



L OVELY
P ROFESSIONAL
U NIVERSITY

JAVA PROJECT REPORT

TOPIC:

Winged Adventure: Flappy Bird

Submitted by

Abhishek kumar (12112336) RK21STA32

Course Code CSE 310

Under the Guidance of :

Dr Ranjith Kumar A (28108)

TABLE OF CONTENTS

INTRODUCTION
PROPOSED TECHINQUES
MODULES
SAMPLE CODE ATTACHED
CONCLUSION
FUTURE ENHASMENT

INTRODUCTION

Flappy Bird is a popular and addictive mobile game that has captured the attention of players worldwide. The game involves controlling a bird to fly through a series of pipes without hitting them. It requires quick reflexes, timing, and coordination to navigate through the obstacles successfully.

In this Java project, we will create a Flappy Bird game using Java programming language. We will use Java's swing library to create the user interface and implement the game's mechanics. The game will have a simple design, and the objective is to score as many points as possible by passing through the pipes.

The project will involve creating two classes, Bird, and Tube. The Bird class will be used to create a bird object in the game and to move it around the screen. The Tube class will be used to create a wall object in the game and to move it around the screen.

We will also use various Java libraries, including the AWT library, to display images, the Timer class to update the game's state, and the Random class to generate random numbers. The project will also include handling user input to allow the player to control the bird's movement.

Overall, this project will demonstrate how to create a simple game using Java programming language and will provide a fun and engaging learning experience for those interested in game development using Java.

WHAT YOU WILL LEARN?

- 1) Building logic with the help of functions, loops, conditionals, and variables
- 2) Handling Classes and Object creations
- 3) Java Swing and Java AWT for creating a user-friendly GUI

FEATURES:

- 1) Use the spacebar key to fly high
- 2) Create obstacles and birds with the help of Java Swing
- 3) Keep updating the current score and the high score as the bird passes by an obstacle

MODULES:

CREATING THE GAME ELEMENTS:

To create the shape of the bird, wall, and background we have used images.

INITIALIZATION OF GAME OBJECTS

```
class Bird extends GameObject {
```

```
    private ProxyImage proxyImage; // ProxyImage object used to load the  
image of the bird
```

```
    private Tube[] tube; // Array of Tube objects used to create the walls in the  
gameV
```

BY THIS THE IMAGES IS LOADED AND GET TO THE GAME

```
proxyImage = new ProxyImage("io.jpg"); // Load the image
```

```
    background = proxyImage.loadImage().getImage(); // Get the image
```

DETECTING COLLISIONS BETWEEN THE BIRD AND WALL OR GROUND:

To achieve this nested if-else conditions are used to compare each object's coordinates

```
private void checkWindowBorder() {
```

```
    if(this.x > Window.WIDTH) { // If the bird has moved off the right side  
of the screen
```

```
        this.x = Window.WIDTH; // Set the x position of the bird to the right  
side of the screen
```

```
}  
if(this.x < 0) { // If the bird has moved off the left side of the screen  
    this.x = 0; // Set the x position of the bird to the left side of the screen  
}  
if(this.y > Window.HEIGHT - 50) { // If the bird has moved off the bottom  
of the screen  
    this.y = Window.HEIGHT - 50; // Set the y position of the bird to the  
bottom of the screen  
}  
if(this.y < 0) { // If the bird has moved off the top of the screen  
    this.y = 0; // Set the y position of the bird to the top of the screen
```

OUTPUTS:





Profile of the Problem. Rationale/Scope of the study (Problem Statement)

The purpose of this study is to investigate the problem being addressed in the Flappy Bird project is the challenge of developing a simple, yet engaging and addictive game that can provide hours of entertainment to players. The game is designed to test the player's skills and reflexes as they navigate a bird through a series of obstacles. The goal is to achieve the highest score possible, which can be a difficult task due to the fast-paced gameplay and the ever-increasing difficulty level. The project aims to provide an enjoyable gaming experience for users while also showcasing the developer's skills in game development.

The scope of the study includes The problem being investigated in the Flappy Bird project is the difficulty in achieving a high score due to the challenging and fast-paced nature of the game. The boundaries of the problem are the constraints of the game itself, such as the limited vertical movement of the bird and the constant movement of the pipes. Additionally, the problem includes the limitations of the player's reaction time and ability to accurately control the bird's movements. The project aims to explore and develop strategies and techniques to overcome these challenges and achieve higher scores.

The problem statement is based on the recognition of The rationale behind the problem statement for the Flappy Bird project is the game's addictive nature and its potential negative impact on players. Flappy Bird gained massive popularity upon its release in 2013, and many players reported spending long hours playing the game, often neglecting other

responsibilities. Furthermore, the game's high level of difficulty could lead to frustration, which could have a negative impact on the mental and emotional wellbeing of players. This project aims to explore ways to improve the gameplay experience of Flappy Bird while addressing the potential negative consequences of its addictive nature.

The proposed solution or objective of the Flappy Bird project is to create a game that provides an entertaining and engaging experience for players. Specifically, the game will involve controlling a bird character to navigate through a series of obstacles, with the objective of achieving the highest score possible. The game will be designed to be simple and intuitive, with easy-to-understand gameplay mechanics that can be quickly learned by players of all ages and skill levels. Additionally, the project aims to leverage modern technologies and programming practices to create a stable and scalable game that can be enjoyed across a range of devices and platforms.

The target audience or stakeholders who will benefit from the Flappy Bird project are casual gamers and game developers who want to learn and understand game development concepts, particularly in the area of mobile game development. Additionally, individuals who are interested in creating mobile games as a hobby or for personal projects can also benefit from this project.

Flappy Bird project aims to address the problem of player frustration and low engagement due to the game's high difficulty level. By implementing an adaptive difficulty algorithm, the project seeks to enhance the user experience and increase player retention. The project's target audience includes mobile gamers, game developers, and researchers interested in game design and user experience. Through this project, we hope to provide insights into the potential of adaptive difficulty algorithms in improving user engagement in mobile gaming applications.

Existing System

Existing System: The existing system for the Flappy Bird game is the original version that was developed by Dong Nguyen in 2013. This game became very popular and was available on both iOS and Android devices. The game involves a bird that flaps its wings to fly through a series of pipes without touching them. The game is simple to play but can be challenging to master.

Existing Software: The existing software for the Flappy Bird game is the original version that was developed by Dong Nguyen in 2013. This software was written using the programming language Cocos2D-x and was available on both iOS and Android devices. The game was simple to play and was free to download.

What's new in the system to be developed: The system to be developed for the Flappy Bird game may have new features and improvements compared to the original version. For example, it may have updated graphics, improved sound effects, additional levels, and new obstacles. The system may also be developed using a different programming language or game engine. Overall, the aim of the new system would be to provide an enhanced and enjoyable experience for the users.

Problem Analysis:

Product Definition:

The product under analysis is a replica of the popular mobile game, Flappy Bird. The game involves navigating a bird through a series of obstacles by tapping the screen to make the bird flap its wings and gain altitude. The objective is to achieve the highest score possible by passing as many obstacles as possible.

Feasibility Analysis:

The development of a Flappy Bird game replica is feasible, as there are many existing similar games available in the market, indicating a demand for such games. Additionally, the required technology and tools for developing the game are readily available.

Project Plan:

The project plan for the development of the Flappy Bird game replica will include the following phases:

Planning Phase: This phase will include outlining the project's objectives, goals, and timelines. It will also involve setting up the project team, defining roles and responsibilities, and creating a project plan.

Design Phase: In this phase, the game's design will be created, including the game mechanics, artwork, and user interface. The design phase will also involve creating wireframes, storyboards, and mockups.

Development Phase: This phase will involve the actual coding and development of the game. The development phase will include programming, integration, testing, and bug fixing.

Deployment Phase: In this phase, the game will be deployed to the intended platform, and any final adjustments or bug fixes will be made.

Maintenance Phase: This phase will involve maintaining the game by fixing bugs and updating the game as required to ensure that it remains relevant and functional.

Software Requirement Analysis for Flappy Bird Game:

Introduction:

The software requirement analysis is a process of identifying, defining and documenting the requirements that must be satisfied by the software. The software requirements analysis for Flappy Bird Game is done to determine the necessary requirements for the development of the game.

General Description:

- 1.Flappy Bird is an endless side-scrolling game that requires the player to control a bird, attempting to fly between pipes without hitting them. The game will have the following general features:
- 2.The game will be played on a mobile device and will be available for both Android and iOS platforms.
- 3.The game will have a simple user interface, which will be easy to understand and operate.
- 4.The game will have a scoring system based on the number of pipes the bird passes through.

Specific Requirements:

The specific requirements for Flappy Bird Game are as follows:

1.User Interface Requirements:

The user interface must be simple and intuitive.

The game should display the score and high score.

2.Functional Requirements:

The bird must be able to fly and move forward.

The bird must collide with the pipes if it comes in contact with them.

The game should end if the bird hits a pipe or touches the ground.

Performance Requirements:

The game should be smooth and responsive with no lag or delay.

The game should have minimum loading time and quick game startup time.

The game should run on low-end devices as well.

4.Compatibility Requirements:

The game should be compatible with both Android and iOS platforms.

The game should be tested on multiple devices to ensure compatibility.

5.Security Requirements:

The game should not access any personal information of the user.

The game should be free from any malware or viruses.

6.Legal Requirements:

The game should comply with all the legal requirements and regulations.

7.Documentation Requirements:

The game should have a user manual, which will describe the game's features and how to play it.

The game should have a technical manual that will provide information about the software, its architecture, and its design.

8.Support and Maintenance Requirements:

The game should have a help desk or a support team to handle user queries and issues.

The game should be maintained regularly to ensure its performance and usability.

In conclusion, the software requirement analysis of Flappy Bird Game has identified the necessary requirements for the development of the game. The specific requirements have been outlined, and it is essential to meet them to ensure the game's success.

System Design:

The system design for the Flappy Bird project would include the overall architecture of the game. It would involve deciding on the game engine to be used, the programming language to be used, and the design of the game environment, such as the background, obstacles, and characters.

Design Notations:

Design notations would be used to create a visual representation of the game's structure, such as flowcharts, UML diagrams, and data flow diagrams. This would make it easier for developers to understand the game's structure and implement the code.

Detailed Design:

The detailed design of the game would involve breaking down the system design into smaller components and defining their functionalities. This would include designing the game's user interface, such as the start menu, gameplay screen, and end menu. It would also involve designing the game's mechanics, such as the physics of the bird's movement and the generation of obstacles.

In addition, the detailed design would specify the code structure and algorithms to be used. This would ensure that the code is efficient and easy to maintain. The detailed design would also include testing plans to ensure that the game works as expected and that any issues are identified and resolved.

Pseudo code for Flappy Bird game:

1. Initialize game variables (score, high score, game over flag, etc.).
2. Load game assets (background, bird image, pipe image, etc.).
3. Create bird object with initial position, velocity, and gravity.
4. Create pipe object with random position and gap between upper and lower pipes.
5. Loop while game over flag is false:
 - a. Check for player input (jump command).
 - b. Update bird position and velocity based on gravity and jump command.
 - c. Update pipe position and check for collision with bird.
 - d. Draw game elements (background, bird, pipes, score) on screen.
 - e. Check for game over conditions (collision with pipe, hitting top/bottom of screen).
 - f. Update score and high score as needed.
6. End game loop and display game over message.
7. Allow player to restart game or quit.

Functional Testing:

Functional testing ensures that the application functions as expected from the end-user perspective. In the case of the Flappy Bird game, functional testing includes testing the game mechanics such as the ability to make the bird flap its wings, obstacles appearing on the screen, scoring mechanism, and game over scenarios.

Structural Testing:

Structural testing checks the internal structure of the application, and the test cases are designed based on the code. In the case of the Flappy Bird game, structural testing includes testing the code for any syntax errors, checking for any infinite loops, and ensuring that the code adheres to the best coding practices.

Levels of Testing:

The testing of the Flappy Bird game can be done at various levels, such as unit testing, integration testing, system testing, and acceptance testing.

Testing the Project:

To test the Flappy Bird game, a range of test cases can be designed that simulate different gameplay scenarios to ensure the game is working as expected. The game can be tested on various devices and platforms to ensure compatibility. Testing can be done both manually and using automated testing tools to ensure maximum coverage and efficiency.

FUTURE IMPROVEMENT

In the future, we would like to add more features to our Flappy Bird game, such as different levels with varying backgrounds and obstacles.

We would also like to implement a high score system that saves the player's best scores and displays them on the start screen.

We would add the sound and music effect in this and also we would add something related to AI which will make this game futuristic and treading for coming era as well.