

DecorIn: An innovative way of designing homes

Pragyan Jaiminy

Pushpak Prateek

Manish Saini

Kumar Pratyay

L Ashwin

19/11/EC/028

19/11/EC/047

19/11/EC/039

19/11/EC/063

19/11/EC/035

Abstract: Many interior designing apps have been made after the advent of AR based technology. Here, in this paper we would discuss how to build a AR app and image rendering process. The app is built using Flutter framework along with the use of OpenGL for image rendering. The AR based process was implemented using the ARCore, developed by Google. The final app has a pre-loaded list of various home décor products like Wall lamp, Table lamp, Sofas, Table and much more. The app is an android-based app and would run on the phones supporting ARCore [1].

I. Introduction

Interior designing apps are getting hugely popular these days. This is helping the end-users to have greater control into what updates they can get into their rooms, without actually making those changes. This enhances the experience of the customers and saves the unnecessary expenses which they face due to wrong judgement of the object to be placed at a particular place. At the same time, developing android app is a popular way of testing one's development knowledge. Owing to this, the app helped us understand how **Augmented Reality** works in real life. The app is written in **Flutter** framework on **Dart Programming Language** and **ARCore SDK** which helps to implement the AR experience in the app. The app building

process can be divided into parts: **1.** Making the Blueprints for the app including deciding the interface and features of the app. **2.** Implementing and interconnecting each component and feature. **3.** Enhanced the user experience and added authentication feature for personalized feel.

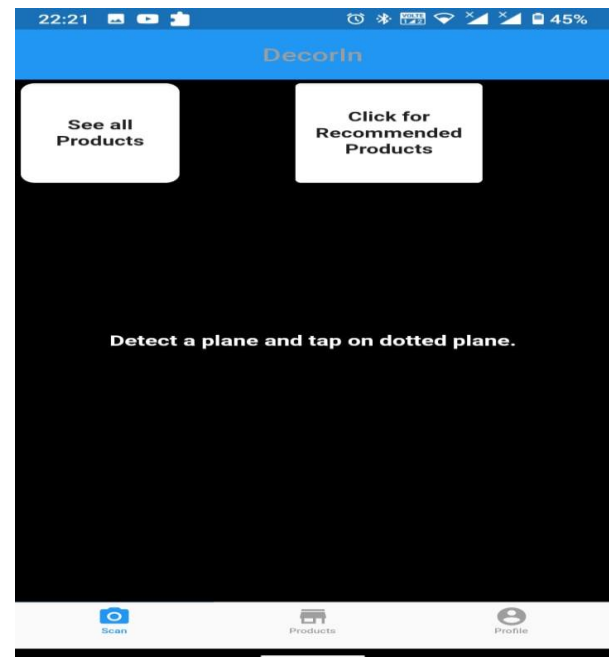


Fig.-1: Start-up screen of the app

II. Discussion

A significant amount of time was spent into deciding what all and how features must be added into the app. For each feature, we had to keep into mind would that enhance the user experience as this is intended to self-service app. We had imported some of the elements of the app from various open-source projects

available. The **ARCore APIs** helped in improving the image rendering on the app. Most image which will be rendered on the app were taken from the web. The authentication feature of the app was developed using **Firebase**.

A humongous task for us was to accurately judge on how the products will be rendered and displayed to the users. The positioning of the products or rather objects must be accurate so as to ensure that no discrepancy remains. A lot of objects were difficult to position as their size is quite big and it makes quite complex to render the same.

Majority of the time went into designing and creating the blueprints of the app. The entire process had to redone several times before reaching the final built design. It was also necessary to regularly take feedback from general audience so that the app may not deviate from the target i.e., **User friendly Interior Designing app**.

III. Methodology

There are many approaches to developing an Augmented Reality Application. We decided on the basic approach any beginner would use. We had used 'Figma' for the UI design of the app. The app was built using **Dart based Flutter framework**. Further, to enhance the personalization of the app, we have added authentication factor in which user can register oneself and save some of the product images which he/she has rendered on the app. This help to save the user's time and efforts to do the same things again and again.

Normal Maps were used as they use the RGB information that corresponds directly with the X, Y and Z axis in 3D space. This RGB information tells the 3D application the exact direction of the surface normal are oriented in for each and every polygon.

Used for texture mapping. Entities have no appearance, functionality or behaviour and it will not render anything until you attach a component with it.

1. Making Blueprints:

Each feature including the products list, recommendation feature, login feature and other minor features were individually developed. Several APIs of the **OpenGL** library were tested in order to get the best out of the app. Several UIs were rejected for being too simple or too unesthetic. Many apps were studied and researched in order to develop the final package. The delete feature of the app wherein you can remove one product once you have rendered was added later when we realized that every time you would want to select a new product you would have to close and reopen the app in order to view them.

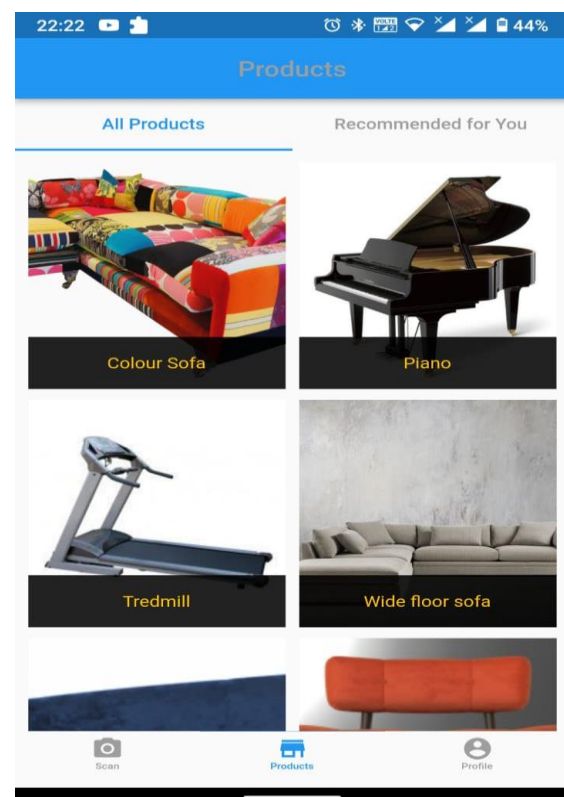


Fig.-2: Displaying the list of product and option to recommend products

A separate icon for the list of products was added so that the users won't have difficulty in locating the desired products. Inside the same products icon, we had added feature so as to recommend the developers of the more products which the users want to see. The added product would have the name of the person who has recommended the product just as a token of respect for the contributor.

2. Implementing and interconnecting features:

After individual classes of features were made, the major uphill task was to implement and stitch them together so as to make them a better experience for the user. Several hours of debugging took place in order to make the app look glitch free and engaging for the user.

3. Enhancing User Experience and Adding Authentication feature:

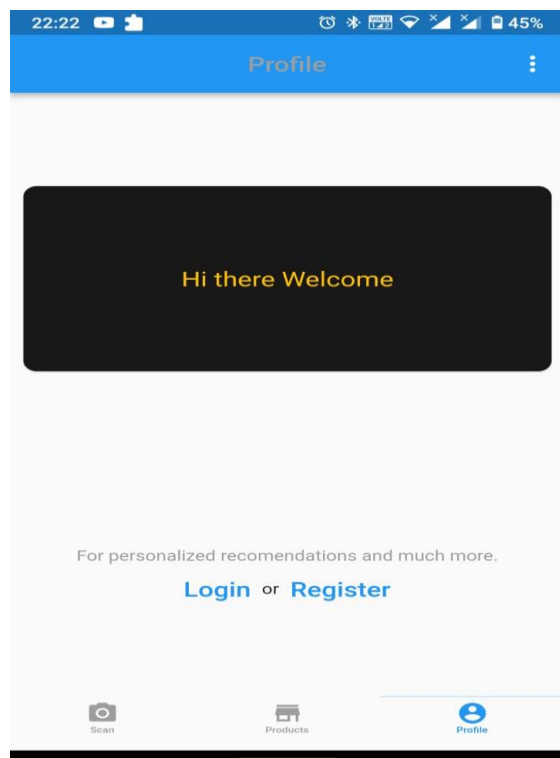


Fig.-3: Authentication or Login Screen

Post connecting and adding several features in the app. The next step towards personalizing the app was adding authentication. A login step was added for users who want to save their stuff and keep records of their products rendered. It was kept in mind that the everyone can still access the app without being forced to login in the app.

IV. Results

The final release of the app has 9 pre-loaded products which would help the user into getting a n overview of the app. Further, there's option to recommend some products to the developers which the users believe should be there.

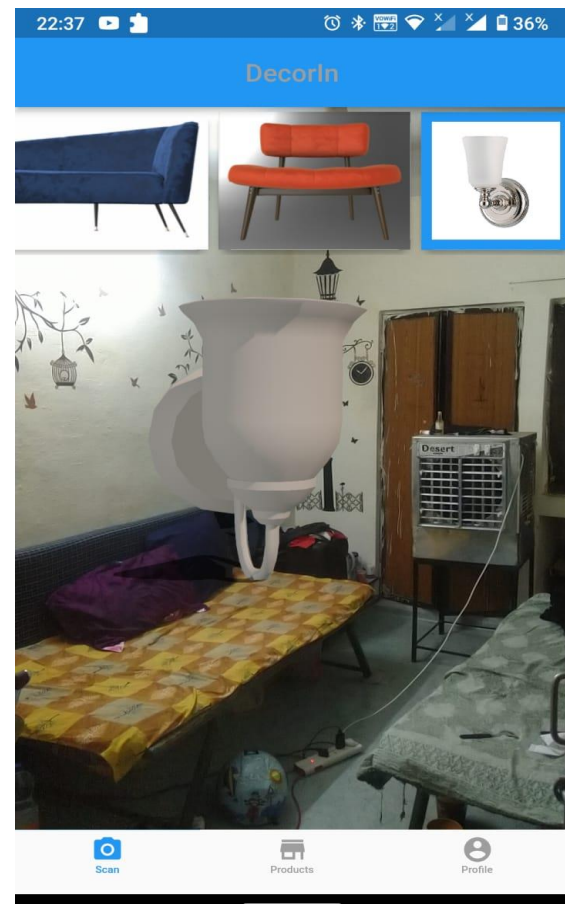


Fig.-4: Render of the Wall Lamp

The file size of the source code is approx. 632 Mb. The size of the APK file for the app is 76 Mb.

V. Conclusion

The final package and release of the app has been done for android-based phones. An APK for the same has been released can be downloaded from the repository. The app is not heavy and can be run under simpler configurations provided that the phone supports ARCore requirements which include the *Accelerometer, Gyroscope and Camera*.

VI. Further Work

The final app released has several bugs like delayed design loading, distorted image rendering at times, errors when multiple logins take place. These issues need to be fixed in order to improve the app's user experience.

The latency which occurs several times need to be reduced so as to accurately predict the objects' positioning. Also, the app can be scaled to load more than one objects at the same time. Better UI and more features can make the app go online for the masses to try. The source code of the app can be used as boiler template for general public to enhance the functionality of the app and make it more-simpler to

use. Also, a *voice command* feature can be added to allow *visually-impaired people* to use this app.

VII. Acknowledgement

We express our sincere gratitude and vote of thanks to our Professor- Dr. Prerana Mukherjee for her constant support and insightful guidance during the duration of the project.

VIII. References

1. <https://www.xda-developers.com/arcore/#:~:text=This%20means%20it%20only%20relies,to%20do%20what%20it%20does.>
2. <https://developers.google.com/ar/devices>
3. <https://learnopengl.com/Advanced-Lighting/Normal-Mapping>
4. <https://www.figma.com/ui-design-tool/>
5. https://en.wikipedia.org/wiki/Augmented_reality
6. <https://firebase.google.com/docs>
7. <https://docs.flutter.dev/>
8. <https://dart.dev/guides>
9. <https://foyr.com/learn/interior-design-basic-concepts-and-principles/>