GitHub Username: Hello-xhemajl

Sneakerverse

Description

Sneakerverse allows users to browse through sneakers they might like. They also can read the story behind the design(sneakers detail view). The app also shows the user's collection of favourited sneakers.

Intended User

The app is intended for cool people of all ages who like to wear sneakers and especially for those who like to find ones that correlate with their character.

Features

Main features include:

- Firebase authentication
- Mark sneakers as favorite
- Show favorite collection
- Show sneakers details
- Browse sneakers
- Free version will show ads

Languages build on

App is written solely in the Java Programming Language

User Interface Mocks

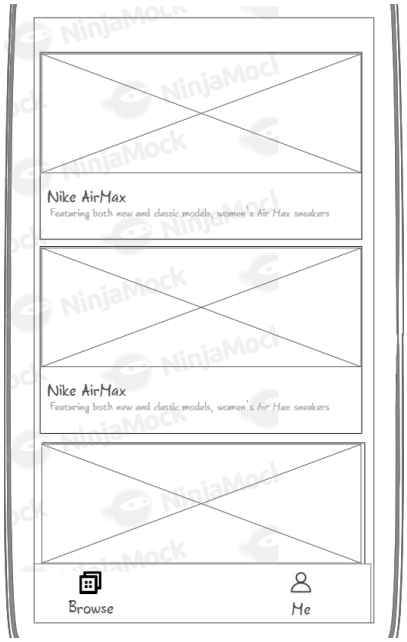
Login Screen

The login screen will allow users to sign-in with Google or Facebook

Favorite Collection Screen

Users will be able to see their favorite sneakers collection and their profile picture

Browse Sneakers Screen

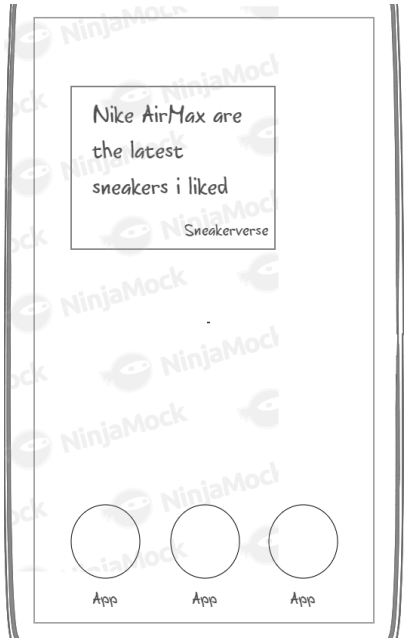


Sneakerverse allows users to browse through sneakers they might like.

Sneakers Detail Screen

Users can read the story behind the design(sneakers detail view)

App widget



App widget will show the latest favourited sneakers and with a tap the user will see the sneakers detail view

Key Considerations

How will your app handle data persistence?

As a start sneakers will be added manually into a SQL database using Room. Also users favorite collection will be persisted locally. In the future(**not for the project submission**) it will be considered to store data on the backend or services from firebase will be used.

Describe any edge or corner cases in the UX.

None.

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

Firebase Authentication will be used since it supports immediate sign-up/sign-in with Google, Facebook or Twitter.

For the free version. Add mob from Google will be used for the app ads. Google has demonstrated that it is a good choice for ads through the years.

For loading images Picasso will be used since it provides an intuitive and easy to use API. It also performs well.

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

To support sign-up with Google and Facebook, Firebase Authentication will be used. Users will be able then to sign in with one tap. App will request name and a picture from the users Google or Facebook account.

The free version will have ads and those will be shown sparingly using Admob service.

All libraries and IDE

- Android Studio version 3.3.2
- gradle version 4.10.1
- firebase-core version 16.0.8
- firebase-auth version 16.2.1
- firebase-ads version 17.2.0
- lifecycle-extensions version 1.1.1
- room-runtime version version 1.1.1
- work-runtime version version1.0.0

Next Steps: Required Tasks

Task 1: Project Setup

To begin:

- Create a Firebase account
- Import Firebase Authentication Library
- Import Admob library
- Configure and connect app with Firebase
- Configure and connect app with Admob
- Import Picasso library
- Create a free and paid flavors
- Import Room and ViewModel libraries

Task 2: Design and Implement local database

App will store sneakers locally using SQLite database. To support sneakers and user data:

- Design and create tables for the sneakers
- Design and create tables for the users
- Design and create tables for user's favorite collection

Task 3: Manually populate local database with sneakers

Write a database loading script that inserts sneakers data like the design story and sneakers image URI.

SIGN-UP/SIGN-IN FEATURE

Task 4: Design and Implement UI for LoginActivity

Users will be able to sign up and sign in using Firebase Authentication:

- Build layout for LoginActivity
    - Promote sign-up with Google or Facebook.
    - Provide sign-in with Google or Facebook.

Task 5: Wire Firebase Authentication and LoginActivity

Users will be able to sign up and sign in using Firebase Authentication:

- Implement sign-up Firebase Authentication
- Implement sign-in Firebase Authentication
- Persist logged-in user after firebase says OK

BROWSE FEATURE

Task 6: Design and Implement UI for BrowserActivity

Browser Activity will enable users to browse through sneaker.

- Build layout for BrowseActivity
- Build layout for sneaker item.

Task 7: Implement BrowseActivity's ViewModel

A ViewModel will be responsible for providing sneakers data. For getting the sneakers data:

- Build ViewModel that provides sneakers data
- Make BrowserActivity work with the VIewModel
- Plan for handling sneakers data returned by the ViewModel

Task 8: Implement and wire RecyclerView, Adapter and LayoutManager

Users will be able to scroll through available sneakers presented as a list. To support that:

- Build Adapter that will provide sneakers items
- Create a LayoutManager that should arrange provided items in a vertical list.
- Make RecyclerView work with Adapter and the LayoutManager

Task 9: Support navigation to sneakers detail view.

Handle user's intent to see sneakers detail:

- Support navigation to sneakers detail view

Task 10: Design and Implement UI for SneakersDetailActivity

Sneakers detail view will show the design story of the sneakers.

- Build layout for SneakersDetailActivity
- Configure ImageView that will show the sneaker's image.

Task 11: Support Mark as Favorite feature

Sneakers detail view will allow users to mark sneakers as favorite.

- Handle user's intend to mark sneakers as favorite:
    - Persist sneakers to user's collection on the local database.

FAVORITE COLLECTION FEATURE

Task 12: Design and Implement UI for FavoritesActivity

Favorites Activity will show user's favorite snekears.

- Build layout for FavoritesActivity
- Build layout user's avatar and info.

Task 13: Implement FavoritesActivity's ViewModel

A ViewModel will be responsible for providing favorite sneakers data. For getting favorite sneakers data:

- Build ViewModel that provides favorite sneakers data
- Make FavoritesActivity work with the VIewModel
- Plan for handling sneakers data returned by the ViewModel

Task 14: Implement and wire RecyclerView, Adapter and LayoutManager

Users will be able to scroll through their favorite collection presented on a grid. To support that:

- Build Adapter that will provide favorite sneakers items
- Create a LayoutManager that should arrange provided items in a grid.
- Make RecyclerView work with Adapter and the LayoutManager

Task 15: Provide remove capability for the favorite collection

Users will be able to remove sneakers from their favorite collection.

APP WIDGET

Task 16: Build app widget

App will show the latest user favorited sneakers.

- On widget click go to sneakers detail view.

OTHER ASPECTS

Task 17: Create app flavors

App will have free and paid flavors.

- Configure app flavors
- Configure keystore for signing app versions

Task 18: Implement ads on the free version

App will have free and paid flavors. Ads will be shown on the free flavor:

- Build layout that will show the ad

Implementation aspects

- App will keep all strings in the strings.xml
- The app includes support for accessibility like content descriptions
- Will implement Worker Manager as an alternative to JobDispatcher for future syncing with backend