

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: ahmedgamalaglan

Matchatak

Description

This great app 😊 will help all football fans all over the world, you can search for matches as well as other things like competitions, teams and players.

You can do all these things depending on the area you choose to search.

Intended User

This app is for all football fans all over the world at all ages who are interested in the football.

Features

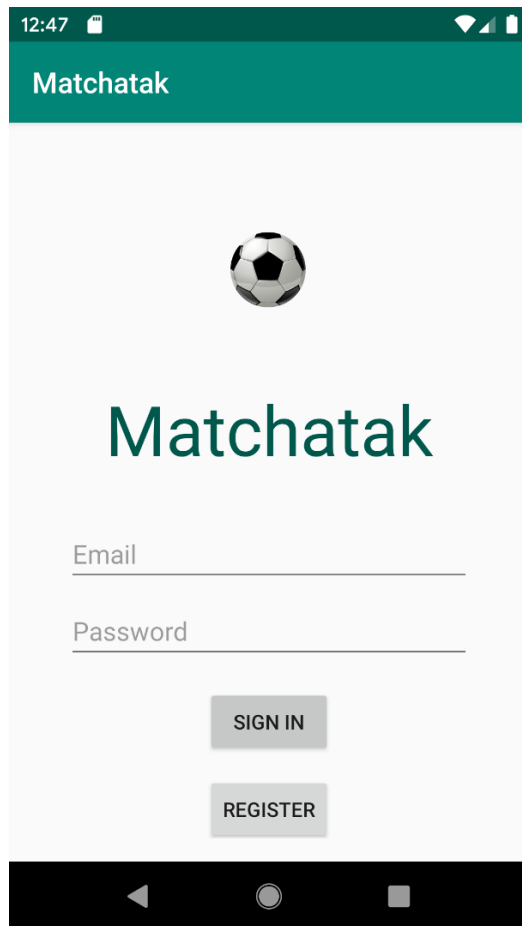
List the main features of your app. For example:

- Show Football Competitions
- Add Favorite Competitions to Database
- Show matches and results
- Show teams and results
- Add Favorite teams to Database
- Show team details
- Show Players details

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 Google Drawings, www.ninjamock.com, Paper by 53, Photoshop or Balsamiq.

Screen 1



12:47

Matchatak



Matchatak

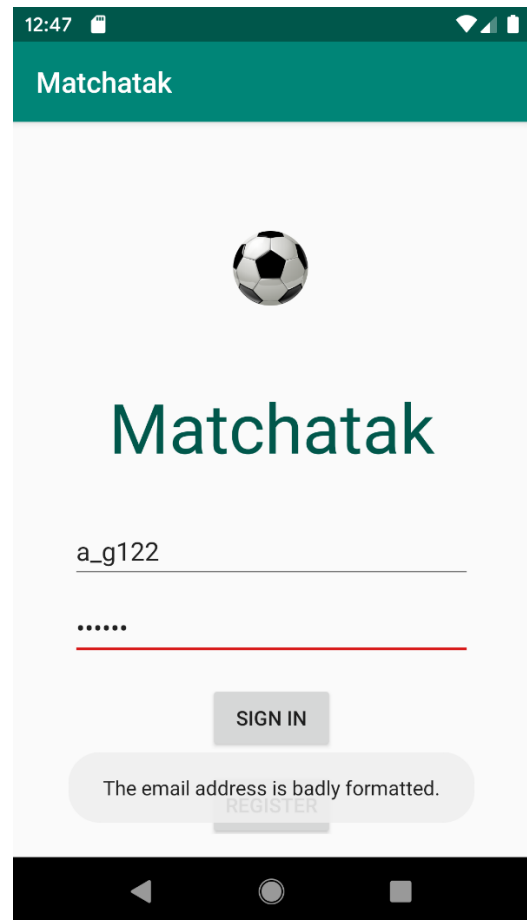
Email

Password

SIGN IN


REGISTER

This screen shows the Matchatak login interface. It features a teal header with the time 12:47 and the app name 'Matchatak'. Below the header is a soccer ball icon and the app name in a large teal font. There are two input fields for 'Email' and 'Password'. Below these fields are two buttons: 'SIGN IN' and 'REGISTER'.



12:47

Matchatak



Matchatak

a_g122

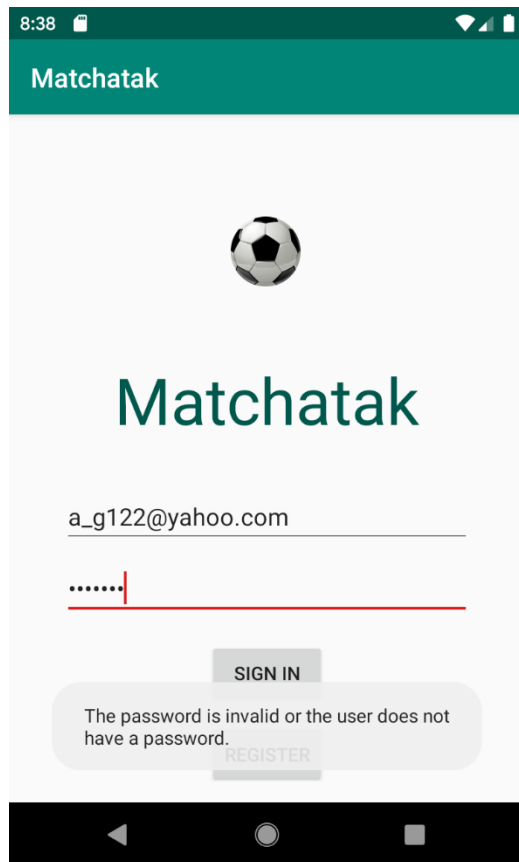
.....

SIGN IN

The email address is badly formatted.

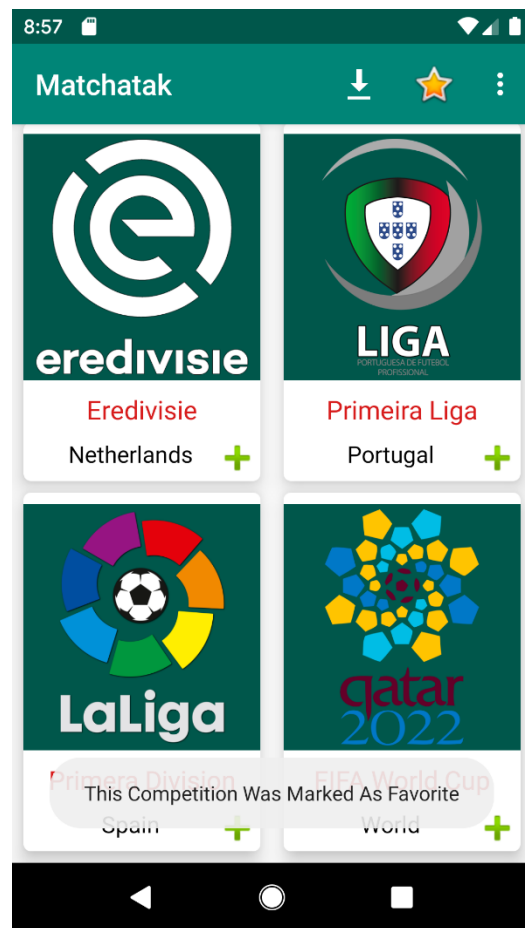
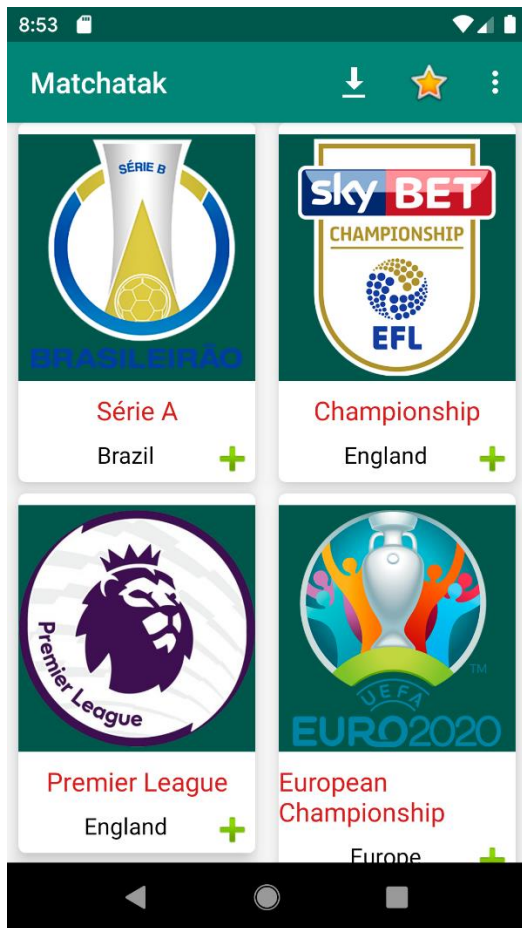
REGISTER

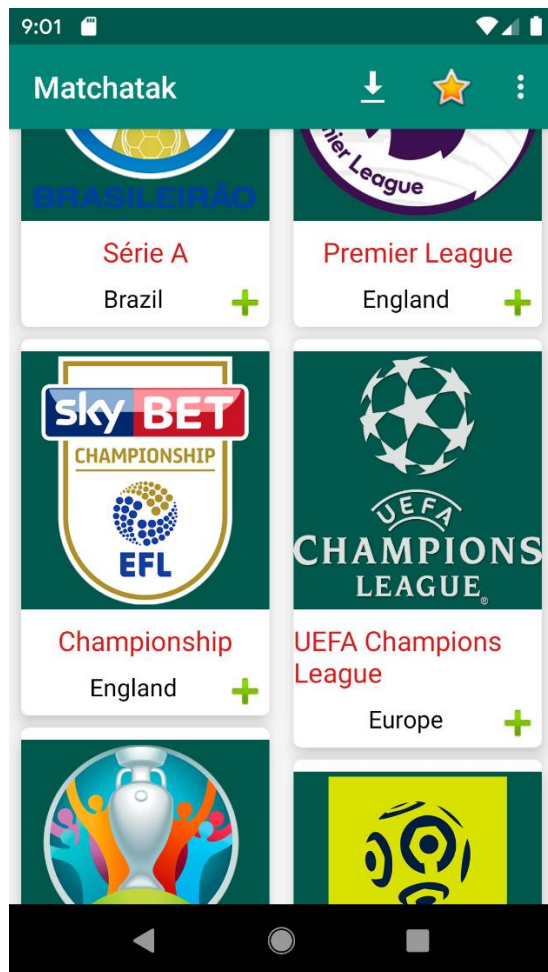
This screen shows the Matchatak login interface with an error message. The 'Email' field contains the text 'a_g122' and the 'Password' field contains six dots. A red error message 'The email address is badly formatted.' is displayed below the 'SIGN IN' button. The 'REGISTER' button is also visible.



This screen is used to login and register users and add authentication through firebase authentication, the user must first register and then login to the app so he can use it as he wants, in cases of wrong password or invalid email format or other error ,the error message is shown in a toast or in the edit Text set error method.

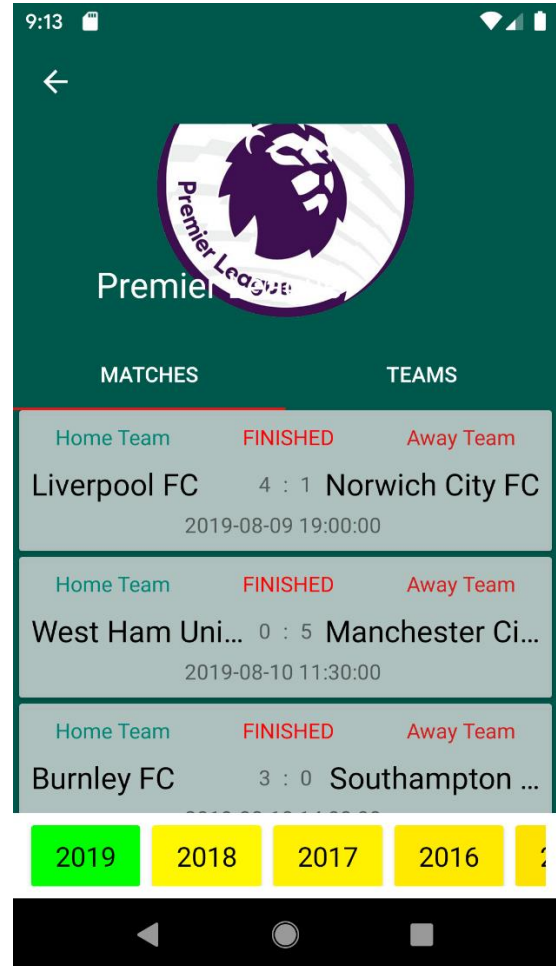
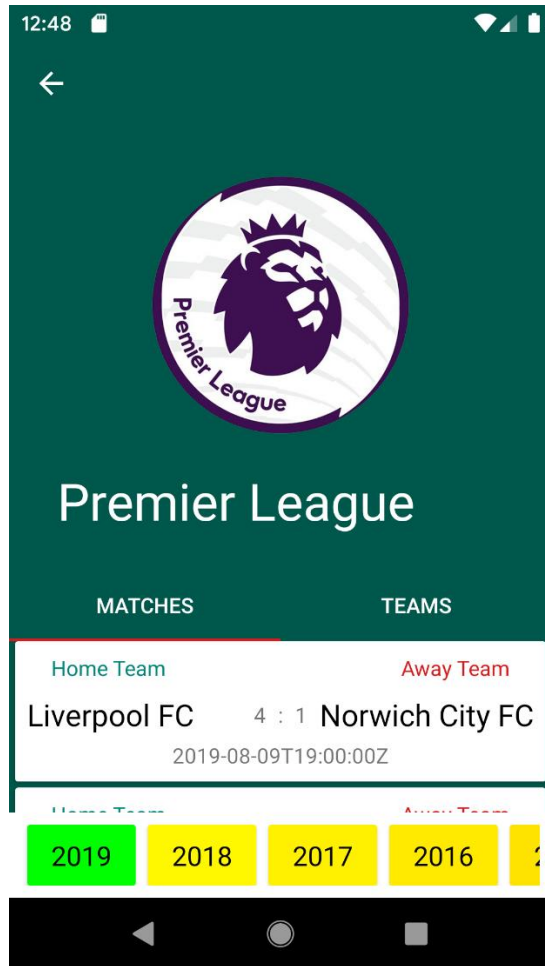
Screen 2





That screen shows as a default all the free competitions in the API that I use so the user can see and open all the shown competitions, The users also can add a certain competition to his favorites throw the + sign in every item and also can show only the favorite ones by clicking the star icon in the action bar, and can reload all the competitions again throw the download icon I the action bar also, And by clicking a certain competition item it opens a competition details in the next activity. You can logout of your account by clicking on options and choose "Sign Out"

Screen 3



9:14

Premier League

MATCHES	TEAMS
Arsenal FC	2 : 2 Tottenham Ho...
2019-09-01 15:30:00	
Home Team	SCHEDULED
Liverpool FC	: Newcastle Unit...
2019-09-14 11:30:00	
Home Team	SCHEDULED
Wolverhampto...	: Chelsea FC
2019-09-14 14:00:00	
Home Team	SCHEDULED
Tottenham Hot...	: Crystal Palace ...
2019-09-14 14:00:00	
Home Team	SCHEDULED
Manchester Un...	: Leicester City FC

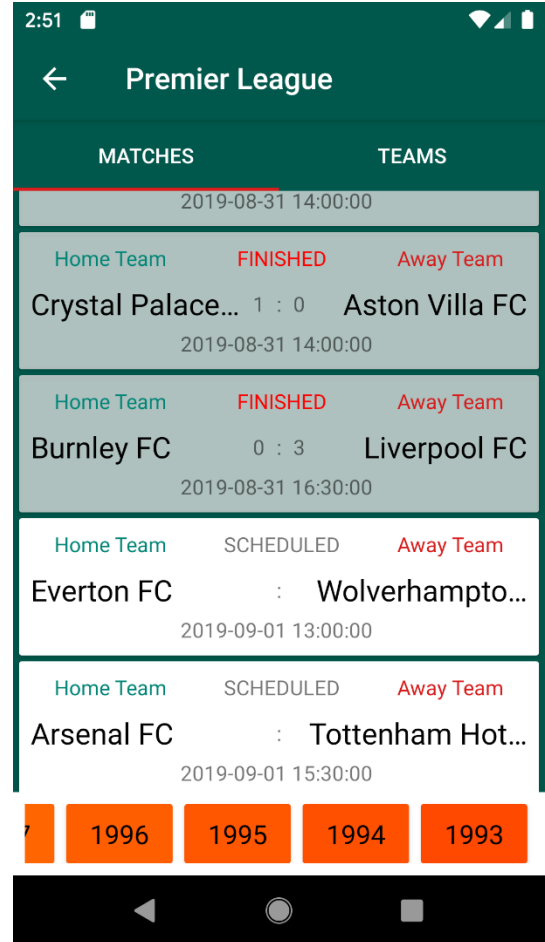
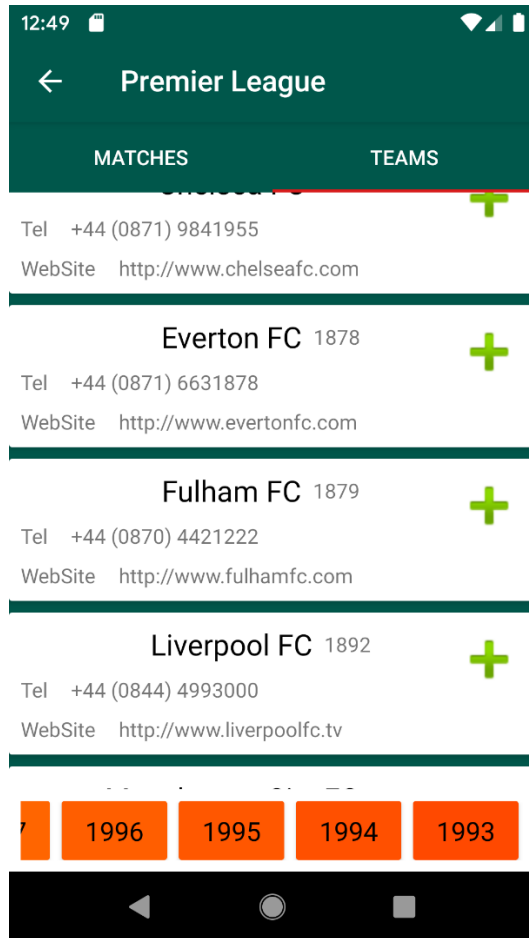
2019 2018 2017 2016

2:51

Premier League

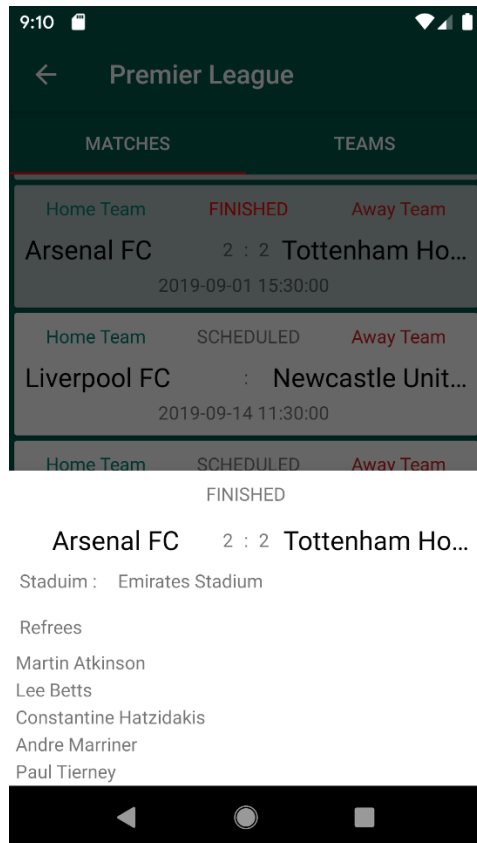
MATCHES	TEAMS
2019-08-31 14:00:00	
Home Team	FINISHED
Crystal Palace...	1 : 0 Aston Villa FC
2019-08-31 14:00:00	
Home Team	FINISHED
Burnley FC	0 : 3 Liverpool FC
2019-08-31 16:30:00	
Home Team	SCHEDULED
Everton FC	: Wolverhampto...
2019-09-01 13:00:00	
Home Team	SCHEDULED
Arsenal FC	: Tottenham Hot...
2019-09-01 15:30:00	

2019 2018 2017 2016



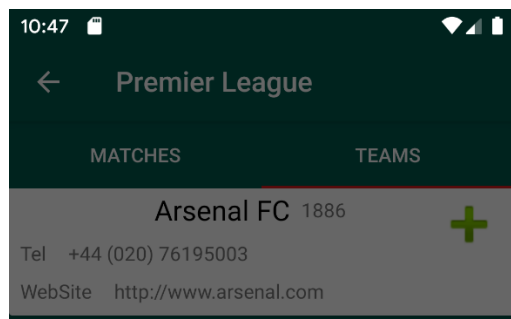
That screen shows the details of the certain competition that the user clicked before, It shows the matches and its results and the teams participated in that competition, The app can show all available seasons of that competition by clicking the season item in the bottom navigation view and scroll it till the wanted season. By clicking a certain match or a team a very detailed view appears with all needed information about the item clicked.

Screen 4



That screen comes up to the screen by clicking on a certain match in a certain competition as it holds the given details of the match, it is implemented as a bottom sheet dialog fragment.

Screen 5



Arsenal FC

Stadium : Emirates Stadium
 WebSite <http://www.arsenal.com>
 Phone +44 (020) 76195003
 Email info@arsenal.co.uk
 Club Colors Red / White

Players

Emiliano Martínez	26
Bernd Leno	1
Matt Macey	0
Sokratis Papastathopoulos	5



Kieran Tierney	0
Dani Ceballos	8
Mattéo Guendouzi	29
Lucas Torreira	0
Mesut Özil	10
Granit Xhaka	34
Mohamed El Neny	0
Ainsley Maitland-Niles	15
Henrikh Mkhitaryan	7
Martinelli	35
Alexandre Lacazette	9
Joe Willock	28
Reiss Nelson	24
Pierre-Emerick Aubameyang	0
Nicolas Pépé	0
Unai Emery	0



Arsenal FC

Stadium : Emirates Stadium
 WebSite <http://www.arsenal.com>
 Phone +44 (020) 76195003
 Email info@arsenal.co.uk
 Club Colors Red / White

Players

Emiliano Martínez	26
Bernd Leno	1
Matt Macey	0
Sokratis Papastathopoulos	5
Nacho Monreal	18
Héctor Bellerín	0
Rob Holding	0
Shkodran Mustafi	0
Sead Kolašinac	31
Calum Chambers	21



That screen also is show by clicking on a certain team item in the teams recyclerView
It show the details as possible I can get from the APIs.

Add as many screens as you need to portray your app's UI flow.

Key Considerations

How will your app handle data persistence?

The app can handle data persistence only when the user clicks the add to favorite icon (+)
The app stores the data throw a room database.

Describe any edge or corner cases in the UX.

In the main activity after logging in when the user chooses to open the favorite competition,
He can back to all available competitions by clicking the download icon again

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

Picasso, is used to load the images from the URLs.
Retrofit, is used to handle networking and calling APIs.
Room, is used to handle database operations.

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

Firebase Auth, is used to register and login and provide necessary authentication.
Firebase Analytics, is used to report the commonly used accounts or any category.

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

The project will require football-data APIs and use other libraries so the next tasks will be

- Add dependencies to gradle and sync them.
- Implement the code to do its purpose.

If it helps, imagine you are describing these tasks to a friend who wants to follow along and build this app with you.

Task 2: Implement UI for Each Activity and Fragment

Making the home screen a login and register screen, so its not necessary to open it every time the app starts:

- Build UI for LoginActivity.
- Build UI for MainActivity.
- Build adapters for RecyclerView.
- Build fragments UI and adapters and Implement click listeners.
- Pass data through intents and uses callbacks in the listeners.
- Start activities with intents and try the cycle of the UI.

Task 3: Understanding APIs and handling calls

Understand the APIs the make a complete cycle with postman and then:

- Add dependencies for the libraries that will handle API calls
- Signup to get Api-Key
- Try calling the first API through the app
- If it works, I will go on calling the rest of needed APIs.

Task 4: Handling ViewModels and Repository

Create my view-models and repositories and log the results to validate them:

- Create new Class Called Repository
- Implement methods that returns the wanted data

- Implement ViewModels in the activities and fragments
- Bind the data to the views after observing them from attached view models.

Task 5: Handling Databases

Implement room database in the app:

- Create MyDatabase class that introduces a singleton instance of DB
- Use the database in the repository and handling actions manually to test them
- If it succeeds in adding and retrieving data from database.
- Implement Database operations with UI events.

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**"