

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Sign in screen](#)

[Home screen](#)

[Recipe List Screen](#)

[Recipe Detail Screen](#)

[Snap 'n' Cook Screen](#)

[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.](#)

[Describe how you will implement Google Play Services.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Implement Google play services](#)

[Task 4: Database design and implementation](#)

[Task 5: Detail screen development](#)

[Task 6: Snap 'n' Cook development](#)

[Task 7: Widget development](#)

[Task 8: Testing and error handling](#)

GitHub Username: riddhik84

Brunchy

Description

Brunchy is an intelligent cooking app that runs on the game changing theories of artificial intelligence and deep learning. It has a wide array of handpicked recipes of delicacies from every part of the globe, tailored to meet the tastes and preferences of our users.

User can set search preference for Veg or Non-veg recipes.

The app supports unique can do snap 'n' cook feature, where users can either upload a food photo or can click a pic and the app will identify dish as well as its ingredients.

The app also shows nutrition information of each recipe in the recipe detail screen so that users can plan a healthy meal by gauging your calorie intake.

Intended User

Food lovers around the globe who love to explore and cook new dishes.

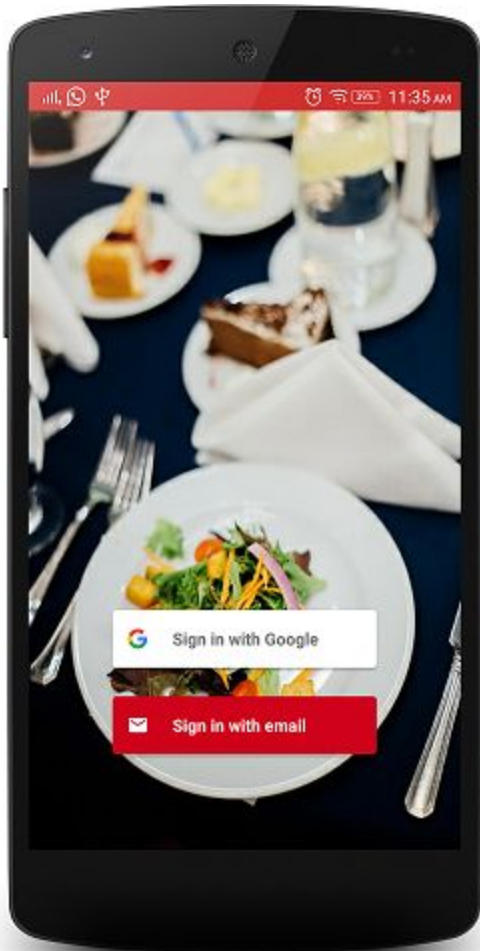
Features

Main Features:

- Predefined categories to explore recipes
- Search recipes
- Snap 'n' cook feature to recipe search
- Recipe detail view with nutrition information
- Mark favourite and Save recipes
- Set calendar reminder to cook a recipe
- Push notifications to explore new recipes
- Set search preference as Veg or Non-Veg
- Search widget

User Interface Mocks

1. Sign In Screen



Launch screen as Sign In screen. User Sign In with Google or with Email. User can also register with Email if there is no existing account.

2. Home Screen



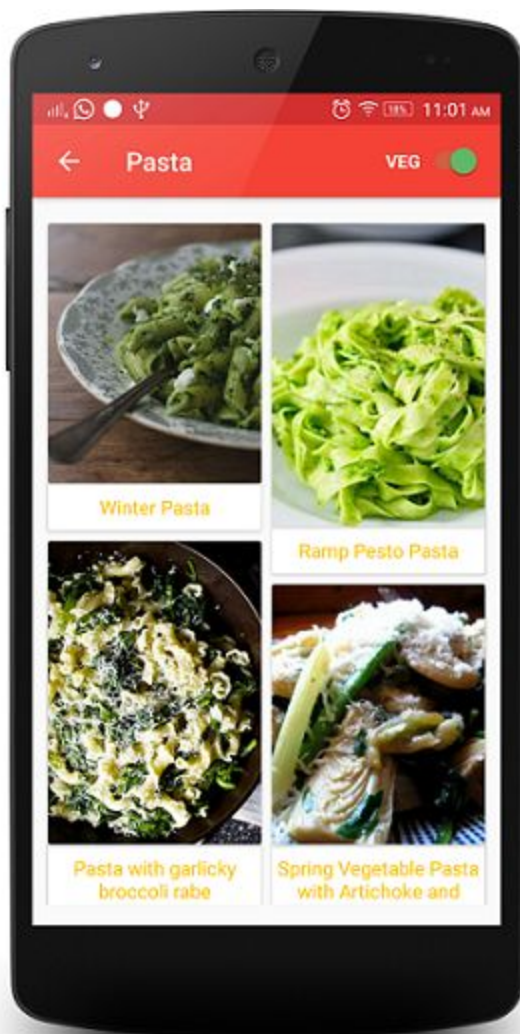
Home screen shows predefined recipe categories. User can select a category and go to next screen where list of recipes will appear.

User can also search a recipe from the search bar at the top of the screen

User can go to Snap 'n' Cook screen by selecting fab button.

User can open navigation drawer by selecting hamburger icon.

3. Recipe List Screen



The recipe list screen will appear when user search a recipe or click on one of the predefined recipe categories.

The screen will also show a switch to set preference as Veg or Non-Veg search.

Clicking on any of the list item will open the recipe detail screen.

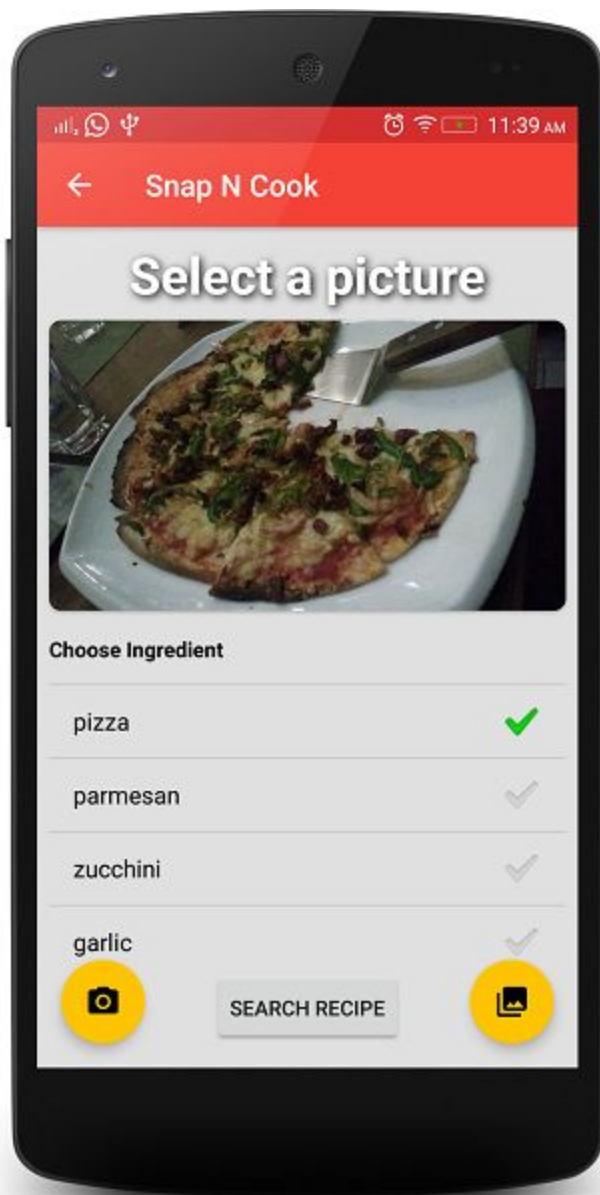
Pressing back will show Home screen of the app.

4. Recipe Detail Screen



The recipe detail screen will show recipe photo, ingredients, calorie and nutrition information and action buttons like favorite, share and reminder.

5. Snap 'n' Cook Screen



The snap n cook screen will show button to upload a recipe image or click a food image. The screen will show image analysis result in a selectable list view. User can select an ingredient and select search recipe to look for similar recipes.

Key Considerations

How will your app handle data persistence?

User will be able to save recipes which they will mark as favourite to the SQLite database.

Describe any corner cases in the UX.

None

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

I will use below libraries in my project:

- Picasso image library to load the recipe images smoothly.
- I will use clarifai API to support snap 'n' cook feature.
- I will use retrofit to make network calls.

Describe how you will implement Google Play Services.

- I will use firebase authentication to for app login.
- I will use firebase notification to send periodic notification to users to explore new recipes. Clicking on the notifications will open recipe list.

Next Steps: Required Tasks

Task 1: Project Setup

- Android studio will be used for app development.
- Obtain API keys and required permission to use edamam recipe APIs and Clarifai SDK in the app.
- Create firebase project and add support for firebase in the app.
- Declare all the required libraries in the build.gradle file and resolve if any errors.

Task 2: Implement UI for Each Activity and Fragment

List the subtasks. For example:

- Build UI for MainActivity

- Build UI for Recipe List view
- Build UI for Recipe detail screen
- Build UI for snap 'n' cook screen
- Build UI for settings
- Build UI for favorite screen

Task 3: Implement Google play services

- Implement Firebase Authentication service for user Sign In
- Implement Firebase Notification service for periodic recipe suggestions

Task 4: Database design and implementation

- Design database to store recipes marked as favorite by user
- Implement recipe data saving for favorite recipes

Task 5: Detail screen development

- Develop recipe detail screen with action buttons like favorite, share and reminder
- Display recipe photo, calories, servings, total ingredients, detailed ingredients, nutrition info

Task 6: Snap 'n' Cook feature development

- Develop snap 'n' cook feature using Clarifai sdk

Task 7: Widget implementation

- Develop a widget to search recipes

Task 8: Testing and Error handling

- Test the app functionalities and fix any bugs
- Implement error handling