

Project Proposal

Team SubjectToChange

Team Membership:

- Michelle Duan (904744846)
- Lauren Fromm (404751250)
- David Lee (804728489)
- Pierson Marks (304742143)
- Kevin Tolby (104666258)

Project Description:

Medio is an application that helps solve the inefficient process of coordinating medical appointments. The primary feature is a collaborative calendar for coordinating appointment times. Medical practices upload calendars with availabilities that patients can then use to schedule and confirm times without the need for back-and-forth communication between the practice's secretary. Additionally, our platform will have the ability to host many different medical practices with each having its own profile page. For example, say a user wants to schedule an appointment at Ronald Reagan Hospital. They would log in, find the calendar for the hospital, and mark a free region of the calendar. When the medical practice logs in, they would then see the patient's marked appointment on the calendar they uploaded.

Other possible features include patient profiles containing personal medical information and the ability of patients to view medical records online, including reports from their previous doctor's visit. This eliminates data-siloing and democratizes medical records, allowing for the seamless transfer of records between practices and an organized timeline of medical history.

We will build our platform using a WebApp with Ruby on Rails as our framework. Our team members seem to have the experience necessary to build this application in the allotted time frame, with an emphasis on secure coding practices. At early stages of development, we may just run the application locally for testing purposes. Later, we plan to deploy our application via Heroku (or another free/easy-to-use platform) so other teams can easily interact with our program.

Privacy and Security Issues:

As with most web applications, our program has many privacy and security issues we need to address. Because our program deals with sensitive medical information, we must ensure that this data is only accessed by users authorized to view that data. Patients should only be able to view their own medical data (not anyone else's) and medical practices should only be able to view their own patient's medical data. The patient may also be able to selectively authorize what data is available to what practice. These concerns will be answered via two-factor authentication to minimize the possibility of accounts being compromised, relying on already developed secure

authentication solutions instead of developing our own, proper functional dependencies that will correctly associate medical data with the patient, and PK encryption to protect sensitive information from being transmitted in plaintext over a (possible) insecure network.

Other issues we potentially may see are race conditions that occur when two users select the same appointment timeslot simultaneously for a certain medical practice or the race condition when a user changes medical practices and needs to revoke the former practice's access to their data. A solution would be to check both on read and on write the permissions of the medical practice/patient. Additionally, we may have to sanitize all inputs into our application (via parameterized queries) to prevent attacks such as SQL-injections or cross-site scripting. For further security, we plan to initialize all variables upon declaration, keep our implementation simple, constantly refer to security standards, include risk analysis at every sprint's retrospective, check return codes to have better error handling, and not use libraries/languages that are known to have a high likelihood of security issues.

Schedule:

- **End of Week 2:**
 - Finalize application architecture.
 - Create the database design/schema.
 - Get a decent grasp of Ruby on Rails.
 - Deploy a blank rails application locally.
- **End of Week 3:**
 - Deliver design document.
 - Complete basic skeleton of app.
 - Review security/privacy issues.
- **End of Week 4:**
 - Complete patient functionality.
 - Complete collaborative calendar feature.
 - Review security/privacy issues.
- **End of Week 5:**
 - Complete medical practice functionality.
 - Review security/privacy issues.
- **End of Week 6:**
 - Complete document upload/sharing feature.
 - Review security/privacy issues.
- **End of Week 7:**
 - Final tests/security review.
 - Demo prototype.

Work Division

- Michelle: Patient profiles
- Lauren: Collaborative calendar
- David: Doctor profiles
- Pierson: Authentication/Login, security reviews of features, database schema
- Kevin: Document upload/sharing