GitHub Username: Salild1011

Food Buddy

Description

A Food Recipes App that lets users search for various recipes, ingredients and food products with great flexibility. The detailed instructions for each recipe along with preparation time, nutritional information will also be provided in this app. The user can log the recipes he/she had made and consumed to keep a track of the daily nutritional intake and keep a track of its data over a period of time.

Intended User

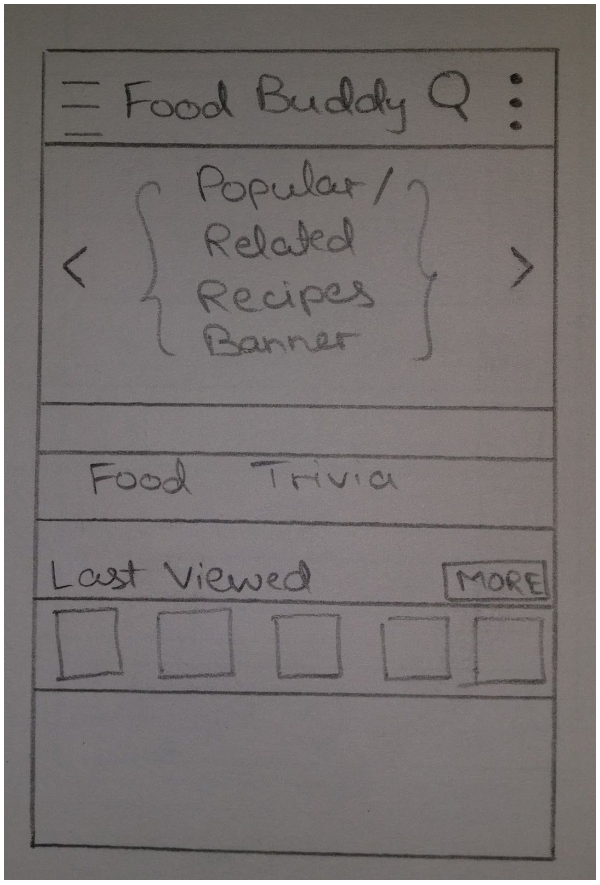
Intended for users who want to search for recipes and for health-conscious people.

Features

- Search for recipes, ingredients, and food items using extensive amount of filters
- Get detailed recipe instructions
- Get nutritional value of a recipe of ingredients
- Store data of daily nutritional intake
- Display historical data of user's nutritional intake in the form of a chart
- Save the links to specific recipes on your Google Drive account

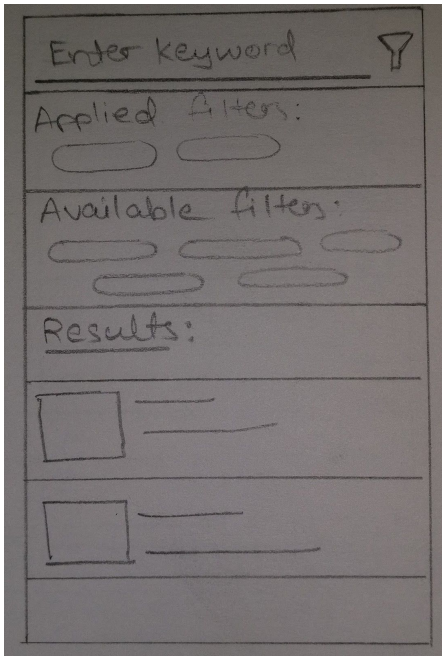
User Interface Mocks

Screen 1



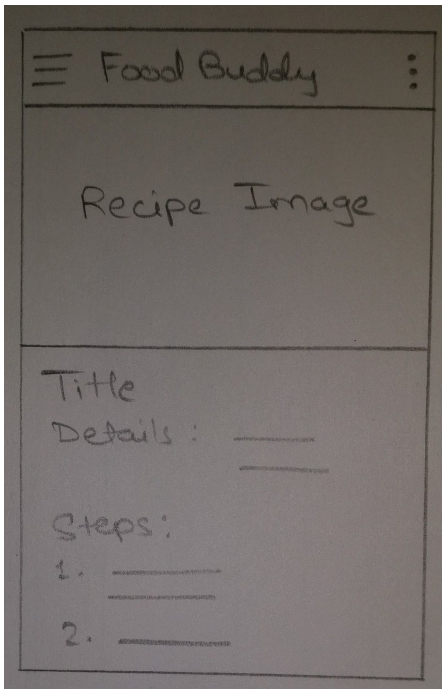
The home screen for the app is as shown above. A banner at the top will display images for random recipes or recipes similar to the user's previous viewed recipes.
Below the banner, a horizontal list of recipes last viewed will be shown.
At the bottom, the nutritional intake of previous days will be displayed.

Screen 2



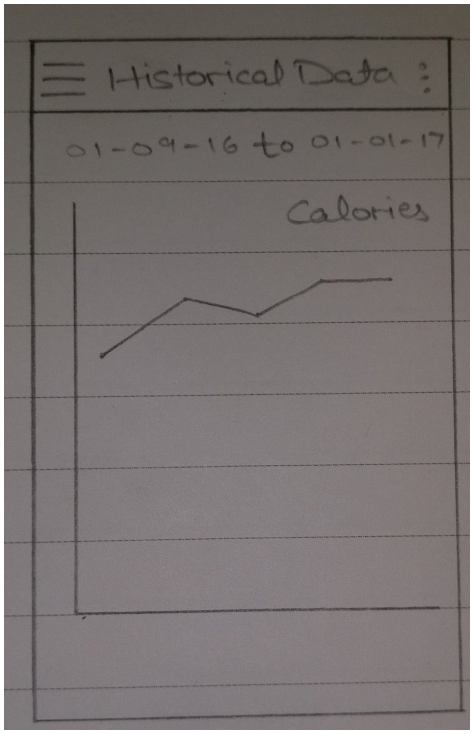
The search screen for the app. Allows to search for recipes and ingredients using the array of filters available. An ad will be displayed as an item within the RecyclerView.

Screen 3



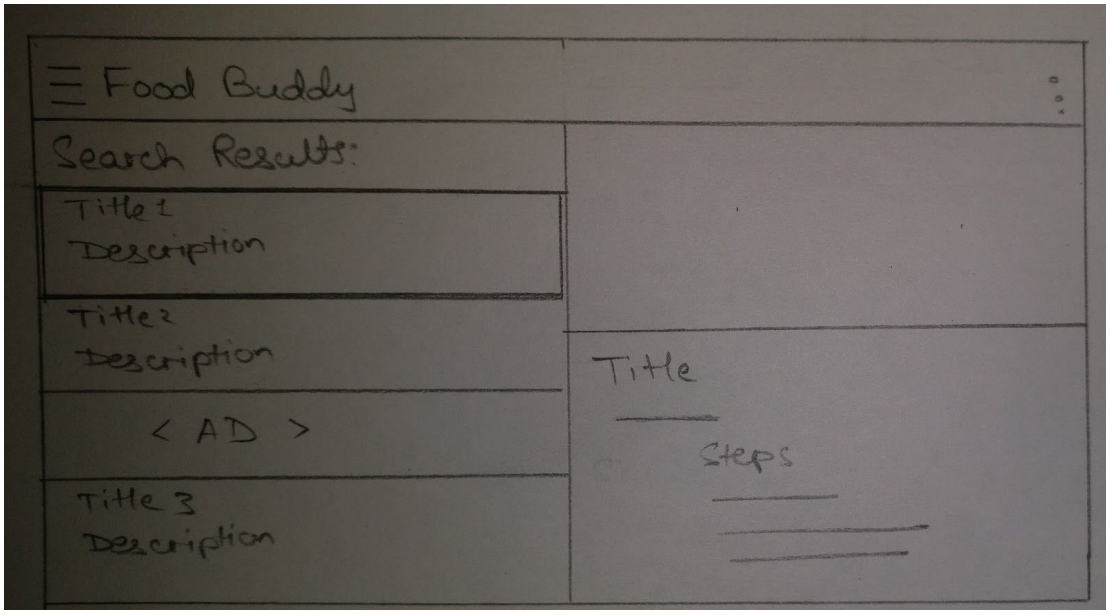
The screen mock for viewing a recipe. Will allow the user to save its link on his/her drive account, and add this recipe as “made and consumed” to add its details to that day’s nutritional intake.

Screen 4



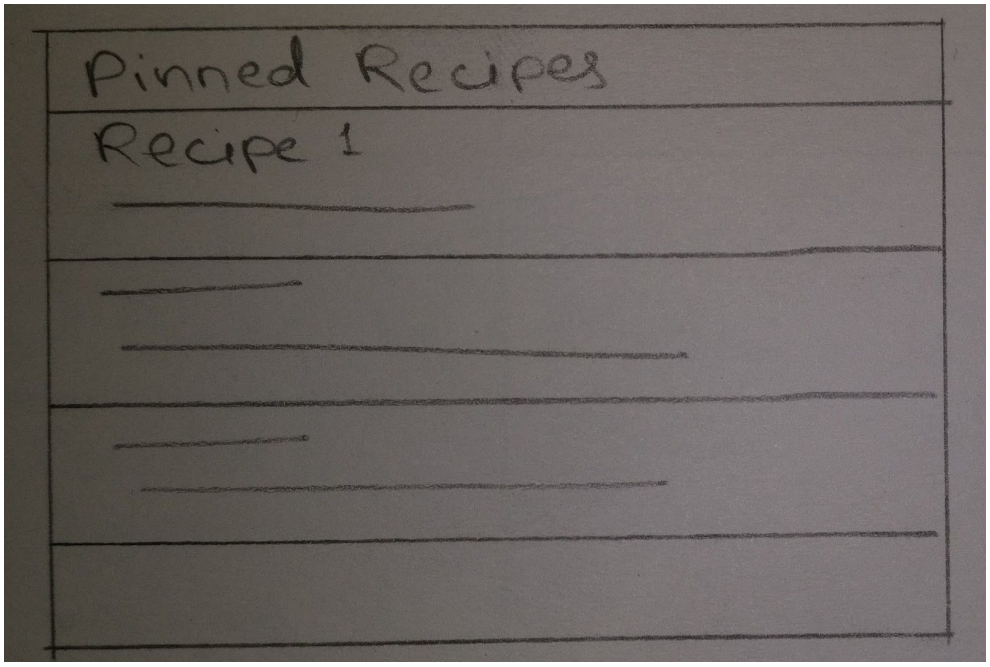
The nutritional intake of user presented in the form of a chart.

Screen 5(Layout for tab with Master-Detail Flow)



The results will be displayed in a master-detail format on landscape mode on tablets. An ad will be displayed as every nth item within the RecyclerView. The selected recipe will be highlighted as shown and its corresponding detailed view will be displayed on the right hand side.

Screen 6(Widget Layout)



The widget will display ‘pinned’ recipes, each of which can be viewed within the app when clicked.

Key Considerations

How will your app handle data persistence?

AsyncTaskLoader will be implemented for fetching detailed data for recipes and food items from the spoonacular API.
For fetching search data, AsyncTask will be implemented

For the nutritional data of user being stored, local SQLite database will be used, using a content provider on top of it.

Describe any corner cases in the UX.

On closing a recipe, or on adding the recipe details to user's Google Drive, the user will be able to navigate back to the main screen

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

- WilliamChart: To create a chart from user data
- Picasso: To handle the loading and caching of images.

Describe how you will implement Google Play Services.

- Google Drive: To store links to certain recipes for personal use
- AdMob: To monetize the app

Next Steps: Required Tasks

Task 1: Project Setup

- Create spoonacular account and get API key
- Configure the libraries required for the project

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
- Build UI for Search mechanism
- Build UI for each option in navigation panel
- Build UI for SettingsActivity

Task 3: Implement food data provision(Online)

- Bind the Activities with data from the spoonacular API
- Fetch data for search operation using AsyncTask
- Fetch data for detailed recipe data using AsyncTaskLoader

Task 4: Implement food data provision(Offline)

- Implement SQLite database and content provider for accessing and storing user's nutritional data on device storage

Task 5: Implement widget

- Create widget UI and display 'pinned' recipes if any
- Handle widget's interaction for appropriate user inputs

Task 6: Implement Google Drive storage and AdMob services

- Implement Google Drive Services
- Bind the app and user's Google Drive account to store the required data
- Implement AdMob Services to display dummy ads