

How to Use this Template

1. Create a new document, and copy and paste the text from this template into your new document [Select All → Copy → Paste into new document]
 2. Name your document file: “**Capstone_Stage1**”
 3. Replace the text in green
-

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any corner cases in the UX.](#)

[Describe any libraries you'll be using and share your reasoning for including them.](#)

[Describe how you will implement Google Play Services.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

GitHub Username: [MercyMutuku](#)

Patient Tracker

Description

Patient Tracker is an App intended to aid in hospital / dispensary / clinic patient data management.

Intended User

This app is meant to be used by people with the receptionist, doctor and pharmacist roles in a health facility.

Features

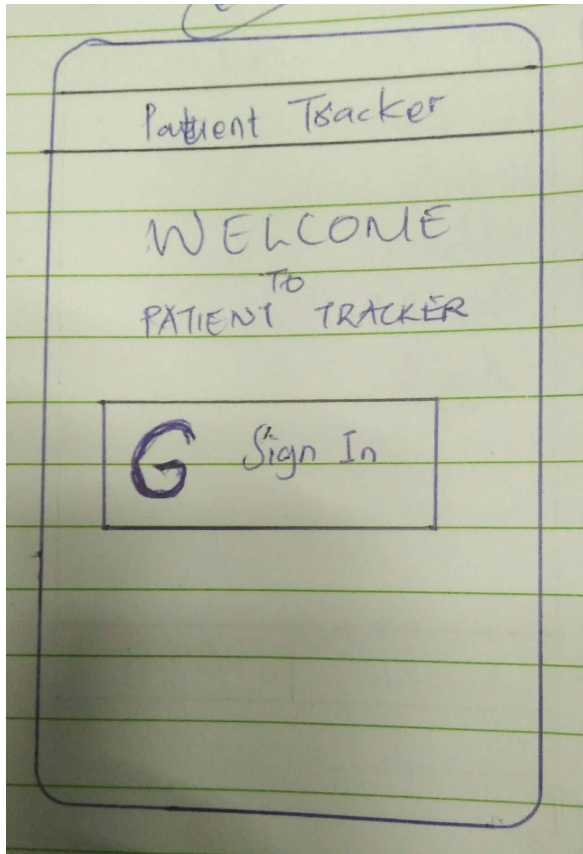
With the Patient Tracker App, users will be able to:

1. Sign in with a google account
2. Add details of a new patient
3. Assign a patient to a doctor

4. Record a patient's illness
5. Prescribe a drug(s) to a patient who has already been assigned a doctor
6. View patient treatment history
7. Confirm issuance of a prescription
8. Share a patient's medical history
9. Add doctors and the days they are available

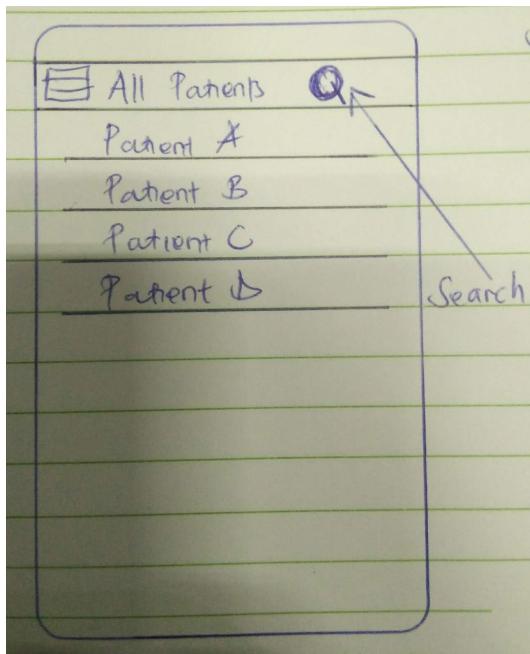
User Interface Mocks

Sign In Screen



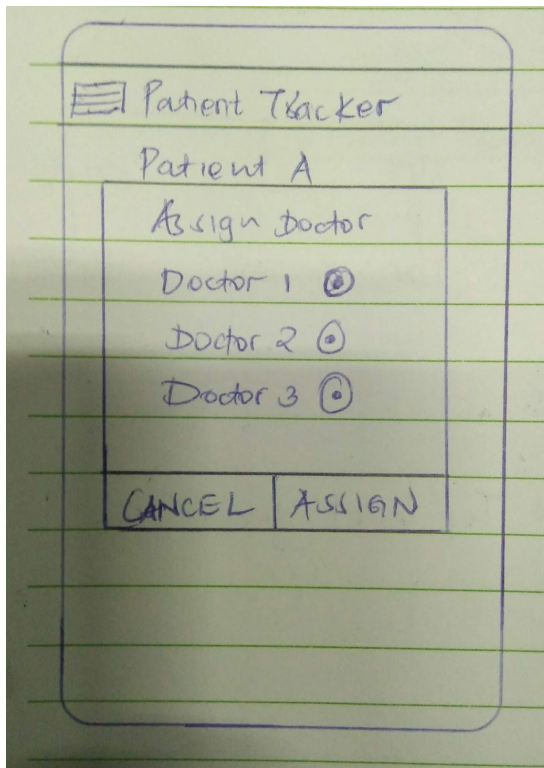
This screen allows a user to sign in to the app with their google account

Home Screen



The Home Screen is a list of all the patients and the last visit dates

Assign Doctor Dialog



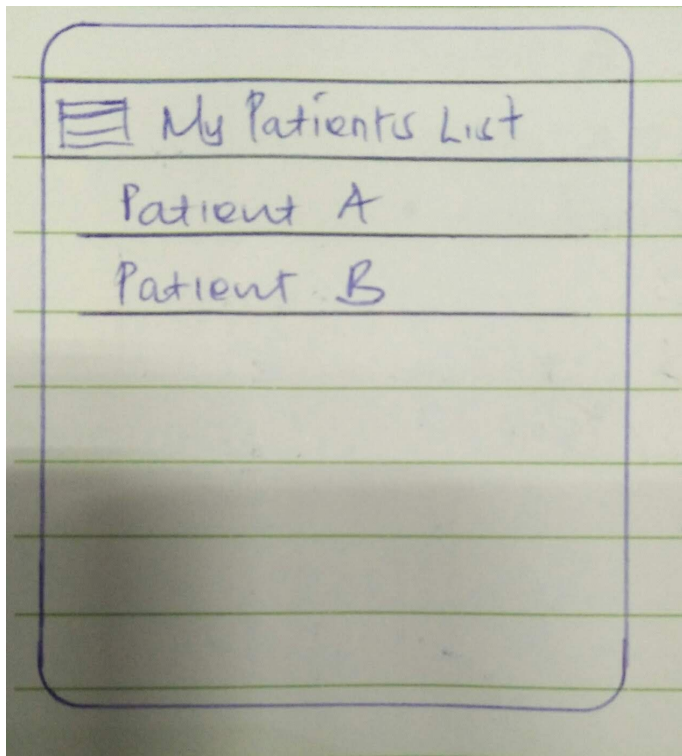
- Lists all the available doctors
- Allows user (receptionist) to assign an available doctor to a patient

Add Patient Screen

Hand-drawn sketch of an "Add Patient" screen on lined paper. The screen features a title bar with a menu icon (three horizontal lines) and the text "Add Patient". Below the title bar, there are five input fields with labels: "Name:", "ID No:", "Address:", "Phone #:", and "Age:". At the bottom of the screen, there are two buttons labeled "CANCEL" and "SUBMIT".

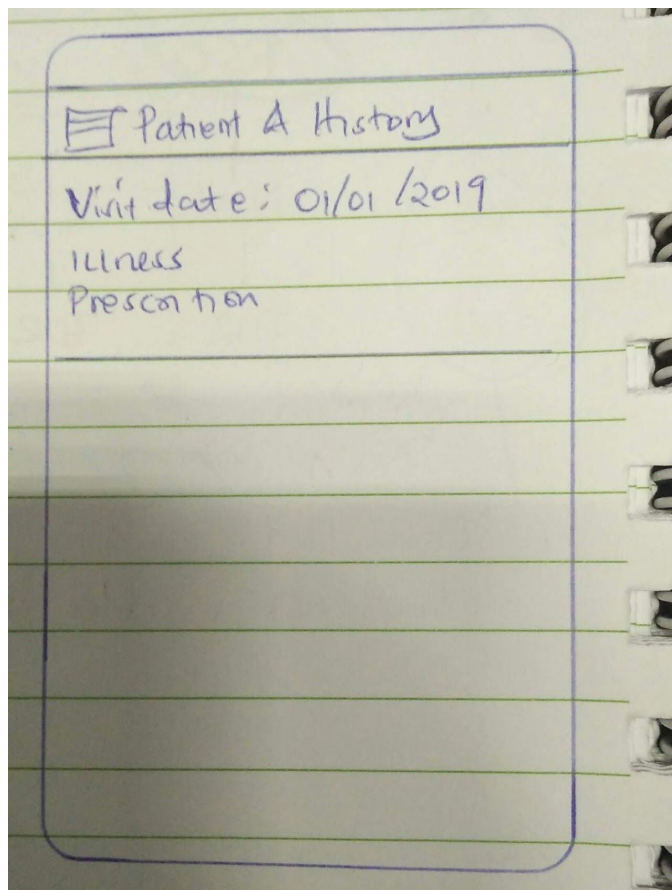
- Allows user (receptionist) to add a new patient's details
- Allows user (receptionist) to edit details of an existing user

Patients Assigned to Doctor Screen



- Lists all the patients assigned to the logged in user (doctor)

Patient Treatment History Screen



A handwritten patient treatment history screen on lined paper. The screen is titled "Patient A History" and contains the following information:

- Visit date: 01/01/2019
- Illness
- Prescription

The screen is designed to record patient visits, including the date of visit, illness, and prescribed drugs.

- Shows a record of all the previous patient visits (Date of Visit, Illness, Drugs(s) prescribed and their quantities, the cost of the visit)

Prescribe Drug Dialog

A hand-drawn sketch of a 'Prescribe Drug Dialog' form. The form is a rounded rectangle with a title bar at the top containing a window icon and the text 'Add Prescription'. Below the title bar is a large rectangular area for input, divided into four horizontal sections: 'illness description', 'select drug ▾', 'Quantity', and 'Duration'. At the bottom of the form is a horizontal bar with two buttons: 'CANCEL' and 'ADD'.

- Allows user to add a short description of what the patient is suffering from
- Allows user to select the drug the patient will be issued with and the dosage

Confirm Drug Issuance Screen

☰ Issue Drug(s)

Patient A

Drug Prescribed:
Panadol

Quantity: 10 tabs
Duration: 3 days

CANCEL | ISSUE

- Allows user to confirm issuance of the prescribed drug to the patient

Key Considerations

How will your app handle data persistence?

I plan to build a ContentProvider

Describe any edge or corner cases in the UX.

All activities / fragments will navigate back to the previous screen when the back button is clicked

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

Picasso: To handle loading images.

Jake Wharton's Butterknife: to enhance code readability and easier views and resources lookups

AppCompat, Design and CardView: for UI compatibility and support

Android support library: for backward compatibility
Timber: for logging

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

Google Play Services for Google Sign-In

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

Task 1: Project Setup

Set up the project on android studio (set name, build tools version, min and target sdk versions)
Setup permissions
Implement all external libraries
Setup vcs for project

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
- Build UI for all fragments
- Build UI for the other activities
- Build UI for alternative screen sizes

Task 3: Database Setup

- Create Room db and classes

Task 4: Create RecyclerView Adapters

- Create adapters for recycler views, taking data from Room
- Set Adapters for recycler views

Task 5: Polish UI

- Create layouts for different screen sizes

- Implement material design guidelines
- Generally polish the UI

Task 6: TESTING

- Test connection to DB
- Test UI
- Test app on devices with different screen sizes

Add as many tasks as you need to complete your app.

Submission Instructions

- After you've completed all the sections, download this document as a PDF [File → Download as PDF]
 - Make sure the PDF is named "**Capstone_Stage1.pdf**"
- Submit the PDF as a zip or in a GitHub project repo using the project submission portal

If using GitHub:

- Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
- Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"