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: arkyuan

Go Cook Me

Description

Design a food receipt app that allow user to search recipes and ingredient required to make the meal according to their preference. User is able to follow the checklist or video on the app to complete their receipt instead the traditional way using a book or watch the TV.

Intended User

Everyone who need to cook their own food

Features

List the main features:

- Search receipts based on preference
- Following a checklist or video to complete their cooking
- Reviews from other users

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

This is the main menu. The user can view the receipts image and descriptions based on the search options (by Default, it's sorted by popularity)

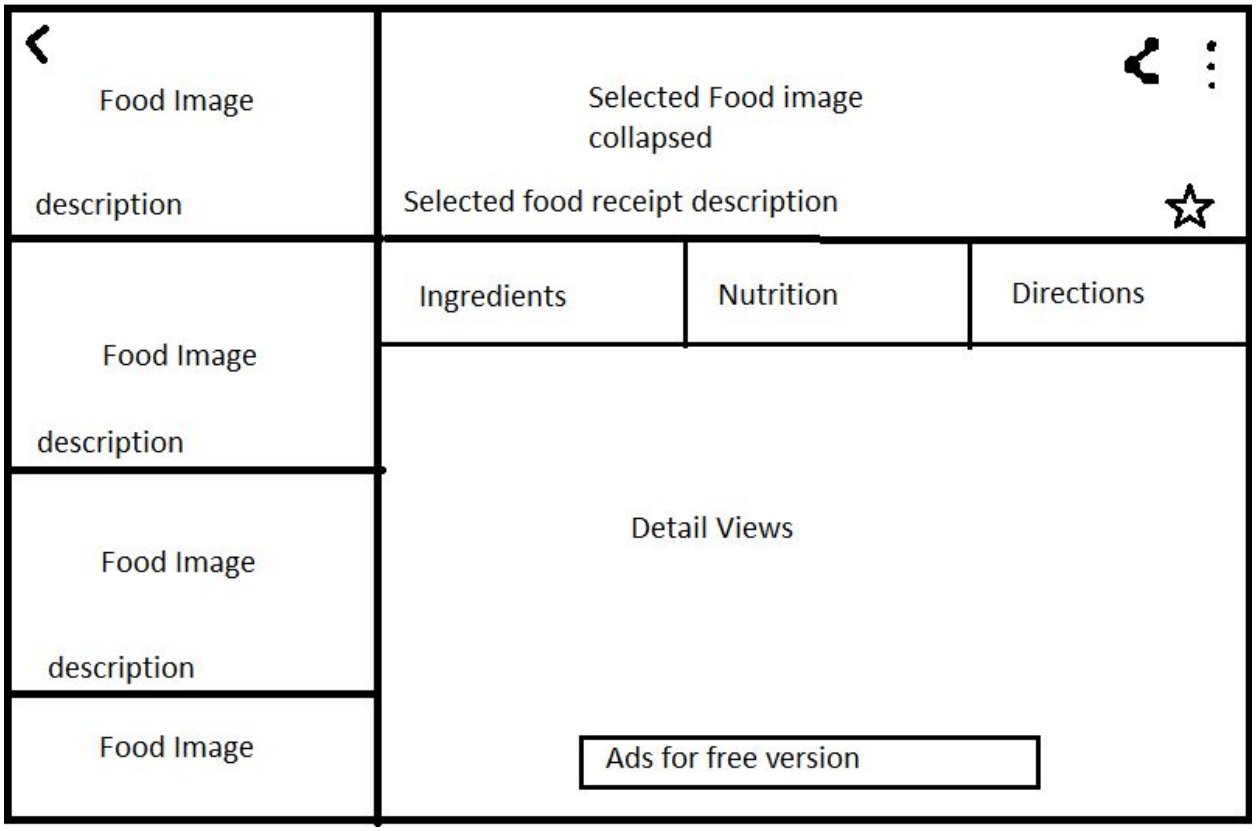
Screen 2

Detail descriptions of the selected food receipt. Will use parallax effect on the Image. You also have three tabs to select from.

1. Ingredients will show you a list of ingredients used for this receipt.
2. Nutrition will show you the nutrition value per serving.
3. Directions will show you a web page or video on how to cook the receipt.

You can also add to favorite by clicking on the star icon. Or Share the receipt's web page with you friends.

Tablet Screen

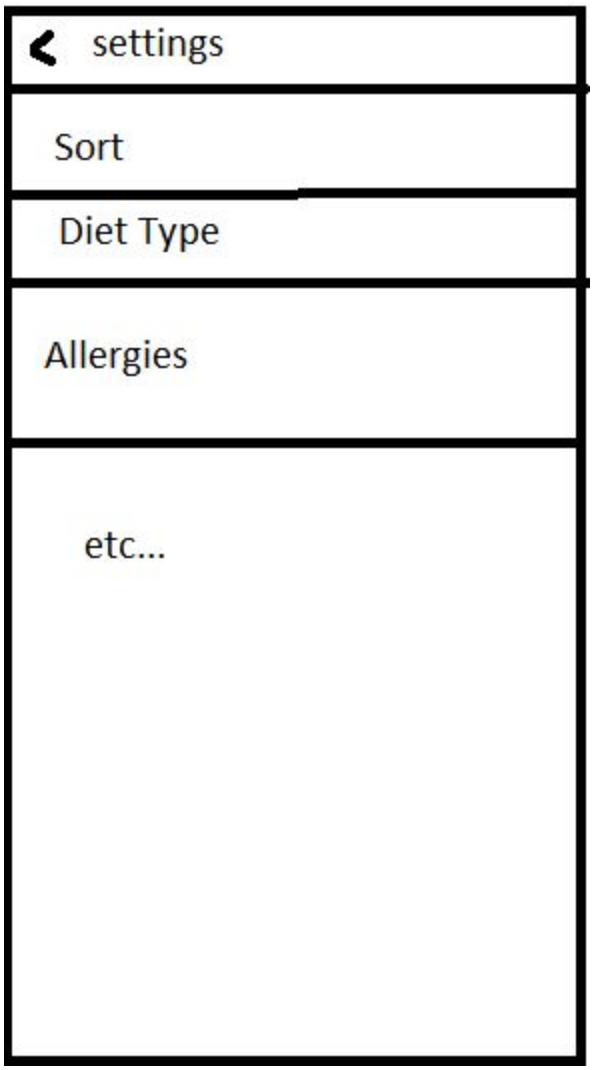


Tablet View, Left side will be Main Activities that shows a list of food receipts.
The right side will be the detail views of selected food receipts. The detail view implementation is the same as above

Widget Screen

Widget will show a list of add to favorite recipes.

Setting Screen



Setting page will show a list of customizable options for search receipts. Like type of diet or filter out certain allergies.

Key Considerations

How will your app handle data persistence?

The app will use a sync adapter to pull the data from Yummly API and store it in the content provider. Then the activities class will use a loader to retrieve the data in the content provider. Then the data will be sent to a 3rd party library (which I will create) to display data on the UI.

Describe any corner cases in the UX.

N/A everything will be stored in the save instance when onPause is called, so when the user come back to the apps, it will be at the same place where they left out. If the user closed the app, it will go back to main screen when they open again.

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

Retrofit will be used to easy maintain the API calls
Stetho will be used in the development stage to check the database data
Picasso might be used to load image from URL (or store URL image as byte[] in the database)
Admob for adds displayed on free version of the app
AppInvites to invite new user to use the app

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

Create a new empty project in Android studio.
Include gradle dependencies in the above libraies.
Targeting the latest sdk

Task 2: Implement UI for Each Activity and Fragment

List the subtasks:

- Build UI for MainActivity/Fragment (main Menu)
- Build UI for Detail Activity/Fragment (detail of the receipts)
- Build UI for Tablet View
- Build UI for Widget
- Build UI for setting page
- Build RTL Layout

Task 3: Create Build Variant

- Create Debug folder and implementations
- Create Release folder and implementations / keystore

Task 4: Implement Content Provider

- Create FoodReceipeContract
- Create FoodReceipeDbHelper
- Create FoodReceipeProvider

Task 5: Implement MainActivity/Fragment

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

Task 6: Implement Detailed Activity/Fragment

- Create Detail Activity
- Create Detail Fragment
- Implement content descriptions
- Empty Views

Task 7: Implement Setting Activity

- Create setting activity

Task 8: Implement Cursor Adapter

- Create Cursor Adapter

Task 9: Implement Sync Adapter

- Create Authenticator
- Create AuthenticatorService
- Create Sync Adapter
- Create Sync Service

Task 10: Implement Widget

- Create Widget Provider
- Create Widget Remote Views Service
- Empty View

Task 11: Implement App Invite (Google services)

- Create App Invite

Task 12: Implement Admob (Google services)

- Create Ad mob

Task 13: Implement Material Design

- Create element transitions

Task 14: Implement ShareActionProvider

- Create Share Action Provider

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**"