



iJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 Issue: XII Month of publication: December 2020

DOI: <https://doi.org/10.22214/ijraset.2020.32678>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

A Research Paper on a Pet-Friendly Application using Flutter and Firebase

Yeshwin Anil¹, Archana Sahoo², Powel Shoby³

^{1, 2, 3}B.E. Student, Department of Computer Engineering, Xavier Institute of Engineering, Mahim, Mumbai, Maharashtra, India

Abstract: Every time a pet owner wants to visit a place he has to sit down and check if their pets are allowed over there or not, where to get all the pet products etc. So, the main motive behind our project is to make the life of pet owners hassle-free and provide them with a one stop solution. The idea is to develop a cross platform app (which can be accessed from both Android and iOS). Technologies used are: Flutter and Firebase, and the language used is DART. The various features incorporated are pet friendly places, parks, hotels, products etc. It would be similar to an e-shop app but it would be oriented towards pets. This application can save a lot of time and effort of the users and provide a more efficient way to handle things out.

Keywords: Flutter, Dart, Firebase, Android Studio, Application Development

I. INTRODUCTION

This is an application used for the convenience of pets and pet owners, where it is easier for an individual to check if a particular place is pet friendly i.e., cafes, shops and parks where they are allowed with their pets. The application acts as a standalone software for the pet owner for buying pet food and accessories. Using the application user shall be able to get directions to the pet friendly places and also be able to purchase the basic necessities for their pets. Developer can get information of all shops and places that allow pets within given pin code by searching different websites online. The application can help all the pet owners, since they don't have to manually call and enquire every place they are to visit to check if their pets are allowed with them

The main objective of our project is to develop an application that implements the Options of places where pets are allowed and directions to it, Availability of Necessary pet products like pet food, toys and accessories and to design and develop an easy and friendly User Interface. There does not exist such an application which is specifically deals with the wellbeing and care of pets. This application will be a one stop answer for pet needs.

II. CONCEPTS (THEORY)

A. Flutter[6]

Flutter is platform-agnostic as it has its own widgets and designs, which mean you can have the exact same app on two platforms. Flutter is an open-source UI software development kit created by Google. It is used to develop applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase. Flutter has its very own user interface parts, alongside a motor to deliver them on the iOS and android stages. The majority of those user interface parts, directly out of the container, fit in with the rules of Material Structure. As opposed to using web views or depending on the device's OEM gadgets, Flutter renders each view parts utilizing its own superior rendering motor. This nature gives plausibility to assemble apps that are as superior as local apps can be. Flutter bolsters state full hot reload while creating, which is look as a main consideration to support advancement cycle.

B. Dart[5]

Dart is a client-optimized programming language for apps on multiple platforms. It is developed by Google and is used to build mobile, desktop, server, and web applications. Dart is an object-oriented, class-based, garbage-collected language with C-style syntax. In Flutter all apps are composed with Dart.. It is generally utilized within Google. It has been demonstrated to have the ability to create enormous web apps.

C. Firebase[4]

Firebase is considered as web application platform. It helps developers build high-quality apps. It stores the data in JSON format which doesn't use query for inserting, updating, deleting or adding data to it. It is the backend of a system that is used as a database for storing data. Firebase is a platform developed by Google for creating mobile and web applications. Firebase Storage was designed for application developers who need to store and serve user-generated content, for example photos or any other file. It gives secure document transfers and download for Firebase applications, regardless of network quality.

D. Android Studio

Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. Android Studio is Android's official IDE. Fast and feature-rich Emulator. Compared to a real device, Android Emulator helps to install and start running the application at much higher speed and testing it under different Android configurations. We will be using XCode and MacBook as far as iOS testing are concerned because there are several security norms to be met.

E. Widgets

In Android, the View is the establishment of each item that appears on the screen. toolbars, Buttons and inputs, each item is a View. In Flutter, View is a Widget. To begin, widgets have an alternate life expectancy: they are unchanging and just exist until they should be changed. At whatever widgets and their case change, Flutter's structure makes another plant of widget occasion. Widgets are light weight, to a limited extent because of their immutability. Since they are not seeing themselves and are not straightforwardly drawing anyone, but instead are a depiction of the user interface and it is semantics that fetch "expanded" into genuine view questions in the engine. Widgets are the fundamental structure squares of a Flutter app user interface. Every widget is a changeless affirmation of section of the UI. Dissimilar to different structures or local stage apparatuses that different perspectives, see controllers, designs, and different properties, Flutter has a steady, brought together article model. Widgets are the components that effect and especially the view, sense of utilization and its appearance.

III. REVIEW OF LITERATURE

A. *Thesis by Wenhao Wu from Metropolia University of Applied Sciences on React Native vs Flutter, Cross platform mobile application frameworks, March 18.^[1]*

1) *Description:* In React Native after every OS UI update, the app components will be instantly upgraded, as well. That said – this can break the app's UI, also applications written in React Native must be able to run JavaScript code. Android doesn't have this functionality by default. Whereas Flutter is cross platform and can build apps for multiple platforms using a single codebase. Not every aspect of the platforms is discussed in detail. For instance, render process, which both React Native and Flutter have put great effort into optimization, has not been deliberated. Furthermore, comparisons between native and cross-platform applications are not mentioned exhaustively.

2) *Installation:* React Native: Can be installed using Node Package Manager (NPM) while using macOS you need to have the HomeBrew package manager as well. So for developers having NPM installed the process is really easy whereas other developers would need to learn the ins and outs of NPM. • Flutter: Supports binary download straight from GitHub for a particular platform. Also, in the case of macOS after downloading the file needs to be added as a PATH variable.

3) *UI Component and Development API*

a) *React Native:* Relies heavily on third-party libraries to access most of the native modules.

b) *Flutter:* Rich set of components comprising of each native module required which excludes the necessity of a third-party library. Also, Flutter is full of WIDGETS.

4) *Testing Support*

a) *React Native:* Again, heavily relies on third-party tools for integration and UI testing.

b) *Flutter:* Flutter has its own features to test apps at 3 levels which are unit test, widget test and integration test.

B. *Thesis by Chunnu Khawas and Pritam Shah from Department of Computer Applications Sikkim University Gangtok, India on Application of Firebase in Android App Development, June 18^[2]*

1) *Description:* Firebase is Google's platform for mobile and web development The platform offers a set of tools that make the product development process a lot easier. That's why it's suitable for building an MVP as it shortens the time and effort put in by developers to actually create it. In this paper various pros and cons of firebase has been discussed. Firebase is NoSQL based.

2) Pros

- a) **Database Options:** Firestore and Realtime are database options one can choose to build their product. They both are cloud-hosted, NoSQL databases that are flexible and scalable in terms of size.
- b) **Free for Beginners:** The platform doesn't charge for most of its services and requires choosing a pricing plan only after reaching a certain amount of database memory.
- c) **Community:** Firebase has good community support.
- d) **Ease of Integration:** The ease of integration and quick setup. The pre-made APIs that the platform offers enable you to add new functionalities within just a few clicks.

3) Cons

- a) **Data Storage is Inconvenient:** The main problem is limited querying capabilities. You can't query for more than one key at a time and the service doesn't provide a way to filter your data. It's because the whole DB is JSON file and it's nothing near the storage format of SQL. The format also excludes the option to model the data.
- b) **Limited Data Migration:** You don't host the data; all data is hosted on Firebase. Unless Firebase provides a migration tool to enable easy transfer of user's data, it strongly limits data migration.
- c) **Android Centered:** Firebase doesn't provide the same capabilities for Android and iOS apps. Its still more Android centered and it's the one system that gets most of dedicated services and facilities

It is difficult for Relational Database Management System (RDBMS) to handle the unstructured data. Firebase is a relatively new technology for handling large amount of unstructured data. It is very fast as compared to RDBMS. This paper highlights on the study about Google provided Firebase API and its unique features. This paper helps in studying how to use Firebase in the Android application according to the developer requirement. This also helps in making android apps faster and efficient as no PHP is required as a third-party language to communicate with the database. The study material is based on the data provided online and referring to the examples given. Google has been updating Firebase on regular basis, AdSense is the beta phase of Firebase. It can not only be used in Android but also to connect cross platform. The work can be further extended by adding new features and exploring new possibilities

IV. WORK PLAN

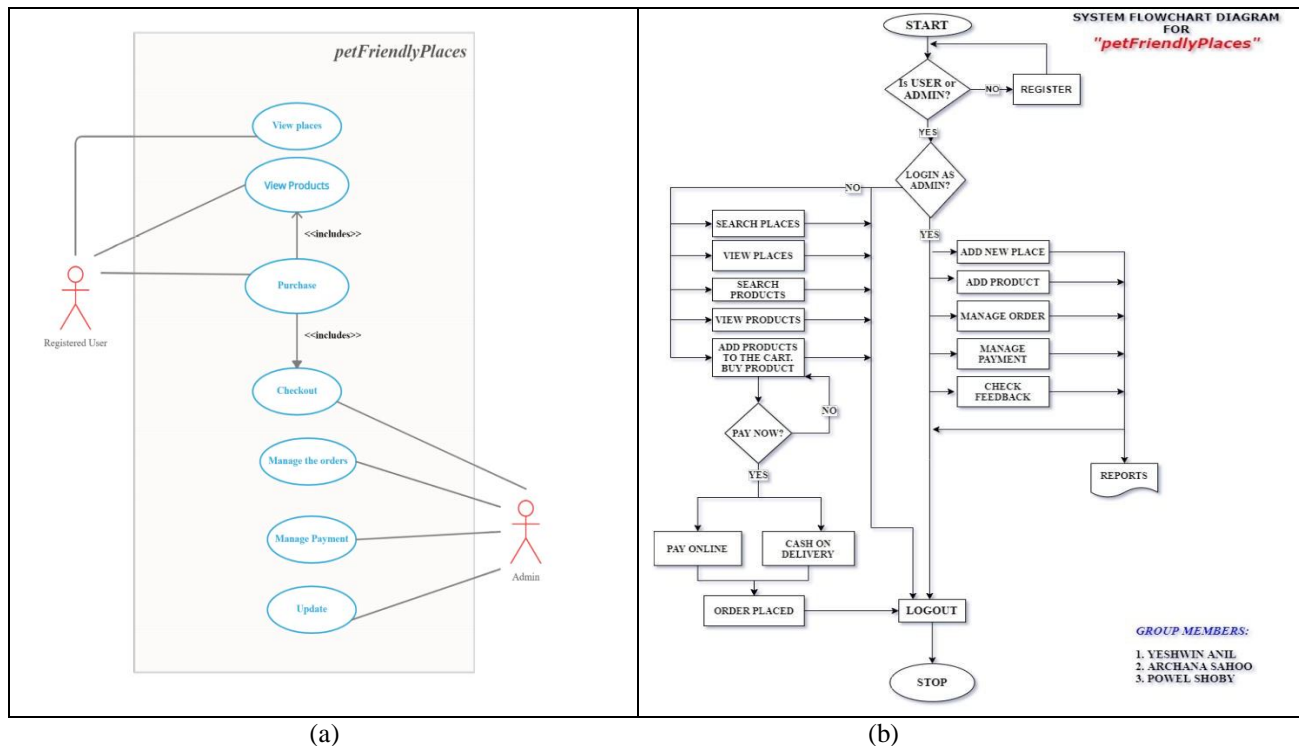


Fig. 1 (a) USE CASE DIAGRAM, (b) FLOW DIAGRAM

- A. As seen in the flowchart, on starting the application, the user needs to register and navigate through the constructed authentication process.
- B. After the authentication process is done with, the application will check whether the person has logged in as an admin or a user. Depending upon this, the functionalities offered to the person will differ which can be shown in the use case diagram.
- C. If the person logs in as an admin, they can do the following functions: add new places and outlets, add new products, manage orders received, oversee payments made, assess reviews and feedbacks.
- D. Logging in as a user will allow you to search different places and find out whether pets are allowed there, explore different pet products and their availability, add those products to your cart and buying them.
- E. We plan on integrating two modes of payment. Payment of the selected product can be done either by online methods or the user can opt for Cash-On-Delivery (COD). After payment, the order gets placed.
- F. If the user's needs are met, user may logout.

V. IDEA AND MOTIVATION

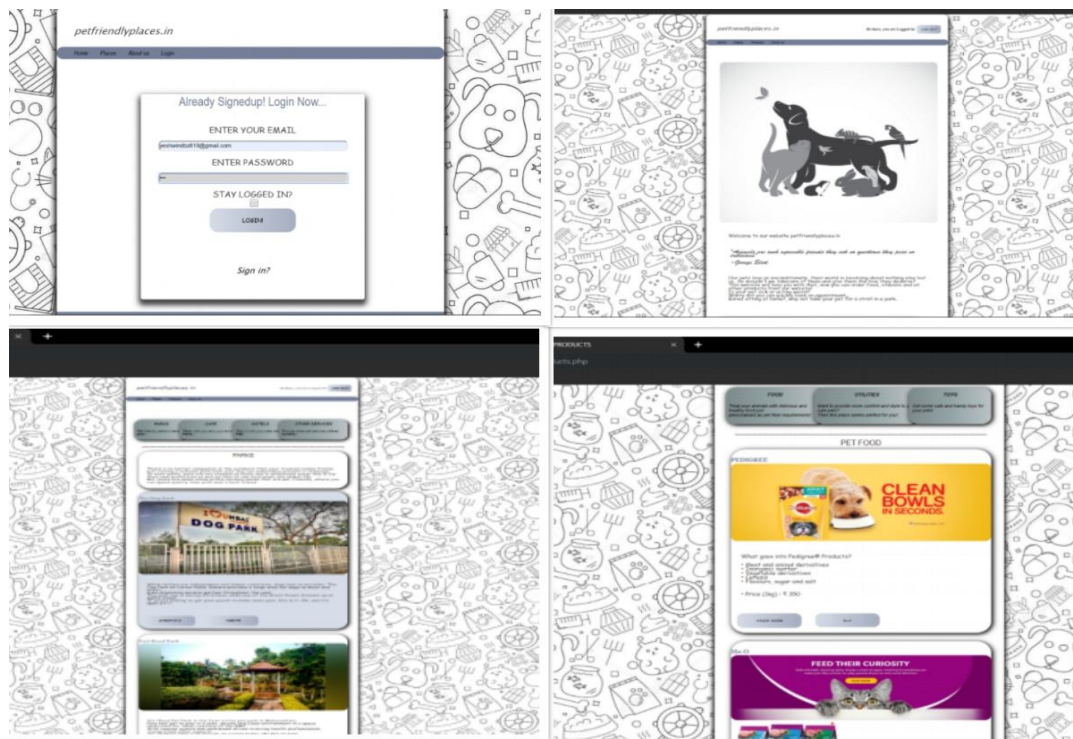


Fig. 2 Snippets of our website

These are some snippets from our previously built website. That was the foundation of this idea. After making a website we thought of expanding this concept to a full fledged application that would run on both android and iOS.

In this project, we will be building up a stage free application by utilizing just one code base which will run easily on both android and iOS. Flutter, Dart and Firebase will help us achieve this initiative

A. Flutter Framework^[6]

As everything is as widgets, so the changed viewpoint proportions and screen sizes isn't an issue. The advantages of picking Flutter are:

- 1) Same UI and Business Logic in All Platforms
- 2) Reduced Code Development Time
- 3) Custom, Animated UI of Any Complexity Available
- 4) Own Rendering Engine
- 5) The Potential Ability to Go Beyond Mobile

B. Dart^[5]

Dart (since this is the programming language utilized with Flutter Framework). Dart is an intriguing programming language with highlights to encourage Web, mobile and

Command line applications. Its significant points of interest are its solidness and backing base. The SDK is continually refreshed and, simultaneously, is truly steady.

C. Firebase^[4]

It offers a bunch of devices that make the item improvement measure much simpler. That is the reason it's appropriate for building a MVP as it abbreviates the time and exertion put in by developers to really make it.

VI. CONCLUSIONS

As you can see from the above research, we are looking forward to implementing this project concept using flutter and firebase. Even though we had planned of using React Native, we fixed Flutter as our final framework because of its pros over the former framework. Thus, we will be developing a platform independent petFriendlyPlaces application by using just one code base which will run smoothly on both android and iOS.

VII. ACKNOWLEDGEMENT

It gives immense pleasure in bringing out this research paper. Firstly, we would like to thank our guide “Ms. Ziany Alpholicy .X” who gave us her valuable suggestions and ideas when we were in need of them. We would also like to thank Prof. Omprakash Yadav for allowing us to take up this idea for our project. Also, a great thanks to Google for providing us with these high-end technologies and also for the well-maintained libraries.

We are immensely grateful to all involved in this project as without their inspiration and valuable suggestion it would not have been possible to come up with this idea.

REFERENCES

- [1] <https://www.theseus.fi/bitstream/handle/10024/146232/thesis.pdf?sequence=1>
- [2] <https://llibrary.net/document/y96dm1vy-application-firebase-android-app-developmentstudy.html>
- [3] <https://flutter.dev/docs/resources/books>
- [4] <https://firebase.google.com/>
- [5] <https://dart.dev/>
- [6] <https://flutter.dev/>
- [7] Alberto Miola. (2020). Flutter Complete Reference
- [8] Eric Windmill. (2019). Flutter in Action
- [9] Ashok Kumar S. (2018). Mastering Firebase for Android Development



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)