

ABSTRACT

Our Aim is to design and create a product rental app that enables the user to rent a product with people in their neighbourhood or require a service, so that the product can be used by the requestor. This system could also help in building an amicable relationship among the neighbours and solve the problem of people not owning non-essential things readily.

Table of content

Chapter no	Title	Page no
	ABSTRACT	i
	LIST OF Table	ii
1	Introduction	1
1.1	Outline of the project	1
1.2	proposed system	1
1.3	Scope of the project	2
2	AIM & SCOPE OF THE PROJECT	3
2.2	AIM	3
2.1	Software requirement	3
2.2	Hardware requirement	3
3	EXPERIMENTAL OR MATERIALS AND METHODS	5
3.1	Flutter	5
3.2	Feature of Flutter	5
3.3	Architecture of Flutter Application	6
3.4	Flutter Packages	10
4	RESULTS AND DISCUSSION, PERFORMANCE ANALYSIS	13

4.1	Discussion	13
4.2	Result	14
5	SUMMARY AND CONCLUSIONS	15
5.1	Reference	15
5.2	APPENDIX	15
5.2A	Screenshots	15
5.2B	Source code	20

Chapter 1

Introduction

1.1 Outline of the project

Renting and exchanging the goods is a method of using technology to bring people together" despite the geographical barriers The technology has been available for years but the acceptance of it was quite recent. Our project to make neighbourhoods close and earn money . It is made up of flutter made by Google. When we post the product ,other users can get the notification. If they have the product they can communicate with the user. Some may give the product for free rental for the neighbourhoods or pay service ,it can be chosen by the user. First This app sends the notification to a 6 km radius from the user given location. This app was designed to be very user friendly. This software can have further posence , such as delivering the product ,map location.

1.2 proposed system

Resolving the names of all the systems connected in a network and enlisting them.

- Used for communication between multiple systems enlisted in the resolved list.
- The GUI operates in two forms, the list form & the chat form.
- The List form contains the names of all the systems connected to a network.
- The chat form makes the actual communication possible in the form of text.

Advantage :

- User friendly
- Fast replay
- More clients can use at a time
- Cross platform

Disadvantage :

- Can see post of only near by location
- Some product cannot be rented

1.3 scope of the project:

Where we rent the product from nearby neighbours ,so where they can earn money and make friends with unknown neighbours . By getting the product, the rental person can save money from buying it.

Chapter 2

AIM & SCOPE OF THE PROJECT

2.1 AIM

This is the simple product rental app where the person can rent all simple item with the neighbourhood .So they can save money instead of buying it .By renting the rarely used product they can also earn the money. In this process neighbour can know about ,they can be good friends also.

2.1 Software requirements

Software can be defined as programs which run on our computer. It acts as petrol in the vehicle. It provides the relationship between the human and a computer. It is very important to run software to function on the computer.

Various software are needed in this project for its development. Which are as follows-?

- Operating system- more than Windows 7
- Android studio
- Flutter

Languages : Dart language is used.

2.2 Hardware Requirements

In hardware requirements we require all those components which will provide us the platform for the development of the project.The minimum hardware required for the development of this project as follows-

- Ram minimum 128 MB
- Hard disk -minimum 5 GB

- Processor Pentium 3
- Floppy drive 1,44 inch

These all are the minimum hardware requirements required for our project. We want to make our project to be used in any. Type of computer therefore we have taken minimum configuration to a large extent. 128 MB ram is used so that we can execute our project in the least possible RAMS GA hard disk is used because the project takes less space to be executed or stored. Therefore the minimum hard disk is used. Other enhancements are according to the needs.

Chapter 3

EXPERIMENTAL OR MATERIALS AND METHODS

3.1 FLUTTER

Flutter is an open-source UI software development kit created by Google. It is used to develop cross platform applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase

Flutter is Google's **portable UI toolkit for crafting beautiful**, natively compiled applications for mobile, web, and desktop from a single codebase. Flutter works with existing code, is used by developers and organizations around the world, and is free and open source.

3.2 Features of Flutter

Flutter framework offers the following features to developers:

- Modern and reactive framework.
- Uses Dart programming language and it is very easy to learn.
- Fast development.
- Beautiful and fluid user interfaces.
- Huge widget catalog.
- Runs the same UI for multiple platforms.
- High performance application

Advantages of Flutter

Flutter comes with beautiful and customizable widgets for high performance and outstanding mobile application. It fulfills all the custom needs and requirements. Besides these, Flutter offers many more advantages as mentioned below:

- Dart has a large repository of software packages which lets you extend the

capabilities of your application.

- Developers need to write just a single code base for both applications (both Android and iOS platforms). Flutter may be extended to other platforms as well in the future.
- Flutter needs less testing. Because of its single code base, it is sufficient if we write automated tests once for both the platforms.
- Flutter's simplicity makes it a good candidate for fast development. Its customization capability and extendibility makes it even more powerful.
- With Flutter, developers have full control over the widgets and its layout.
- Flutter offers great developer tools, with amazing hot reload.

Disadvantages of Flutter

Despite its many advantages, flutter has the following drawbacks:

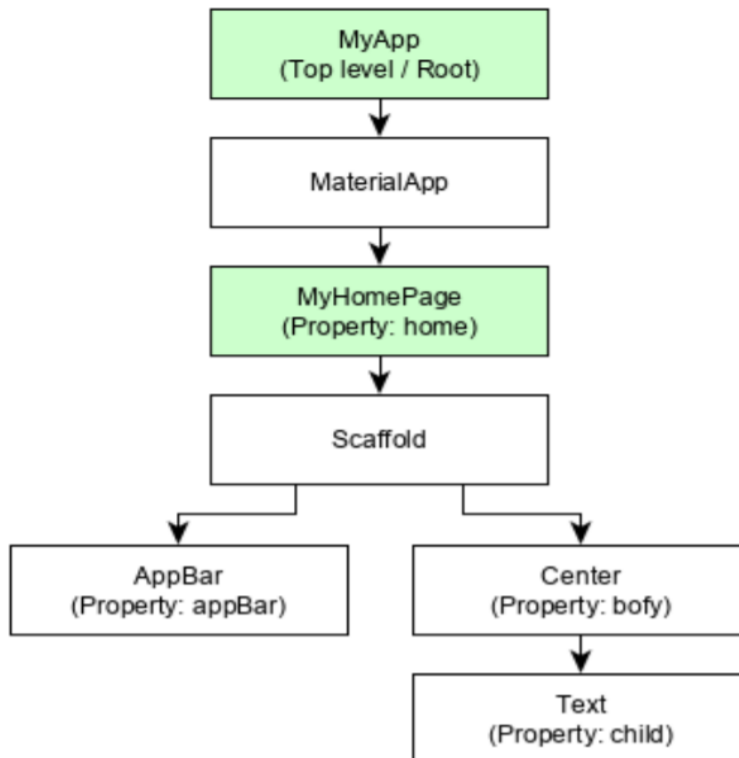
- Since it is coded in Dart language, a developer needs to learn a new language (though it is easy to learn).
- Modern frameworks try to separate logic and UI as much as possible but, in Flutter, user interface and logic is intermixed. We can overcome this using smart coding and using high level modules to separate user interface and logic.
- Flutter is yet another framework to create mobile applications. Developers are having a hard time choosing the right development tools in highly populated segments.

3.3 Flutter – Architecture of Flutter Application

In this chapter, let us discuss the architecture of the Flutter framework.

Widgets

The core concept of the Flutter framework is In Flutter, Everything is a widget. Widgets are basically user interface components used to create the user interface of the application. In Flutter, the application is itself a widget. The application is the top- level widget and its UI is built using one or more children (widgets), which again build using its children widgets. This composability feature helps us to create a user interface of any complexity. For example, the widget hierarchy of the hello world application (created in previous chapter) is as specified in the following diagram:

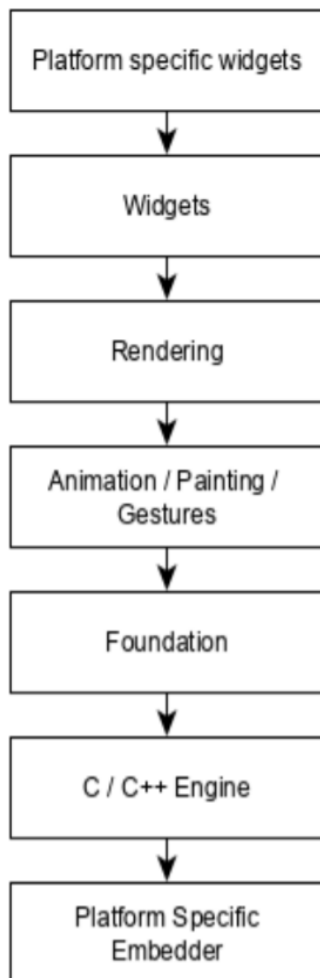


Here the following points are worth notable:

- MyApp is the user created widget and it is build using the Flutter native widget, MaterialApp.
- MaterialApp has a home property to specify the user interface of the home page, which is again a user created widget, MyHomePage.

decreasing complexity. A layer is build using its immediate next level layer. The top most layer is widget specific to Android and iOS. The next layer has all flutter native widgets. The next layer is the Rendering layer, which is a low level renderer component and renders everything in the flutter app. Layers go down to core platform specific code.

The general overview of a layer in Flutter is specified in the below diagram:



The following points summarize the architecture of Flutter:

- In Flutter, everything is a widget and a complex widget is composed of already existing widgets.
- Interactive features can be incorporated whenever necessary using GestureDetector widget.