one other by highlighting the conflicting drugs in yellow with a “!” caution marker. Selecting a medication and then clicking the “Remove” button will remove the medication from the patient’s prescribed medication list.

Figure 5.9 – Current Medication List view
Selecting a medication in the list will also enable the “Detail View” button below the list. Clicking the button will display a window, shown in Figure 5.10, which contains all details of the medication, including common side effects, known drug-drug conflicts, etc.

![Medication Details](image)

**Figure 5.10 – Medication Detail view**

Below the medication table is an optional section for displaying any relevant clinical trials associated with the diagnosis, shown in Figure 5.11. Double-clicking a listed trial will navigate the user to the selected trial's information page. If there are no applicable trials, then this section will be grayed out and disabled.

![Trials](image)

**Figure 5.11 – Relevant Trials List view**

Template based on IEEE Std 830-1998 for SRS. Modifications (content and ordering of information) have been made by Betty H.C. Cheng, Michigan State University (chengb at chengb.cse.msu.edu)
6 References


7 Point of Contact

For further information regarding this document and project, please contact Prof. Betty H.C. Cheng at Michigan State University (chengb at cse.msu.edu). All materials in this document have been sanitized for proprietary data. The students and the instructor gratefully acknowledge the participation of our industrial collaborators.