

Bamboo Mobile Health App Proposal

Team

Kalpesh Padia, Robert Dates, Pankti Desai, Yu-Ching Hu, Shreyas Zagade, Pranav Firake

Tagline

The interaction of a patient and a doctor is vital when making shared decisions about healthcare. This is even more important in the case of patients suffering with chronic diseases like Multiple Sclerosis. Existing mobile technology allows patients to create a log of symptoms and their efforts in alleviating them. However, a gap still exists in care and management of symptoms due to the inability of the patients to effectively express their symptoms to their doctors and healthcare providers.

We aim at solving this problem by developing a mobile application that will create and present a succinct view of the recorded symptoms and activities to both the patient as well as his doctor. This will enable more efficient communication between them and allow the patient to obtain care that will improve their quality of life.

Approach

Patients suffering from chronic diseases like Multiple Sclerosis often install health monitoring and symptom tracking applications to enable them to keep track of their day to day activities and symptoms. However, they find it difficult to convey their symptoms and activity schedule to doctors during their regular visits for managing their disease.

We believe that this can be attributed to the inefficient presentation of health information and inadequate progress reporting to the user by the application. This further creates a challenge when the patient wants to share a record of his activity and symptoms with his healthcare provider to aid in diagnosis and devise an appropriate care plan.

In this project we will build a prototype of an application that will present patient symptom and activity data gathered from an existing mobile health application "Bamboo Mobile Health". Instead of focusing on data entry and collection, we are focusing on designing effective presentation and reporting of patient data using visualizations on a mobile device. This decision was taken based on the client requirements and given time-constraints.

Our mobile application will receive data from a backend server populated with existing timestamped patient data for symptoms: nature, severity, and occurrence; activities: type and frequency; and personal goals (pain management, increased activity, etc.). Based on this data, our application will create personalized visual reports that will allow the users to track and view the history and progress of their symptoms and activities undertaken to manage those symptoms.

References

Below is a small list of references that we take inspiration from for the design and development of the mobile application in this project.

1. Visual elements and visualizations:
 - a. [Visualizing information on mobile devices](#)
 - b. <https://www.pinterest.com/timoa/mobile-ui-progress-bar/>
 - c. <https://chaione.com/blog/tips-data-visualization-mobile-devices/>
2. UI Design Guidelines
 - a. <https://developer.apple.com/design/tips/>
 - b. <https://developer.apple.com/design/>
 - c. <https://medium.muz.li/top-9-ui-design-trends-for-mobile-apps-in-2018-14b4fa350d3a>

Technologies

Mobile Application: Swift

Backend: Google Firebase

Development Milestones

Our project will require the development of two components, a mobile application, and a backend. Below we have provided a high-level breakdown of how each component will be developed.

Mobile Application

A1: Wireframes and UI Design

- Identify the views and screens the app will contain
- Identify the various visualizations presented in the app
- Create wireframes for the various screens
- Design and sketch visualizations for progress and history views

A2: User login, authentication, and user profile

- Develop the views for user login
- Connect to backend for authentication
- Develop the views for creating user profile and capturing relevant one-time information

A3: Record and save audio clips

- Develop the views for allowing user to record and save audio clips as part of daily activity record

A4: Progress view (static)

- Develop the views for showing and comparing progress of actual v/s planned exercises and activity minutes
- Develop and show static visualizations

A5: Progress view (data from DB)

- Develop and show visualizations based on data pulled from a backend

A6: History view (static)

- Develop the views for showing a historical view of exercises and symptoms
- Show static visualizations

- Playback recorded audio clips

A7: History view (data from DB)

- Develop and show visualizations based on data pulled from a backend

A8: Extra Credit: Reward Gems

- Award reward “gems” based on activity and exercise history. Store these gems as part of user profile and display to user

Backend

B1: Backend Design

- Identify data attributes
- Design database schema

B2: APIs for user authentication and user profile

- Author necessary APIs to support user authentication and save profile information

B3: Import mock data

- Check client-provided data to ensure data types correspond
- Potentially write script to import data

B4: APIs for progress data retrieval

- Author necessary APIs for progress data retrieval

B5: APIs for history data retrieval

- Author necessary APIs for historical data retrieval

B6: Extra Credit: Speech to Text conversion

- Use relevant cloud APIs to perform Speech to Text conversion on the recorded audio files, and store the transcripts on backend
- Provide access to transcripts

B7: Extra Credit: Analysis on converted text

- Perform appropriate analysis on transcripts to identify common patterns
- Provide access to results of analysis in the form of textual/graphical reports.

Grading Milestones

Milestone	Tasks Implemented	% Credit (of final grade)
Milestone 1	A1, B1	5 + 5 = 10
Milestone 2	A2, B2	5 + 5 = 10
Milestone 3	A3, B3	5 + 5 = 10
Milestone 4	A4, B4	10 + 10 = 20
Milestone 5	A5, A6, B5	10 + 10 + 10 = 30
Milestone 6	A7	10

Lightweight Evaluation	Evaluation of the developed app by the client and select alpha testers	10
Milestone 7 (Extra Credit)	A8, B6, B7	5 + 10 + 10 = 25

Gantt Chart of Tentative Deadlines

