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

GitHub Username: RegNex

FAMILY TRACKER

Description

The aim of this App is to lets its users track family members, friends, etc. to know where they are at any time. The App also helps users save some basic health information which can serve as a reference when in an Emergency to the Emergency service personnels. The Emergency service personnel can lookup these details to know what medication to administer to the user and what not to.

Intended User

Family Tracker can be used by anyone with a smartphone who is ready to share his/her location with friends and family.

Features

Family Tracker Will:

1. Save health information
2. Family Contact number
3. Save personal information about user
4. Display location on google map
5. Route and navigate to user

User Interface Mocks

Some mock screens of App

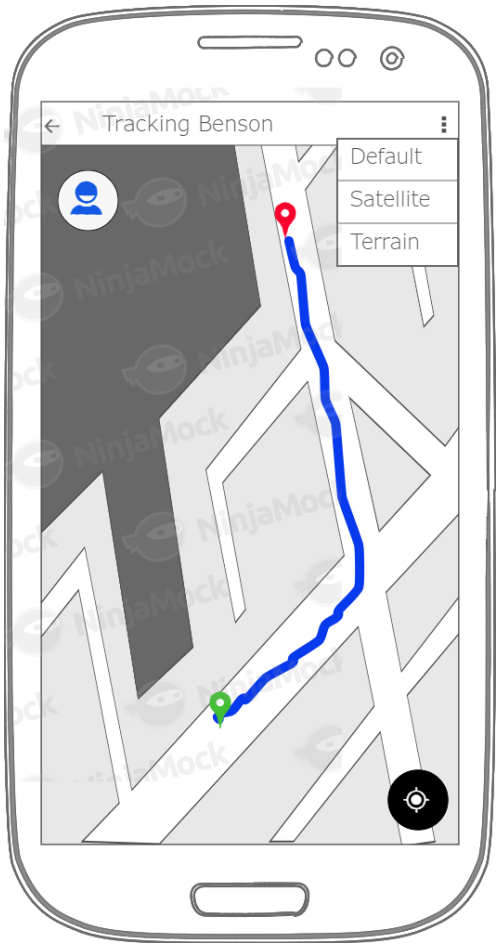
Screen 1

Login Screen with Google login option

Screen 2

Profile Screen for user to enter and save basic information

Screen 3



Tracking Screen showing route from current location to person being tracked.

Screen 4

Setting Screen showing basic setting options of App

Screen 5

Setup Screen where user selects contacts to track

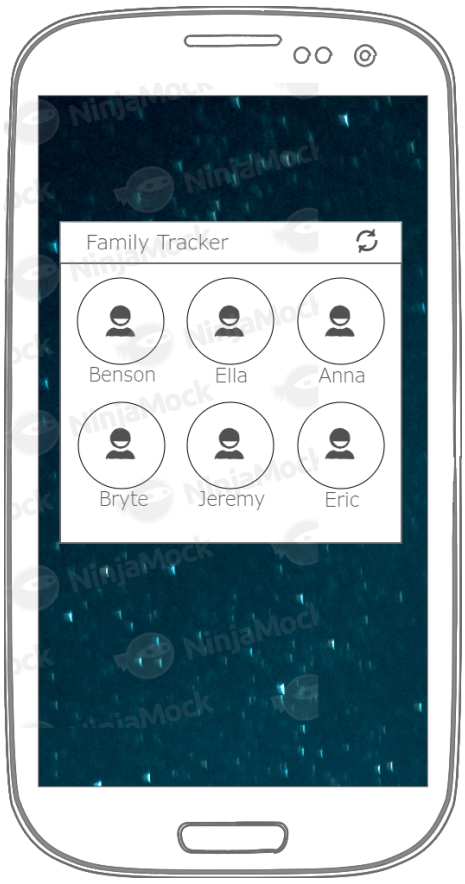
Screen 6

Health Information Screen where users will enter basic health information about them

Screen 7

App Lock Screen where user can set Pin protection to the App

Screen 8



App Widget to display basic actions of app

Key Considerations

How will your app handle data persistence?

User's data will be stored in firebase Realtime database and related images will be stored in Firebase Storage.

Describe any edge or corner cases in the UX.

Internet connection detection will be implemented to alert the user (using snackbar or a toast) when there is no internet connection to track or display map.

In order to save user internet bundle usage, user's profile picture will be downloaded in the background on the device using AsynTask so that instead of the App always fetching the image online, it will fetch it from the user's local storage.

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

Picasso for loading and caching of images.
CircularImageView for displaying images in circular manner.
Firebase realtime database for storing user data
Firebase Storage for storing user images
Google Maps for displaying user location and routing
PincodeView for providing lock feature to App

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

I will be using google authentication in firebase to authenticate users and grant them access. Also, google maps and location will be used to get the person been tracked current location and that location is sent to firebase realtime database then sent back to the the user that is doing the tracking.

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

- Create a new empty project in Android studio.
- Include gradle dependencies in the above libraries targeting the latest SDK
- The App will be programmed solely in Java programming language

Task 2: Add Required Files

- Add google_json file to project

Task 3: Implement UI for Each Activity and Fragment

List the subtasks:

- Build UI for MainActivity/Fragment (main Menu)
- Build UI for ContactActivity/Fragment (contact display Screen)
- Build UI for ProfileActivity/Fragment (Profile Screen)
- Build UI for HealthActivity/Fragment (Health Information Screen)
- Build UI for MapsActivity/Fragment (Map Screen)
- Build UI for Tablet View
- Build UI for Widget
- Build UI for setting page
- All strings will be in a strings.xml file
- RTL layout switching on all layouts.

Task 4: Implement Services

- Create LocationPusher service
- Create LocationTracker service

Task 5: Implement MainActivity/Fragment

- Create Main Activity
- Create Main Fragment
- Implement content descriptions
- Handle Error Cases

Task 6: Implement ContactActivity/Fragment

- Create Contact Activity
- Create Contact Display Fragment
- Retrieve user's saved contacts
- Create Contact Edit Fragment
- Save user's selected contacts

Task 7: Implement Setting Activity

- Create setting activity
- Create Lock feature

Task 8: Implement Map Activity

- Create Map Activity
- Retrieve user current location
- Display user current location

Task 9: Implement Widget

- Create Widget Provider
- Create Widget Remote Views Service

Task 10: Implement App Invite (Google services)

- Create App Invite

Task 11: Implement Material Design

- Create element transitions

Task 12: Implement ShareActionProvider

- Create Share Action Provider

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**"