GitHub Username: jfleurent

Find'N'Dine

Description

Have you ever felt pressured to pick a restaurant? Felt the constant sense indecisiveness of finding a restaurant that would meet your preferences? Or wanting to find that hidden gem in your local town? For these questions, Find'N'Dine will solve problems and more. With features like searching for a place based on their price, distance, or food type, and having the option to categorize your favorites, Find"N'Dine has made it easy for you to find the places you love and find that one place that satisfies you cravings.

Intended User

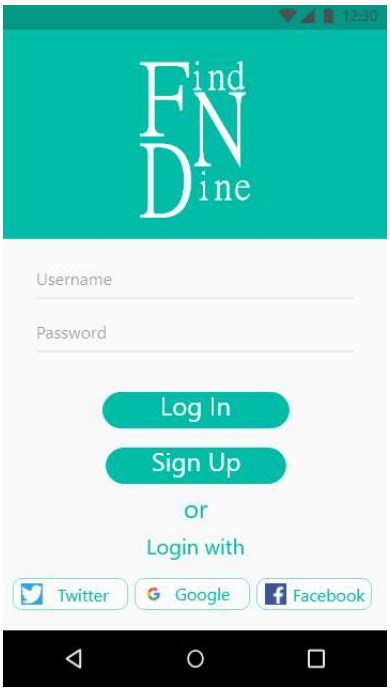
Anyone who wants to find local restaurant in their area.
Anyone who is indecisive on where to go for their meals.

Features

- Users will be given a randomly selected restaurant based on their preferences
- Users will be given a history of restaurants viewed
- Users will have the ability to query for restaurants based on distance, price, and other parameters
- Users will be able to login through a personal account with app, Facebook, or Google.
- Users will be able to customize their profile page
- Users will be able to see reviews and photos of selected restaurants

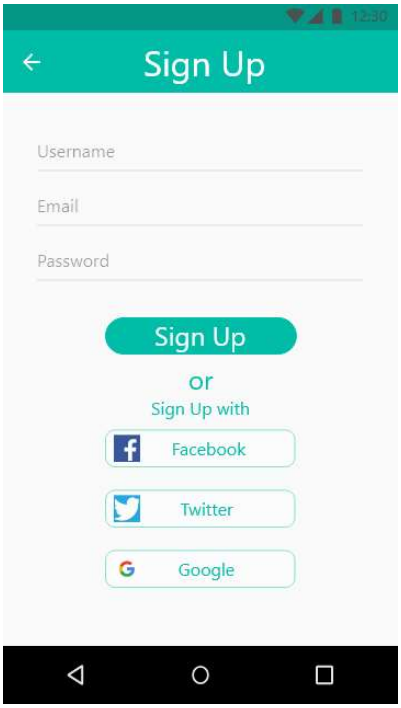
User Interface Mocks

Login Screen



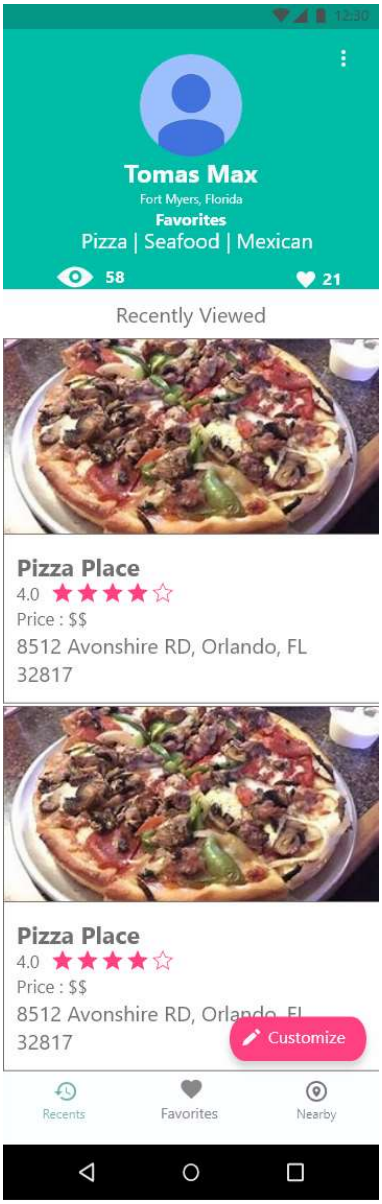
Used for logging in the user with a twitter, facebook, google, or an account with the app. Also provides a sign-up option for users that don't have an account.

Sign Up Page



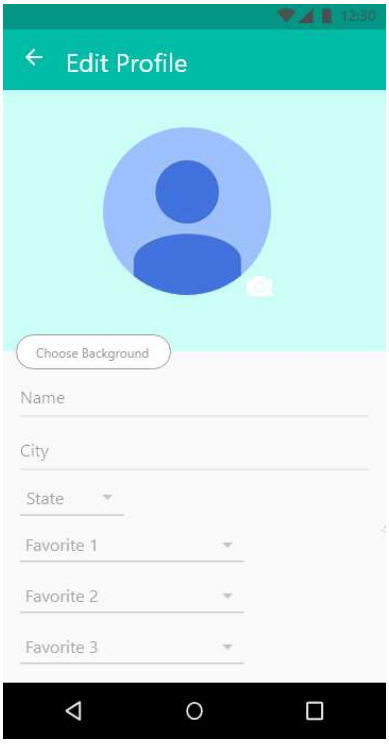
Used for signing up users to the app.

User Home Page



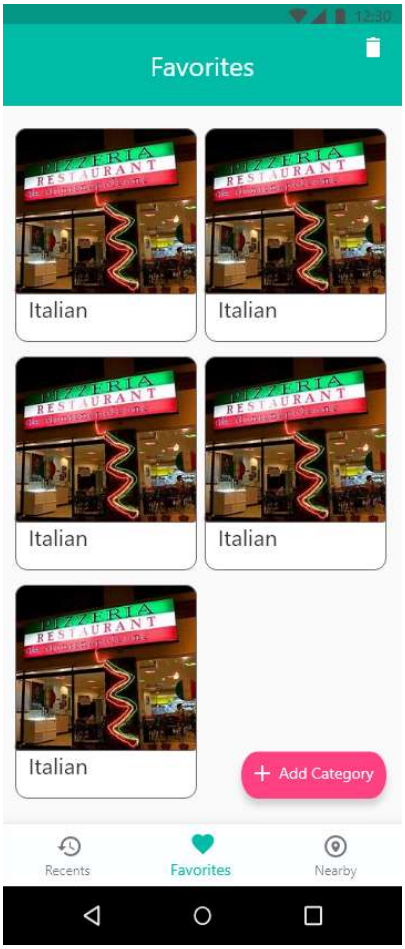
This page provides basic information about the user and shows a list of recent places the user's has viewed in the app. There's also an option to customize the page with the action button.

Edit Profile Page



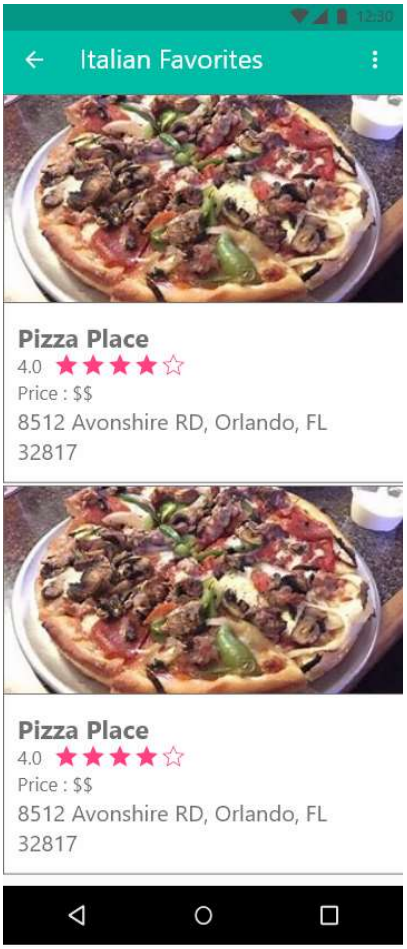
This page is used for customizing the profile image, background color, and other information about the user.

Favorites Page



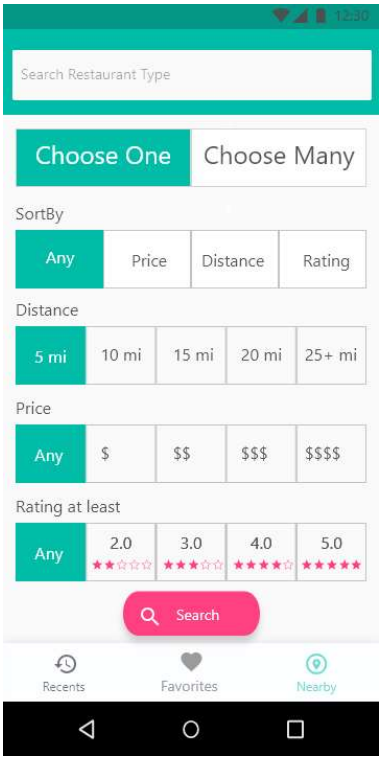
This page contains a list of categories favorites that link to another page that contains a list of restaurants. The user could add new categories by clicking on the action button and delete categories by clicking on the trash can icon at the top.

Detailed Favorites Page



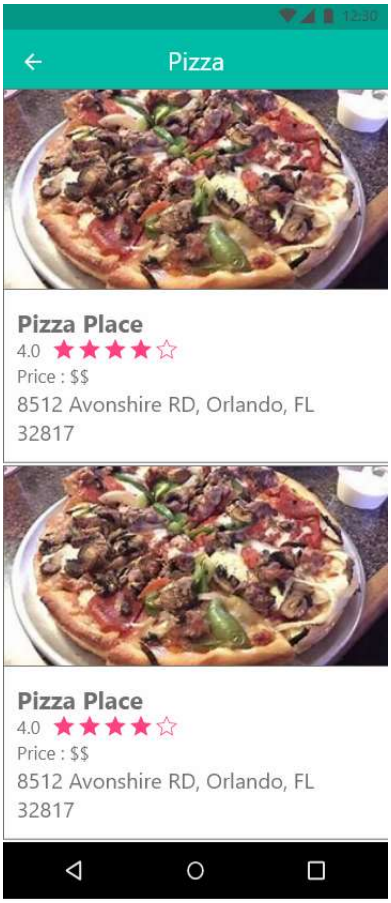
This is page that will open when a user clicks on one of the categories in the Favorites Pages. This page also contains a list of restaurants that the user favorited for a category and there's an option to change the title of the category in the options icon at the top.

Search Page



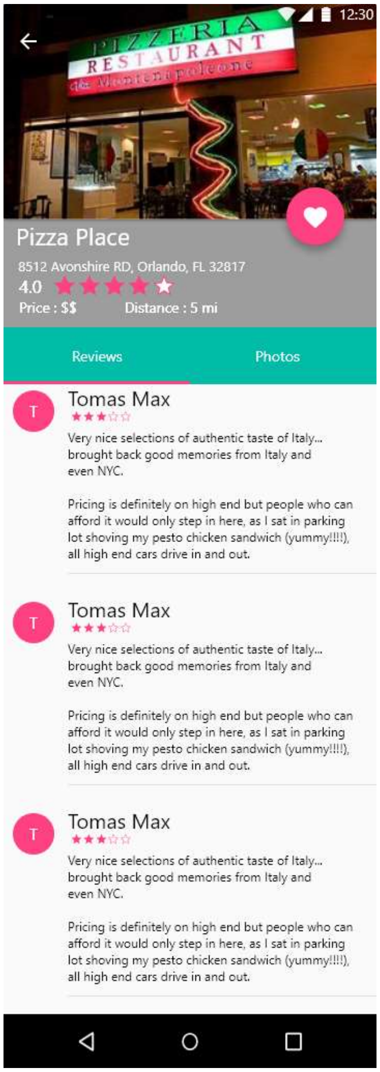
This page is used to search for varying restaurant types with the parameters for price, distance, ratting being selectable by the user. The search bar will create a chip for each keyword the user enters which will be used to generate results in the next activity.

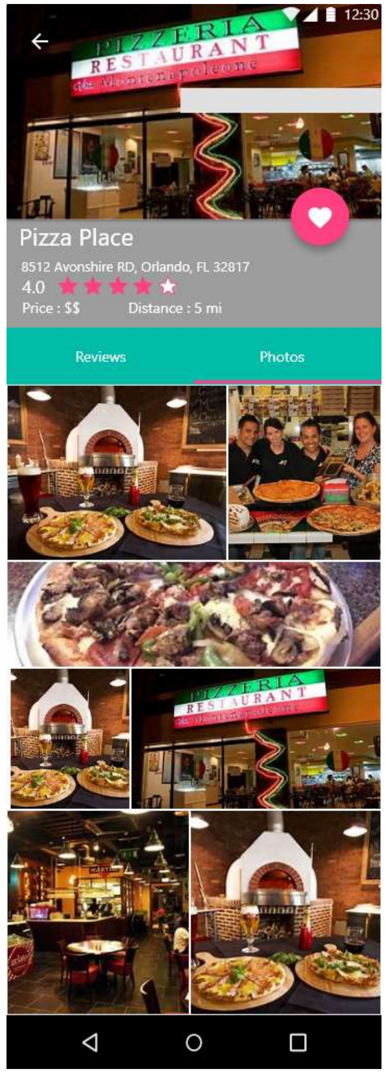
Result Page



This is the page that will appear right after the user enters their search in the Search Page.

Detailed Restaurant Page





This is the page that will display the detailed info about the restaurant, review and images of the restaurant.

Key Considerations

How will your app handle data persistence?

Firebase Realtime Database will be used to store user's profile information and user's favorited restaurants.

Describe any edge or corner cases in the UX.

There will be animated transition when user goes from home page to selected restaurants
.

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

Picasso (https://github.com/JakeWharton/butterknife)

- For ease of loading images to views through the web

TapTargetView (https://github.com/KeepSafe/TapTargetView)

- Adds directions to UI components in app

Retrofit (https://github.com/square/retrofit)

- Type-safe HTTP client for Android

Butterknife (https://github.com/JakeWharton/butterknife)

- Ease the process of creating view objects

Android Material Color Picker Dialog (https://github.com/Pes8/android-material-color-picker-dialog)

- Color Picker for profile page

Gson (https://github.com/google/gson)

- Used to convert JSON strings into java classes

Timber (https://github.com/JakeWharton/timber)

- Simplify logging

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

The Google Play Services that are going to be used are Location and Identity.

Next Steps: Required Tasks

Task 1: Project Setup

- Configure libraries
- Set-up Firebase Realtime Database
- Get Account for Google Places API and Yelp API

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity in XML (Login Page)
- Build UI for SignupActivity in XML (Signing Page)
- Build UI for NavigationActivity (User Home, Favorites, and Search Page)
- Build UI for HomepageFragment in XML (User Home Page)
- Build UI for FavoritesFragment in XML (Favorite Page)
- Build UI for SearchFragement in XML (Search Page)
- Build UI for RestaurantListFragment in XML (User Home, Favorites, Result Page)
- Build UI for CustmizedPageActvity in XML (Edit Profile Page)
- Build UI for ReviewsFragment in XML (Detailed Restaurant Page)
- Build UI for PhotosFragment in XML (Detailed Restaurant Page)
- Build UI for DetailedRestaurantActivity in XML (Detailed Restaurant Page)
- Build UI for RestaurantItem in XML (User Home, Favorites, Result Page)
- Build UI for FavoritesCategoryItem in XML (Favorite Page)
- Build UI for ChangePhotoDialog in XML (Edit Profile Page)
- Build UI for ChangeBackgroundDialog in XML (Edit Profile Page)
- Build UI for CreateCategoryDialog in XML (Favorites Page)
- Build UI for AddFavoriteDialog in XML (Detailed Restaurant Page)

Task 3: Create Classes

- Create class for restaurant
- Create class for user
- Create class for restaurant images
- Create class for category favorites
- Create class for retrofit client
- Create interface for retrofit API calls
- Create classes for recyclerview adapters

Task 4: Implement Functionality to Activities and Fragments

- Implement functionality to MainActivity (Login Page)
- Implement functionality to SignupActivity (Signing Page)
- Implement functionality to NavigationActivity (User Home, Favorites, and Search Page)
- Implement functionality to HomepageFragment (User Home Page)
- Implement functionality to FavoritesFragment (Favorite Page)
- Implement functionality to SearchFragement (Search Page)
- Implement functionality to RestaurantListFragment (User Home, Favorites, Result Page)
- Implement functionality to CustmizedPageActvity (Edit Profile Page)
- Implement functionality to ReviewsFragment (Detailed Restaurant Page)
- Implement functionality to PhotosFragment (Detailed Restaurant Page)
- Implement functionality to DetailedRestaurantActivity (Detailed Restaurant Page)
- Implement functionality to ChangePhotoDialog (Edit Profile Page)
- Implement functionality to ChangeBackgroundDialog (Edit Profile Page)
- Implement functionality to CreateCategoryDialog (Favorites Page)
- Implement functionality to AddFavoriteDialog (Detailed Restaurant Page)

Task 5: Test and Debug Features of the App

- Test behavior of the app when there’s no internet connection
- Test the animations in the app
- Test the data that is loaded in the view of the app
- Test functionality of UI components in the app
- Test the functionality of the database