Mobile Apps 2024

Assignment 2

Due: 11 April 2024 23:59

Gabriel BARAZA (EPITA Group)

Student no. 74522

Plan :

I - Introduction:

The purpose of this project is to come up with a contemporary mobile application which fits perfectly into the current expectations of smartphone users. The app has been developed in such a way that it provides easier navigation and practical features that address users' everyday needs on managing their products and transactions. For this reason, Kotlin and Android Studio were used in order to ensure effective user interface and robust database solutions integration for improved user's experience. This paper, therefore, outlines the methodologies employed in developing the application, the challenges faced as well as solved resulting into a functional and efficient product. Parts of this report were written in French and translated into English thanks to Deepl/ChatGpt.

Requirements Checklist:

1. Authentication

    - [X] Allow User to Signup

    - [X] Log In using email and password

    - [X] Store userID once logged in to keep the user logged in (even after restarting the app)

2. Product Listing

    - [X] List Product Categories

    - [X] On clicking a Category, list Products in that Category

    - [X] On clicking a Product, show Product description, show buy button and controls to change quantity

3. Cart

    - [X] Show cart summary

    - [X] Show total amount

- [X] Purchase button to place an order, show order notification

4. Show order history

    - [X] List users orders

    - [X] On clicking an Order, show Order details and Products ordered

    - [ ] On clicking a Product, take them to Product description page created for 2.3

5. Show User details

    - [X] Use the stored userID to show user details

    - [ ] Show a random circular profile image

    - [X] Show Logout button, on click take back to Signup / Log In page (Restart should not auto login after logout)

6. UI/Implementational Requirements

    - [X] Lazy lists to be used for all Lists: Categories, Products, Orders

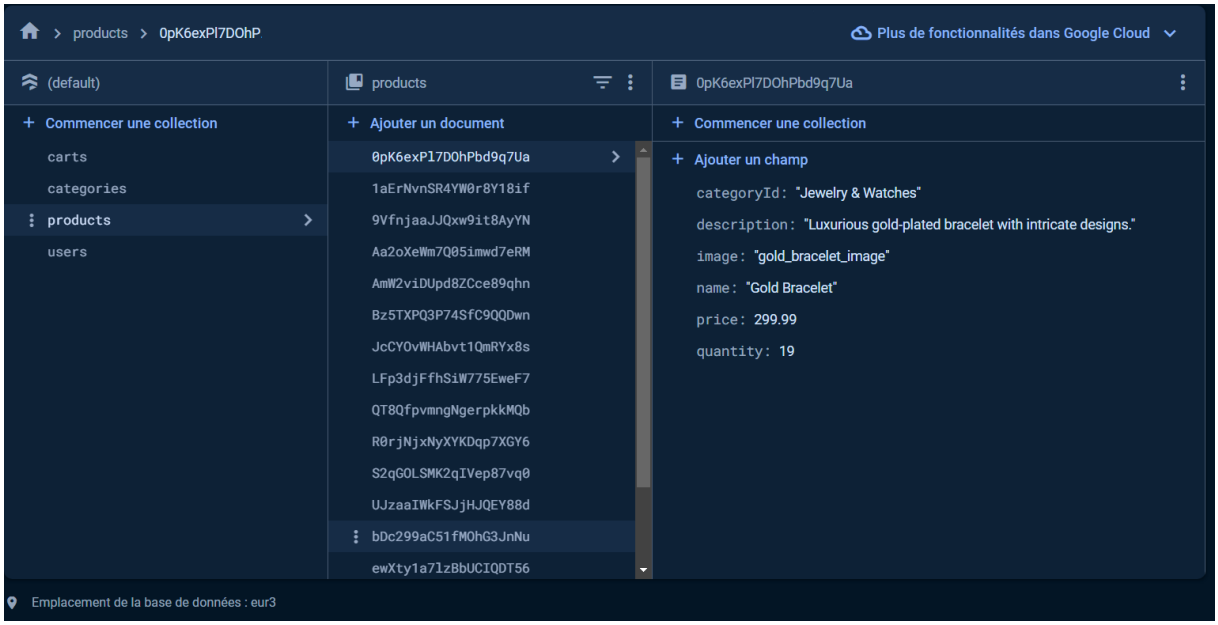
    - [ ] If logged in, attach authentication token to all requests until logout

    - [X] Add a small "About this app" button in the profile page, that shows a page on click with your copyright details and credits

II - Requirements Analysis :

The analysis of the project requirements highlighted the need for a robust solution for user management and data storage. Firebase was identified as an ideal platform due to its comprehensive set of features tailored for mobile application development. Opting for Firebase allows for straightforward user authentication processes that support various login methods, which is crucial for enhancing user accessibility and security.

Additionally, Firebase Firestore was selected for data storage due to its real-time capabilities and ease of integration with mobile applications. Firestore provides a flexible, scalable database solution for efficiently storing and syncing data across user interactions in real time.



Here is the main structure of the database, you can directly access to it if more information is required.(I was too late to use the image field and put images in my project.)

III) Design :

In the implementation phase of this mobile application project, establishing a coherent and user-friendly navigation structure was crucial for ensuring a smooth user experience. Utilizing Jetpack Compose and the modern tools provided by AndroidX Navigation, I set out to establish a clear path for users to navigate through the various features and screens of the application seamlessly.Here is the navigation :

Login Screen (LoginScreen):

→ SignUp Screen

→ Screen 0

SignUp Screen (SignUpScreen):

→ Login Screen

→ Screen 0

Screen0 :

→ Product Categories Screen (Named MainScreen in the project)

→ User Information Screen

→ Cart Screen

Product Categories Screen (ProductCategoriesScreen):

→ Screen 0

→ Products Screen

→ Cart Screen

Products Screen (ProductsScreen):

→ Product Categories Screen

→ Screen 0.

→ Product Details Screen

→ Cart Screen

Product Details Screen (ProductDetailsScreen):

→ Products Screen

Cart Screen (CartScreen):

→ Screen 0

→ User Historic Screen

→ Product Categories Screen

User Historic Screen (UserHistoricScreen):

→ Cart Screen

User Information Screen (UserInformationScreen):

→ Screen 0

→ About Screen

→ Product Categories Screen

→ Cart Screen

About Screen (AboutScreen):

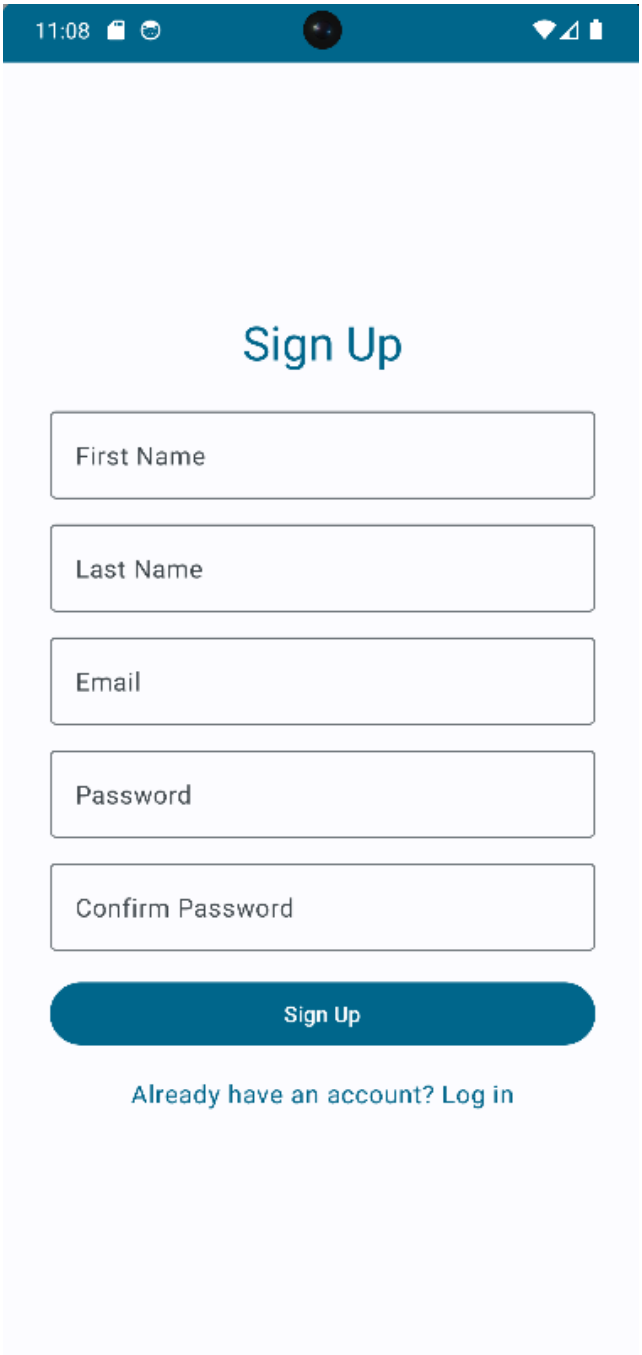
→ User Information Screen

This navigation structure, implemented using NavHost in Jetpack Compose, organizes the application's routes in a clear and logical manner. By defining each screen transition within the NavHost, I ensure that users can navigate through the application intuitively, enhancing user satisfaction and engagement.

The application's appearance has remained rather simple, due to the fact that the focus has been on the application's efficiency.

Part IV – Development

I'm going to talk about the most important screen developments, or at least the ones that seem to get regular use in the application.



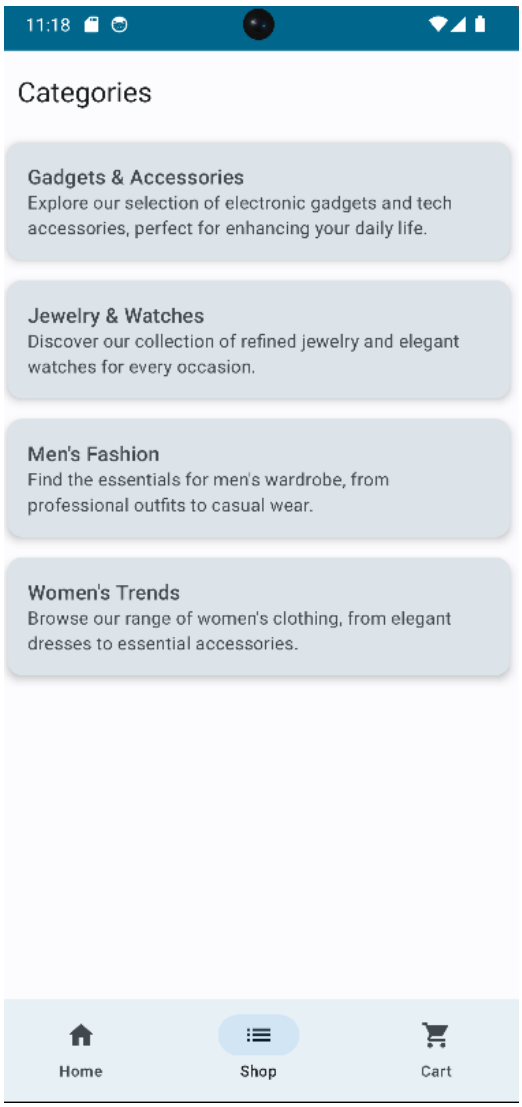
SignUp Screen

Functionality: This screen is designed for new user registration, enabling users to sign up with their email and password, along with optional fields for additional personal details such as name and phone number.

UI Elements: It features input fields for the necessary registration information, mirroring the straightforward design of the LoginScreen but includes additional fields to accommodate the registration requirements.

Data Handling: The information entered by users is securely stored in Firestore under a dedicated user collection. Other user fields can be filled in elsewhere in the application.

Product Categories Screen

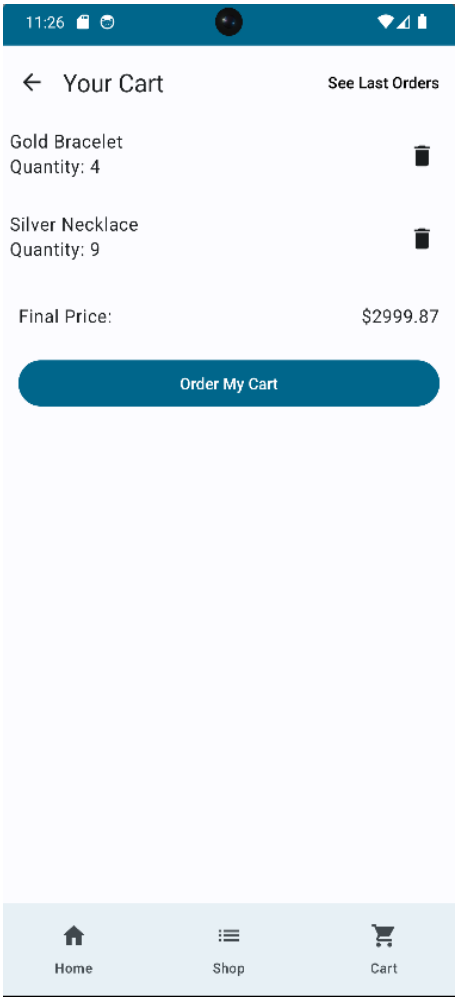


Functionality: Allows users to navigate through various product categories, making it easier to find products they are interested in.

UI Elements: Lazy List, Categories are displayed using a lazy list, which loads elements as they scroll into view, improving performance and user experience.

Data Handling: Category data is fetched in real-time from Firestore, ensuring users see the most current information.

Cart Screen



Functionality :

The Cart Screen displays all items a user has added to their cart, allowing them to adjust quantities, remove items, or proceed to checkout.

UI Elements:

This screen implements a lazy list to dynamically display cart items. Each cart item is associated with interactive elements that allow users to remove items directly from their cart.

Data Handling :

Retrieving Product Details: On the Cart Screen, each item's product name from the CartItem structure is used to query the Firestore database. This involves fetching the corresponding product details where the name field in the database matches the product name in the cart.

Stock Adjustment: When a user decides to purchase items from the cart, the system dynamically adjusts the available stock in the database. This is done by querying the product by its name, retrieving the current stock level, and then decreasing it by the quantity specified in the user's cart.

V – Conclusion :

In conclusion, this project was very comprehensive and enabled us to master new aspects of mobile application development. Indeed, this project linked the creation of a UI, the mastery of application navigation and the use of Firebase. It gave me the opportunity to practice these concepts. Calls and requests to Firebase were at times complicated to

use or apply, but overall the result was consistent with what was required and I was able to achieve a rendering that I liked.