

UKRAINIAN CATHOLIC UNIVERSITY

BACHELOR THESIS

Flutter app for picking and ordering tiles for interior design using AR

Author:
Luka TRISKA

Supervisor:
Serj Miskiv

*A thesis submitted in fulfillment of the requirements
for the degree of Bachelor of Science*

in the

Department of Computer Sciences
Faculty of Applied Sciences



APPLIED
SCIENCES
FACULTY ●

Lviv 2020

Declaration of Authorship

I, Luka TRISKA, declare that this thesis titled, "Flutter app for picking and ordering tiles for interior design using AR" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

“And I knew exactly what to do. But in a much more real sense, I had no idea what to do.”

Michael Scott

UKRAINIAN CATHOLIC UNIVERSITY

Faculty of Applied Sciences

Bachelor of Science

Flutter app for picking and ordering tiles for interior design using AR

by Luka TRISKA

Abstract

Now more than ever, it is a huge advantage for businesses that specialise in interior design, to be able to showcase their products to customers at a tap of button. With great advancements in the field of Augmented Reality, you can now model and style your apartment using only your phone. This was our biggest motivation while working on this application ([link for the demo video](#)) – to be able to make life easier for our clients.

Acknowledgements

I would like to thank my family, the university teachers, my supervisor, and all of my friends for helping me along this way.

Contents

Declaration of Authorship	i
Abstract	iii
Acknowledgements	iv
1 Introduction	1
1.1 What is Flutter?	1
1.2 Why Flutter	1
2 Existing Solutions	3
2.1 Housecraft	3
2.2 Planner 5d	3
3 The Proposed Solution	5
3.1 Flutter	5
3.2 User Flow	6
3.3 REST API Integration	6
3.4 AR	7
3.5 Choosing different tile ornaments	9
3.6 Cart	10
3.7 Checkout	11
4 Conclusion	12
Bibliography	13

Chapter 1

Introduction

In the field of interior design, it is becoming more and more popular to be able to use AR technology to see what a certain detail would look like in real life. This was my main motivation for developing this application, for the users and potential customers of Leoti to be able to see the way our tiles look in their home, before even making an order. The application also makes it much easier for the buyer to make an order for the chosen tiles, and for us to be able to respond to that order.

1.1 What is Flutter?

Flutter is a free and open source mobile UI framework created by Google in 2017. It provides a fast and expressive way for developers to build native apps on both iOS and Android with only one codebase. One of the handy and modern features of Flutter is Hot Reload – the ability to update the application in a matter of seconds on a simulator or a real device.

Flutter consists of two main parts:

- A Software Development Kit (SDK): A collection of tools that are going to help you develop your applications (e.g. tools to compile your code into native machine code – for iOS and Android).
- A Framework (UI Library based on widgets): A collection of reusable UI elements (buttons, text inputs, sliders, and so on) that you can personalize for your own needs.

To develop on Flutter, you have to use Dart – a typed object programming language, where everything is a widget (akin components in React). It was created by Google in 2011, and has been improved a lot over these past years.

1.2 Why Flutter

Here are a few main reasons why we chose to use Flutter for this application:

- **Fast Development** – It is very easy to see the smallest changes in the design of the app and workflow using the Hot Reload feature.
- **Expressive and Flexible UI** – Flutter provides layered architecture, which results in incredibly fast rendering and expressive design. The various pre-built widgets make it a lot easier to design the app.
- **Similar to Native App Performance** – since Flutter does not rely on any intermediate code interpretation, and is built directly into the machine code, any performance bugs of the build process are eliminated.

- **Own Rendering Machine** – Flutter uses Skia[*Skia Engine*] for rendering itself onto a platform-provided canvas. Because of this, UI built in Flutter can be launched on virtually any platform.
- **A Vast Library of Plugins** – easy access to advanced OS-level features, such as fetching GPS coordinates, Bluetooth connection, interacting with default OS apps etc. In case a plugin has not yet been developed for a specific feature, you can establish communication between platform-native code and Dart through **platform channels**.

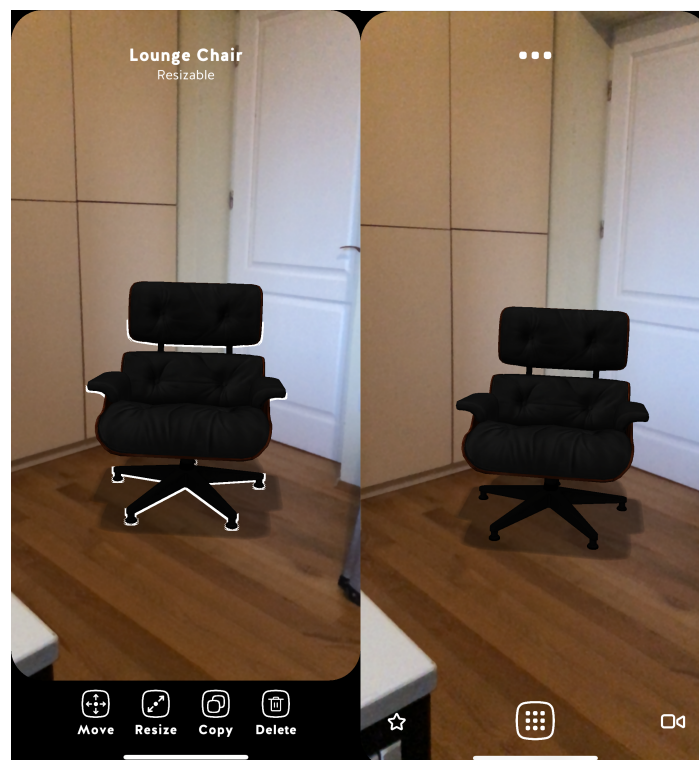
Chapter 2

Existing Solutions

Applications that were the closest to my goal were Housecraft [*Housecraft App*] and Planner 5D [*Planner 5D App*] on the AppStore.

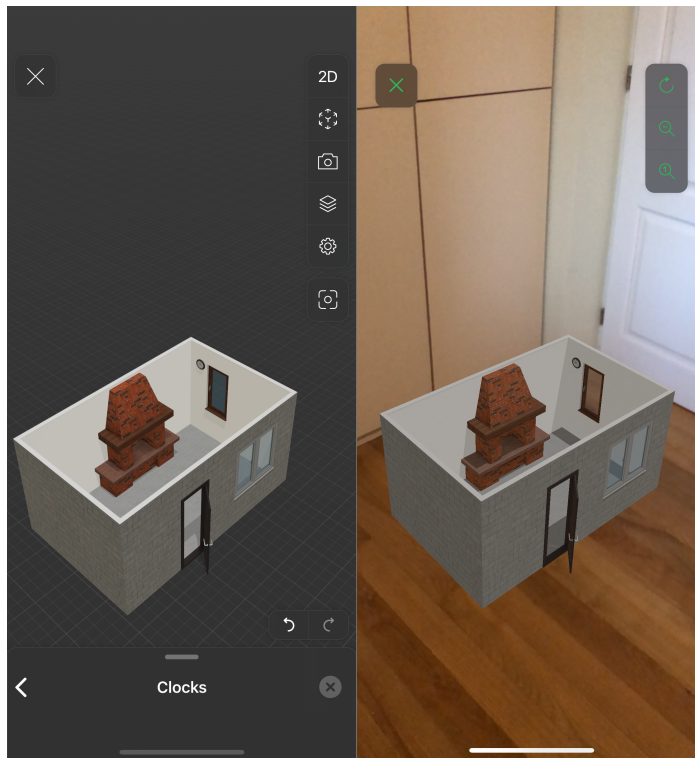
2.1 Housecraft

This app is rather minimalist and easy to use. You have to scan the floor, and then choose from a dozen different objects to place on it. Afterwards, you can rotate, resize and delete objects.



2.2 Planner 5d

Planner 5d is a much more complex application. Users can map out an entire building, with various different rooms and also the outside territory. Then you can place various items in the building – from windows, arches, doors and stairs to furniture, electrical appliances, and exterior elements (trees, plants, garages, pools etc).



However, none of them allowed for projecting different flooring material, such as tiles.

Chapter 3

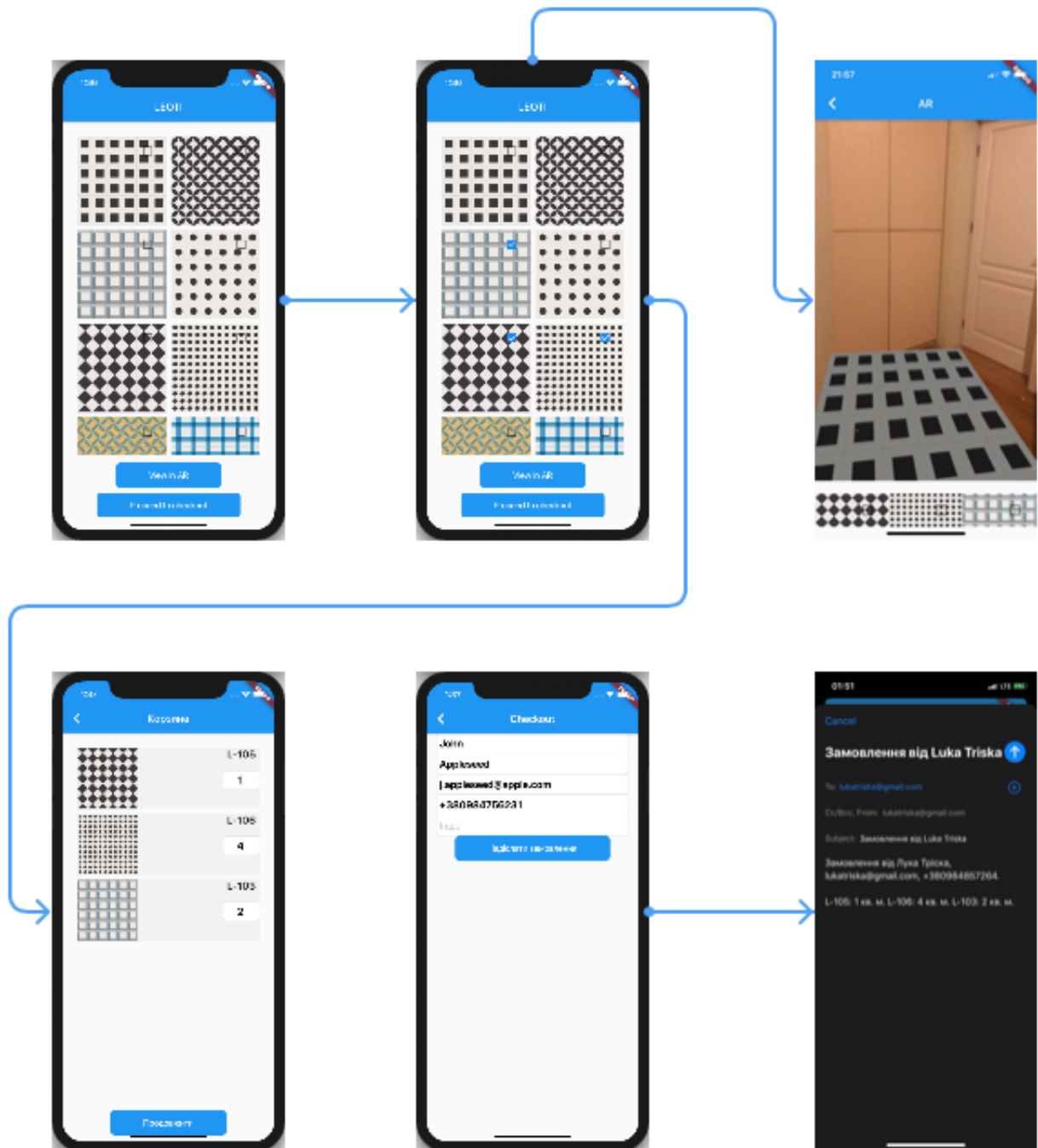
The Proposed Solution

3.1 Flutter

The reason why I chose Flutter to develop this application was that using this technology, it is very straight forward to develop a multi-platform program, writing the same code. Sure, some parts of the code had to be fitted to the specific platform (iOS or Android), but in general, the app turned out to be rather robust. Here are some of the Flutter plugins I used while developing this app:

- **flutter_email_sender**[*Flutter Email Sender Plugin*]-allows to send emails from flutter app using native platform functionality. Uses the built-in Gmail app for Android, and the Mail app on iOS using MFMailComposeViewController.
- **arkit_flutter_plugin**[*Flutter Arkit Plugin*] – allows for straightforward integration of ARKit for iOS into the Flutter app.
- **arcore_flutter_plugin**[*Flutter Arcore Plugin*] – built on top of the above-mentioned plugin, for integrating ARCore into Flutter.

3.2 User Flow

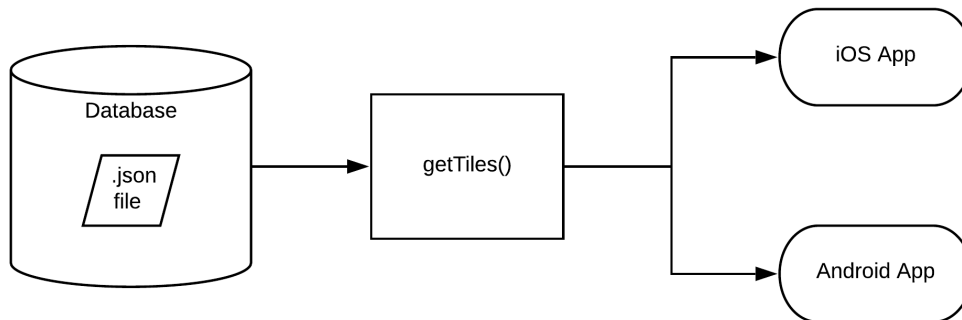


3.3 REST API Integration

When building an application that involves different data, that might be changed in the future, it is a good solution to create the database where you can create, update and delete the data. To implement this, we decided to use REST API.

We run a simple http server from the directory where data (specifically – tile ids, names and images) is stored on our computer. Later on this server will be running on one of the cloud computing machines. Whenever the app is launched, we use

an Http GET method implemented in the `getTiles()` function in the `main.dart` file to fetch the data from the server, and store it in a local variable. From then on the data is sent accordingly to the corresponding widgets. Here's a simple diagram of the system.



3.4 AR

For implementing AR, I used the ARKit and ARCore plugins for flutter. Those are community developed plugins, that allow for seamless integration of AR platforms for both Android and iOS into a Flutter app.

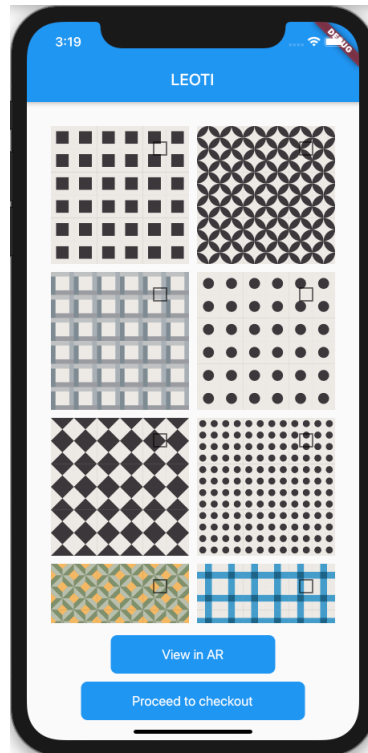
Here is how the AR Screen looks like:



The program automatically identifies any horizontal plane, and paints the chosen tile image on it. On the bottom of the AR screen users can choose which tile of the ones they chose they would like to see projects onto the floor.

For the time being, all orders will be processed manually once they are sent by the users. Since this is a niche business, there is no need to scale it for now.

3.5 Choosing different tile ornaments

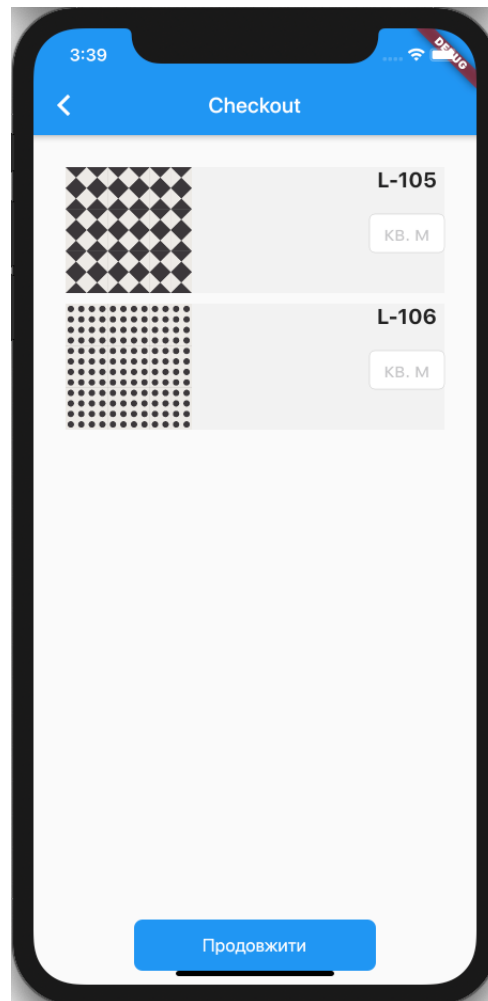


This is the first screen that users see when they launch the app, and we decided to make it rather straightforward. In order to achieve this grid layout, we used the following code snippet.

```
Widget gridViewSelection = GridView.count(  
  childAspectRatio: 1.0,  
  crossAxisCount: 2,  
  mainAxisSpacing: 10,  
  crossAxisSpacing: 10,  
  padding: EdgeInsets.symmetric(horizontal: 30, vertical: 30),  
  children: loftTiles.map(...)) // here we map out the tiles
```

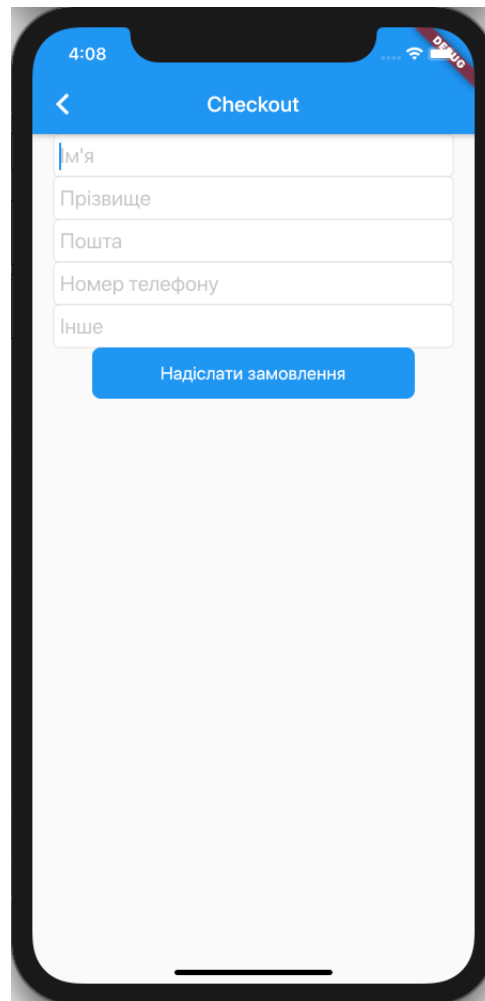
Users can choose one or more ornaments that they like, and see them in AR, where the images will spread onto surfaces.

3.6 Cart



In the cart screen users can see the tiles they have chose, choose the amount of square meters for each tile they would need, and proceed to the order confirmation screen. These are just estimates, all of the final details will be sorted out manually after we receive the order and contact the client.

3.7 Checkout



The screenshot shows a mobile application interface for a checkout process. At the top, the status bar displays the time 4:08, signal strength, Wi-Fi, and cellular data (D2000) icons. The app's navigation bar is blue with a white back arrow on the left and the title "Checkout" in the center. Below the navigation bar, there is a form with five input fields: "Ім'я" (Name), "Прізвище" (Surname), "Пошта" (Postcode), "Номер телефону" (Phone number), and "Інше" (Other). A blue button with the text "Надіслати замовлення" (Send order) is positioned below the form fields.

Here users are prompted to fulfil their personal information, which we will later use to contact them and finalise the order.

Chapter 4

Conclusion

In conclusion, we have achieved most of what was planned for the application, even though there still are a few finishing touches left. We will publish the application in the near future, promote it using our social media pages, and see whether or not it proves to be successful.

Bibliography

- Francesco, Gian Marco Di. *Flutter Arcore Plugin*. URL: https://github.com/giandifra/arcore_flutter_plugin.
- Inc, Skia. *Skia Engine*. URL: https://uk.wikipedia.org/wiki/Skia_Graphics_Engine.
- Leuschenko, Oleksandr. *Flutter Arkit Plugin*. URL: https://github.com/olexale/arkit_flutter_plugin.
- LLC, Sirvo. *Housecraft App*. URL: <https://apps.apple.com/us/app/housecraft/id1261483849>.
- Planner5D, UAB. *Planner 5D App*. URL: <https://apps.apple.com/us/app/planner-5d-interior-design/id606173978>.
- Šidlauskas, Tautvydas. *Flutter Email Sender Plugin*. URL: https://github.com/sidlatau/flutter_email_sender.