How to Use this Template

1. Make a copy [File → Make a copy...]
2. Rename this file: "**Capstone_Stage1**"
3. Replace the text in green

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
3. Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"

GitHub Username: Sugamsm

STAR15 FM

Description

It is an Internet based FM Radio App. Major problem that it solves, user can request a song of his favour, and soon the song is played in the playlist. User can Dedicate and Request Songs to and for Listener Buddies and to Friends. Listener Buddies can chat live with RJ Star.

Intended User

Anybody from anywhere! For all the Music lovers out there, this app is your own Music Player.

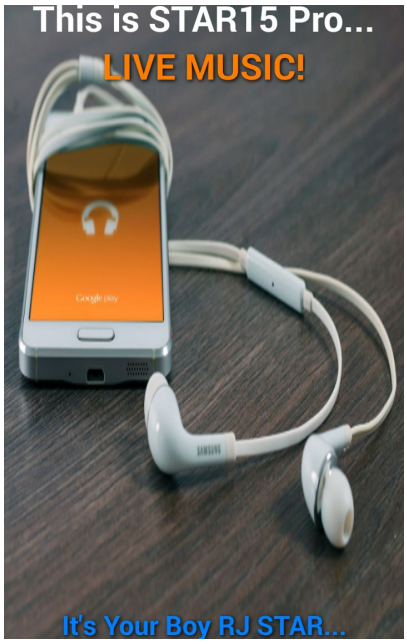
Features

- Play Live Music from Online Stream
- Request Song
- Dedicate Song to anyone
- Live Group Chat
- Live Notifications

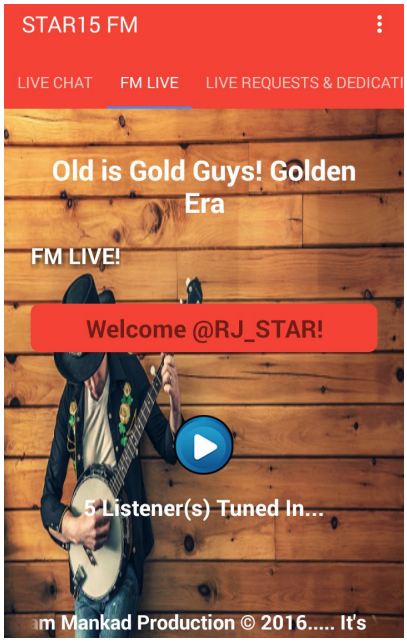
User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

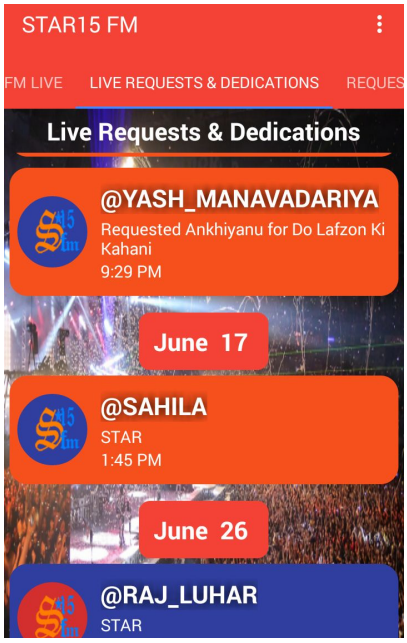
Screen 1



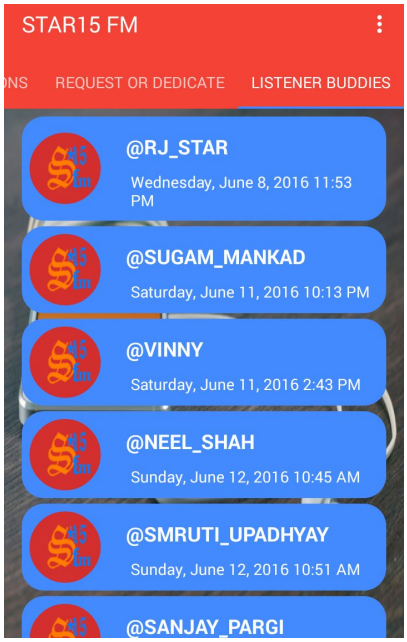
Screen 2



Screen 3



Screen 4



Key Considerations

How will your app handle data persistence?

Will build a Content Provider to store data locally.

Describe any corner cases in the UX.

There is a foreground notification running when the FM is streamed by the user. OnClick of notification opens up the Currently Playing screen of the app via a splash screen, if the screen is not already showing.

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

Picasso to handle the loading and caching of images, as a 3rd party library.
App uses Firebase Cloud Messaging, so the same library is used too.

Describe how you will implement Google Play Services.

Google Play Services are used as Firebase Cloud Messaging.

Next Steps: Required Tasks

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

Task 1: Project Setup

- Configure libraries
- Set up and Integrate App with Firebase Console Project for Cloud Messaging
- Material Design Implementation for UI
- Configure Local Data Storage

Task 2: Implement UI for Each Activity and Fragment

- Build UI for Splash Screen/Registration Screen
- Build UI for MainActivity
- Build separate UI for Each Fragment used as a separate Page
- ViewPager and Swipe Tabs for Various Features

Task 3: Startup UI Function

Build UI for Startup screen that behaves as a :

- Splash Screen
- Registration Screen
-

Followed by :

- Creating layout for Startup Activity
- Check for User registration

Task 4: UI for MainActivity

MainActivity will contain 5 Fragments in a ViewPager and Swipe Tabs Implementation.

The 5 Fragments are separate features as:

- ❏ Live Chat
- ❏ FM Live Currently Playing (Now Playing Screen)
- ❏ Buddies' Live Activities like Requests And Dedications
- ❏ Request or Dedicate a Song
- ❏ Registered Buddies

Above is Implemented as follows :

- Layout creation for each separate features

Request or Dedicate Fragment will contain two sub Fragments respectively.

Task 5: Widget Implementation

Implementation of Widget as follows:

- ★ Create Layout for Widget
- ★ Manage Data to be displayed on the Widget.

Task 6: Target Screens

Each and every UI's availability for maximum screens.

- Various res/layout folders for different sizes

Task 7: Data Handling

The App Implements Firebase Cloud Messaging, implement the :

- ❖ User's Chat storage
- ❖ Requests and Dedications storage
- ❖ Registered Buddies Storage

using Content Provider.

Task 8: Error Handling

Implement Error handling for :

- ➔ Network Operations :
  - ◆ Live Stream Playing
  - ◆ Send Chat Message
  - ◆ Request or Dedicate Mechanism
  - ◆ Loading Data from the Server

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
3. Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"