

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[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: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

**GitHub Username:** smoke275

# Food Scout

## Description

Hungry? Stuck at a new place and can't find a restaurant? Well don't worry. Food Scout is here to help. It finds you the best restaurants in the vicinity, so that you don't have to stay hunger for long.

## Intended User

Tourists

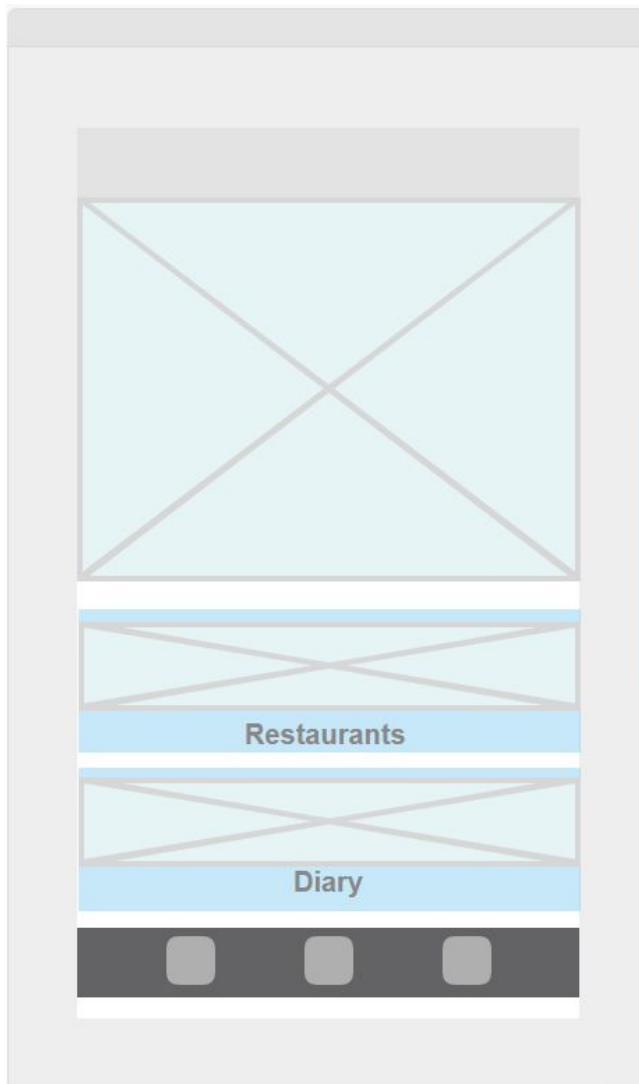
## Features

- Find restaurants to eat according to their reviews, prices and other parameters
- Stay safe with GPS tracking
- Find routes to the selected restaurants

- Facility to create personal diary

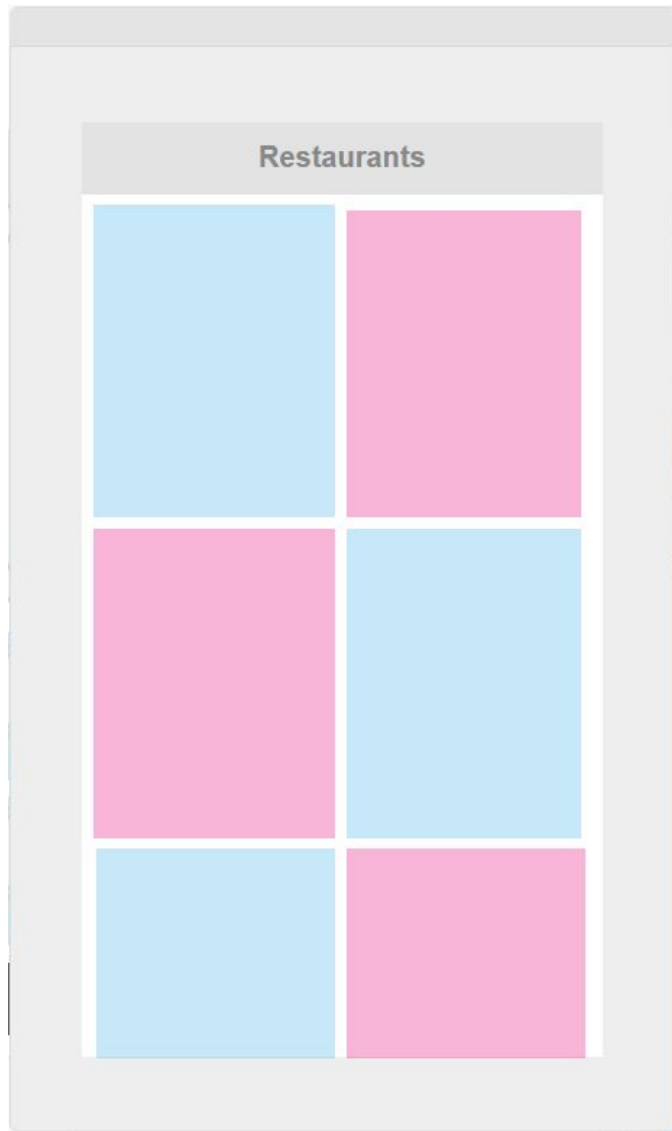
## User Interface Mocks

### Screen 1



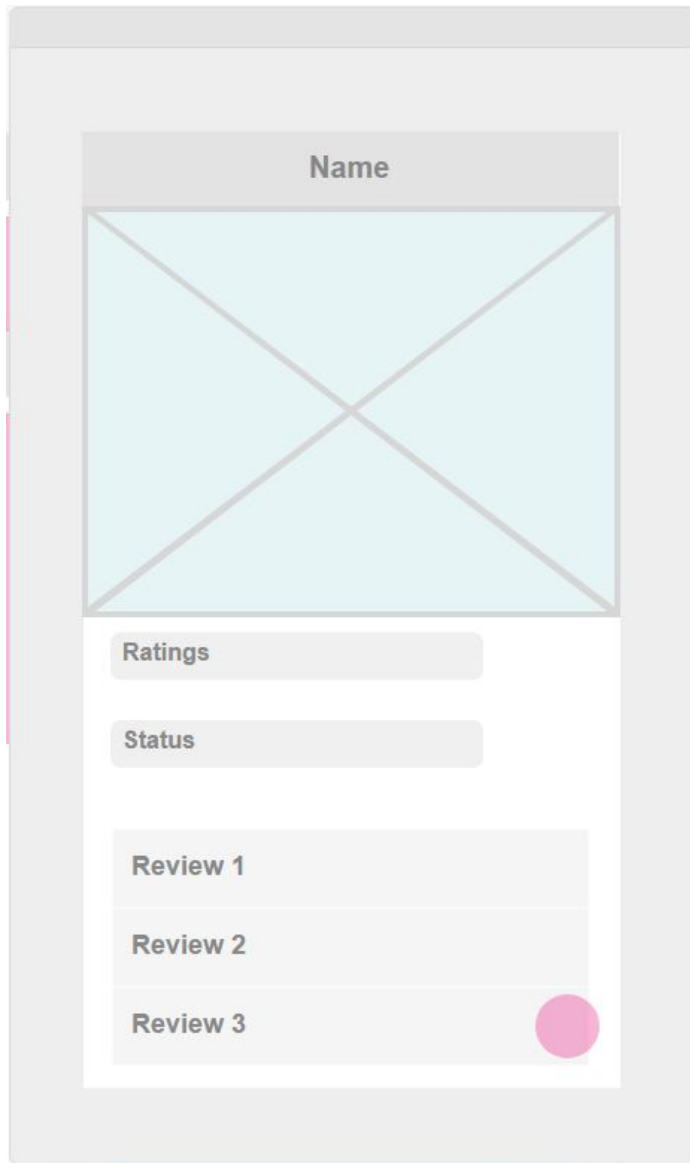
Find places to visit the local restaurants near the person's GPS location or after getting an input from the user

## Screen 2



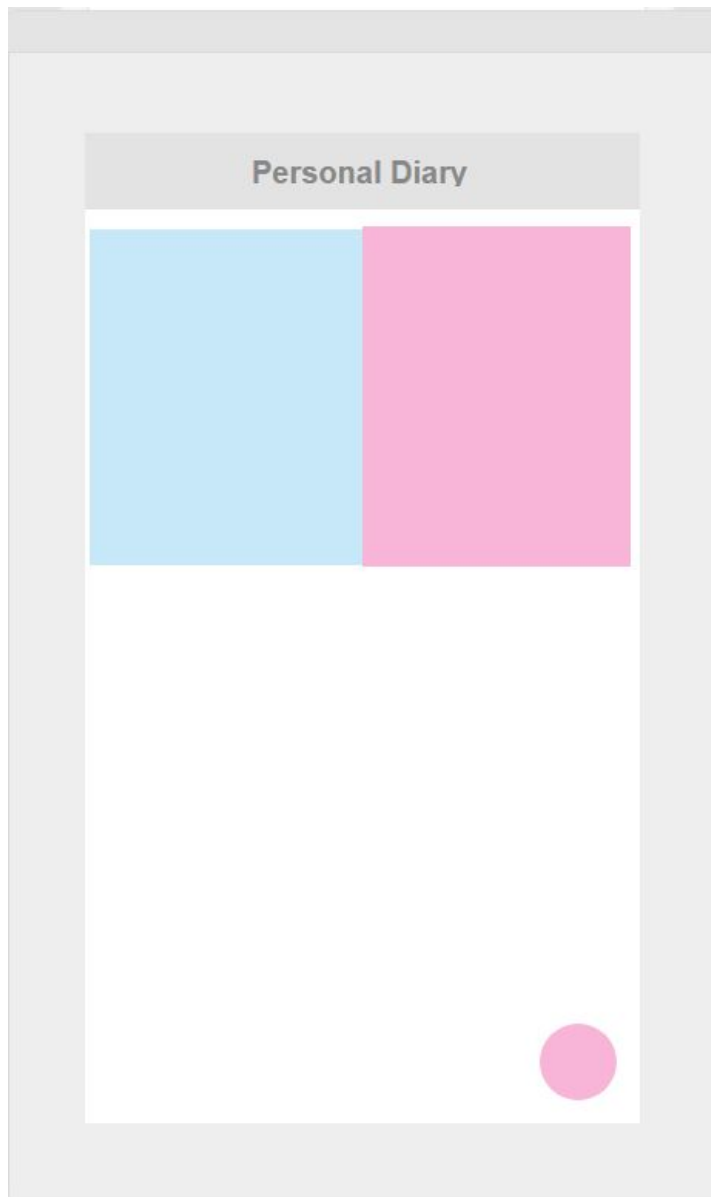
For displaying the restaurants found as cards

### Screen 3



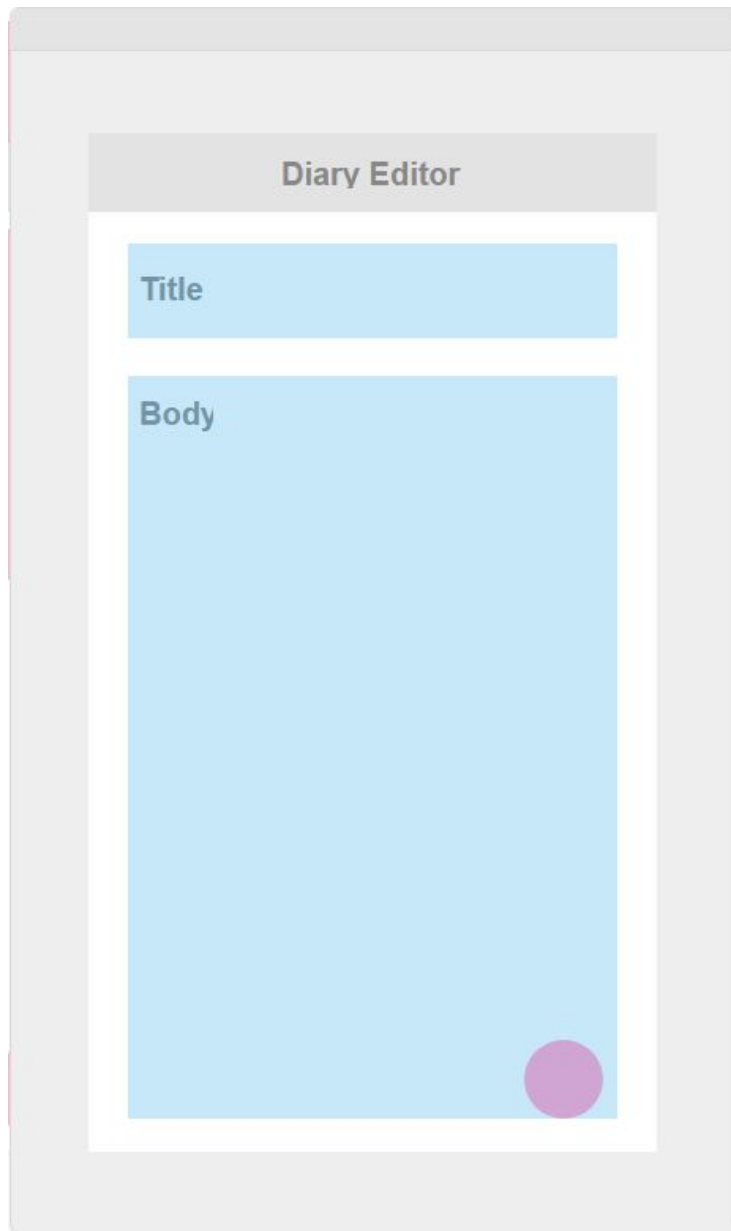
A detailed view of the restaurant shown in the card

#### Screen 4



Personal Diary screen to show the list of memories created

## Screen 5



Editor Screen to edit or add memories

## Key Considerations

How will your app handle data persistence?

Data persistence will be handled using a new Content Provider to store Personal Diary data.

### **Describe any corner cases in the UX.**

If the GPS data is unavailable, the app will ask the user to select the their place of interest.

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

- Retrofit for parsing JSON data returned by the servers
- Glide for loading and caching images
- Okhttp for data transfer
- Butterknife for view binding
- Schematic to reduce content provider boilerplate codes
- Google play services for fetching maps, places and Admob

### **Describe how you will implement Google Play Services.**

- Google Places to get nearby places of interest
- Google Maps to get map data
- Admob to get advertisements

## **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 initial project in Android Studio.
- Create a workflow.
- Obtain API keys for various Google Play services from the developer console.
- Configure gradle build scripts to include relevant libraries.

### **Task 2: Implement UI for Each Activity and Fragment**

- Design loading screen

- Create UIs to obtain location from user, if required.
- Build UI for each activity.
- Build UI fragments to display restaurants etc.

### **Task 3: Design Alternate UI**

- Design UI alternatives for tablet screens and enable them for tablet users.

### **Task 4: Integrate Services to get and store Data**

- Integrate Google Places and Google Maps API to get required information.
- Implement a Content Provider to store and retrieve personal diary data.

### **Task 5: Create Widget**

- Design widgets to add to the home screen.
- Adding event handlers to the widget.