

GitHub Username: GeorgeKorovesis

RecipeMe

Description

This app provides list of best rated recipes based on user selection (e.g. chicken, beef). Also gives the opportunity for bookmarking favorite recipes and share with friends.

Intended User

This app is intended for everyone in search of recipes.

Features

Main features of your app:

- Displays information
- Saves information
- Shares information

User Interface Mocks

Screen 1



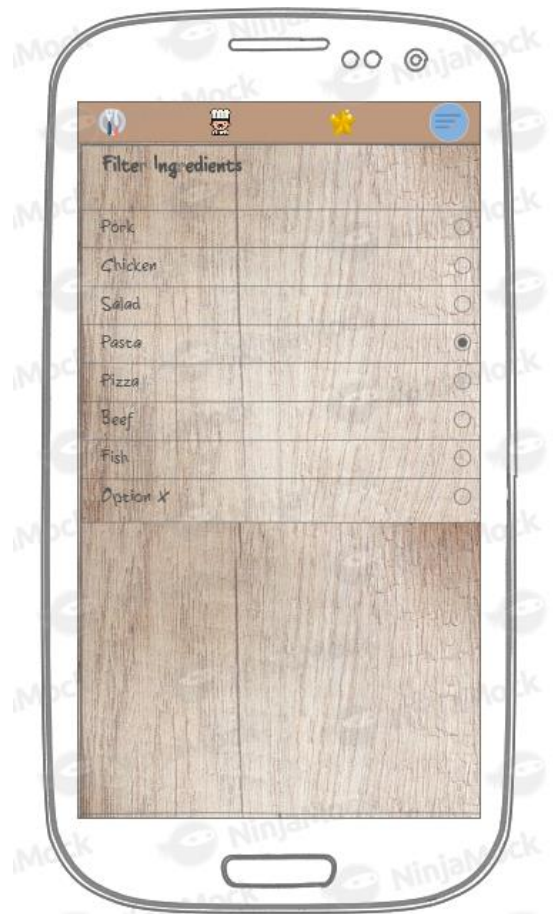
This is the initial screen, where a list of different recipes is presented to user (👤). This screen is also almost identical to the second tab, with the only difference that in second screen recipes are selected base on time of day and rating (🍽️).

Screen 2



This screen appears every time a recipe is selected and contains information about recipe, and functions such as bookmark, share

Screen 3



This image appears every time Option menu item is selected (marked in blue). It contains options to filter the recipes shown in initial screen (🇫🇷)

Key Considerations

How will your app handle data persistence?

Database using Room will be used.

For preferences, SharedPreferences will be used.

Describe any edge or corner cases in the UX.

-

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

Picasso to handle the loading and caching of images.

Retrofit for API calls

Room for accessing db

Butterknife for ease with Android resources

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

Background services will be used for the API calls

Next Steps: Required Tasks

Task 1: Project Setup

- Configure libraries
- Gather necessary icons and images
- Check copyrights
- Add Crashlytics

Task 2: Implement UI for Each Activity and Fragment

There is one main Activity, with an action bar with 4 items, and 4 different Fragments

Task 3: Create models for Room Database

Create models based on the data to be shown to user + extra data such as a column for favorite recipes etc. Should be implemented in parallel with Task 4, in order to define the data needed.

Task 4: Create Retrofit API calls

This will be the [API](#). There is needed a check to see which calls are needed for fetching all recipes, and recipes based on user preference

Task 5: Fill necessary lists with data

Here it shall be defined whether Loaders will be used, or ViewModels.

Task 6: Create scheduler for updates

A scheduler has to be created to trigger for updates in recipes, which combined with task 5 will update data in corresponding lists

Task 7: Handle Error Cases

Handle error cases