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: github.com/ye-lin-aung

Drunky
Description

A cocktail mixing app which provides more than 3000 of cocktails with detail instructions.Young people love to party but hiring a bartender is a bit costly.And people love cocktail.They don't want to go a bar and choose just a small variety of cocktails only.The app will be perfect for

people who wants to learn how to make cocktails for any purposes.The app includes tastes,videos,glasses in put in and also ingrediants.

Intended User

The app is intended for young people,self-learners and everyone who want to make cocktail by themselves.
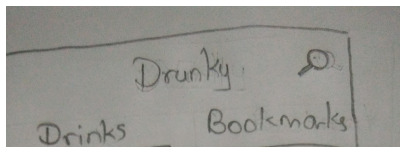
Features

List the main features of your app. Eg:
- Request data from server
- Able to watch tutorials
- Able to bookmarks

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.
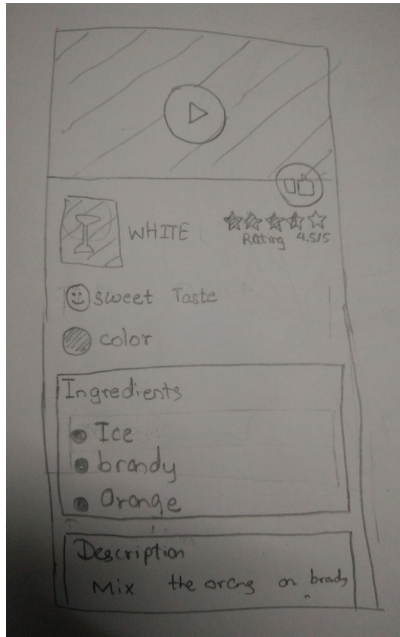
Screen 1

Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]
This is the main screen the user can search,view bookmarks and view drinks

Screen 2

Replace the above image with your own mock [click on the above image, then navigate to Insert → Image…]
This is an description screen.It will contains taste of the drink,glass to place in,how to mix,what ingredients do we need.

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 will use own content provider.

Describe any corner cases in the UX.

If the user long press or press like in item it will be saved into bookmarks.

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

Picasso is use for caching images from online and show cache images after loaded whether it is online or offline.

Dagger is a fully static, compile-time dependency injection framework for both Java and Android.

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 (data binding library,retrolambda and dagger2.etc)
- Generate Api keys from addb api

.

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
- Build UI for MainFragment
- Build UI for BookmarkFragment
- Build UI for DetailActivity

Task 3: Implement Content Provider

- Design database schema
- Create database contract
- Create content provider
- Integrate facebook stheto
- Test content provider

Task 4: Download,Cached and Show

- Create Rest api endpoint services with retrofit
- Create Repository class to save data from online to local,cached it and show back
- Implements Loaders

Task 5: Optimization for large Screen

- Create a Dialog style for detail activity
- Increase RecyclerView row count according to screen size

Task 6: Integrate Google services and fabric

- Integrate google analytics
- Integrate fabric for crashlytics
- Integrate admob for ads

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**"