GitHub Username: aznxed

MyHealth

Description

MyHealth is a health management application. It allows users to track and manage their healthcare through their android device.

Intended User

MyHealth is intended for patient use as well as healthcare provider use.

Features

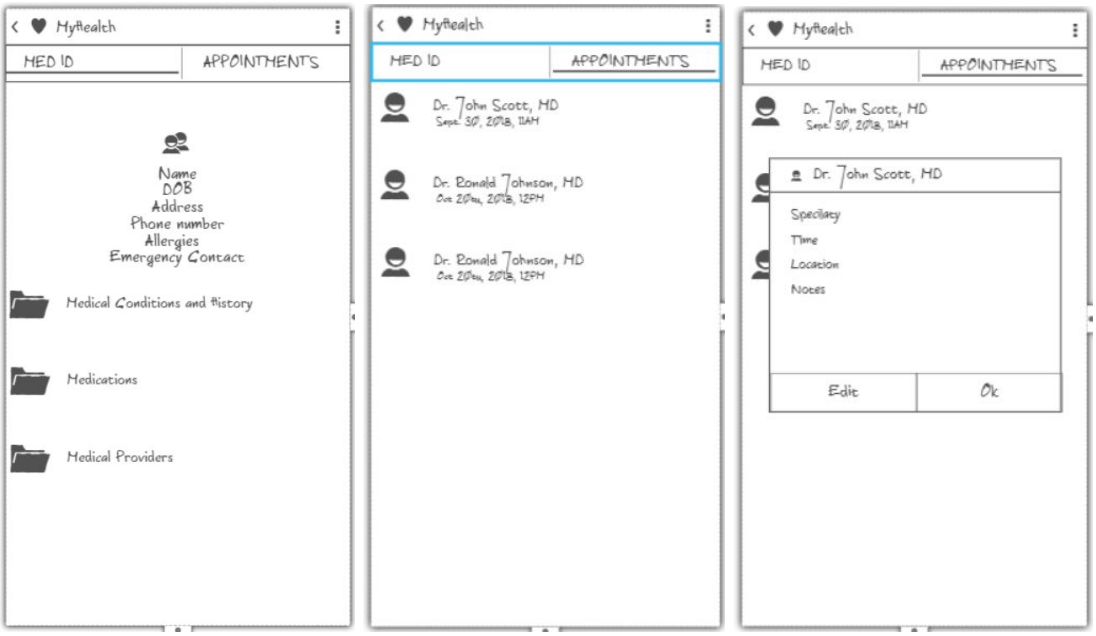
- App is written solely in the Java Programming Language.
- App keeps all strings in a strings.xml file and enables RTL layout switching on all layouts.
- App includes support for accessibility
- App provides a widget that lists scheduled appointments
- Patients will be able to
    - List and track their medications
    - Record their medical issues and history
    - View and request appointments
- Providers will be able to
    - View scheduled patients and their information
    - Respond to appointment requests

User Interface Mocks

Screen 1 - Login Screen

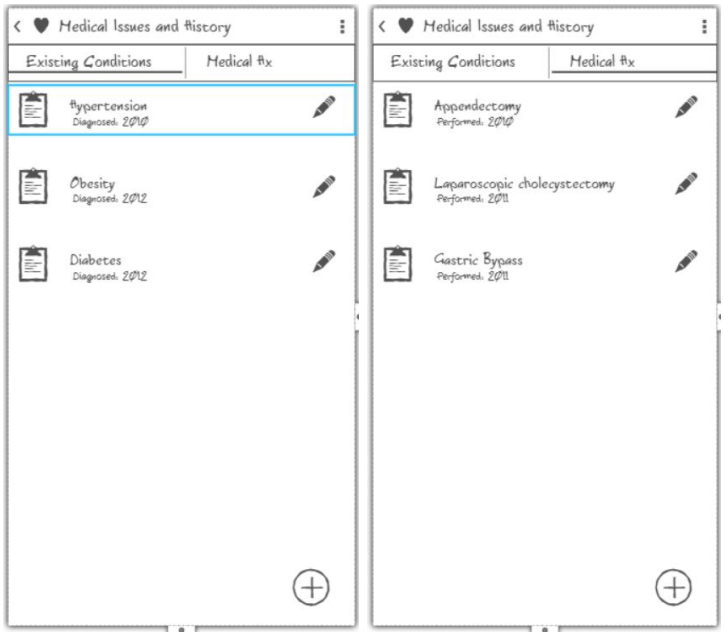
Users will be greeted with a login screen, in which they can login or create a new account.

Screen 2 - Medical ID and Appointments Screen



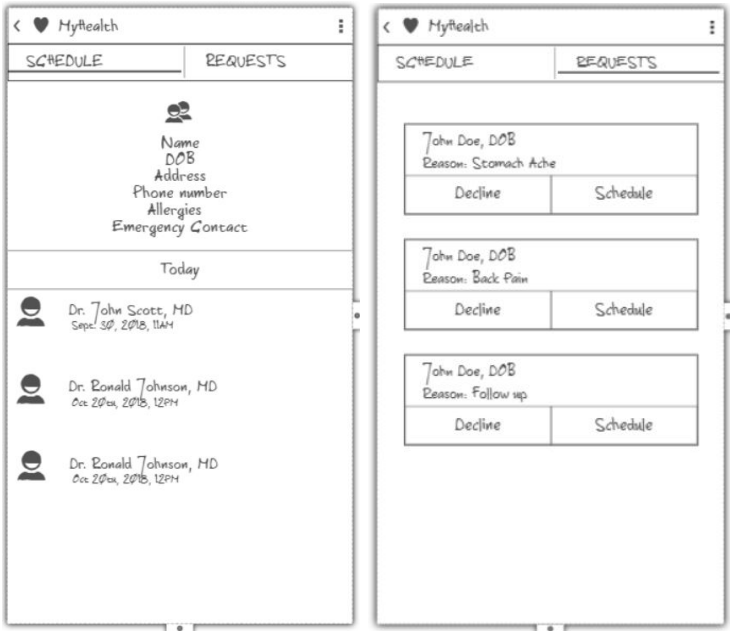
After login, users will be able to view their medical identification, along with navigation to other important medical information. Users can also use the viewpager to swipe to or click to view their appointments.

Screen 3 - Medical Conditions and History Screen



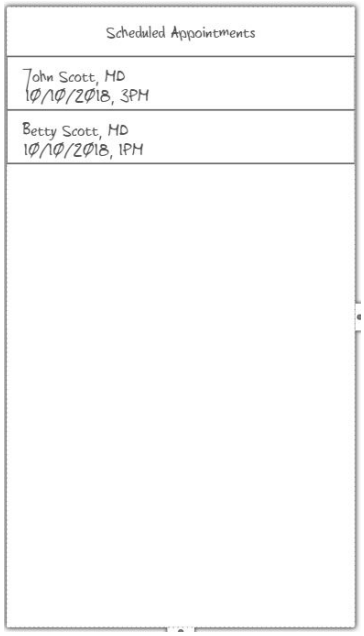
Patients will be able to document and view their existing and historical medical history.

Screen 4 - View Schedule and Requests Screens



Providers will be able to see their patients listed, with a various options available, such as date and time look ahead. They can select the patient to view more details regarding the patient. They can also swipe to the right for appointment requests.

Screen 5 - Widget View Schedule Screen



MyHealth will include a widget that shows a providers schedule for the day or a patient's appointments for the day.

Key Considerations

How will your app handle data persistence?

Patients will be able to keep a copy of their information on their device via a content provider. Data will be pushed to Firebase Realtime Database.

Data will be retrieved via AsyncTasks for the application. An IntentService will be used to fetch data for the app widget.

Describe any edge or corner cases in the UX.

None that I can think of.

Describe any libraries you'll be using and share your reasoning for including them.

- Picasso - for image loading
- Butterknife - for view binding
- Google Design - for building material views

Describe how you will implement Google Play Services or other external services.

- Firebase SDK Authentication - for registering and saving patients
- Firebase Realtime Database - for storing patient information

.

Next Steps: Required Tasks

Task 1: Setup Firebase SDK Authentication

I will be following the directions here for setting up Firebase SDK Authentication on Android.

Task 2: Setup Firebase Realtime Database

I will be following the directions here for setting up Firebase Realtime Database on Android.

Task 3: Setup Multi Variants

I will be setting up two flavors of the app, one for providers and one for patients.

Task 4: Setup Views and Activities

Setup views for all activities

- MainActivity
- AppointmentActivity
- MedicationsActivity
- Etc.

Task 5: Create Widget

Create a widget provider that uses a listView to display upcoming appointments for providers and patients.

Task 6: Make App Material

Style designs per Material Design Guidelines.

Task 7: Create Android Tests and Java Tests

Create tests to ensure correct interactions and application logic.

Wireframes created using NinjaMock.