GitHub Username: ksheremet

Cocktails Pro

Description

Are you preparing for a party or just want to cook something yummy for yourself? It means that this is the right app for you. Enjoy refreshing non-alcoholic drinks, explore and create different excellent cocktails, relish hot chocolate drinks. Add whatever you like to your favourites to have an offline access.

Intended User

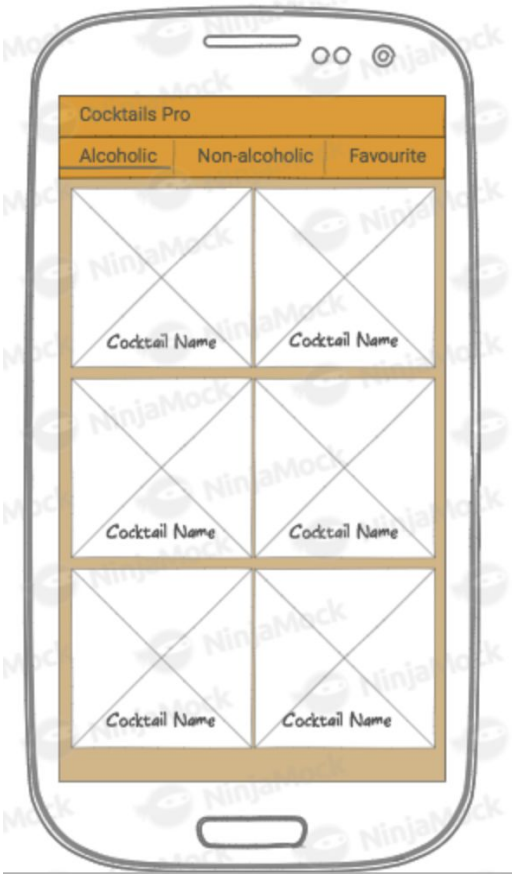
This app is for party-makers, families, for people who want to diversify their beverages, cocktails lovers, hot chocolate drinkers.

Features

- List of non-alcoholic beverages
- List of alcoholic cocktails
- List of Cocoa drinks
- List of Favourite drinks added by user
- Detailed information about a drink: ingredients, preparation, picture
- The app has a companion homescreen widget, that displays ingredients of a beverage.

User Interface Mocks

Screen 1

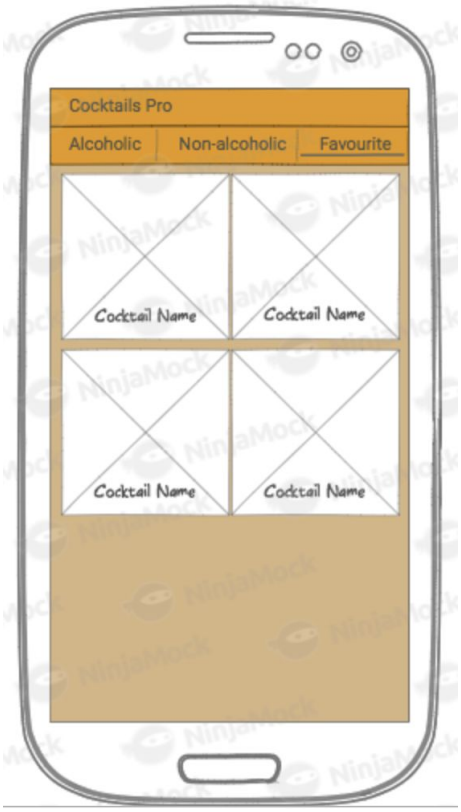


The screen contains a grid of beverages. User can click on tabs in Action Bar to choose between alcoholic, non-alcoholic, favourites, cocoa beverages. Every item in Grid has picture and name of a cocktail. When user taps on a picture, opens a screen with detailed information.

Screen 2

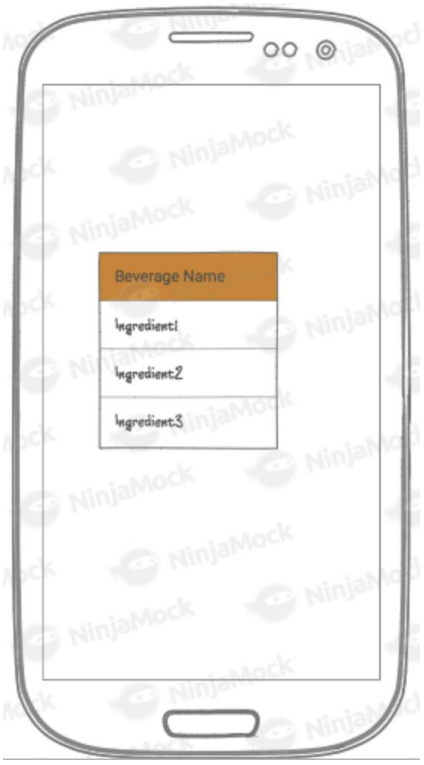
The screen contains a detailed information about cocktail: name, ingredients, preparation, picture. User can add a beverage to favourites by clicking on the “Add to Favourites” button.

Screen 3



The screen contains an information about favourite drinks of the user. This information is available even when the user is offline.

Screen 4



Widget displays name and ingredients for the last marked favourite beverage. By clicking on the widget the Cocktail Pro app opens.

Key Considerations

How will your app handle data persistence?

The user’s favourite beverages are stored using Room. Data is updated whenever the user favourites or unfavourites a beverage. No other persistence libraries are used.
When the "Favourite" tab is selected, the main view displays the entire favourites collection based on beverages stored in the database.

Describe any edge or corner cases in the UX.

UI contains tabs to change content on the main screen: alcoholic, non-alcoholic, favourite, cocoa beverages. Beverages are displayed via a grid of their corresponding thumbnails. When a user changes the tab (non-alcoholic, alcoholic, cocoa, favourite) the main view gets updated correctly.

UI contains a screen for displaying the details for a selected beverage.
Beverage details layout contains name, ingredients, beverage poster, preparation instructions.

When a beverage thumbnail is selected, the beverage details screen is launched. In the beverage detail screen, a user can tap a button (for example, a star or heart) to mark it as a Favourite. Tap the button on a favourite beverage will unfavourite it.

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

- **Picasso** to handle the loading and caching of images (http://square.github.io/picasso/)
- **Retrofit** to fetch JSON data from API (http://square.github.io/retrofit/)
- **Google Gson** to convert JSON to Java Object (https://github.com/square/retrofit/tree/master/retrofit-converters/gson)
- **Android Architecture Components** for managing UI component lifecycle and handling data persistence
- **ConstraintLayout** to create large and complex layouts with a flat view hierarchy
- **Dagger2** will be used for dependency injection
- **The Cocktails API** for fetching an information about different beverages (https://www.thecocktaildb.com/api.php)

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

- **AdMob** is used to display advertisement in a free version of the app.
- **Firebase Crashlytics** is used as a crash reporter that helps track and fix stability issues.
- **Google Analytics** will be used to measure user activity to named screens.

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

Task 1: Project Setup

- Create a project in the Java Programming Language
- Use stable release versions of all libraries, Gradle, and Android Studio

Task 2: Implement fetching cocktails data

- Fetch list of non-alcoholic, alcoholic, cocoa beverages using Retrofit
    - Alcoholic: https://www.thecocktaildb.com/api/json/v1/1/filter.php?a=Alcoholic
    - Non-alcoholic: https://www.thecocktaildb.com/api/json/v1/1/filter.php?a=Non_Alcoholic
    - Cocoa: https://www.thecocktaildb.com/api/json/v1/1/filter.php?c=Cocoa
- Fetch a beverage by id using Retrofit
    - Ex:https://www.thecocktaildb.com/api/json/v1/1/lookup.php?i=13060

Task 3: Implement saving favourite beverages to DB

- Setup Android Architecture Components library for persistency
- Implement adding favourite beverage to DB
- Implement removing favourite beverage to DB
- Implement fetching list of favourite beverages from DB

Task 4: Implement UI for Each Activity and Fragment, Widget

- Build UI for MainActivity and display thumbnails of beverages
- Implement tabs for different search options
- Build UI for detailed beverage activity
- Implement Widget

Task 5: Implement error reporting using Firebase Crashlytics

- Setup Firebase Crashlytics
- Report errors in Crashlytics

Task 6: Create Product Flavors

- Create free and paid product flavors
- Setup AdMob library
- Show advertisement in free version

Task 7: Implement Google Analytics

- Setup Google Analytics
- Add screen tracking