PatientBank: CPSC 490 Proposal Fall 2014

Student: Feridun Mert Celebi

Group Members: Abhishek Chandra, Paul Fletcher-Hill, Mikayla Thompson

Advisor: Ruzica Piskac

PatientBank is a platform for individuals to aggregate their own health information. Especially for high utilizers of health care, comprehensive health records are incredibly valuable. While electronic health records have become more prevalent, patients’ access to their own information remains limited. PatientBank solves this problem for the patient by requesting, gathering, and maintaining a comprehensive and unified record on his or her behalf.

Background
The first version of PatientBank was developed for Professor Piskac’s CPSC 439 Software Engineering class in the spring of 2014. The project included a flexible framework for storing health records, a personal health record web interface, and a simple XML-based data integration algorithm. The original team included Feridun Mert Celebi, Paul Fletcher-Hill, and Rafi Khan. Over the summer of 2014, Celebi and Fletcher-Hill continued to develop parts of the application, including two satellite applications for administrators to view patient information and providers to upload health records.

In the fall of 2014, Abhishek Chandra and Mikayla Thompson will join the project as collaborators. Chandra gained exposure to the project in the spring of 2014, when he worked on another team in the same CPSC 439 Software Engineering class, and Thompson spent the summer of 2014 at the electronic medical records giant, Epic Systems, where she developed an interest in health information technology. The team’s intentions for CPSC 490 are to continue developing the core PatientBank functionality, while each focusing on a single part of the growing platform. These individual projects will be detailed later in this proposal.

One might wonder why four students have chosen to work as a group rather than on four individual projects. The answers lies in the fact that PatientBank is a uniquely positioned project that balances practical context with technical complexity. Collaboration is an exciting and enjoyable process that often lends spectacular results, and the project is at a point in its development where there are enough unique, isolated projects to keep four students interested and busy for the semester.

(Current) System Design
Representing and storing health records is a very challenging task, given the wide variety in health record formats. Patients often have vital signs, medications, and more, but the details of how each of these components is manifested in a database is not standardized. One approach could have been to explicitly define tables for each section (a medications table, for instance) of a record and hard-code the nested relationships between them all. This method seemed highly redundant to us, so we decided to abstract away the details of this graph into
two types of objects: record sections and record section fields. Each record section can be thought of as a table, where the record section fields are columns in that table. The rows in the record section “table” are record updates—timestamped data objects, each associated with a patient.

This data model also allows us to differentiate between generalized data and patient-specific data. The actual structure of a health record—the nested record sections and their fields—is not unique to a patient, while the record updates. Storing these record updates in their own table simplifies the design and increases performance (due to easy indexing).

(Current) Implementation Details
The current implementation of the project utilizes the following technologies:

- Ruby on Rails: *API and backend framework*
- PostgreSQL: *Primary relational database*
- Redis: *In-memory database for storing events and notifications*
- Solr: *Indexing data for full-text search*
- Angular.js: *Frontend framework*

Projects Overview
When patient health records are requested, providers’ responses come in two forms: PDF and XML files. In order to be useful to all patients, PatientBank should be able to accommodate all forms of records, so the first three projects include methods for integrating these distinct health record forms. The final project has to do with the security of the platform—especially in health care, an industry with high stakes and strict laws, security is incredibly important. Feridun Mert Celebi will study encryption techniques and develop a framework for formally verifying HIPAA compliance.

Project 4: Security (Feridun Mert Celebi)
Problem and motivation
PatientBank deals with vast amounts of sensitive, personal information (i.e. Personal Health Information). Dealing with such information raises well-deserved concerns regarding the security of the system. Hence, it is of utmost significance to maintain high standards for the security of PatientBank, while preserving the performance of database read and write operations.

Health Insurance Portability and Accountability Act (HIPAA) provides a framework for the security of systems that deal with sensitive information. So far, PatientBank has been utilizing basic encryption techniques to meet HIPAA security and privacy requirements. Unfortunately, vast use of encryption techniques introduces a variety of database performance disadvantages. One of the to-be-implemented features that HIPAA mandates is encryption for in-transit data to prevent disclosure of sensitive information during data transfer from our Core API to our Angular.js applications.
Description

There are both practical and academics aspects of this project. The practical part of the project involves three sub-parts that would narrow down the focus of work to be implemented and make it more accomplishable:

1. Explore and implement various encryption techniques for data in-transit without sacrificing our database performance.
2. Although it is ideal to have complete security, it would be unrealistic to assume that would be the case as there are many techniques to cause security breaches. Hence, as HIPAA mandates, it is also crucial to improve our auditing and logging capabilities to detect any security breaches.
3. Testing the security and performance of features to-be-implemented. This part would involve learning new penetration testing techniques and the creation of performance benchmarks, along with investing time to produce bi-monthly risk analysis reports of PatientBank.

The academic part of the project would focus on formal verification methods to measure the compliance of the PatientBank platform, given HIPAA security and privacy rules as a set of constraints. With the help of our advisor, Ruzika Piskac, we have had the opportunity to explore the papers and projects that have focused on formal verification methods for health care problems and pinned down three researchers: Anupam Datta from Carnegie Mellon University, David Kotz from Dartmouth College and Helen Nissenbaum from New York University. Two main papers that Professor Piskac pointed out lay a solid foundation for us to tackle formalizing HIPAA security and privacy rules1,2.

Goals of the project:

● Explore and implement different encryption techniques for in-transit data that would also yield the maximum database performance.
● Improve the auditing and logging capabilities of PatientBank to detect sources of data loss.
● Test the security of the application via different penetration testing mechanisms, create performance benchmarks/data, and start producing bi-monthly risk analysis reports.
● Develop formal verification techniques for proving PatientBank meets HIPAA requirements.

---

1 [http://www.andrew.cmu.edu/user/danupam/TschantzDattaWing12.pdf](http://www.andrew.cmu.edu/user/danupam/TschantzDattaWing12.pdf)
2 [http://link.springer.com/chapter/10.1007%2F978-3-642-22438-6_4#page-1](http://link.springer.com/chapter/10.1007%2F978-3-642-22438-6_4#page-1)