

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI – 590018**



**An Internship Report on
“Flappy bird App”**

**Submitted in partial fulfilment of the requirements for the degree of
BACHELOR OF ENGINEERING**

**IN
COMPUTER SCIENCE AND ENGINEERING**

Subject: INTERNSHIP [18CSI85]

Submitted By

RITHESH NK

USN: 4JK19CS039

Internship carried out at

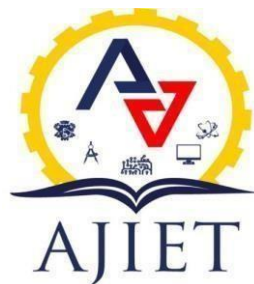
“Klientship Technologies Pvt. Ltd.”

Internal Guide

**Mr. Chanchal Antony
Associate Professor
AJIET ,Mangaluru**

External Guide

**Mr. Likith J Poojary
Managing Director
Klientship Technologies, Mangaluru**



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

A. J. INSTITUTE OF ENGINEERING & TECHNOLOGY

NH-66, KOTTARA CHOWKI, MANGALURU – 575006

2022 - 2023

A. J. INSTITUTE OF ENGINEERING & TECHNOLOGY

NH – 66, Kottara Chowki, Mangaluru - 575006

A Unit of Laxmi Memorial Education Trust (R)

(Affiliated to Visvesvaraya Technological University, Belagavi & Approved by AICTE, New Delhi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



CERTIFICATE

Certified that the Internship Report titled “**FLAPPY BIRD APP**” is carried out by **Mr. Rithesh NK**, USN: **4JK19CS039**, a bonafide student of A.J. Institute of Engineering & Technology Mangalore, at **Klientship Technologies Pvt. Ltd., Mangalore** in partial fulfilment for the award of **Bachelor of Engineering in Computer Science and Engineering** of **Visvesvaraya Technological University, Belagavi** during the year 2022- 2023. It is certified that all corrections /suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library.

The report has been approved as it satisfies the academic requirements in respect of Internship prescribed for the said Degree.

Mr. Chanchal Antony
Internship Guide

Prof. Snitha Shetty
Internship Coordinator

Dr. Antony P J
Head of the Department

Dr. Shantharama Rai C
Principal

Examiners

Signature with Date

1.

2



06/10/2022

CERTIFICATE OF INTERNSHIP

THIS CERTIFICATE GOES TO

Rithesh NK

has successfully completed an App Development internship of 4-weeks. He has worked on technologies such as Dart, Flutter, Firebase, and Firestore. We found the student to be a keen and enthusiastic candidate during the course.

A handwritten signature in black ink, appearing to read "Harshith J Poojary".

Harshith J Poojary

CO-FOUNDER

A handwritten signature in black ink, appearing to read "Vinyas Jagannatha Poojary".

Vinyas Jagannatha Poojary

CO-FOUNDER

ACKNOWLEDGEMENT

First and foremost, we thank our parents for what we are and where we are today, without whose hard work and sacrifice we would not be here today.

Hearty thanks to **Klientship Technologies Pvt. Ltd.**, for providing all the facilities that helped us in timely completion of Internship work.

We deem it a privilege to place on record the deep sense of gratitude to our internship coordinator **Prof. Snitha Shetty, Assistant Professor**, Department of Computer Science and Engineering, and internship guide **Mr. Chanchal Antony, Associate Professor**, Department of Computer Science and Engineering, who always stood behind us and supported in each and every step of the internship work.

We are grateful to **Dr. Antony P J**, Vice Principal & Head of the Department, Computer Science and Engineering for their support and encouragement.

We are indebted to our respected Principal **Dr. Shantharama Rai. C** and the management of **A. J. Institute of Engineering and Technology, Mangaluru** for providing all the facilities that helped us in timely completion of this internship report.

Finally, we would like to thank all the teaching and non-teaching staff of Department of Computer Science and Engineering for their valuable help and support.

Name: RITHESH NK

USN: 4JK19CS039

TABLE OF CONTENTS

CHAPTER NO	CHAPTER	PAGE NO
1	Executive Summary	1
2	Company Profile	3
	2.1 Company History	3
	2.2 Vision	4
	2.3 Mission	4
	2.4 Services	4
3	Problem statement and objectives	6
	3.1 Problem Statement	6
	3.2 Objectives	6
4	Weekly overview	7
5	Task Performed	9
	5.1 About task performed	9
	5.1.1 What is Flutter	9
	5.1.2 Techniques	12
	5.1.3 Packages	14
	5.2 Tool used	17
	5.2.1 Software Requirement	17
	5.2.2 Hardware Requirement	17
	5.3 Methodology	17
6	Result analysis	18
	6.1 Screenshots	18
	6.2 Result analysis	19
7	Discussion	20
	7.1 Discussion	20
	7.2 Future	21
8	Conclusion	22
	8.1 Outcome of the internship	22
	8.2 Scope for the future	22
9	SWOT Analysis	23
10	Bibliography	24

LIST OF FIGURES

FIGURE NO.	FIGURE NAME	PAGE NO.
2.1.1	Company Logo	3
5.4.1	Waterfall Model	17
6.1.1	Home Page	18
6.1.2	gameplay Page	19

LIST OF TABLES

TABLE NO.	TABLE NAME	PAGE NO.
Table 4.1	Weekly overview	7
Table 9.1	SWOT Analysis	23

CHAPTER 1

EXECUTIVE SUMMARY

This report refers to work completed during my internship with Klientship Technologies Pvt. Ltd., from 03 September 2022 to 02 October 2022. During my internship at Klientship, I had the opportunity to learn about several exciting technologies such as GitHub, Git, Flutter, and Firebase. At first, it was quite challenging to grasp the concepts and understand how to use them. However, with time and practice, I managed to overcome the difficulties and gain confidence in my skills.

One of the main challenges I faced was understanding Git and how to use it effectively. At first, I found it confusing to understand the different branches, commits, and merges. However, with the help of my mentor and team members, I was able to understand how Git works and how to use it to manage code versions effectively. I also learned how to use GitHub, a web-based platform for hosting and sharing code repositories, to collaborate with team members on projects.

Working on the flappy bird App project was a great experience for me. It was my first time building a complete mobile application using Flutter, and I faced several challenges along the way. One of the biggest challenges was implementing the login functionality using Firebase Authentication.

Another interesting aspect of the project was building the user interface for the app. I had to carefully design the different screens to ensure that they were user-friendly and easy to navigate. I also had to ensure that the design was consistent across different screen sizes and orientations. Flutter made it easy for me to build beautiful and interactive UI components such as buttons, text fields, and images. To manage the codebase, we used Git and GitHub. I learned how to create branches, commit changes, and merge code changes. This allowed us to work collaboratively on the project and avoid conflicts. We also used GitHub to track issues and bugs, which made it easy for us to prioritize and address them in a timely manner.

In addition to the technical skills, I also gained valuable insights into the working culture of a fast-paced tech company. My mentors and colleagues were always willing to help and guide me, and I learned how to work effectively under tight deadlines and manage my time efficiently. This experience taught me the importance of being adaptable and agile in a constantly evolving technological landscape.

Moreover, the internship gave me exposure to the importance of continuous learning in the tech industry. I realized that technology is constantly evolving, and it is essential to stay up-to-date with the latest trends and tools to remain relevant in the industry. This realization has motivated me to pursue additional learning opportunities and participate in various online communities and events to stay connected with the latest technological advancements.

Overall, my internship experience at Klientship was a great learning opportunity. I gained hands-on experience with several cutting-edge technologies and developed valuable skills that will be useful in my future career. I also learned the importance of collaboration and effective communication in a team environment. I am grateful for the opportunity to work on the Facebook clone project and look forward to applying the skills and knowledge I gained in future projects.

Throughout my internship experience, I learned several valuable skills and gained a better understanding of how these technologies work in a real-world scenario. I also developed strong problem-solving skills and learned how to communicate effectively with team members. This experience has had a significant impact on my personal life and my career. It has allowed me to gain confidence in my skills and inspired me to continue learning and exploring new technologies. I am grateful for the opportunity to work with such a great team and learn from experienced mentors.

CHAPTER 2

COMPANY PROFILE

In this chapter, the brief introduction of the company logo, its major milestones, and the best services provided by the company. In further chapter it briefs about the specific function involved in company, roles and responsibility of the company and their domain in detail.

2.1 Company History

The logo and brand in its entirety are a fusion of a strong core rooted in the Spirit of Values, and a strong sense of connectedness with stakeholders; shape the future together, with them.



Figure 2.1.1: Company Logo

Klientship Technologies Pvt Ltd is a technology company that provides a wide range of services, including web development, app development, machine learning, cloud computing, search engine optimization (SEO), digital marketing, and robotic process automation (RPA) services. The company was founded in 2018 under the name VAwebsites by two brothers, Likith Jagannatha Poojary and Harshith J Poojary.

In 2021, the company was registered as a private limited company and changed its name to Klientship Technologies Pvt Ltd. The company has its headquarters in Bangalore, India. Klientship Technologies Pvt Ltd has several subsidiaries, including Social Bubble, which provides social media management services; The Smart Learn, an e-learning platform; Easy Tourism, a travel and tourism company; and Accounting, which provides accounting and financial management services.

The company has a team of experienced developers, designers, and digital marketing experts who work closely with clients to deliver customized solutions that meet their specific needs. Klientship Technologies Pvt Ltd has worked with clients from various industries, including healthcare, finance, retail, and e-commerce.

2.2 Vision

Klientship Technologies Pvt Ltd's vision is to become a global leader in technology services and to help businesses of all sizes succeed in the digital age. The company aims to achieve this by providing innovative and cutting-edge solutions to its clients that help them achieve their business goals.

The company believes in delivering high-quality services that meet the specific needs of its clients. Its team of experts works closely with clients to understand their business objectives and provide customized solutions that address their unique challenges.

Klientship Technologies Pvt Ltd's vision is to be at the forefront of technology innovation and to leverage the latest advancements in technology to provide its clients with the best possible solutions. The company aims to create a culture of continuous learning and improvement among its employees to ensure that it stays ahead of the curve in the fast-changing technological landscape.

2.3 Mission

Our mission is to make it easy for you to build the software you need. Everyone should be able to do it – doesn't matter what your background, tech knowledge or budget is.

- **Web Development:**
Web Design & Logo - Web designers craft the overall vision & plan for a website layout. Professional logo development: Business, Company, or Personal.
- **App Development:**
IOS & Android - Design Software applications to run on mobile devices. Modern and mobile-ready application that will help you reach all of your marketing.
- **Advertising:**
SEO - We offer services that include keyword research, on-page optimization, linkbuilding, and content marketing.

Google Ads - Advertising services include: Google search result pages, gmails, YouTube and other websites participated in Google Ads program.

- Graphic:

Graphic Design - Klientship provides professional graphic design services that cater to a variety of needs, including branding and marketing materials.

CHAPTER 3

PROBLEM STATEMENT AND OBJECTIVES

3.1 Problem Statement

Flappy Bird was developed by Vietnamese programmer Dong Nguyen and released in 2013 for iOS and Android devices. Despite its simplistic design, the game quickly became a viral sensation and gained a massive following. It was downloaded over 50 million times and was reportedly earning \$50,000 per day from in-app advertising. The game's difficulty and addictive nature were key factors in its popularity. It was incredibly challenging, and many players found themselves spending hours trying to beat their high score. However, this also led to some negative consequences, as players reported feeling frustrated, stressed, and anxious while playing the game. Some even claimed that they were unable to stop playing, even when it was affecting their daily lives. In conclusion, Flappy Bird was a game that gained massive popularity due to its addictive gameplay, but it also faced criticisms for its potential negative impact on players' mental health.

3.2 Objectives

- The objective of Flappy Bird is simple: to guide a small bird through a series of pipes without touching them. The player taps the screen to make the bird flap its wings and fly upwards, and releases the screen to make it descend.
- The bird must pass through the gaps between the pipes to score points. The game ends if the bird hits a pipe or falls to the ground.
- The main objective of the game is to achieve a high score by successfully navigating the bird through as many pipes as possible.
- The game's mechanics are designed to make this difficult, as the bird is constantly moving forward, and the gaps between the pipes are small and require precise timing to pass through.
- As a result, the game can be frustratingly difficult, and players may find themselves playing repeatedly in an attempt to beat their high score.
- Overall, the objective of Flappy Bird is to challenge the player's hand-eye coordination, timing, and reflexes, while also providing a simple and addictive gameplay experience.

CHAPTER 4**WEEKLY OVERVIEW OF INTERNSHIP**

This chapter lists the tasks performed per day throughout the internship period at Klientship

Week - I	Date	Day	Task/ Topic Completed
	03/09/2022	Sat	Introduction About Their Company
	07/09/2022	Wed	Installation and Required Apps
	08/09/2022	Thu	Working of Flutter and Visual Studio
	09/09/2022	Fri	Working of GitHub
	10/09/2022	Sat	How to Pull Request from GitHub
	12/09/2022	Sun	CMD Command Prompts

Week - II	Date	Day	Task/ Topic Completed
	13/09/2022	Mon	How to Use Flutter
	14/09/2022	Tue	Visual studio and git Use
	15/09/2022	Wed	Learning about the documentation flutter
	16/09/2022	Thu	Creating a New File and Using Flutter
	17/09/2022	Fri	Basic project in flutter
	19/09/2022	Sun	Learned About how to add flutter app in your phone and website

Week - III	Date	Day	Task/ Topic Completed
	20/09/2022	Mon	Learned About Firebase
	21/09/2022	Tue	Firebase Uses
	22/09/2022	Wed	Flutter and firebase implementation
	23/09/2022	Thu	Install the Firebase SDK
	24/09/2022	Fri	Learned How to use Authentication in your Firebase console
	26/09/2022	Sun	Learned about different frameworks in firebase

Week - IV	Date	Day	Task/ Topic Completed
	27/09/2022	Mon	How To Run a Code in GitHub
	28/09/2022	Tue	Discussion About the Project Topic
	29/09/2022	Wed	Implementation of the Project
	30/09/2022	Thu	Project Presentation in the Office
	01/10/2022	Fri	Project Presentation in the Office
	02/10/2022	Sat	Internship Project Completion

Table 4.1: Weekly Overview

CHAPTER 5

TASK PERFORMED

This chapter discusses the duties we completed throughout our internship at Klientship Technologies Pvt. Ltd., as well as the knowledge we received during that time. The chapter also discusses about the technical and non-technical abilities I gained during my internship at Klientship Technologies Pvt. Ltd., as well as a full account of the activities and work I did each day.

5.1 About Tasks Performed

The task performed in this internship includes learning the basics of flutter, flutter packages such as image picker, flutter fire, URL launcher, git commands, and others package.

5.1.1 What is Flutter

Flutter is an open-source mobile application development framework created by Google, which allows developers to build high-quality, cross-platform mobile applications for iOS and Android platforms. Flutter uses the Dart programming language and provides a rich set of pre-built widgets and tools that make it easy to create visually appealing and responsive user interfaces. Some of the key characteristics and points that make Flutter stand out:

- **Widgets:** As mentioned earlier, Flutter's UI is built using widgets. Widgets are essentially building blocks that developers can use to construct their UIs. Flutter offers a rich library of pre-built widgets, as well as the ability to create custom widgets. This widget-based approach allows for highly customizable UIs, and makes it easy to reuse UI elements throughout an app.
- **Performance:** Flutter is designed to be fast and efficient. It achieves this through a number of features, such as its use of native code, its efficient rendering engine, and its ability to perform complex computations in the background using Isolates. Flutter's focus on performance makes it a popular choice for building apps that require high performance, such as games and multimedia apps.

- **Hot Reload:** One of the most unique features of Flutter is its hot reload feature. Hot reload allows developers to make changes to their code and see the results instantly, without having to restart the app. This can save developers a significant amount of time and make the development process more efficient.
- **Cross-platform support:** Flutter allows developers to build apps for multiple platforms using a single codebase. This includes support for both Android and iOS, as well as other platforms such as the web and desktop. This can save time and resources, as developers only need to write and maintain a single codebase for multiple platforms.
- **Developer tools:** Flutter includes a robust set of developer tools that make it easy to develop and test apps. This includes tools such as the Flutter DevTools, which provide real-time performance profiling and debugging capabilities, and the Flutter Test Framework, which allows for automated testing of UI elements and interactions.
- **Customization:** Flutter offers a high degree of customization for developers. This includes the ability to create custom widgets, use custom fonts and icons, and create custom animations and transitions. This flexibility allows developers to create unique and visually appealing UIs.
- **Community:** Flutter has a growing and active community of developers, who contribute to the framework through code, tutorials, and support. This community provides a wealth of resources and expertise for developers who are new to Flutter or who need help with specific issues.

Firebase

Firebase is a mobile and web application development platform created by Google. It provides a suite of backend services such as authentication, real-time database, cloud storage, and messaging, among others, that can be used to build scalable and feature-rich applications.

Flutter and Firebase can be integrated together to build mobile applications with real-time data synchronization, user authentication, and other backend features. The integration of Flutter with Firebase is facilitated by the Flutter Fire library, which provides a set of Flutter plugins for Firebase services.

Here's how Flutter and Firebase work together:

- **Flutter sends requests to Firebase using Firebase plugins:** Flutter uses Firebase plugins, which are a set of packages that provide an interface between the Flutter application and Firebase services. Each plugin corresponds to a specific Firebase service, such as authentication, database, storage, or messaging. Flutter plugins can be added to the project by adding dependencies to the pubspec.yaml file.
- **Firebase handles requests and responses:** Firebase handles requests from the Flutter application and sends back responses. Firebase uses secure authentication methods and APIs to ensure that only authorized users can access the Firebase services.
- **Firebase sends real-time updates to Flutter:** Firebase uses a real-time database to send real-time updates to the Flutter application. This allows the application to be updated in real-time as changes are made to the database.

Motivation for the project:

The motivation behind the development of Flappy Bird was likely to create a simple and addictive game that could be enjoyed by a wide range of players. The game's developer, Dong Nguyen, has stated in interviews that he initially developed the game as a hobby, with no expectation of it becoming as successful as it did. Overall, the motivation behind the development of Flappy Bird was likely a combination of creating a fun and addictive game, and standing out in a crowded mobile gaming market.

Resources used:

The following resources were used in the development of the Facebook clone app:

- **Flutter:** Flutter is an open-source UI toolkit for building high-performance, high-fidelity mobile, web, and desktop applications.
- **Android Studio:** Android Studio is the official integrated development environment for building Android applications.
- **Dart programming language:** Dart is a client-optimized language for fast apps on multiple platforms.

- Flutter plugins: Flutter plugins are pre-written code libraries that can be imported into your Flutter application to provide additional functionality.

5.1.2 Techniques

1. Planning:

- a. This is a crucial phase that sets the foundation for the entire project. It involves defining the objectives, scope, and requirements of the app. The project team works together to identify the target audience, the app's purpose, and the expected outcomes. During this phase, the team creates a project plan, including timelines, budgets, and resource allocation. They also define the app's technical requirements, such as the development platform, tools, and libraries.
- b. The planning phase should include a thorough analysis of the market and competition, as well as user research and feedback. The team should identify the app's unique selling points and value proposition to stand out from similar apps. It's also important to consider legal and regulatory requirements, such as privacy and security policies, and ensure compliance.

2. Design:

- a. In this phase, designers create the visual and interactive aspects of the app. They use various tools to create wireframes, mockups, and prototypes. The design team defines the app's UI elements, including typography, color schemes, icons, and imagery. They also determine the navigation flow and interaction patterns of the app. The design phase typically involves several iterations of feedback and revision until the final design is approved.
- b. The design phase should also consider accessibility and usability principles, such as designing for different screen sizes and input methods. The design team should work closely with the development team to ensure the designs are technically feasible and implementable. User testing and feedback can also be incorporated into the design phase to validate the designs and improve the user experience.

3. Development:

- a. This phase involves coding the app's functionality using the Dart programming language. Developers build the app's architecture, implement the UI designs, and

integrate various libraries and APIs..

- b. It's essential to follow best practices for coding, such as using version control, writing unit tests, and documenting the code. The development phase should also include continuous integration and deployment (CI/CD) practices to ensure fast and reliable releases. The development team should also consider security and privacy issues, such as encryption, authentication, and data storage. They should also test the app's performance under different network conditions and device configurations.

4. Testing:

- a. In this phase, the app is tested to ensure it is functional and free of bugs. The testing team creates test plans and scripts to validate the app's behavior under various conditions. Testing can include unit tests, integration tests, and acceptance tests. Testers try to find defects and report them to the development team. The development team then fixes the defects and verifies the fixes. The testing phase is iterative and can continue throughout the project's life cycle.
- b. The testing phase should also include security and penetration testing to identify vulnerabilities and prevent attacks. The testing team should also use automated testing tools and techniques to increase test coverage and efficiency. The testing phase should be collaborative between the testing and development teams to ensure effective communication and issue resolution.

5. Deployment:

- a. This phase involves publishing the app to the app store or distributing it to users. Developers configure the app's settings, sign the app, and upload it to the app store or a distribution platform. They also create app store listings and promotional materials. Once the app is live, developers monitor its performance, such as the number of downloads, ratings, and user feedback.
- b. They may also release updates and bug fixes to improve the app's functionality and user experience. The deployment phase should also include app store optimization (ASO) practices to increase the app's visibility and downloads. The deployment team should also consider user

5.1.2 Packages

Flutter packages are pre-written code modules that can be integrated into a Flutter app to add new functionality, streamline development, and reduce coding time. Here are some popular Flutter packages and their uses:

1. Flutter Material Design:

Flutter Material Design is a package that provides a rich set of UI components for building visually appealing apps that follow Google's Material Design guidelines. The Material Design package includes various widgets such as AppBar, BottomNavigationBar, FloatingActionButton, Card, and many more. The package is easy to use, and its widgets are highly customizable, allowing developers to design apps that are consistent with the Material Design principles. This is used in the project to create navigation bars, buttons and search bars.

2. Flutter HTTP package:

The Flutter HTTP package enables Flutter apps to interact with HTTP APIs and fetch data from servers. It simplifies the process of making HTTP requests and handling responses. The HTTP package also provides support for various HTTP methods such as GET, POST, PUT, and DELETE. It also supports custom headers, timeouts, and response parsing.

3. Flutter Firebase:

Firebase is a platform that provides various services such as authentication, real-time database, cloud storage, and more. The Flutter Firebase package allows developers to integrate Firebase services into their apps easily. Firebase offers an easy-to-use interface, and the package provides methods for initializing Firebase services, configuring the Firebase instance, and accessing various Firebase services such as authentication, database, and storage.

4. Flutter Bloc package:

The Flutter Bloc package provides a way to manage the state of a Flutter app by breaking it down into small, manageable units called blocs. It enables developers to separate the business logic from the UI and makes it easier to test the app.

5. Flutter Provider package:

The Provider package is another state management solution for Flutter apps. It provides a way to pass data between widgets efficiently and makes it easier to handle complex UI interactions. The Provider package uses the InheritedWidget widget to provide data to child widgets, allowing widgets to access data without the need for global variables. With the Provider package, developers can create apps with a scalable architecture and avoid boilerplate code

6. Flutter Image Picker package:

The Image Picker package enables Flutter apps to access the device's camera and gallery and select an image. This package simplifies the process of selecting an image and uploading it to a server. The Image Picker package also provides support for customizing the picker UI and handling errors. With this package, developers can create apps that allow users to select images quickly and efficiently.

7. Flutter Shared Preferences package:

The Shared Preferences package provides an easy way to store and retrieve key-value pairs in a Flutter app. It allows developers to persist data on the device and use it across app sessions. The Shared Preferences package is ideal for storing app settings, user preferences, and other small data sets that need to persist across app sessions. With this package, developers can create apps that remember user preferences and settings.

Pros:

- **Modular architecture:** Flutter's modular architecture allows developers to create independent, reusable components, which can help to reduce code duplication and improve maintainability.
- **Strong developer tools:** Flutter comes with a set of powerful developer tools, including a visual debugger, profiler, and command-line tools, which can help developers to identify and fix issues quickly.
- **Integration with other technologies:** Flutter can easily integrate with other technologies, such as Firebase, GraphQL, and more, which can help to enhance app functionality and performance.
- **Customizability:** Flutter's flexible widget system allows developers to create custom widgets and animations that can be tailored to the specific needs of an app.
- **Rapid development cycles:** Flutter's fast development cycles and hot reload feature can enable developers to create and test new features more quickly, which can lead to faster time-to-market.

Cons:

- **Learning curve:** Flutter uses its own programming language called Dart, which may require some time for developers to learn.
- **Limited libraries:** While Flutter's community is growing, it may still be limited in terms of the number of libraries and plugins available compared to other mobile app development frameworks.
- **Large app size:** Flutter apps tend to have larger file sizes compared to native apps due to the inclusion of the Flutter engine and framework.
- **Limited access to native features:** While Flutter can access many native features of the device, some features may still require native development to access.
- **Unproven long-term stability:** Flutter is still a relatively new framework and its long-term stability and viability are yet to be proven.

5.2 Tools Used

5.2.1 Software Requirement

- IDE: Visual Studio
- Language: Dart, Flutter & Firebase
- OS: Windows 10

5.2.2 Hardware Requirement

- Processor: intel core i5 or AMD Ryzen 5
- RAM: minimum 8GB
- Disk Space: 108GB

5.3 Methodology

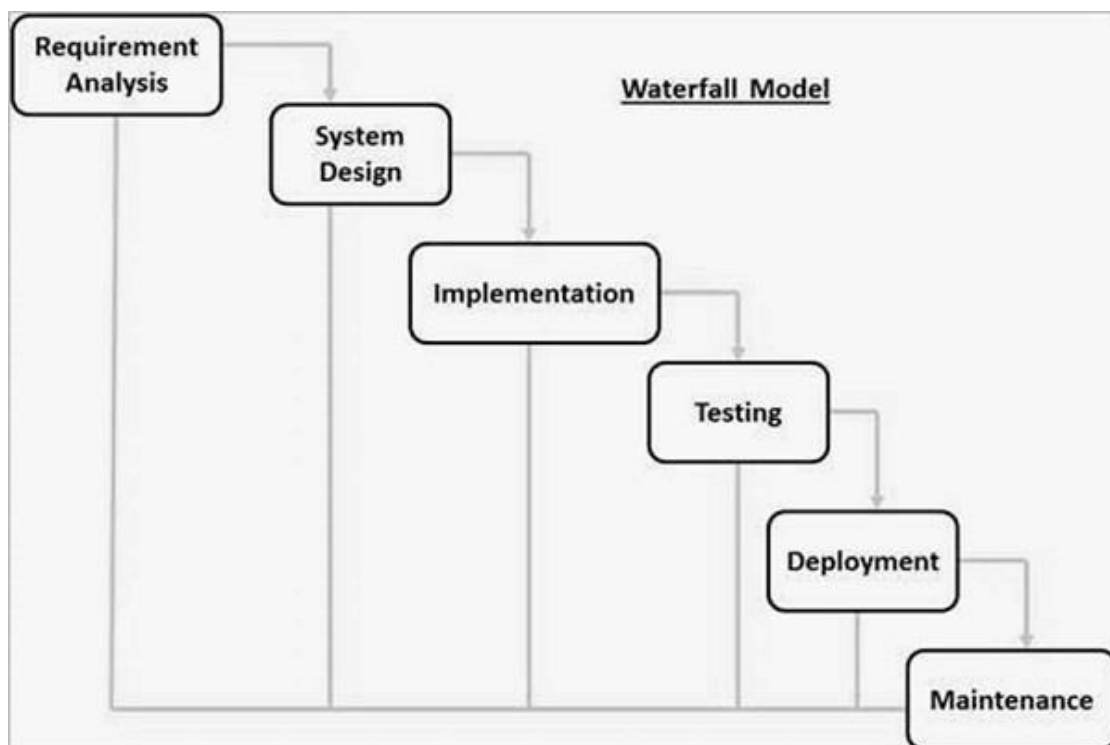


Fig 5.4.1 Waterfall model

The waterfall methodology, also known as the Waterfall model, is a linear sequential approach to software development that follows a strict, step-by-step process. It is a well-established methodology that has been widely used for decades and is characterized by its rigid structure and the emphasis on planning and documentation.

CHAPTER 6

RESULTS AND ANALYSIS

6.1 Screenshots

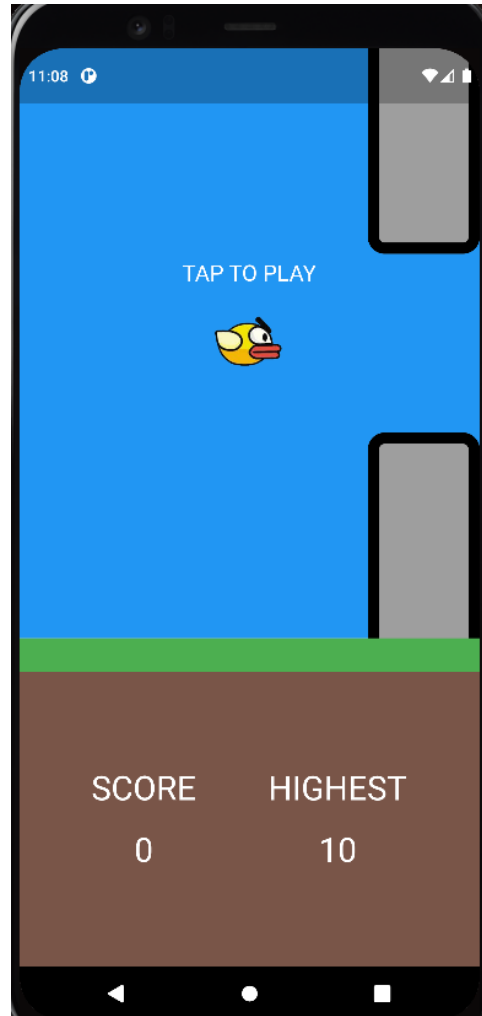


Figure 6.1.1 Home Page

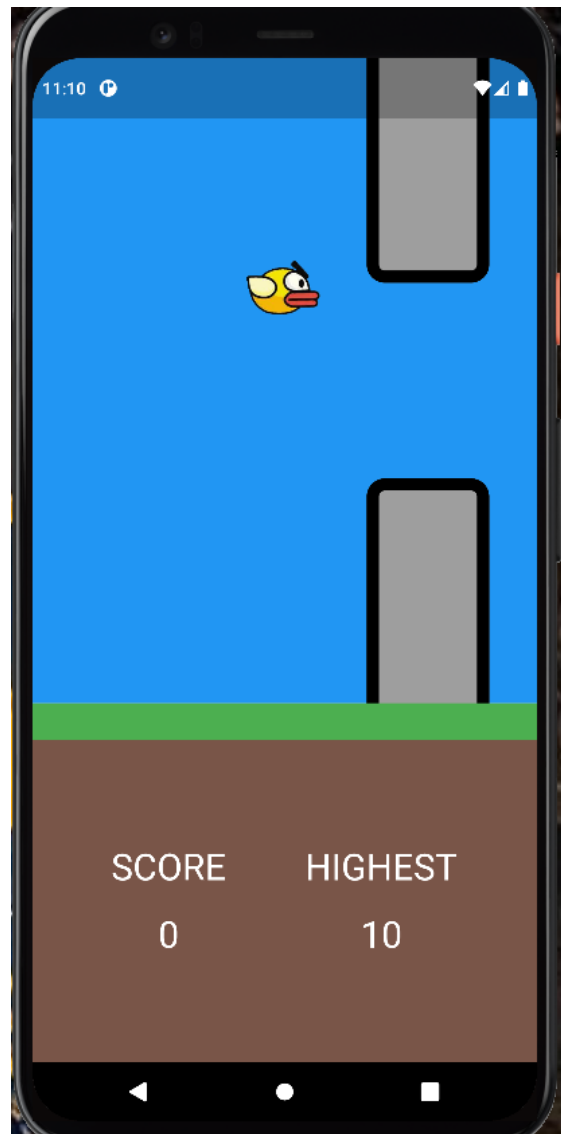


Figure 6.1.2 Gameplay Page

6.2 Result analysis

This project's main goal is to make a working flappy bird App.

CHAPTER 7

DISCUSSION

7.1 Discussion

With Flutter, developers can write one codebase for both iOS and Android platforms, which significantly reduces development time and costs. Flutter's "hot reload" feature also enables developers to see the results of their changes almost instantly, making the development process faster and more efficient.

Another significant use of Flutter is in the development of custom UI components. Flutter provides a rich set of widgets and tools that allow developers to create beautiful, customized UIs with ease. With Flutter's customizable widgets, developers can create unique user experiences that stand out from other apps.

Flutter is also an excellent choice for creating high-performance apps. The framework provides an optimized architecture that enables fast rendering and smooth animations, which is crucial for apps that require high-performance, such as games and AR/VR apps.

But the Flutter's reliance on the Dart language can also be seen as a disadvantage. While Dart is an easy-to-learn and powerful language, it is not as widely used as other programming languages, such as JavaScript. This can make it difficult to find developers with experience in Dart, which can lead to challenges in hiring a skilled development team.

Another con of Flutter is that it is a relatively large framework, which can increase app size. This can be a disadvantage for apps with low storage or devices with slow internet connections. However, Flutter's tree shaking feature can help reduce app size by removing unused code.

Lastly, Flutter's reliance on its own set of widgets and UI components can be a disadvantage for developers who prefer to use pre-built components or third-party libraries. While Flutter provides a rich set of widgets, it may not be as extensive as other frameworks.

7.2 Future

Flutter is quickly becoming one of the most popular frameworks for mobile app development, with many businesses and developers using it to create high-quality applications that are efficient, user-friendly, and visually stunning. Its potential impact on the app development industry in the coming years is immense, as it provides developers with the ability to create engaging and visually appealing applications that run seamlessly across multiple platforms.

Flutter is also evolving and changing rapidly. Google is continually updating the framework with new features and capabilities, making it more powerful and efficient with each update. For example, Flutter 2.0 introduced new features such as sound null safety, improved support for desktop and web applications, and new widgets for building beautiful user interfaces.

Flutter's adoption by developers and businesses is also growing rapidly. Many developers are switching to Flutter due to its flexibility, ease of use, and robust development tools. Businesses are also recognizing the potential of Flutter to reduce development costs, speed up delivery times, and create high-quality applications that run seamlessly across multiple platforms.

Flutter is transforming the future of mobile app development in a significant way. Its potential impact on the app development industry is immense, and its popularity is growing rapidly. Flutter's flexibility, ease of use, and powerful development tools are making it a popular choice for developers and businesses alike. As the framework continues to evolve and change, we can expect even more exciting features and capabilities to be added in the coming years

CHAPTER 8

CONCLUSION

8.1 Outcome of the Internship

During my internship, I learned various new concepts about Flutter. During an internship, you can learn a lot about your own strengths and weaknesses. Accept the mistakes you will make as an intern and the many things you will not know. To get the most out of your internship training experience, ask questions, observe, and take risks. Even though I knew a bit about the basics of android development concepts, this internship taught more on it to me. After this internship, working with flutter appears to be a little easier than before and interesting when learnt how firebase and flutter can be used together for better product.

8.2 Scope for future work

Flappy Bird has not been updated or re-released since it was removed from app stores in 2014. Therefore, it is unlikely to have any future scope as a game in its current form.

However, the game's popularity and success have inspired many developers to create similar games with similar mechanics, such as Splashy Fish, Swing Copters, and others. These games have also gained some level of popularity, but none have reached the same level of success as Flappy Bird.

Additionally, the legacy of Flappy Bird has had a lasting impact on the mobile gaming industry. It highlighted the potential for simple, addictive games to become viral sensations, and it also brought attention to the issue of addictive gameplay and its impact on mental health.

Furthermore, the controversy surrounding Flappy Bird's development and marketing practices has led to increased scrutiny of the mobile gaming industry and its practices. This has resulted in changes to the way games are marketed and monetized, with many developers now focusing on more ethical and transparent practices.

In conclusion, while Flappy Bird itself may not have any future scope as a game, its legacy has had a lasting impact on the mobile gaming industry and has inspired changes in how games are developed and marketed.

CHAPTER 9

SWOT ANALYSIS

<p style="text-align: center;">Strength</p> <ul style="list-style-type: none">• Flexibility for multiple platforms• Safety and security• high performance	<p style="text-align: center;">Weakness</p> <ul style="list-style-type: none">• Learning curve• Large app size• Limited access to native features
<p style="text-align: center;">Opportunities</p> <ul style="list-style-type: none">• Faster development• Large community• Compatibility with other technologies	<p style="text-align: center;">Threats</p> <ul style="list-style-type: none">• Immature technology• Limited support for older devices• Limited access to native feature

Table 9.1: SWOT Analysis

BIBLIOGRAPGHY

- [1] Dong Nguyen. (2013). Flappy Bird. [Mobile application software]. Retrieved from <https://flappybird.io/>
- [2] Stuart, K. (2014, February 11). Flappy Bird: the story behind the game that drove the world insane. The Guardian. Retrieved from <https://www.theguardian.com/technology/2014/feb/11/flappy-bird-story-behind-game-dong-nguyen>
- [3] Yarow, J. (2014, February 10). The Flappy Bird frenzy has ended. Here's what it all means. Business Insider. Retrieved from <https://www.businessinsider.com/the-flappy-bird-frenzy-has-ended-heres-what-it-all-means-2014-2>
- [3] Linder, B. (2014, February 10). Flappy Bird and the frustrating, seductive simplicity of mobile gaming. The Verge. Retrieved from <https://www.theverge.com/2014/2/10/5392828/flappy-bird-and-the-frustrating-seductive-simplicity-of-mobile-gaming>
- [4] Wawro, A. (2014, February 10). Flappy Bird's removal could signal major changes for mobile game developers. Gamasutra. Retrieved from https://www.gamasutra.com/view/news/210628/Flappy_Birds_removal_could_signal_major_changes_for_mobile_game_developers.php
- [5] .01

Good, O. (2014, February 11). Flappy Bird and the dark side of indie game development. Wired. Retrieved from <https://www.wired.com/2014/02/flappy-bird-indie-game-development/>