GitHub Username: **lawrence-gichuki**

StepApp

Description

Daily step counter, pedometer and easy calorie counter to help you lose weight

Intended User

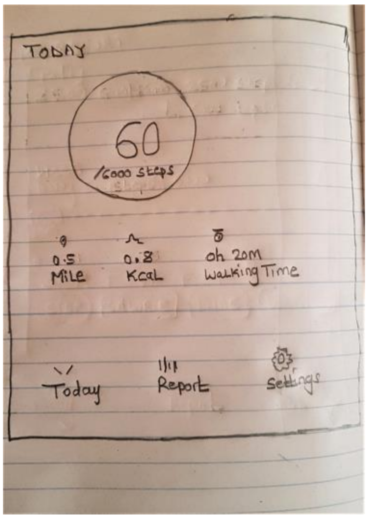
This App is for fitness enthusiasts who would wish to shed off that extra kilo and maintain a healthy lifestyle

Features

Main features of the app:

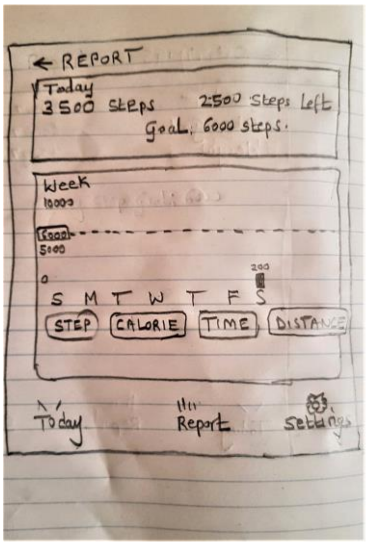
- Count steps
- Tracks burned calories
- Tracks walking distance and time

Screen 1



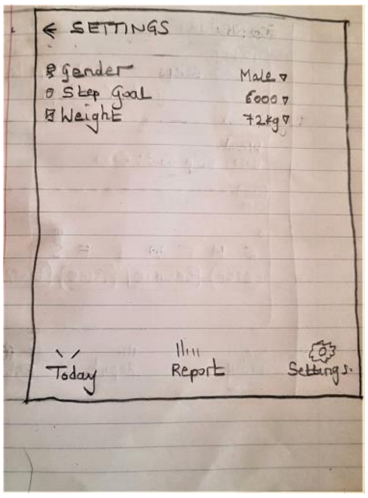
This is the home screen. It displays the steps, miles, calories burnt and walking time in the current day

Screen 2



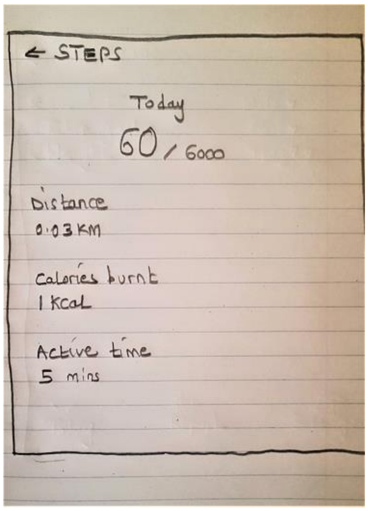
This screen displays a week's view of the steps covered, calories burnt, walking time and the distance covered

Screen 3



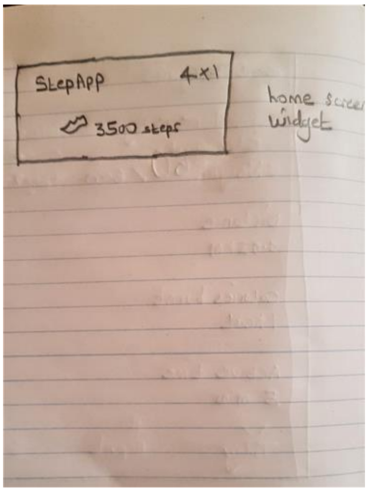
This screen enables the users to customize the app making it fit for the intended use

Screen 4



This screen is displayed when a user clicks on the steps button in the report screen.

Screen 5



This is the home screen widget

Key Considerations

This App is written solely in Java Programming Language

How will your app handle data persistence?

I will use the Google fit's Recording API to record data in the background with persistent subscriptions

App will use fitness store content provider to access locally stored data

Describe any edge or corner cases in the UX.

Because of device power saving, some devices stop counting steps when the screen is locked.

Other devices may not have a built-in pedometer sensor

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

MPAndroidChart is a powerful Android chart view / graph view library, supporting line- bar- pie- radar- bubble- and candlestick charts as well as scaling, dragging and animations

This library will simplify graphical reporting of the information captured from the built-in sensor

NAME	VERSION
Android Studio	3.5.3
Gradle	6.0.1
Gradlew	5.4.1
MPAndroidChart	3.1.0
design	28.0.0
Play-services-fitness	18.0.0

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

I will create a google developer account to gain access to the google fit APIs

I will use the following Google fit APIs:

i. The Sensors API – This will read raw sensor data in the app in real time

ii. The Recording API – This API will record data in the background with persistent subscriptions

iii. The History API - provides access to the fitness history and lets apps perform bulk operations, like inserting, deleting, and reading fitness data. Apps can also import batch data into Google Fit

Next Steps: Required Tasks

Task 1: Project Setup

- Create a google developer account
- Install the latest google play services
- Create a new Android project using Android Studio
- Configure libraries in the build.gradle file
- Add App permissions in the Manifest file
- Build the App feature described earlier
- Test the App
- Deploy the App

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
- Build UI for the Report Activity
- Build UI for the Settings Activity
- Build the home screen widget

Task 3: Your Next Task

- Implement the Sensors API
- Implement the Recording API
- Implement the History API
- Implement the other App logic
- Handle Error Cases

Task 4: Your Next Task

- Complete the Strings.xml file
- Complete the colors.xml
- Make the App Material

Task 5: Your Next Task

- Test the App

- Deploy the App to the Play Store