

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”**

[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.](#)

[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: mmublackpirate

1Mark

Description

1Mark is a quiz app for Matriculation Students in Myanmar. It will help prepare students for their upcoming exam.

Main Features

- Custom or Quick Test
- Two question type (True/False and Fill in the blank)
- Analysis of accomplishment

Intended User

Matriculation Students in Myanmar

Features

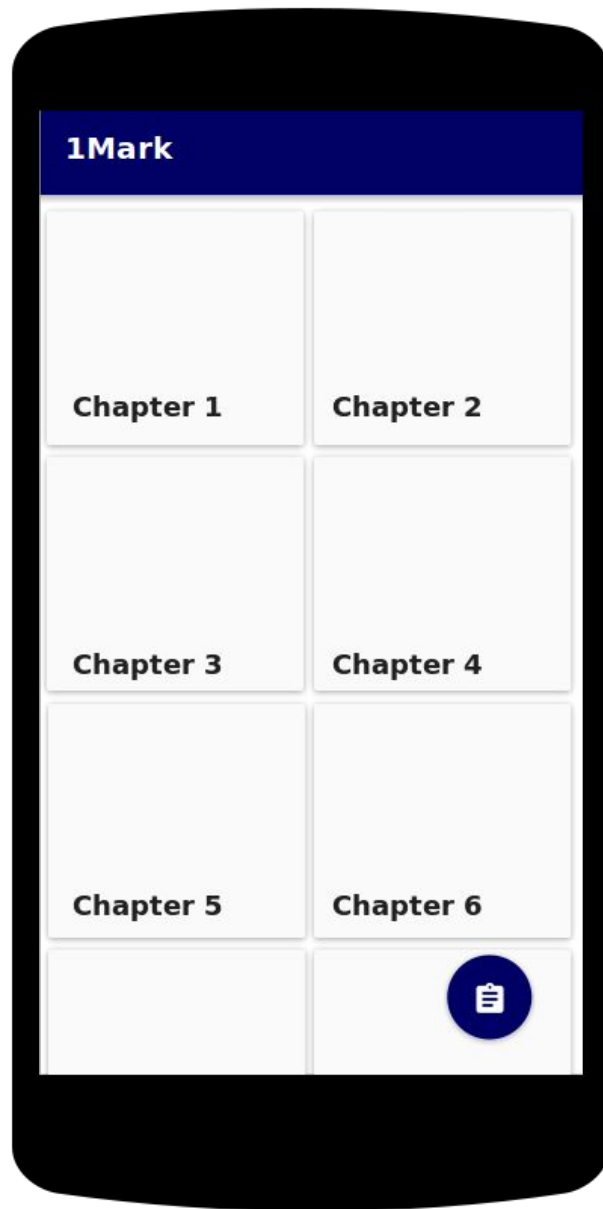
Main Features

- Custom or Quick Test
- Two question type (True/False and Fill in the blank)
- Analysis of accomplishment

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



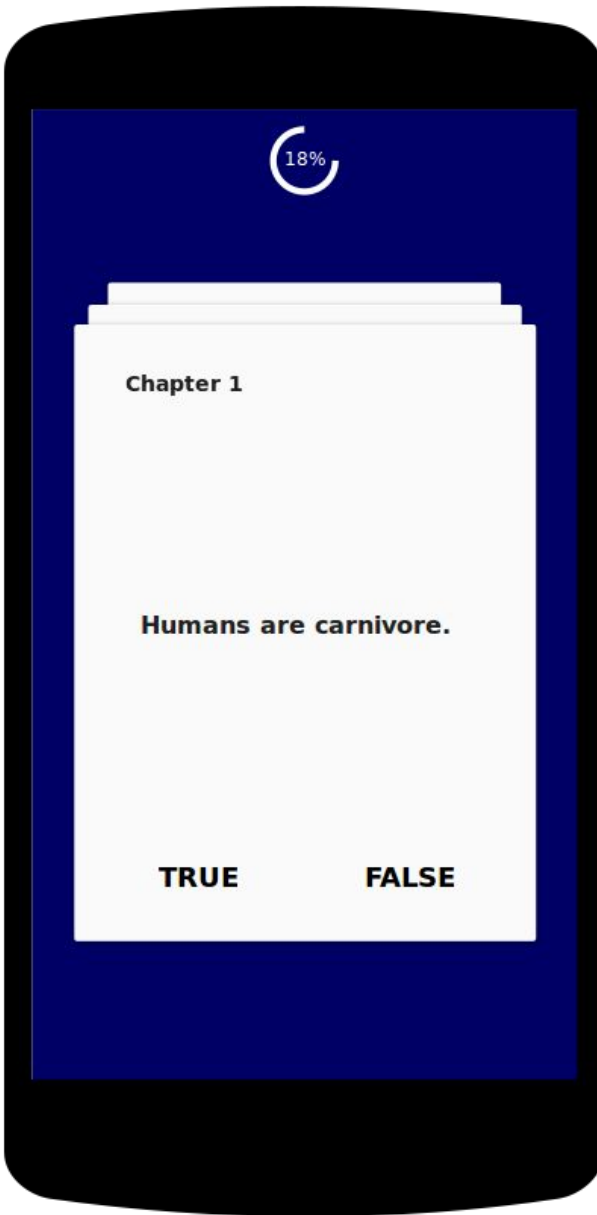
Main Screen

There will be background image for each card

Clicking the FAB button will open up a dialog asking user to choose Quick Test or Custom Test.

If user chooses any of the test, it will go into Quiz Screen.

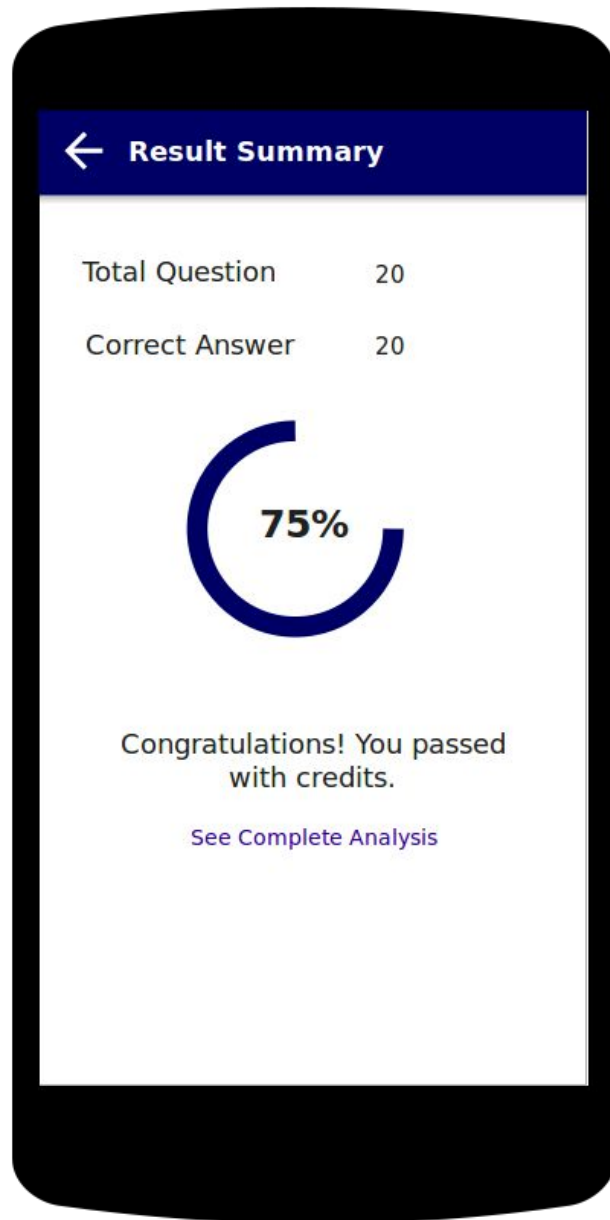
Screen 2



Quiz Screen

It will use a Flippable Cards UI.

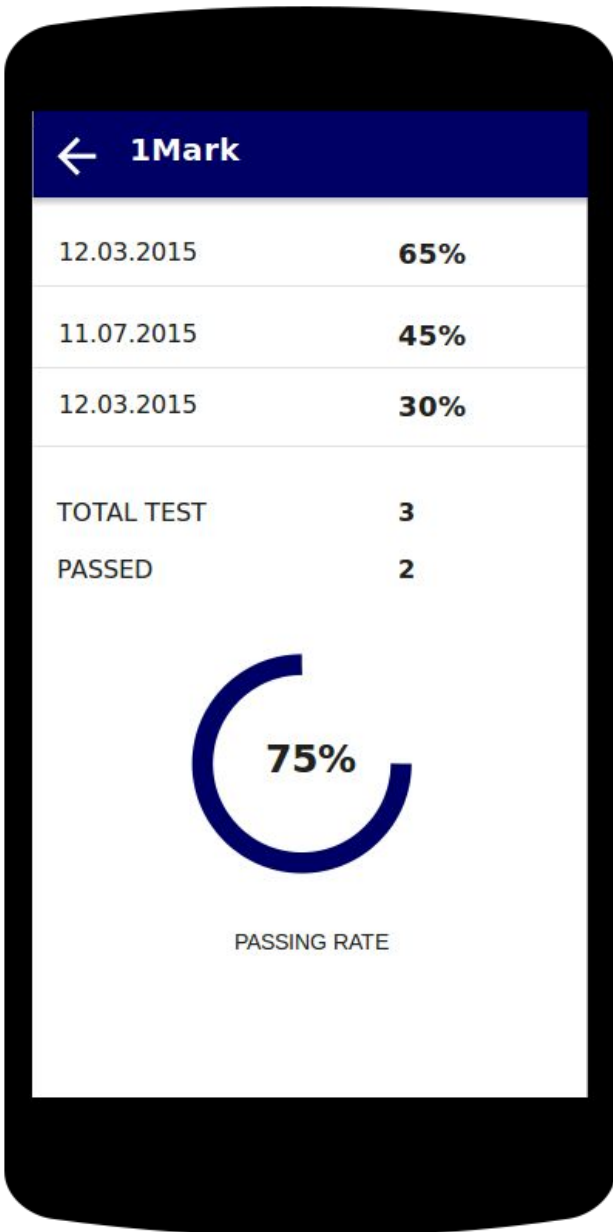
Screen 3



Analysis Screen

At the end of each test, user will see the result summary for the respective test. He/she can choose to see the complete analysis of all the tests too.

Screen 4

**Complete Analysis Screen**

User can see the complete analysis of all the tests he/she has taken and the passing rate.

Key Considerations

How will your app handle data persistence?

The app will use a Sqlite Database with a custom content provider + loader to show items.

Describe any corner cases in the UX.

If the user press back button during each test, the state of the test will be saved automatically and when user choose to test again, the app will ask whether he will want to resume the previous test.

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

<https://github.com/blipinsk/FlippableStackView> for Flippable cards

Glide for image caching and loading.

Retrofit for networking.

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 the Library.
- Configure Syncing and Account
- Configure the POJO model and Sqlite Db

Task 2: Implement UI for Each Activity and Fragment

- Build UI for Home Screen
- Build UI for Quiz screen
- Build UI for Analysis Screen
- Build UI for Complete Analysis Screen

Task 3: Your Next Task

- Implement Adapters
- Implement DB and Caching

Task 4: Your Next Task

- Configure loader and recycler view
- Implement GA and Crashlytics

Task 5: Your Next Task

- Test and Debug
- Configure Proguard

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

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**"