STICKYTOONS

Hammad Naseer (1812247)
Muhammad Khushnood Ali (1812274)

Supervised by: Mr. Muhammad Azeem

In partial fulfillment of requirement for the degree
Bachelor of Science (Computer Science)
Department of Computer Science
Faculty of Computing and Engineering Sciences
Shaheed Zulfikar Ali Bhutto Institute of Science and Technology
Islamabad, Pakistan

Fall 2022

STICKYTOONS

By

Hammad Naseer (1812247)
Muhammad Khushnood Ali (1812274)

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF BACHELOR OF SCIENCE (COMPUTER SCIENCE)

(Supervisor) Muhammad Azeem	(FYP Committee Head) Muhammad Nadeem Khokhar
(Program Manager BS(CS)) Dr Shahzad Latif	

Department of Computer Science
Faculty of Computing and Engineering Sciences
Shaheed Zulfikar Ali Bhutto Institute of Science and Technology
Islamabad, Pakistan

Fall 2022

DECLARATION

We, the candidates of Bachelor of Science (Computer Science) at Shaheed Zulfikar Ali Bhutto Institute of Science and Technology, Islamabad do hereby certify that this report titled **STICKYTOONS**, submitted as partial fulfillment of Bachelor of Science (Computer Science) degree requirements, is our original work and we are its sole author. All the employed materials, references to the literature and the work of others have been referred to and duly cited. This report has not been presented for examination anywhere else.

Hammad Naseer
(1812247)

Muhammad Khushnood Ali
(1812274)

Project Overview

STICKYTOONS is a problem-solving game that aims to enhance player's thinking skills. One of the main challenges in developing this game is the lack of similar games available in Pakistan, which motivated the team to create a unique and engaging experience for players. The proposed solution is to use Unity 3D and C# to create a game that is initially available for the Windows operating system, with the possibility of expanding to other platforms in the future. The project team will utilize resources such as the Unity asset store and open source sites to assist in the development and programming of the game. In terms of scope, the game will be accessible to players of all ages and will focus on problem-solving through various levels and challenges. It will be compared to existing systems through user testing and feedback to evaluate its effectiveness in enhancing thinking skills.The conclusion of this project is that STICKYTOONS has the potential to be a successful and unique game that addresses the lack of problem-solving games in Pakistan, while also providing an enjoyable and intellectually stimulating experience for players. In the future, the team hopes to expand the game to other platforms and continue to improve and update it based on user feedback

Dedication

Firstly, we dedicate our project to the creator Allah Almighty and dedicate to whom the world owes its existence Muhammad (Peace Be Upon Him) and dedicate this to our beloved parents, our extremely dedicated and generous teachers and supportive friends, their prayers always pave the way to success for us.

Acknowledgment

We are grateful to Allah Almighty for giving us the opportunity to complete this Final Year Project named STICKYTOONS. During the process of finalizing this report, We came across a number of people whose contributions, in various ways, aided in its creation, and they deserve a special note of thanks. It brings us joy to express my thanks to each and every one of them.

First and foremost, We want to express our gratitude to my university's FYP Panel for accepting our request and allowing us to work on this project. Next, We owe a debt of gratitude to our project supervisor, Mr. Muhammad Azeem, who helped me throughout the process. He inspired me to give his support on all matters related to this project, and as a result, We achieved the finest possible outcome. Above all, his invaluable and rigorous monitoring at every stage of the project inspired us in countless ways.

We also want to recognize our professors, who played a significant role in our success because they taught us what we know and enabled us to work and finish the project.

Revision History

Compiled By	Checked By	Date	Reason for Change	Version
Khushnood	Mr. Ali Raza	23 Feb 2022	Initial Version	1.0
Khushnood	Mr. Ali Raza	28 Feb 2022	Project Overview	1.2
Khushnood	Mr. Azeem	04 Mar 2022	Introduction Reviewed	1.1
Hammad	Mr. Azeem	14 Mar 2022	Application Reviewed	1.2
Hammad	Mr. Azeem	17 Mar 2022	Literature Reviewed	1.3
Hammad	Mr. Azeem	21 Mar 2022	System Description Reviewed	1.4
Khushnood	Mr. Azeem	02 Apr 2022	Proposed Solution Reviewed	1.5
Khushnood	Mr. Azeem	11 Apr 2022	Software Process model Reviewed	1.6
Khushnood	Mr. Azeem	20 Apr 2022	Sprint Limitation reviewed& References	1.7
Hammad	Mr. Azeem	25 Apr 2022	Functional/Non functional Requirement Revised	1.8
Hammad	Mr. Azeem	02 Jun 2022	Revised Use Cases	1.9
Hammad	Mr. Azeem	20 Sep 2022	User Interface Reviewed	2.3
Hammad	Mr. Azeem	28 Sep 2022	Revised UML Diagrams	2.3
Hammad	Mr. Azeem	02 Oct 2022	Test Plan discussed	2.3
Hammad	Mr. Azeem	06 Oct 2022	Unit Testing Reviewed	2.0
Hammad	Mr. Azeem	13 Oct 2022	Revised Unit Testing	2.1
Khushnood	Mr. Azeem	25 Oct 2022	Revised Mistakes	2.2
Khushnood	Mr. Azeem	31 Oct 2022	Integration Testing Revised	2.3
Khushnood	Mr. Azeem	02 Nov 2022	System Testing Reviewed	2.4
Hammad	Mr. Azeem	08 Nov 2022	Revised Grammar Mistakes	2.5
Hammad	Mr. Azeem	16 Nov 2022	Conclusion/Future Work Discussed	2.6
Hammad	Mr. Azeem	25 Nov 2022	Future Work Reviewed	2.7
Khushnood	Mr. Azeem	02 Dec 2022	User Table Reviewed	2.8
Hammad	Mr. Azeem	09 Dec 2022	UML Diagram Revised	2.9
Khushnood	Mr. Azeem	16 Dec 2022	User Manual Reviewed	3.0
Hammad	Mr. Azeem	22 Dec 2022	User Manual Reviewed	3.1
Khushnood	Mr. Azeem	02 Jan 2023	Revised User Manual	3.2
Hammad	Mr. Azeem	09 Jan 2023	Revised Use cases	3.3

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Proposed game is a desktop application which uses windows operating system and run on computers and laptops. The game contains different environments each with unique puzzles and obstacles players can work together to get through these levels .The purpose of this game is solely entertainment and which often test the player's IQ, coordination and problem solving skills.It incorporates many sub-types that have the shared trait of zeroing in on the activities of the person utilizing a props to pass the stage. Usually these props can include any kind of boxes, wires, chains and many other objects to drag pick or climb onto. Most commonly, goal is to find the exit of these exciting levels by solving puzzles with nothing but you and your in game partners. This game intends to improve brain functioning such as cognitive abilities, coordination, team work and other learning skills. This game will have positive effects on brain functioning of all kind of age group, but without the violence and potentially negative effects unlike other games.

1.1 Product Purpose

Regardless of the pandemic which caused such a lot of monetary insecurity and emergency that affecting the entire world, the investigators after research inferred that the game business has developed at a pace of 57% shockingly. Indeed, even as these words are being composed millions of individuals use their PCs. The justification for this development might be legitimized by the way that the game business can draw in numerous clients with various preferences be it a youngster or a grown-up.The proposed game is being made by keeping all these things in mind and making it interesting so people with all group of ages can play it.

The proposed game is a Third-person Perspective game created using Unity game engine. It is developed for the windows operating system. In this game, the player has the ability to stick to the objects present in the environment and use them cooperatively with the help of other team members to walk through different levels. The game is designed to have different maps and levels, with many missions for the player to complete either solo or with a team. The ultimate goal is to find the exit of these exciting levels by solving puzzles and overcoming obstacles. Local co-op for two players and online multiplayer with up to five friends is also available.

Gameplay offers a unique and challenging experience. The player must use their sticking ability in creative ways to overcome obstacles and defeat enemies. This could include sticking to walls and ceilings to traverse dangerous areas or using their powers to attack enemies. The game also includes various power-ups that can enhance the player's abilities and make the journey through the levels even more exciting.

In addition to the main campaign, the product also offers a competitive multiplayer mode where players can compete against each other in various challenges and mini-games. This adds an extra level of replay value to the game and allows players to test their skills against their friends or other players online.

The proposed game is a fun and engaging game that is sure to provide hours of entertainment for players of all ages. The combination of unique gameplay mechanics, challenging levels, and multiplayer features make it a standout game in the industry. It is developed with the intention of appealing to a wide range of players, with the hope that it will become a beloved and enduring title.

1.2 Product Scope

The importance of a product scope lies in its ability to help guide the development process and ensure that the final product meets the needs and expectations of the intended users. A clear and well-defined scope helps to minimize scope creep, which is the tendency for a project to grow beyond its original scope. This can lead to delays, budget overruns, and other problems. By defining the scope up front, the development team can stay focused on the specific goals and objectives of the project, which helps to ensure that the final product is delivered on time and within budget [1].

The product scope for the proposed game includes a range of features and requirements designed to provide a fun and engaging gameplay experience for players. These features include third-person perspective gameplay, a sticking mechanic that allows the player to adhere to objects in the environment, cooperative multiplayer for two players and up to five online friends, a variety of maps and levels with different themes and challenges, solo and cooperative missions to progress through the game, puzzle-solving gameplay that challenges the player's problem-solving skills, a cute and colorful art style, intuitive controls, and smooth gameplay. The scope also includes replay value through a wide range of levels and missions, providing players with plenty of content to keep them entertained. These features combine to create a unique and engaging gameplay experience that will appeal to players of all ages and skill levels.

Module:Basic Gameplay

- Develop basic gameplay mechanics
- Movement, jumping, and object grabbing mechanics implemented and tested, simple level for testing purposes created.
- Player health and damage system implemented
- Basic enemy AI and combat mechanics implemented
- Player character design finalized and implemented
- Basic sound effects and music added

Module: Solo Play

- A variety of levels and missions that challenge the player to solve puzzles and progress through the game.
- Levels designed to be played alone.
- Range of challenging levels with different themes and obstacles.

- Puzzle-solving gameplay that challenges the player's problem-solving skills.
- Attractive and colorful art style, appealing to players of all ages.
- High level of replay value.

Module: Cooperative Play

- Cooperative multiplayer for up to 2 players and up to 5 online friends
- Range of levels and missions that challenge players to work together to overcome obstacles and reach their goals.
- Innovative sticking mechanic that allows players to use the environment and objects around them in creative ways.
- Attractive and colorful art style.
- Smooth gameplay.
- High replay value with a wide range of levels and missions to tackle.

Module: Art and Design

- Create and implement art assets
- Design and create character models for the game
- Design and implement environment assets
- Add particle effects to the game
- Develop cohesive art style for the game

Module: User Interface

- Develop user interface
- Menus and controls implemented, HUD designed and tested, smooth and intuitive user experience created.

Module: Sound and Music

- Create and implement sound effects and background music.
- Develop a library of sound effects and background music that match the game's theme and ambiance. Implement sound effects and background music in the appropriate places throughout the game. Ensure that the sound design enhances the overall gameplay experience and adds to the game's atmosphere.

Module: Polishing and Optimization

- Perform final testing and optimization
- Final testing performed, game optimized to run smoothly on various platforms, additional features added and tweaks made to create a polished final product.

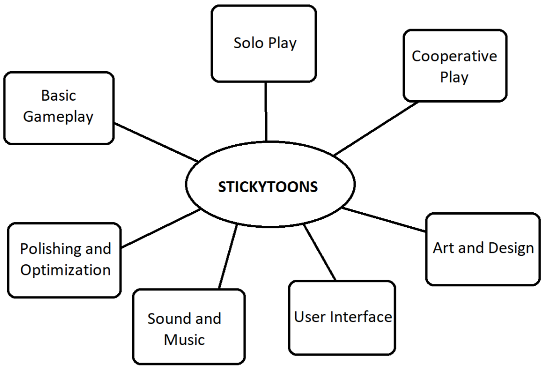


Figure 1.1: Core Functionality of the proposed game

1.3 Objectives

The objective of this project is to plan and execute a three-dimensional game using C# programming language with the help of Unity3D software. The project aims to develop a complete game level along with documentation that includes everything that should be available in a three-dimensional puzzle game experience. The purpose of this game is to entertain and sharpen the player's senses. It is designed to be easy and accessible for players of all ages without requiring any instructions or foreplay [2].

The importance of having clear objectives for proposed game cannot be overstated. It will provide a sense of purpose and direction for the efforts and resources being invested in the project. Additionally, it will help to ensure that the resources are being utilized effectively and efficiently.

Once completed, the proposed game will have various features such as puzzle-solving gameplay and obstacle courses to progress through levels. The physics-based controls will enable players to interact with the environment and objects in the game, and use their creativity and problem-solving skills to figure out the best way to reach the end of each level. The game will also offer open-ended environments with hidden secrets and collectibles to discover, and players can choose to play alone or with friends in local or online multiplayer modes.

The proposed game will require players to use their character's unique abilities, such as the ability to grab and move objects, to overcome challenges and solve puzzles. Collaborating with other players to complete levels and achieve goals will be another exciting feature of the game. Players will be able to experiment with different approaches and strategies to solve puzzles and progress through the game, and they can also customize their character's appearance by unlocking new skins and costumes.

The proposed game's wobbly, physics-based movement will add a new dimension to the player's experience as they navigate through levels, and they will be able to explore various environments and discover new paths and hidden areas. Additionally, the game's grappling hook and different objects and tools such as boxes and explosives will enable players to solve puzzles and overcome challenges.

1.4 Intended Market of Product

A market analysis is a crucial part of a business plan, as it allows you to understand the market and target audience for your product. Here is a detailed market analysis summary for my proposed project:

The proposed game targets children aged between eight to sixteen who are avid gamers, with a focus on addressing social issues and self-protection. The game's design is tailored to suit the needs and interests of teenagers who spend a significant amount of time using handheld devices to play games. With a highly competitive video game market, the project differentiates itself with its innovative sticking mechanic, cooperative gameplay, and its unique social focus. A well-executed marketing strategy and a strong product have the potential to achieve solid sales and boost the country's income.

The gaming industry in Asia, particularly in Pakistan, is among the fastest-growing in the world. The project addresses global issues with a specific focus on Pakistan, resulting in a significant economic impact on Asian countries. With an estimated contribution of Asian gamers reaching eighty-two billion US dollars in 2022 and accounting for fifty-five percent of the global market, there is immense potential for growth in this industry.

Game Industry: The project's unique features and focus on social issues create a distinct identity, attracting interest from a vast population of gamers. As the game is developed with the terrain of Asian countries in mind, it has a significant impact on Asian society, making it a valuable asset to the rapidly growing gaming industry. By marketing the game globally, developers can leverage positive feedback to enhance its popularity beyond Pakistan, further solidifying its place in the industry.

Revenue Industry: From a revenue perspective, selling the game abroad can generate substantial income for the country, thus increasing Pakistan's per capita income. As a third-world country with a per capita income below regional and global averages, the gaming industry offers a promising opportunity to improve the nation's financial situation. With the gaming industry being the largest generator of revenue worldwide, having a revenue of two hundred and three billion in 2020, and expected to grow at a CAGR of thirteen percent, the project holds the potential to make a significant positive impact on Pakistan's economy.

1.5 Intended Users of the Product

The intended user for the product is broad and includes players of all ages who enjoy puzzle-solving and cooperative gameplay. This includes casual gamers, hardcore gamers, puzzle fans, and cooperative gamers. In addition to these players, there are also several stakeholder groups who will be directly or indirectly customers of the proposed game. These stakeholders include:

Players: As a Player, you will receive in game rewards at success of the product. The proposed game is a game with a unique and engaging gameplay mechanic, making it a standout title in the crowded video game market. Its cute and colorful art style, challenging levels, and multiplayer features make it a game that will appeal to a wide range of players, increasing its potential sales. Its accessibility and replay value make it a game that players will keep coming back to, potentially driving repeat sales for your business. By stocking the product in your store, you can offer your customers a unique and entertaining gaming experience that they won't find elsewhere.

Distribution platforms: The proposed game is a multi-platform game, which means it is available on a variety of distribution platforms to reach a wider audience. The game can be played on desktops, laptops, and mobile devices such as tablets and smartphones. One of the most popular distribution platforms for the proposed game is Steam, a digital distribution platform that offers a vast collection of games to its users. Steam is known for its easy-to-use interface and community features, which allow players to connect with others and share their experiences.

Marketing partners: The proposed game is looking for marketing partnerships with companies and organizations that share our passion for gaming and creativity. We believe that our game can be a great addition to the portfolio of any company that targets the gaming market. As a marketing partner of the proposed game, you will have access to our marketing materials, including game trailers, screenshots, and promotional artwork. You will also receive support from our team to create custom marketing campaigns that fit your brand's voice and target audience.

Casual Gamers: The game is the perfect for players who enjoy casual gaming sessions. Its short levels and easy-to-learn mechanics make it perfect for quick play sessions, even if you don't have a lot of time to devote to gaming. Its cute and colorful art style is sure to appeal to casual gamers of all ages, and its challenging but not overwhelming gameplay makes it a great game for players of all skill levels. Whether you're a seasoned gamer or just starting out, you'll find something to like in the product. Its intuitive controls and smooth gameplay make it a joy to play, whether you're playing solo or with friends. Its replay value is high, with plenty of levels and missions to keep you entertained. The product is the perfect game to unwind with after a long day.

Hardcore Gamers: It is a must-play for hardcore gamers who enjoy a challenging and rewarding gameplay experience. Its difficult levels and competitive multiplayer mode provide a high level of depth and replay value, making it a great choice for players who enjoy a good challenge. Its innovative sticking mechanic and creative level design offer a refreshing twist on traditional puzzle-platformer gameplay, making it a standout title in the crowded video game market. Whether you're playing solo or with friends, the product is sure to provide hours of entertainment for hardcore gamers. Its intuitive controls and smooth gameplay make it a joy to play, even at high skill levels. With its wide range of levels and missions, the product offers a high level of replay value for hardcore gamers who want to keep coming back for more. Its cute and colorful art style adds an extra layer of charm to the gameplay, making the product a game that hardcore gamers of all ages can enjoy.

Chapter 2

Background and Literature Review

The reason for a writing Literature review is to comprehend existing exploration and discussions significant to a specific subject or area of study and to introduce this information as a composed report. Directing a writing survey will assist you with building information in your field. You will find out about significant ideas, research strategies and exploratory procedures utilized in your field. You will likewise acquire understanding into how specialists apply the ideas you learn in your unit to certifiable issues. One more incredible advantage of writing audits is that by perusing you will acquire a superior comprehension of how examination results are introduced and talked about in your specific discipline. Assuming you focus on what you read and attempt to accomplish a comparative style, you will climb to a higher level recorded as a hard copy for your discipline [3].

In the proposed game, conducting a thorough literature review is essential to analyze and evaluate existing games in the market. The background study involves researching and examining relevant ideas that can be used for comparison in the project. By identifying gaps and weaknesses in the existing games, the literature review helps to fill those gaps and address conflicts that may have arisen in previous products. It also sheds light on the problems and failures of previous games, providing valuable insights for the development of the proposed game. The literature review in proposed game serves as a foundation for understanding the current state of the gaming industry and informs the decision-making process to create a game that stands out in terms of efficiency, gameplay, and user experience.

2.1 Existing System Description

The gaming industry offers a wide range of games that cater to various audiences and preferences. However, there are only a few games that truly emphasize cooperative puzzle-solving and non-violent gameplay. It is essential to analyze the existing games in the market, focusing on their strengths and weaknesses, to understand the gap that our proposed game can fill. By examining these existing systems, we can identify opportunities to create a unique and engaging gaming experience that appeals to a broader audience, including children and adults looking for non-violent entertainment that challenges their cognitive abilities and teamwork skills [4].

The gaming landscape was analyzed to identify gaps in the market for the proposed game. A comprehensive analysis of existing games, including their strengths and weaknesses, was conducted to understand the potential opportunity that the proposed game could fill. Through this analysis, opportunities were identified to create a unique and engaging gaming experience that appeals to a broader audience, including children and adults seeking non-violent entertainment that challenges their cognitive abilities and teamwork skills. The proposed game has been designed to provide a refreshing and innovative gaming experience that stands out in the market by offering cooperative puzzle-solving gameplay that promotes teamwork, critical thinking, and a fun and engaging experience for players of all ages.

Human: Fall Flat [5] is an engaging, three-dimensional puzzle-solving game developed by No Brakes Games, an independent game development studio known for its innovative and creative approach to game design. Featuring a third-person perspective, the game immerses players in a unique gameplay experience. In this game, players are tasked with navigating through diverse maps and solving a variety of puzzles to reach the level's finish point. This dynamic gaming experience offers both single-player and multiplayer modes, making it accessible and enjoyable for various types of gamers. The game's captivating environments, creative puzzles, and cooperative gameplay elements provide endless entertainment, while challenging players to think strategically and work together to overcome obstacles. With its combination of innovative mechanics, captivating level design, and seamless integration of single-player and multiplayer gameplay, Human: Fall Flat, created by the talented team at No Brakes Games, offers an exceptional gaming experience that appeals to a broad range of players.The features for Human fall flat are listed below.

- Hardware friendly.
- Smoother game-play(with FPS cap of 60frames/second).
- Customizable controls/key-binds.
- Multiplayer.
- Customizable graphics.

The limitations of the Human fall flat are listed below.

- Repetitive puzzles.
- Bad graphics and textures.
- Camera shaking (which appears to be tied to the movement of the character's head).
- Bug and glitches (not interacting enough with the community).
- Character stutters.

Gang Beasts [6] is a highly engaging, physics-based multiplayer party game where players find themselves in wild, chaotic brawls as they strive to knock each other off a variety of hazardous stages. This captivating game has been skillfully developed by the creative minds at Boneloaf and expertly published by Double Fine Presents. Gang Beasts showcases an array of quirky, gelatinous characters that come to life with unique physics-based controls, creating an unpredictable and entertaining gaming experience in every match. The core concept of the game revolves around utilizing the character's distinct physical abilities, such as grabbing, punching, and lifting, to effectively knock opponents out before ultimately hurling them off the stage to secure victory.

Gang Beasts has managed to capture the hearts of players with its vibrant, colorful art style and whimsical, slapstick humor. The game offers a diverse selection of stages, each presenting its own set of unique challenges, environmental hazards, and visual appeal, ensuring that no two matches feel the same. Players can further enhance their experience by customizing their characters with a variety of hilarious costumes and accessories, adding a touch of personal flair to each brawl.The features of the Gang Beasts are listed below.

- Many levels (small).
- Skins.
- In game purchases.
- Good physics.
- Interesting Animations.

The limitations of Gang Beasts are listed below.

- Small maps.
- Boring single-player mode.
- Cant play locally(LAN).
- Not optimised.
- Crashes alot.

Crab Game [7] is an exhilarating, first-person perspective, multiplayer three dimensional game that takes its inspiration from the widely acclaimed TV show ”Squid Game.” Crab Game thrusts players into a high-stakes, life-or-death competition where only the most cunning and agile can survive and claim the ultimate prize. In order to emerge victorious, players must expertly navigate a series of increasingly challenging levels, each featuring its own distinctive mini-game designed to test their dexterity, problem-solving abilities, and quick thinking.

The allure of Crab Game lies in its captivating premise and the intense atmosphere it creates. Players are pitted against one another in a desperate struggle for survival, forcing them to adapt quickly to their surroundings and devise creative strategies to outwit their opponents. The game's fast-paced, unpredictable nature ensures that no two matches are ever the same, keeping players on the edge of their seats and constantly craving more.The features of the Crab Game are listed below.

- Up to three5 players.
- 28 mini maps.
- 9 game modes.
- Number of servers.
- Updates are regular.

The limitations of the Crab Game are listed below.

- Low resolution textures
- Game not optimized
- Alot of bug abuse.
- No difficulty level.
- Voice chat isuues.

Fall Guys [8] is an incredibly engaging multiplayer battle royale game that immerses players in a series of daring obstacle courses, with the goal of reaching the end of each level. The brainchild of developers Mediatonic and publishers Devolver Digital, Fall Guys has won the hearts of gamers with its lighthearted nature, addictive gameplay, vivid graphics, and physics-based mechanics, all set within the context of a thrilling elimination-style competition. Players must tackle a variety of obstacles, some of which are explicitly designed to eliminate them from the game, while most of the courses challenge participants to outpace their rivals and cross the finish line in order to advance to subsequent rounds. The game continues until only one player remains standing, ultimately claiming the title of winner.

To further enhance the gameplay experience, Fall Guys boasts an ever-evolving set of levels, each presenting unique and often unpredictable challenges. This ensures that every match offers a fresh and exciting experience, pushing players to adapt to the constantly changing landscape. In addition, the game features a vast array of customizable character skins and outfits, enabling players to personalize their in-game appearance and express their individuality as they compete in the chaotic, high-energy races. Fall Guys' delightful blend of whimsy, skill-testing gameplay, and fierce competition has solidified its status as a beloved and enduring title in the gaming world.The features of the Fall Guys are listed below.

- Can be played with 60 players at a time.
- Competitive and cooperative.
- Many character customizations.
- Variety of maps.
- Optimised AI.

The limitations of the Fall Guys are listed below.

- No map selection.
- Regional multiplayer (lobby can only start if 60 players are staked).
- Poor Server optimization.
- System Requirement too high.
- Connection issues.

The Academy [9] is an enchanting, puzzle-based adventure game set within the intriguing and mysterious confines of a school teeming with secrets waiting to be uncovered. Developed by Pine Studio and published by Snapbreak, the game immerses players in a rich narrative that weaves together a vast array of challenging puzzles, a beautifully crafted world, and engaging character development.

As players explore the enigmatic environment of The Academy, they must employ their wits and problem-solving abilities to unravel the many riddles and conundrums that lie within its hallowed halls. Each puzzle is designed to test players' intellectual prowess and creativity, pushing them to think outside the box and uncover hidden connections. The captivating storyline, combined with the diverse set of puzzles, ensures that players remain fully engaged and invested in the game's unfolding mysteries. The features of The Academy are listed below.

- Good graphics.
- Majority of the puzzles are fun and interesting.
- Interest setting design.
- Verity of characters.
- Low spec computer support.

The limitations of The Academy are listed below.

- Some puzzles are too easy.
- Puzzles with unclear rules/objectives.
- Controls are difficult to master.
- No multiplayer.
- Texture glitches.

Omno [10] is a captivating third-person adventure game that transports players on an unforgettable journey through a wide range of breathtaking landscapes. Developed by Jonas Manke and published by StudioInkyfox, the game whisks players away to lush forests, sun-scorched wastelands, icy tundras, and even the lofty heights of a long-lost civilization nestled among the clouds.

Throughout their adventure, players will encounter a host of strange and wondrous creatures, uncover intriguing surprises, and perhaps even forge new friendships. Omno's immersive world is designed to be interactive, inviting players to fully engage with their surroundings and become deeply invested in the game's exploration and discovery aspects.

Omno presents players with an array of challenges in the form of puzzles, hidden secrets, and obstacles to overcome. Drawing from classic 3D puzzle-platformer gameplay, Omno expertly combines brain-teasing riddles with skill-based challenges, ensuring that players remain thoroughly entertained and engaged from start to finish.The features of Omno are listed below.

- Gorgeous atmosphere.
- Relaxing game play.
- Multiple abilities and mechanics.
- Variety of creature.
- Fidgeting feature.

The limitations of Omno are listed below.

- Short game play
- Missing voice lines
- Vague story
- No multiplayer
- Hi end specs requirement.

Risk of Rain [11] is a classic multiplayer action game that takes players on an exhilarating journey through numerous levels, each featuring unique dimensions and challenging scenarios. Developed by Hopoo Games and published by Chucklefish, this engaging game can be enjoyed solo or cooperatively with up to three friends, as players team up to battle a diverse array of enemies and strive to unlock valuable loot.

The game's core objective revolves around finding a way to escape the mysterious and perilous planet on which the players find themselves stranded. Risk of Rain seamlessly blends intense combat, strategic decision-making, and resource management, ensuring that every playthrough is a thrilling and distinct experience. With its captivating gameplay, diverse environments, and the opportunity for cooperative action, Risk of Rain remains a timeless favorite among fans of action-packed multiplayer games.The features of Risk of Rain are listed below.

- Beautiful soundtrack.
- Mesmerizing aesthetic.
- Skill based gameplay.
- Unique, diverse characters.
- Easy to use GUI.

The limitations of Risk of Rain are listed below.

- Difficult when playing with friends.
- No customization.
- No puzzles.
- Unfair advantages to AI.
- Frame drops.

Paws and Soul [12] is a serene and introspective third-person perspective, three-dimensional adventure game centered around the theme of rebirth. In this captivating experience, players take on the role of a wolf exploring a mysterious forest, seeing the world through the eyes of an animal while uncovering the story of their past human life.

The gameplay focuses on exploration and the search for collectibles, as players seek out clues about their previous existence and discover bonus stories about other characters in the game. The enchanting forest setting offers breathtaking views, and players can take comfort in knowing they are entirely safe as they roam the landscape, letting their curiosity guide their journey.

As players delve deeper into the game, they will begin to understand the connection between the lives of the wolf and the human, ultimately unraveling the enigmatic secret of reincarnation. Paws and Soul provides a tranquil, immersive gaming experience that encourages contemplation and self-discovery, allowing players to reflect on the interconnectedness of life and the beauty of nature.The features of Paws and Soul are listed below.

- Decent graphics.
- Theme track is good.
- Good story line.

- Game play is fine.
- Interactive props.

The limitations of Paws and Soul are listed below.

- Bad mechanics.
- Boring.
- No multiplayer.
- Only one character.
- Optimization issues.

Etherborn [13] is a polished jump inside the gravity-puzzle type; a natural puzzle plat previous based on investigating and understanding gravity-moving designs. Player a voiceless being that has essentially been conceived solidly into a worldwide where an immaterial voice anticipates your appearance. As Player most memorable contemplations arise, you understand your process has essentially began. Player should achieve this ethereal voice to totally perceive players individual presence. In this worldwide, the laws of physical science act in a manner on the method for bowing your creative mind. Player experience gravity opposite to the floor player's remaining on, considering creative investigation prospects in conditions circumspectly created as route based thoroughly confounds. Visit through exquisite and intriguing landscapes where each level is a normal riddle which player need to find, control furthermore, shift players gravity on so it will interpret the way forward

The features of Etherborn are listed below.

- Gravity-moving platforming and investigation.
- Route based puzzles.
- Surreal levels in an atmosphere.
- Dynamic and captivating original soundtrack.
- Custom controller support.

The limitations of Etherborn are listed below.

- Puzzles are too easy .
- Game is too short.
- Terrible game design.
- No multiplayer.
- Glitched light effects.

The proposed solution has been compared to ten existing solutions developed in the past ten years, and it is evident that the new features added in this game are not present in the aforementioned games

Table 2.1 is a comparison of existing games that share similarities with the current project is presented to highlight the different types of functionalities included in the proposed solution. This comparison helps to demonstrate the differences between the existing and proposed solutions, and identify potential areas for improvement in the new game. To facilitate this comparison, a table can be used to display the relevant features of the different games. By analyzing the strengths and weaknesses of the existing solutions, the proposed solution can be further refined to make it more efficient and enjoyable.

Table 2.1: Applications Comparison

	Applications									
Features	Humans fall flat [5]	Gang Beasts [6]	Crab Game [7]	Fall Guys [8]	The Academy [9]	Omno[10]	Risk of rain [11]	Paws and Soul [12]	Etherborn [13]	Proposed Game
Customizable key binds	✓	✗	✗	✗	✓	✓	✓	✗	✗	✓
Multiplayer	✓	✓	✓	✓	✗	✗	✓	✗	✗	✓
Player Costumes	✓	✓	✓	✓	✓	✗	✓	✗	✗	✓
Checkpoints	✗	✗	✗	✗	✓	✓	✓	✗	✓	✓
Unique puzzles	✗	✗	✓	✓	✓	✓	✗	✓	✓	✓

2.2 Future System Usage Analysis

It is important to ensure that proposed game is a well-designed and engaging game that appeals to players. This can be achieved through thorough testing and iteration during the development process, as well as by gathering feedback from players and using it to improve the game.

It is also important to understand the competitive landscape in the market and how proposed game compares to other games in terms of features, gameplay, and appeal. This can help identify opportunities to differentiate the game and position it as a unique and attractive offering in the market.

It is essential to stay up to date with trends and changes in the market and consumer preferences, and to adapt the game accordingly. For example, if the market is shifting towards mobile gaming, it may be worthwhile to consider developing a mobile version of proposed game.

In order to reach a wide audience and drive adoption of the game, it will be important to invest in marketing and promotion efforts. This could include tactics such as advertising, social media marketing, content marketing, and more.

Providing excellent customer support and regularly releasing updates and new content for the game can help retain players and encourage them to continue playing and recommending the game to others.

2.3 Problem Statement / Limitations

Video games have been found to offer interactive experiences that aid in children's brain development and can positively impact society by addressing specific issues. There is a growing demand for high-quality games that can modify society and teach important lessons to players. While the game industry already holds value in our society, it is essential to continue exploring new ways to encourage morality and teach valuable lessons to young children. Our proposed game aims to engage players and provide depth and immersion while conveying a moral lesson that addresses the prevalent social issue of kidnapping. According to the International Game Developers Association (IGDA), Pakistan's game industry generates approximately $25 million USD annually, while the global game industry generates billions of dollars. The industry is projected to generate around $25 billion USD by 2025.

Proposed game developers will incorporate moral lessons in their creations as games that do not address social issues often lack substance. Proposed game teaches valuable skills on how to survive in kidnapping situations while providing entertainment. Developing video games that address social issues can benefit society, even when played purely for entertainment purposes.

While several survival games are available in the market, not all game developers address social issues or include moral lessons in their games. Proposed game is unique in effectively addressing a social issue and imparting a moral lesson. It is vital for game developers to consider addressing social issues and incorporating moral lessons in their games to benefit society, even if their primary purpose is entertainment.

2.4 Proposed Solution

The proposed solution to the problem of the limited availability of locally developed games in Pakistan is to develop a new game that is specifically designed to appeal to Pakistani gamers. This game will be developed using the latest technologies and design principles in order to create an engaging and enjoyable experience for players.

To ensure the success of the game and address the problem of the limited availability of locally developed games in Pakistan, the project team will utilize resources such as the Unity asset store and open source sites to assist in the development and programming of the game. The team will also conduct user testing and gather feedback from players to evaluate the game's effectiveness in meeting the needs and preferences of Pakistani gamers. By developing a game that is specifically designed to appeal to Pakistani gamers and providing a high-quality and enjoyable experience, the project team hopes to contribute to the growth of the local game development industry and provide a source of entertainment for Pakistani gamers that is competitive with games being developed by larger studios.

By implementing the strategies given in figure 1.1 and developing a game that meets the needs and preferences of Pakistani gamers, the project team hopes to make a significant contribution to the growth and development of the local game development industry, while also providing a high-quality and enjoyable source of entertainment for Pakistani gamers.

2.5 Software Process Model

There are several reasons why it is important to use a process model in the development of software. A process model helps to provide a clear and structured approach to software development, which can help to ensure that the project is completed efficiently and effectively. It also helps to define and document the various phases of the development process, establish clear roles and responsibilities for team members, and provide a way to measure progress and track the status of the project. In addition, a process model helps to establish clear communication channels and protocols, which can help to ensure that all team members are on the same page and working towards the same goals. Overall, using a process model in the development of software can help to ensure that the project is completed successfully and that all stakeholders are aligned in their efforts [15].

The proposed game will also benefit from the use of a process model in its development. By using a structured approach to game development, the project team can ensure that all aspects of the game are thoroughly planned and executed, resulting in a high-quality final product. This will involve defining and documenting the various phases of the development process, such as game design, programming, art and sound development, and user testing. Using a process model can also help the project team to measure progress and track the status of the project. By establishing clear milestones and metrics, the team can monitor their progress and make adjustments as needed to ensure that the project stays on track. This can help to ensure that the game is completed on time and within budget.

2.5.1 Introduction

The agile model in Figure 2.1 helps to manage the project's scope, schedule, and budget effectively by breaking down the project into smaller, manageable tasks. This can help to reduce the complexity and ensure that the project stays on track. Agile's emphasis on collaboration and feedback can facilitate communication between team members, stakeholders, and users, helping to ensure that everyone is aligned on the project's goals and can work together to achieve them. Additionally, Agile's ability to address changes and challenges quickly can be beneficial in game development, where new ideas and feedback may arise frequently. Overall, using the Agile model in the development of the proposed game can help to ensure that the game is developed efficiently, effectively, and with high quality.

2.5.2 Justification

The Agile development model is particularly well-suited for developing the proposed game, due to several advantages that align with game development requirements. This Agile model facilitates rapid progress and iteration, which is essential for delivering a successful game. With game's emphasis on player engagement, the use of short development cycles enables quick delivery of working software and feedback from the team and players, leading to constant improvement and iteration. This ensures that game meets the needs of its target audience, providing an enjoyable and immersive gaming experience.This Agile model is adaptable and flexible, which is crucial for a game like this that requires constant innovation and creativity. With the Agile model, develop-

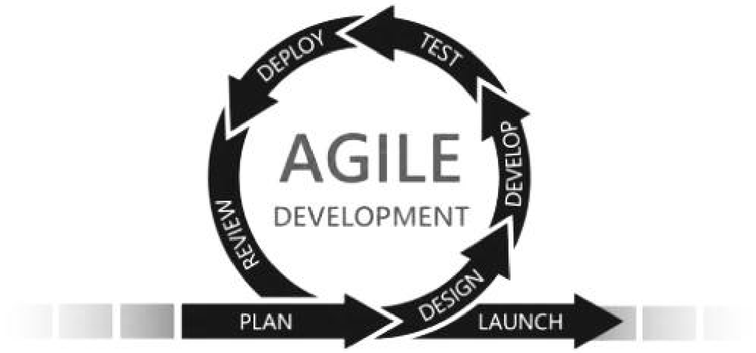


Figure 2.1: Agile Model [17]

ers can quickly adapt to changing requirements and priorities during the development process, enabling them to respond to new ideas and opportunities. This ensures that game remains fresh and engaging, meeting the evolving needs of its audience.This agile model promotes collaboration and communication among team members, which is critical for the successful development of proposed game. Since game development involves multiple disciplines such as art, design, programming, and audio, regular communication and collaboration among team members are necessary to share ideas and feedback. The Agile model ensures that everyone works together towards a common goal, resulting in a cohesive game that meets the needs of its target audience. The combination of these Agile model advantages with game's focus on player engagement and innovation makes it an ideal fit for the development of this game.

2.5.3 Steps

The Agile sprint model is a popular framework for software development that involves dividing a project into shorter development cycles, or sprints, each typically lasting between one to four weeks. For the proposed game game development project, we have identified seven sprints, each with a duration of three weeks, that will be used to deliver the game. The first sprint will focus on defining the scope and objectives of the project, while the second sprint will involve gathering and analyzing requirements for the game. The third sprint will be dedicated to designing the technical solution, including architecture, algorithms, and data structures. In the fourth sprint, the development team will start implementing the technical solution by writing code, creating unit tests, and integrating third-party libraries or frameworks. The fifth sprint will focus on testing the game's functionality and fixing any issues or defects identified. The sixth sprint will be allocated for polishing and adding additional features based on feedback from stakeholders and playtesting, and the final sprint will focus on the final polishing and release of the game. By using the Agile sprint model, the development team will be able to deliver a high-quality game that meets the evolving needs of the target audience within a set timeline.

Sprint 1: Basic Gameplay

- **Objective:** Develop basic gameplay mechanics
- **Deliverables:** Movement, jumping, and object grabbing mechanics implemented

and tested, simple level for testing purposes created.

Sprint 2: Solo Play

- **Objective:** Develop solo play features
- **Deliverables:** Level design with increasing difficulty implemented, puzzle-solving gameplay mechanics added, environment interaction mechanics tested.

Sprint 3: Cooperative Play

- **Objective:** Develop cooperative play features
- **Deliverables:** Online multiplayer mode implemented, level design for co-op play added, character synchronization tested, levels requiring collaboration between players created.

Sprint 4: Art and Design

- **Objective:** Create and implement art assets
- **Deliverables:** Character models created and added to the game, environment assets designed and implemented, particle effects added, cohesive art style created.

Sprint 5: User Interface

- **Objective:** Develop user interface
- **Deliverables:** Menus and controls implemented, HUD designed and tested, smooth and intuitive user experience created.

Sprint 6: Sound and Music

- **Objective:** Create and implement sound effects and background music
- **Deliverables:** Sound effects and background music created, sound design made cohesive and enhancing gameplay atmosphere.

Sprint 7: Polishing and Optimization

- **Objective:** Perform final testing and optimization
- **Deliverables:** Final testing performed, game optimized to run smoothly on various platforms, additional features added and tweaks made to create a polished final product.

Chapter 3

Software Requirements Specification

Software requirements are the specific needs, capabilities, and constraints that a software product must fulfill in order to be successful. These requirements might include functional requirements, such as the specific features or functionality that the software should provide, or non-functional requirements, such as performance, security, or usability. The importance of software requirements is that they serve as the foundation for the design and development of a software product, and ensure that the final product meets the needs and expectations of the users or customers. Without clear and well-defined requirements, it is difficult to create a high-quality software product that meets the needs of the target audience [18].

In the development of proposed game, software requirements play a vital role in ensuring a successful game. These requirements refer to the specific needs, capabilities, and constraints that the game must fulfill to meet the needs of its target audience. Functional requirements for proposed game refer to the game's features and functionality, including gameplay mechanics, level design, and character interactions. Non-functional requirements, on the other hand, relate to performance, security, and usability, such as the game's loading time, graphics quality, and user interface design. By clearly defining and prioritizing both functional and non-functional requirements, the development team can ensure that proposed game meets the expectations of its players and delivers a high-quality gaming experience.

Functional requirements in game development may include player movement, character selection, and puzzle-solving gameplay. Non-functional requirements may include a responsive user interface, optimized graphics for different devices, and compatibility with different operating systems. Clearly defining these requirements is critical to the design and development of the game, as it ensures that the final product meets the needs and expectations of the users or players.

3.1 Introduction

In the proposed game, the Software Requirements Specifications (SRS) document is a crucial tool in ensuring that proposed game adheres to its original vision and meets the needs of its intended users. The SRS document allows developers to communicate the specific requirements of the game to the developing team and tester, while also enabling players to confirm, clarify, and expand on the content over time. This document provides a comprehensive view of the project and serves as an essential component of proposed game development. It outlines the functional and non-functional requirements of the game, including its target audience, game mechanics, visuals, audio, and other critical features. The SRS document will guide the development process and ensure that proposed game meets the expectations of its users.

3.1.1 Document Scope

The scope of the document is centered around the development of a non-violent puzzle-solving game for the Windows operating system. The game is intended to enhance critical thinking and problem-solving capabilities and appeal to gamers of all ages. Players will be challenged to navigate through various mazes and puzzles to reach the end goal. The game will consist of a variety of maps, each with unique challenges and obstacles for players to overcome. A single-player mode will be available for users to play the game on their own, as well as a cooperative multiplayer mode for friends to work together to solve puzzles and complete levels. The multiplayer mode will allow for up to four players to join the game simultaneously, providing an opportunity for collaborative problem-solving and teamwork. Engaging and colorful graphics, immersive audio and visual effects, and a variety of characters and animations will add to the overall appeal of the game and make it more enjoyable to play. To enhance user experience, the game will be designed with an intuitive user interface, making it easy for players to navigate through the game's different levels and menus. Additionally, a tutorial mode will provide new players with a clear and concise overview of the game's mechanics and controls. The scope of the document is focused on creating an entertaining and engaging puzzle-solving game for Windows OS, accessible to gamers of all ages, while also providing opportunities for collaborative gameplay and enhancing critical thinking skills.

3.1.2 Audience

The document is designed to be used by a variety of stakeholders involved in the software development process, including testers, designers, maintainers, and project managers. However, it is important to note that the intended audience should have a basic knowledge of software development and programming or have experience with the games available in the market. The requirements outlined in the SRS document are crucial for understanding the purpose and scope of the game. It is recommended that the document be read from front to back in order to fully comprehend the project and its requirements.

- **Testers:** Testers can use the SRS document to verify if the final product is working as described in the document. They can perform various tests on the game, such as functional and non-functional tests, to ensure that the game meets the specified requirements.
- **Designers:** Designers can leverage the SRS document to model the game's sprites, maps, and characters according to the game's theme. The document can provide them with the necessary information on the game's art style, design, and aesthetics.
- **Maintainers:** Maintainers can refer to the SRS document if the game is not working as described in the document, which might indicate bugs that need to be fixed. The document can help the maintainers to understand the game's functionality and design to locate and fix the bugs.
- **Project managers:** Project managerscan utilize the SRS document to get a comprehensive understanding of the game project's objectives, deliverables, and progress. They can refer to the document to ensure that the game is being developed according to the defined requirements and timelines. The SRS document can also assist project managers in keeping track of the development process and ensuring that the project is on track.

3.2 Functional Requirements

Functional requirements are the specific actions or tasks that a software system is required to perform in order to meet the needs of the users or customers. They describe the functions that the software must be able to perform, and may include items such as input and output data, user interfaces, security, performance, and other capabilities.Functional requirements are needed in order to clearly define the capabilities and behavior of a software system. Without a clear set of functional requirements, it is difficult to design and develop a software product that meets the needs and expectations of the users or customers. Functional requirements provide a basis for creating a detailed design for the software, and for verifying that the final product meets the specified requirements. They also help to ensure that the software is easy to use and provides a good user experience, by defining the required features and functionality in detail [19].

Table 3.1 shows that the proposed game, includes several important function requirements that allow players to interact with the system. Firstly, players will be able to select their desired character from the available options, giving them control over their gaming experience. They will also be able to move the character in different directions, providing them with freedom and control over their gameplay. Additionally, players can pause and resume the game as needed and exit the game at any time. Another exciting feature of the game is that players can select between different levels, each with unique challenges and obstacles. Moreover, players will be able to stick the character with its arms to different objects, providing them with a unique gameplay experience. To enhance the social aspect of the game, players can join different lobbies to play with others. Finally, the game will allow players to collide and interact with different objects, adding depth and complexity to the gameplay. These function requirements will be critical in ensuring an engaging and immersive experience for players.

Table 3.1: Functional Requirements

Requirement ID	**Description**
FR1	Players can Select the character from given options.
FR2	Player can move the character in different directions.
FR3	Player can pause/resume game.
FR4	Player can exit the game.
FR5	Player can select between different levels.
FR6	Player can Stick the character with its arms to different objects.
FR7	Player can join different lobbies to play.
FR7	Player can collide and interact with different objects.

3.3 Non-Functional Requirements

Non-functional requirements are requirements that specify how a system should behave and what qualities it should possess, but they do not describe any specific functionality. They are often referred to as ”constraints” or ”quality attributes” because they constrain the design of a system and dictate the quality of its performance. Non-functional requirements are important because they ensure that a system is usable, reliable, maintainable, and efficient, among other things. They are also important because they can have a significant impact on the overall user experience of a system. Some common examples of non-functional requirements include performance, security, scalability, usability, and maintainability [20].

In game development, non-functional requirements are essential for ensuring the success of the project. While the game may still operate without these requirements, the overall experience may be compromised. These non-functional requirements include aspects such as engaging gameplay, a compelling story, good performance, usability, responsiveness, and stability. For instance, in the case of the game proposed game, developers prioritize non-functional requirements such as maintaining game performance during gameplay and ensuring that the game is responsive and maintainable. By focusing on these non-functional requirements, game developers can enhance the player experience and create a game that is enjoyable and immersive.

3.3.1 Software Quality Attributes

In the development of a proposed game, software quality attributes are essential to ensure that the game functions smoothly and is both sustainable and reliable. Software designers and developers aim to maintain high standards of SQA to guarantee that the game meets the specifications of the client. The quality attributes of the game must ensure that it is safe to use, and that the framework used to play the game is dependable for users during gameplay. For instance, in the game proposed game, SQA measures will be implemented to ensure that the game runs smoothly and without errors, and that it is user-friendly and safe to play. The game will be thoroughly tested to guarantee that it meets the highest standards of software quality.

- **Usability:** The game should be easy to use and navigate for players of all ages. This includes intuitive controls and clear on-screen prompts.
- **Reliability:** The game should be reliable, with minimal downtime or bugs that impact the player's experience.
- **Compatibility:**The game should be compatible with a range of devices and operating systems, to ensure a wide audience can access it.

3.3.2 Performance Requirements

In the case of the proposed game, it is important to ensure that the game performs smoothly and without any lag or glitches, as these can negatively impact the player experience. Performance testing will be conducted to evaluate how the game performs under different scenarios, such as with varying levels of graphics quality or on different hardware configurations. Additionally, load testing will be performed to evaluate how the game handles high traffic or user load. By establishing clear performance requirements and conducting thorough testing, we can ensure that the proposed game provides a seamless and enjoyable gameplay experience for all players.

In the project game, it is important to measure the performance so the software performance can be evaluated based on the following characteristics:

- Users should be able to play in pairs of two in the game.
- Any response time should be no longer than five seconds.

- There shouldn't be more than a 4-second delay between pairs.
- It must be able to function under challenging circumstances, such as extreme heat or cold.
- Reliable connections.
- Rapid data transfer.

3.3.3 Safety Requirements

In developing the proposed game, safety requirements are considered to be of utmost importance. These requirements are necessary to ensure that the game is safe for players to use and does not pose any harm to them physically or emotionally. One of the safety requirements is to ensure that the game's content is appropriate for all ages, so it should not contain any violence, explicit language, or adult themes. Another safety requirement is to ensure that the game's user interface is easy to navigate, and that players are not confused or disoriented while playing the game. The game should also be designed in such a way that it does not cause any physical harm to the players, such as eyestrain or repetitive motion injuries. The safety of the players should be considered throughout the development process, from the design of the game to its implementation and testing. By implementing safety requirements, the proposed game can provide a fun and enjoyable experience for players while ensuring that their safety and well-being are not compromised.

3.3.4 Other Non-Functional Requirements

In the proposed game, the Software Requirements Specification (SRS) document outlines both functional and non-functional requirements. Functional requirements detail the features and capabilities of the game system that are essential to fulfill user needs, while non-functional requirements focus on the quality attributes that enhance the overall performance and reliability of the game. Although the game may still function without non-functional requirements, they are crucial in ensuring smooth gameplay and meeting user expectations during gameplay.

In the proposed game, it is important to measure the Non functional requirement so the software, So the non functional requirements can be evaluated based on the following characteristic:

- The game should have smooth and responsive gameplay, with minimal lag or bugs.
- The game should be compatible with a range of hardware and operating systems.
- The game should have an intuitive user interface and controls
- The game should be able to handle a large number of players and retain good performance.

3.4 Requirements Gathering Techniques Used

Requirements gathering techniques are used to identify and document the specific needs and expectations of stakeholders for a software system. These techniques help to ensure that the final product will meet the requirements of the end users and meet the overall goals of the project. Some common techniques include interviews, focus groups, surveys, and prototyping. The choice of technique will depend on the specific goals and constraints of the project, as well as the preferences and availability of the stakeholders. It is important to use a variety of techniques in order to gather a comprehensive and accurate understanding of the requirements [21].

Several requirements gathering techniques were utilized in the development of the game to ensure that all requirements were gathered effectively and accurately. The techniques used included interviews with potential users and stakeholders to understand their needs and expectations, surveys to collect data from a larger audience, focus groups to get in-depth feedback from a specific target group, and observation of similar games in the market to identify the key features that players expect. Additionally, brainstorming sessions were held with the development team to generate ideas and identify potential requirements that were missed during the initial analysis. All of these techniques were used in combination to ensure that the requirements were gathered comprehensively and accurately. By using a variety of techniques, the team was able to identify and prioritize the most critical requirements, leading to a successful development process that met the needs and expectations of the target audience.

3.4.1 Focus groups

Focus groups are a type of research method that involves gathering a small, diverse group of individuals together to discuss and share their thoughts, opinions, and insights on a particular topic or product. This technique is often used in market research to gather feedback and ideas from a representative sample of the target audience. Focus groups can be conducted in person or online, and typically involve a moderated discussion guided by a set of predetermined questions or prompts. The results of focus groups are often used to inform the development of new products or services, or to assess the effectiveness of marketing campaigns [22].

It was important to involve audience in the game development process. That's why the team used focus groups to gather insights and feedback on the games. Focus groups were a research method that involved bringing together a small group of players to discuss and share their thoughts on the game. By gathering feedback on the gameplay, art style, and overall experience, the team could make informed decisions about how to improve the game. To conduct a focus group, the team brought together players who had experience with similar games and asked them to play an early version of the game. This helped the team to identify what gamers liked and what could be improved. By incorporating this feedback into development process, the team ensured that game is tailored to the needs and preferences of target audience.

- The focus group could help identify the preferred art style for the game, including color schemes and character designs.
- The focus group could provide feedback on the game's mechanics, such as puzzle difficulty, level design, and controls.
- The focus group could suggest additional features or elements to be added to the game, such as multiplayer modes or mini-games.
- The focus group could help identify the target audience's preferences, such as age group, genre interests, and platform preferences.

- If the game has a story or narrative component, the focus group could provide feedback on the storyline, characters, and dialogue.
- The focus group could identify any bugs or glitches encountered during gameplay, providing valuable feedback to the development team.
- Based on their experience playing the game, the focus group could suggest improvements or changes to the game's design, mechanics, or user interface.

3.4.2 Brainstorming

While brainstorming sessions can certainly be helpful in generating new ideas, it is important to keep in mind that they cannot replace the essential process of gathering thorough requirements. By taking the time to gather all necessary information and utilizing techniques such as brainstorming, software developers can ensure that they are creating a product that meets the client's needs and achieves the desired outcome. Brainstorming can help the development team come up with innovative solutions to problems and identify potential roadblocks early on. This can ultimately lead to a more efficient development process and a better final product [31].

Developers of the proposed game often use prototypes to present their game to project supervisors, clients, and the general public to evaluate their level of acceptance in the market and suggest any necessary changes or additions to the product. By using prototyping, developers can create a game that meets the client's expectations and achieves the desired outcome. Like movies, video games require screen mock-ups to showcase different scenes and better demonstrate the game's nature. Therefore, game developers often use prototypes to present their games to project supervisors, clients, and the general public to evaluate their level of acceptance in the market and suggest any necessary changes or additions to the product.

- Clarity on the game mechanics: Through the brainstorming process, the development team may have gained a clearer understanding of how the game mechanics should work and how to make them engaging for players.
- Identification of potential issues: By exploring different ideas and solutions, the team may have identified potential issues with the game and how to address them early on in the development process.
- Improved collaboration: Brainstorming can help foster collaboration and communication among team members, ensuring that everyone is on the same page and working towards the same goals.
- Incorporation of creative ideas: Brainstorming can be a great way to generate new and innovative ideas for the game, leading to a more unique and interesting final product.
- Increased buy-in from team members: By involving team members in the brainstorming process and incorporating their ideas, the team is more likely to feel invested in the project and motivated to see it through to completion.
- Enhanced market acceptance: By incorporating feedback from focus groups and other potential players into the brainstorming process, the team may be able to identify ways to make the game more appealing and increase its chances of success in the market.

3.5 Time Frame

To ensure efficient product development, a schedule is established with a time frame that begins with the feasibility report and concludes with implementation. The software development team determines the estimated number of hours required to fulfill the product owner's requirements. It is essential that all modules are completed within the specified time frame to ensure timely delivery [23].

The table 3.2 describes that, in the proposed game, the developers are motivated by the time frame to work efficiently, resulting in higher productivity in a shorter period. The time frame is divided into eight sprints, each lasting thirty days, with specific tasks assigned to each sprint. These tasks include Game Concept, Character Design and Animation, Level Design and Terrain Creation, Game Mechanics and Programming, Audio and Visual Effects, Play Testing, and Final Polishing and Release. By dividing the development process into sprints and assigning specific tasks to each sprint, the team can efficiently manage their time and meet the project's goals within the given time frame.

Table 3.2: Time Frame

Phase	**Duration**
Sprint 1 (Basic Gameplay)	30 days
Sprint 2 (Solo Play)	10 days
Sprint 3 (Cooperative Play)	40 days
Sprint 4 (Art and design)	60 days
Sprint 5 (User Interface)	30 days
Sprint 6 (Sound and Music)	20 days
Sprint 7 (Polishing and optimization)	20 days

Chapter 4

Software Design Specification

SDS is a document that outlines the specific details of the software being developed, including its functionalities, user requirements, and design specifications. The SDS is an important tool for ensuring that the development process is on track and that the end product meets the desired requirements and specifications. It serves as a reference for the development team and helps to keep everyone on the same page throughout the development process. The SDS plays a critical role in the success of software development projects. It helps to ensure that the software meets the requirements of the end-users and that it is delivered on time and within budget. The document also helps to minimize the risk of project failure by providing a clear roadmap for development and ensuring that all stakeholders are on the same page [24].

The Software Design Specification is a crucial aspect of the proposed game as it outlines key features such as Character Selection, Character Movement, and Exit Game. The Use Case Diagram is equally important as it elaborates on the game's functional requirements. To ensure a comprehensive development process, a document is prepared containing all the necessary information for system development. The software design is created by designing and creating modules. Furthermore, the Use-Case Description is a methodology used to organize, identify, and clarify system requirements.

4.1 Entity-Relationship Diagram

In software development, an Entity-Relationship Diagram (ERD) is a graphical representation used to describe interrelated entities in a specific domain of knowledge. It consists of entity types that categorize the things of interest and relationships that can exist among entities. The ERD provides a visual representation of the structure and relationships between entities, allowing for a clear understanding of the data model and its organization. It is a fundamental tool used in database design and modeling to ensure a logical and efficient representation of the data in the software application being developed [25].

The proposed game does not require the use of an Entity-Relationship Diagram (ERD) due to the game's real-time data management and transient nature of game data. The game elements, such as scores and health, are expected to be cleared once the game session ends, and a new session begins. As a result, alternative data management techniques, such as in-memory data structures or game engine-specific data management systems, may be more suitable for the game's specific requirements. The decision not to use a traditional database and ERD in the game's development approach may reflect the game's unique gameplay and mechanics, where persistent data storage is not needed, and a lightweight and fast-paced gaming experience is desired.

4.2 Use-Case Diagram

A use case diagram is a visual representation of the interactions between actors and the functionality of a system. It is used to identify the primary functions and flows of a system, and to ensure that all necessary functions are included. Use case diagrams are important because they provide a high-level view of the system, allowing stakeholders to understand the overall functionality and to identify any potential issues or gaps. They are also useful for communication and planning, as they provide a clear and concise way to document and discuss the functional requirements of a system [25].

Particularly for the proposed game a use case diagram will allow developers to analyze the system and its different actors or roles. The use case diagram provides a clear illustration of the context of the game, whether as a whole or some of its components. This is particularly useful in modeling complex systems such as proposed game.

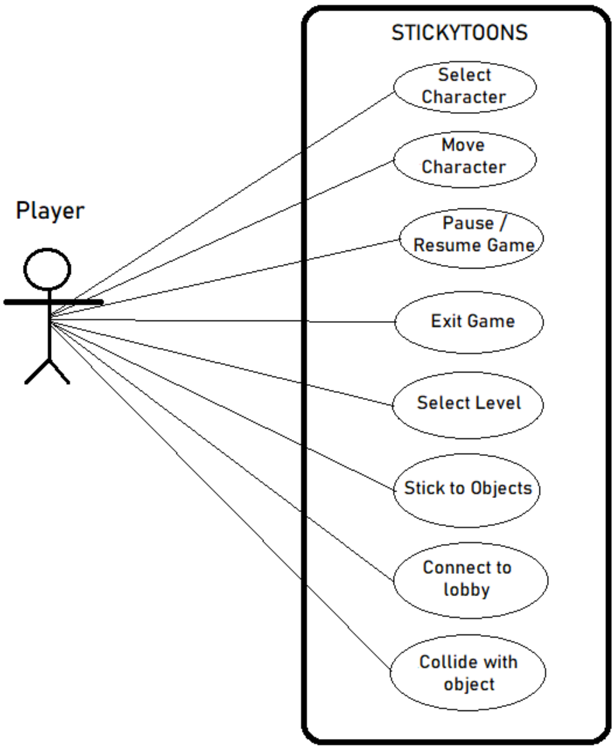


Figure 4.1: Use-Case Diagram for Proposed game

4.3 Use-Case Descriptions

A use case description outlines the steps involved in a specific task or action that a user performs using a system. It typically includes a brief summary of the task, the actors (users) involved, and the specific steps or actions taken. Use case descriptions are used to document and understand the functionality of a system and can help identify potential problems or areas for improvement. They are an important tool in the development and testing of a system, as they provide a clear and detailed outline of the system's intended functionality [26].

The use case description plays a crucial role in developing the proposed game. It helps to identify how the game is played by the player, the sequence of gameplay, and how the different functionalities and features of the game are checked, such as the movement of the user while pressing different buttons. The use case description provides a clear understanding of how the game functions and enables developers to identify any potential issues or areas for improvement. By utilizing the use case description, game developers can create a comprehensive and detailed plan for the development process of the proposed game, ensuring that the final product meets the needs and expectations of the users.

4.3.1 UC01: Select Character

Actors: Player.

type: Primary player.

Brief Description: The use case describes how a player selects character in the game.

Basic Flow:

1. User clicks and request is sent to character select module.
2. Character select modules requests game module for characters.
3. Character select shows characters available.
4. User selects the desired character.
5. Game module confirms and set the selected character.
6. User interface displays the action.

Alternative Flows: None

Pre-condition:

1. The game is running and the player has access to the character selection option.
2. The game module has a list of characters available for selection.
3. The user interface is properly functioning and displays the characters available for selection.
4. The player has some understanding of the game and the different characters available.
5. The player has a device with an internet connection that can send and receive requests to and from the game server

Post-condition: Optimised Performance

Exceptional Scenarios: None.

4.3.2 UC02: Move Character

Actors: Player.

type: Primary player.

Brief Description: The use case describes how a player moves the selected character.

Basic Flow:

1. Request is sent to game module for character movement.
2. Game module receives the request back and shows the movement.
3. Movement is displayed.
4. User acknowledges the movement.
5. Game module moves the character.
6. User interface displays the action.

Alternative Flows: None.

Pre-condition:

1. A character has already been selected and is currently present in the game environment.
2. The game environment allows for movement of the selected character.
3. The player has some understanding of the game controls and how to move the character.

Post-condition: Character is moved in desired direction.

Exceptional Scenarios: None.

4.3.3 UC03: Pause/Resume Game

Actors: Player.

type: Primary player.

Brief Description: The use case describes how a player can pause and resume the game.

Basic Flow:

1. User interface requests the game module to pause.
2. Game module pauses the game and switches to pause menu.
3. User clicks the resume button and request is sent to game module.
4. Game module resumes the game.
5. User interface displays the action.

Alternative Flows: None.

Pre-condition:

1. The game is currently running.
2. The player has access to the ESC key to pause the game.
3. The game module has a pause menu that can be displayed.
4. The player has some understanding of the game controls and how to pause and resume the game.

Post-condition: Game is paused/resumed.

Exceptional Scenarios: None.

4.3.4 UC04: Exit Game

Actors: Player, Multiplayer opponent.

type: Primary player, Secondary player.

Brief Description: The use case describes the controls to quit the game while playing.

Basic Flow:

1. User interface sends request to game module for pause menu.
2. Game module replies to to user interface for pause menue.
3. User presses quit game button.
4. Game module receives the request to quit game.
5. Player acknowledges the end of the game.
6. Game module clears the logs and closes.
7. Game module closes the game.

Alternative Flows:

1. Player presses Alt+f4 key combination.
2. User Interface shows a confirmation message for quitting game.
3. Player clicks yes and request is sent to game module.
4. Game module clears the logs and closes.
5. Game module closes the game.

Pre-condition: None

Post-condition:

1. Game is closed.

Exceptional Scenarios: None.

4.3.5 UC05: Select Level

Actors: Player, Multiplayer opponent.

type: Primary, Secondary player.

Brief Description: The use case describes how a player can select different levels.

Basic Flow:

1. Game module shows the list of levels that can be selected.
2. Player selects the desired level.
3. Request is sent to game module for specified level.
4. User interface shows loading to level.
5. Game module loads the character to level.
6. Game module sets the desired level.

Alternative Flows:

1. User presses ECS to open pause menu.
2. User clicks back to main menu.
3. Game module exits the current level and moves to main menu.
4. User selects level again.

Pre-condition: None.

Post-condition: The game module successfully loads and sets the selected level, allowing the player to start playing at the desired level.

Exceptional Scenarios: None.

4.3.6 UC06: Stick to Objects

Actors: Player.

type: Primary Player.

Brief Description: The use case describes how a character sticks to objects and props.

Basic Flow:

1. Game module checks the input button.
2. Game module moves the character arms as per click.
3. User interface displays the movement.
4. Player confirms the character's movement and sticks to object.
5. Player continues to switch between different controls.

Alternative Flows: None.

Pre-condition:

1. The game is running and the player is controlling their character.
2. The player's character is within range of an object or prop that can be interacted with.
3. The player has a mouse or other input device available to use for clicking.

Post-condition: Player moves and sticks to objects and props.

Exceptional Scenarios: None.

4.3.7 UC07: Connect to Lobby

Actors: Game Engine.

type: Primary: Game Engine.

Brief Description: The use case describes how a player can join a lobby.

Basic Flow:

1. Game module replies by opening lobby selection.
2. Player clicks a lobby and and request is sent to game module.
3. Game module checks if the connection is stable and connects the player to the lobby.
4. Player selects the character and requests is sent to game module.
5. Game module checks the availability of the character and assigns the character to player.
6. Game module connects the player to the lobby's map.
7. Player starts playing the level.

Alternative Flows:

1. Game module checks if the connection is not stable.
2. Player is sent back to main menu if the connection is not stable.

Pre-condition:

1. The player must have a stable internet connection to connect to the lobby.
2. The lobby must have available slots for players to join.
3. The player may need to have a registered account with the game service in order to join a lobby.

Post-condition: Optimised Performance.

Exceptional Scenarios: None.

4.3.8 UC08: Collide with object

Actors: Primary: Game Engine.

type: Primary: Game Engine.

Brief Description: The use case describes how character reacts when it collides with objects.

Basic Flow:

1. Game module checks if the approached object is grab able or rigid.
2. Player will be able to move and grab the grabable object.
3. Player will not be able to cross if the object is rigid.
4. User interface shows the movement to display.
5. Game module reacts according to the game mechanics.
6. Player falls off from the ground.
7. Game module will force the player to spawn back to last saved checkpoint.

Alternative Flows: None.

Pre-condition: None.

Post-condition: Players roams around the level.

Exceptional Scenarios: None.

4.4 Sequence Diagrams

A sequence diagram is a type of UML diagram that shows the interactions between objects or components in a sequential order. It is used to model the behavior of a system and is useful for understanding the interactions between objects or components within a system. Sequence diagrams are useful for identifying the interactions between objects or components, as well as the order in which these interactions occur. They can help to identify potential issues or bottlenecks in the system, and can be used to design and optimize the system to improve efficiency and performance [27].

In a game development project, having a clear understanding of the sequence of events is crucial in order to build specific modules of the game and understand how they are affected by different actions. Sequence diagrams are commonly used as interaction diagrams to help visualize the flow of the game. Without a proper sequence, it can be difficult to understand what is happening in the game. Therefore, maintaining a clear sequence of events is necessary for the development of the game's modules and understanding how they operate in the game.

Figure 4.2 shows the player requests to select a character from the user interface, which then displays the character selection screen and communicates with the character selection module. The character selection module requests the list of available characters from the game module, which provides it. The list is then displayed on the screen. Once the player selects a specific character, the game module confirms the selection to the character selection module, and the user interface displays the selected character.

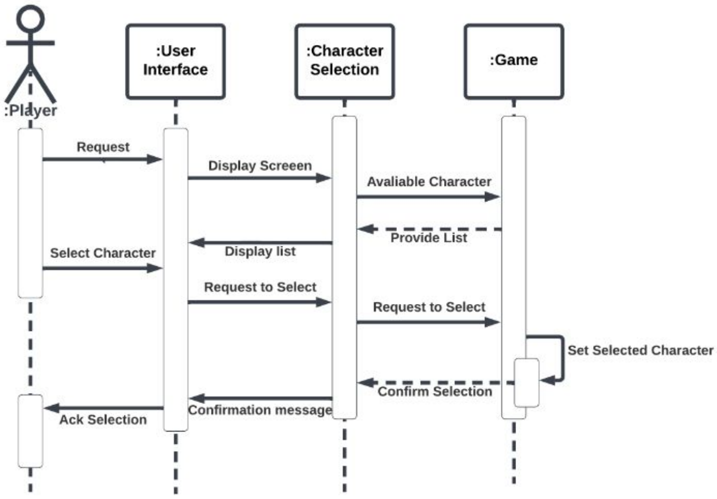


Figure 4.2: Sequence Diagram for Select Character

Figure 4.3 describes that after the player requests to move, the user interface sends a request to the game module to confirm the movement. Once the game module receives the request, it verifies if the movement is valid and possible within the game's rules. If the movement is valid, the game module confirms the movement to the user interface. Then, the game module updates the character's position according to the confirmed movement and displays the updated character movement on the screen. This process ensures that the game is played correctly and follows the rules set by the game module.

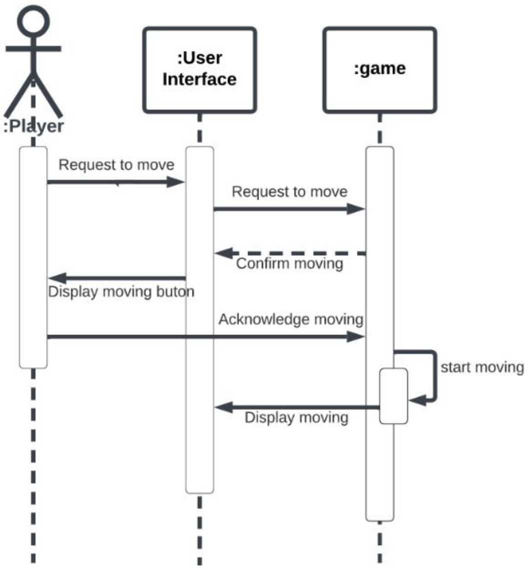


Figure 4.3: Sequence Diagram for Move Character

the Figure 4.4 The player initiates the process by clicking the ”Pause” button in the user interface. The user interface sends a ”Pause” message to the game module. The game module receives the message and saves its progress. Then, the game module sends a ”Pause” message back to the user interface. The user interface receives the ”Pause” message and displays a paused game to the player.

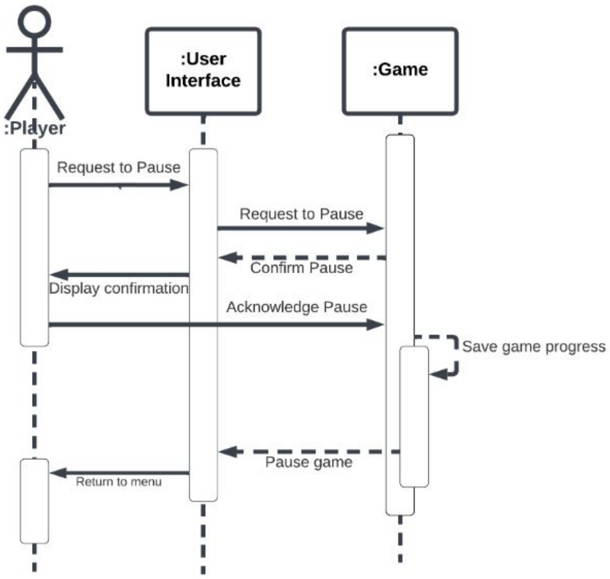


Figure 4.4: Sequence Diagram for Pause

Figure 4.5 shows that thhe player initiates the process by clicking the exit game button in the user interface. The user interface displays a confirmation message. Once the player confirms by clicking exit, the user interface sends an exit message to the game module. The game module receives the message and sends a confirmation message back to the user interface. After the user interface receives the confirmation message, it quits the game.

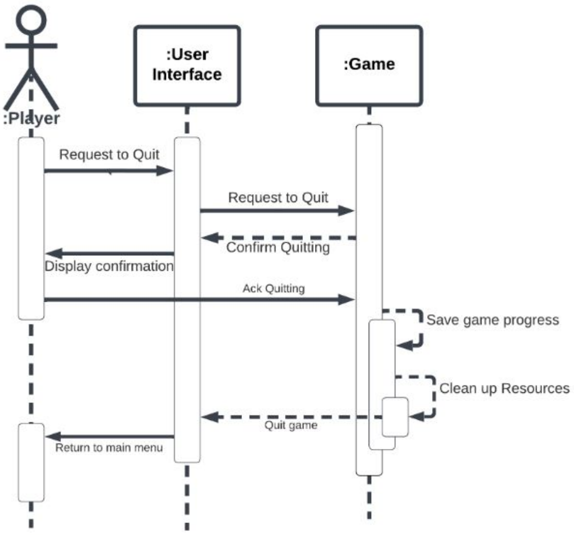


Figure 4.5: Sequence Diagram for Exit Game

Figure 4.6 shows that the process begins when the player clicks the ”Select Level” button in the user interface. Once the player confirms the action, the user interface sends a message to the game module to set the level. The game module then sets the level and sends a message to the user interface to play the selected level. The user interface receives the message and displays the selected level to the player. Finally, the player can play the level of the game.

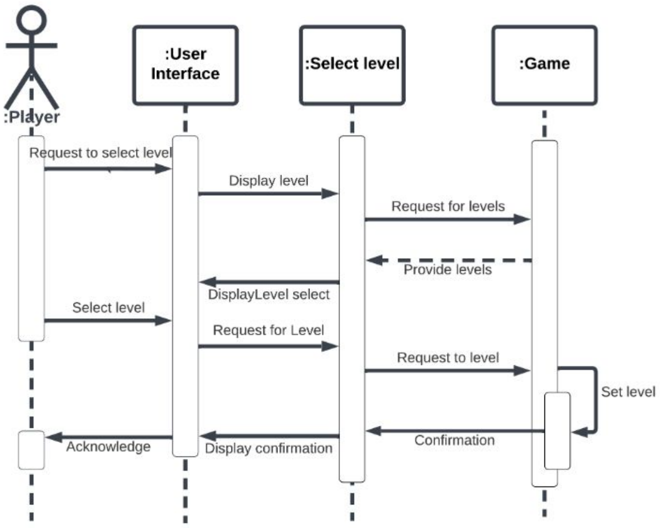


Figure 4.6: Sequence Diagram for Select Level

Figure 4.7 shows that the process begins when the player clicks the ”Select Level” button in the user interface. Once the player confirms the action, the user interface sends a message to the game module to set the level. The game module then sets the level and sends a message to the user interface to play the selected level. The user interface receives the message and displays the selected level to the player. Finally, the player can play the level of the game.

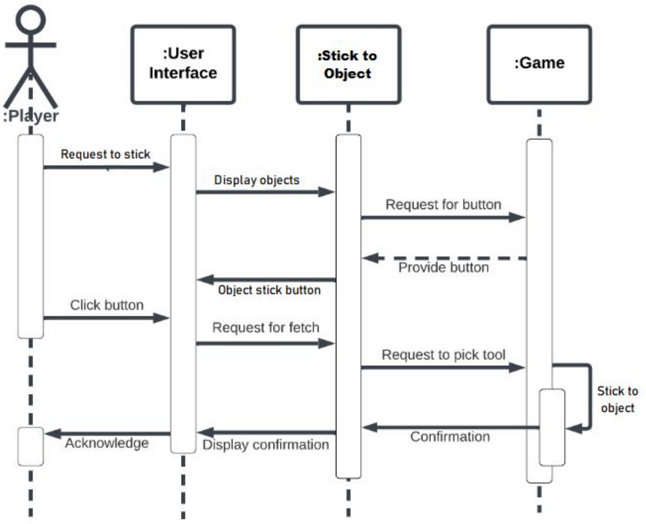


Figure 4.7: Sequence Diagram for Stick to Object

Figure 4.8 shows that the player initiates the process by clicking the ”Connect” button in the user interface. The user interface sends a ”Connect” message to the game server. The game server receives the ”Connect” message and creates a lobby. Then, the game server sends a ”Lobby Created” message to the user interface. The user interface receives the ”Lobby Created” message and displays the lobby to the player. The player can then join or start a game within the lobby.

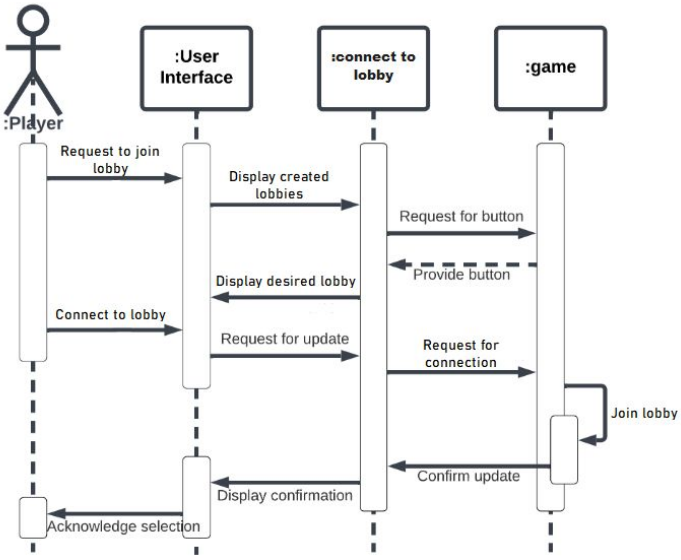


Figure 4.8: Sequence Diagram for Connect to Lobby

Figure 4.9 shows that the player moves the character in the game. The user interface detects a collision with an object and sends a ”Collide” message to the game module. The game module receives the ”Collide” message and plays a collision animation. Then, the game module updates the game state, checks for game over conditions

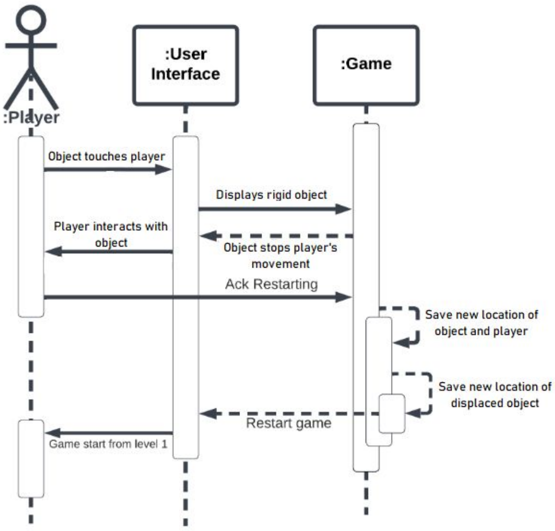

:Player
:User Interface
:Game
Object touches player
Displays rigid object
Player interacts with object
Object stops player's movement
Ack Restarting
Save new location of object and player
Save new location of displaced object
Restart game
Game start from level 1

Figure 4.9: Sequence Diagram for Collide with Object

Chapter 5

Interfaces and Physical Design

It is important to carefully consider and design the interfaces and physical aspects of a system or product because they play a significant role in the user experience. A user-friendly interface and physical design can make it easier for the user to navigate and use the system, leading to higher levels of satisfaction and productivity. On the other hand, a confusing or poorly designed interface and physical design can lead to user frustration and decreased satisfaction with the product. It is therefore important to thoroughly test and refine the interface and physical design during the development process to ensure that it meets the needs and expectations of the user. Additionally, an attractive and visually appealing interface and physical design can help to differentiate the product in the market and increase its appeal to consumers [28].

Interfaces and physical design refer to the visual and physical aspects of a system or product, including the layout and design of screens, buttons, and other user interface elements, as well as the physical characteristics of the product itself, such as its size, shape, and materials. These elements are important because they affect how the user interacts with and perceives the system or product. An effective interface and physical design can improve usability, accessibility, and user satisfaction, while a poorly designed interface and physical design can create obstacles and frustration for the user.

5.1 User Interfaces

User Interfaces (UI) refer to the way that a user interacts with and controls a system or product. This includes the layout and design of screens, buttons, and other visual elements, as well as the method of input, such as touch screens, keyboard, and mouse. The purpose of a UI is to make it easy and intuitive for the user to access and use the features and functions of the system or product. A well-designed UI can improve the user's experience, making it enjoyable and efficient to use the system or product. On the other hand, a poorly designed UI can create confusion and frustration for the user, leading to a negative experience [29].

In the proposed game, the User Interfaces (UI) play a crucial role in shaping the overall player experience. The UI encompasses the visual elements, design, and input methods that players use to interact with the game. This includes the layout of screens, buttons, and other graphical elements, as well as the methods of input such as touch screens, keyboard, and mouse. The goal of the UI in the project is to provide a user-friendly and intuitive interface that allows players to easily navigate the game, access its features, and control the gameplay. A well-designed UI enhances the player's engagement, making it enjoyable and seamless to interact with the game's functionalities. The UI in the project is carefully crafted to ensure that players can easily understand and use the game's controls, resulting in a positive user experience and high player satisfaction.

Figure 5.1 shows how character select screen in the game is a user-friendly interface that allows players to choose their preferred character before entering the game. It features a clean and intuitive design with easy-to-use buttons, including an ”Edit” button for character customization, a ”Toggle Player” button for quick switching between characters, and a ”Back” button for easy navigation. Overall, the character select screen enhances the gameplay experience by providing a sense of personalization and control for players.

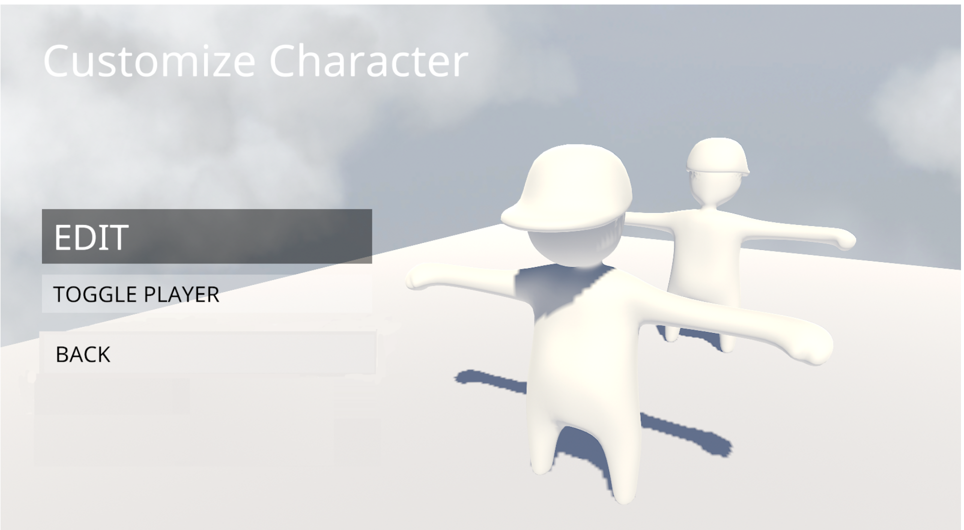


Figure 5.1: Select Character

Figure 5.2 describes character movement is a critical component of any game, and the game is no exception. In the game, players control their characters using a combination of keyboard and mouse controls. The character moves around in a side-scrolling environment, and the controls are straightforward and intuitive. The translucent box on the screen provides instructions to the player, letting them know that they can use the WASD keys to move the character around and the right-click and left-click buttons on the mouse to move each hand independently. This unique control scheme allows for precise movement and control over the character's actions, which is essential for puzzle solving and navigating the game's various obstacles. Overall, the character movement in the game is fluid and responsive, making for an enjoyable and engaging gameplay experience.

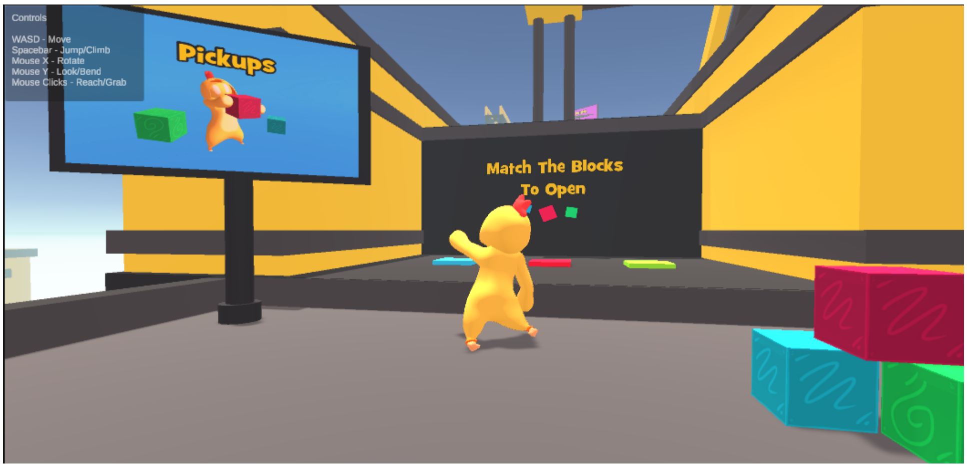


Figure 5.2: Move Character

Figure 5.3 describes pause and resume game menu screen in the game is an important feature that provides players with control over their gameplay experience. When a player presses the "Escape" key, the game pauses and the pause menu appears on the screen. The pause menu includes several options for the player, including the ability to resume the game and continue playing from where they left off. Additionally, players can choose to save the game at the point they paused, allowing them to come back to it later. Finally, the "Quit" button allows players to exit the game entirely and return to the main menu.

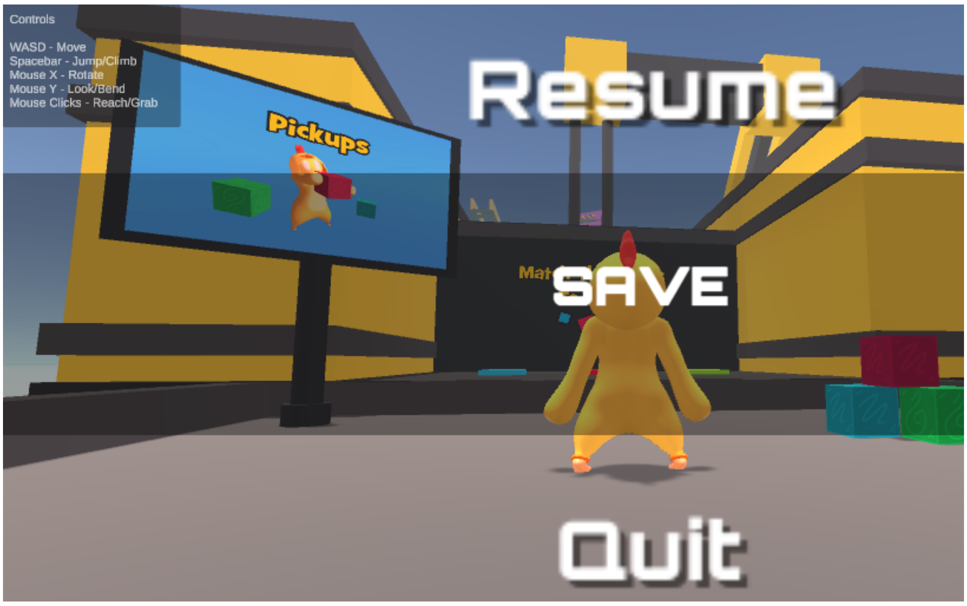


Figure 5.3: Pause/Resume Game

Figure 5.4 shows when a player decides to quit the game in the game, they can do so by clicking the "Quit" button in the pause menu. This will exit the game entirely, closing the application and returning the player to their desktop or operating system. The "Quit" button is clearly labeled and easily accessible in the pause menu, allowing players to exit the game without any confusion or difficulty. Once the player has quit the game, their progress will be saved automatically so that they can pick up where they left off when they next launch the game.

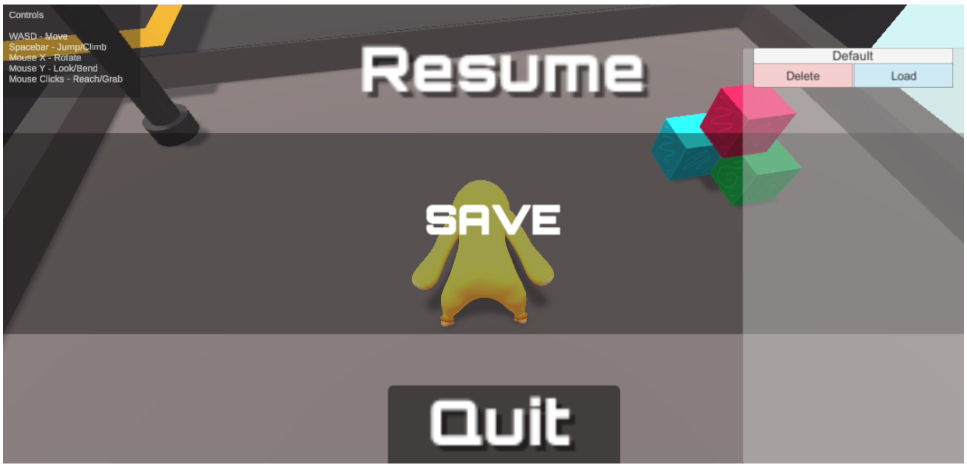


Figure 5.4: Exit Game

Figure 5.5 describes that map selection in the game. Before beginning a game, players are given the option to choose from a variety of maps, each with its own unique terrain, obstacles, and challenges. The map selection screen is designed to be intuitive and user-friendly,

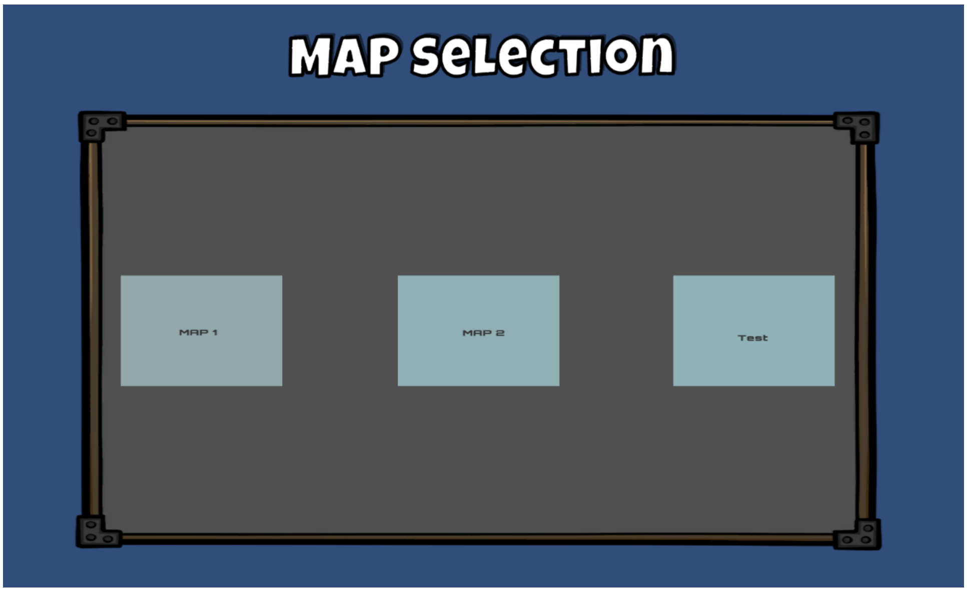


Figure 5.5: Select Level

Figure 5.6 elaborates sticking mechanism in the game is a unique gameplay feature that sets it apart from other games in the puzzle genre. Players control a character who is capable of grabbing and holding onto two boxes simultaneously, one in each hand. This creates a wide range of puzzle-solving possibilities, as players must use their character's sticky abilities to traverse the game's various obstacles and challenges. The mechanism is easy to use, with players using the right mouse button to move one hand and the left mouse button to move the other hand.

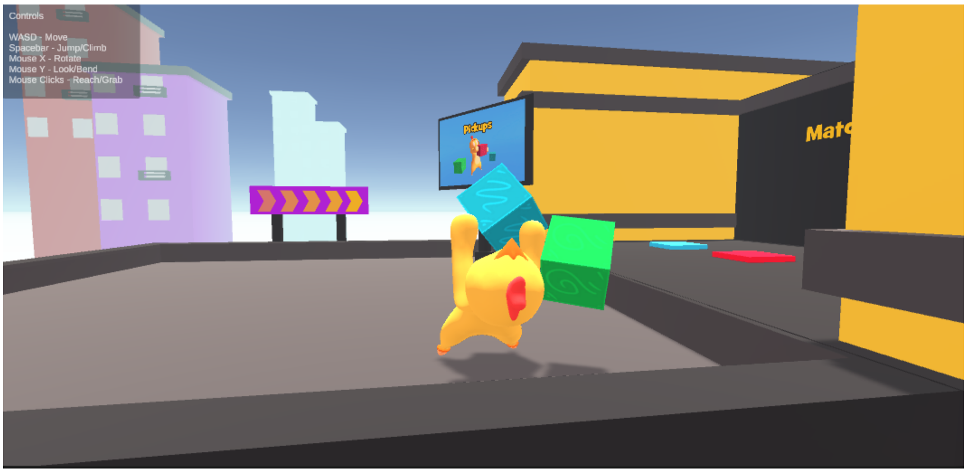
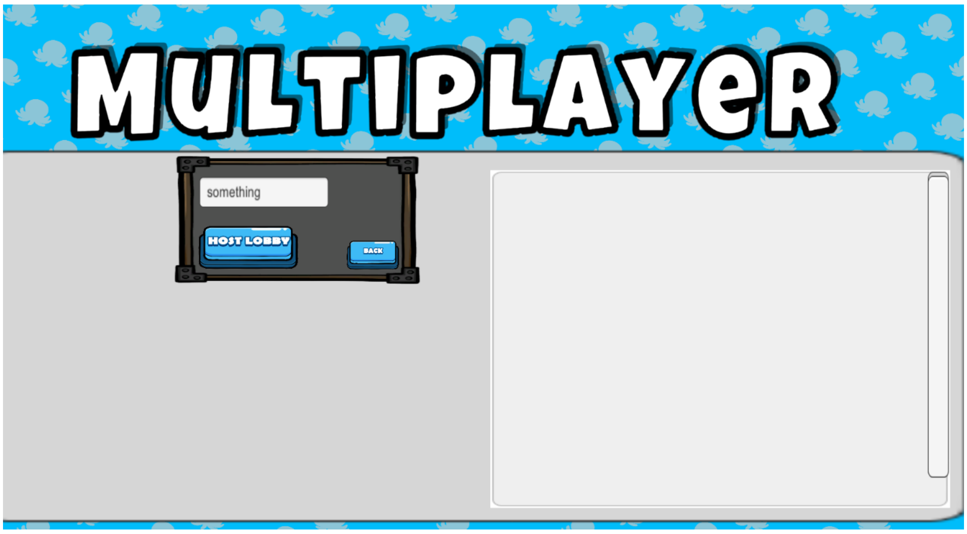


Figure 5.6: Stick to Objects

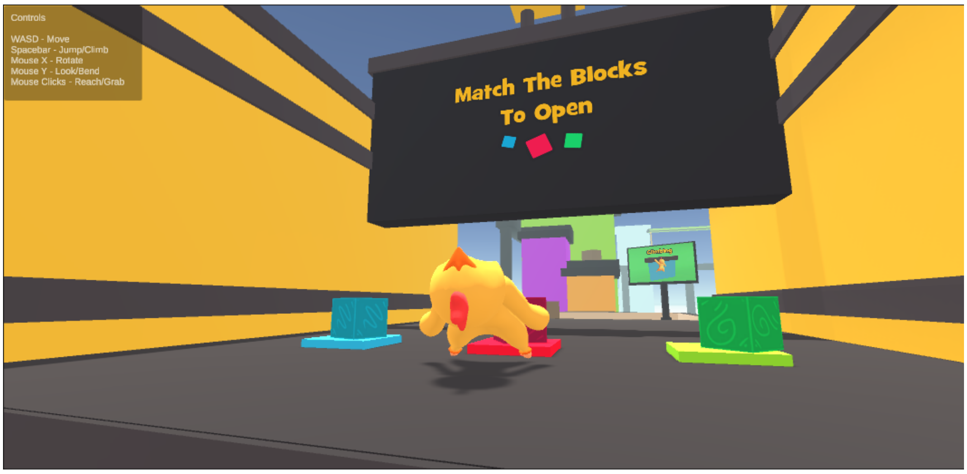
Figure 5.7 shows the lobby screen in the game allows players to host or join a game session. Players have the option to either create a new game session and invite their friends to join, or search for and join an existing game session hosted by someone else.



something

Figure 5.7: Connect to lobby

Figure 5.8 describes object collision in the game. In the game, the player's character can grab and hold onto two boxes at a time using a unique sticking mechanism. The player can then move around with the boxes and place them on designated tiles on the floor. When a box is placed on a tile, it triggers a collision event that can activate various mechanisms and unlock doors. This requires the player to carefully navigate the environment, strategically placing boxes on the correct tiles to progress through the game. Overall, the object collision system adds a layer of puzzle-solving gameplay that challenges players to think creatively and strategically.



Controls

WASD - Move
Spacebar - Jump/Climb
Mouse X - Rotate
Mouse Y - Look/Bend
Mouse Clicks - Reach/Grab

Figure 5.8: Collide with object

5.2 User Tables

Information and data pertaining to users, as required by the application, should be stored in the User table. This includes details such as Customer-Id, First-name, Last-name, Email, and Contact, which are typically organized in a tabular format with the use of keys such as Super key and Local key. The data types utilized in the table may include Var-char and Integer, with specified lengths for the data mentioned. This structured approach to data organization ensures efficient storage and retrieval of user-related information, allowing for effective management and utilization of user data within the application [30].

In the proposed game, it has been noted that the game does not utilize a database. As a result, the inclusion of a user table may not be necessary for the development of the game. Since there is no need to store user-related information or manage user data within a database, the use of a user table can be omitted from the game's development process. This unique characteristic of proposed game, where user data is not stored or managed, presents a different approach to game development compared to other applications that rely on databases to store and retrieve user information.

Chapter 6

Test Plan

A testing plan is a document that outlines the strategy, approach, resources, and schedule for testing a product or system. It is an important part of the development process because it helps ensure that the final product meets the specified requirements and is of high quality. A testing plan helps identify what needs to be tested, how it will be tested, who will be responsible for testing, and when the testing will be performed. It also helps to coordinate the efforts of the testing team and to track the progress of the testing process. In short, a testing plan helps to ensure that the product is thoroughly and effectively tested before it is released to the public [31].

In game proposed game, a test plan is a critical component for creating and developing software applications. The test plan document outlines essential information about product modules, including their functionality and any malfunctions that may be present. This document provides customers and managers with a clear understanding of the testing process. Typically, the test plan is divided into three sub-tests, including unit testing, integration testing, and system testing. All three tests are crucial in assessing the performance of individual parts of the game after joining different modules and systematically testing the game by playing it multiple times.

6.1 Unit Testing

To conduct unit testing for the product, manual testing method is used. Manual testing allows to test more complex units and scenarios that may be difficult to automate. Combination of positive and negative testing approaches are used, testing both valid and invalid inputs and expected and unexpected behavior [32].

In the proposed game, unit testing is an essential step to verify the proper functioning of specific game functions. This process involves testing each functional requirement individually to ensure the quality of the game. The developers perform unit testing from the start to the end of the game, with the software testing team playing the game to test all features thoroughly and ensure that no bugs are left behind.

6.1.1 Select Character

Table 6.1 provides a description of this process, which should be mandatory before starting the game. If the game begins without requiring the player to select a character, it may indicate issues with the game design. It is important to carefully consider the character selection process and its role in the overall gameplay experience.

Table 6.1: Select Character

Use Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass or Fail
UT01	Character Select	1. Shift to main menu 2. Click play button 3. Click on Character select	Press "S","W","A","D"	Character selected	As expected	Pass
UT02	Character Select without confirming	1. Click on play button 2. Click on Character select	Press "S","W","A","D"	Game will not start	As expected	Pass

6.1.2 Move Character

Table 6.2 is to test the movement of the character in a game, ensuring that it moves smoothly and according to the physics engine's design. This test is important to ensure that the character's movement feels natural and responds appropriately to player inputs.

Table 6.2: Move Character

Use Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass or Fail
UT03	Character Select	1. Shift to main menu 2. Click play button 3. Click on Character select 4. Click on movement keys to move	Press "S","W","A","D"	Character moves	As expected	Pass
UT04	Character Select without confirming	1. Click on play button 2. Click on Character select 3. Click on movement keys to move	Press "S","W","A","D"	Player will not move in any direction	As expected	Pass

6.1.3 Pause/Resume Game

Table 6.3 outlines the pause and resume feature in a game, which is important for allowing players to take breaks without losing their progress. When the game is paused, the player is directed to a pause menu that typically includes options to exit to the main menu, resume the game, or restart from the beginning. In any level of the game, the pause button or menu is necessary to account for unexpected interruptions or breaks in gameplay. Without this feature, players may lose interest or become frustrated if they cannot pause the game and resume it later.

Table 6.3: Pause/Resume Game

Use Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass or Fail
UT05	While playing game check whether game is pause or not	1. Open main menu 2. Click Play game 3. Select Level 4. Game starts 5. Click ESC	Press ”S”,”W”,”A”,”D”	Game Paused	Pass	
UT06	While playing game check whether game is resume or not	1. Open main menu 2. Click Play game 3. Select Level 4. Game starts 5. Click Resume button	Press ”S”,”W”,”A”,”D”	Game Resumed	As expected	Pass

6.1.4 Exit Game

Table 6.4 outlines the guide for playing the game, which includes instructions for using the quit function. If the user chooses to quit the game after pausing, they will be directed to the main menu. Should the user wish to change the game's level or settings, they can also quit and return to the main menu. However, if the user quits the game entirely and wants to play again, they will lose their data and the game will reset from the beginning. In such cases, the user must select their desired levelto play again.

Table 6.4: Exit Game

Use Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass or Fail
UT07	While in game go to main menu by exiting the game	1. Move to main menu 2. Click on select level 3. Click on desired level 4. Game starts . Click ESC to pause 1. Click on Exit game	Left Mouse Button	Game Exited	As expected	Pass

6.1.5 Select Level

Table 6.5 describes the process of unit testing game levels in proposed game is explained. As the game levels are open-ended and can be played in any order, it is essential to ensure that all levels function correctly. To do this, unit testing involves selecting each level to verify its functionality.

Table 6.5: Select Level

Use Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass or Fail
UT08	Selecting the desired level	1. Move to main menu 2. Click on play 3. Click on Select level 4. Click on desired level	Press ”S”,”W”,”A”,”D”	Level selection complete	As expected	Pass
UT09	Level selection within the map	1. Press ESC to open menu 2. Click back to main menu 3. Click play 4. Select desired level again	Press ”S”,”W”,”A”,”D”	Level selection complete	As expected	Pass

6.1.6 Stick to Objects

Table 6.6 is a test of the grabbing and sticking mechanism. Grabbing the grabable objects and sticking to the rigid objects is key for the completion of the level. Every level contains rigid objects as well as grabable objects with which the player will interact.

Table 6.6: Stick to Objects

Use Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass or Fail
UT10	Move player near the object to stick or grab	1. Go to main menu 2. Click play 3. Click and select the level 4. Move towards the rigid object and click mouse buttons	Press ”S”,”W”,”A”,”D”, LMB, RMB	Object grabbed or stick to the character	As expected	Pass

6.1.7 Connect to lobby

Table 6.7 outlines the guide for playing the game, which includes instructions for using the join lobbies function. In order to play the multiplayer a user must join a hosted lobby. Then the host will continue and start the game.

Table 6.7: Connect to lobby

Use Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass or Fail
UT011	While in game go to main menu to join lobby	1. Go to main menu 2. Click play 3. Click on multiplayer 4. Click on one of the hosted lobbies in list 5. Select a character 6. Move character to start playing	Press ”S”,”W”,”A”,”D”, LMB, RMB	Successful connection to the lobby	As expected	Pass

6.1.8 Collide with Object

Table 6.8 describes the test of collision of player and objects in the surroundings. Different levels have different obstacles and items to collide with. If there is no ground or object to cling upon the player respawns to last location with floor.

Table 6.8: Collide with Object

Use Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass or Fail
UT12	Move player near the object or ground to collide to	1. Go to main menu 2. Click play 3. Click and select the level 4. Move towards the rigid object, ground, or obstacle	Press ”S”,”W”,”A”,”D”, LMB, RMB	Players collide with objects and grabs	As expected	Pass
UT13	Player does not collides with ground or object	1. Go to main menu 2. Click play 3. Click and select the level 4. Player fells off from ground	Press ”S”,”W”,”A”,”D”, LMB, RMB	Player respawns to the last location it touched the ground	As expected	Pass

6.2 Integration Testing

Integration testing, commonly referred to as integration and testing, is a sort of software testing in which various software application parts, modules, or components are examined collectively. These modules, however, might be created by different programmers. [33].

During the proposed game, the software testing team verified various functional requirements separately and then integrated them to ensure their compatibility. Integration testing was considered essential for developers to test and visualize the application’s bigger picture. The software’s nature made integration testing more useful and critical to the project's success. Additionally, integration testing played a crucial role in ensuring the system's reliability.

6.2.1 Graphical User Interface

Table 6.9 elaborates testing methods used to ensure efficient joint functionality or modules . Integration testing is the primary method used to test the majority of functions together to determine if the module is functioning properly. This is important because users typically use all the features and functionalities of a system. To evaluate the overall functionality of the graphical user interface, graphical user interface testing is utilized in conjunction with integration testing. In a graphical user interface, all the modules are integrated and tested to thoroughly check all the features and functionality.

Table 6.9: Graphical User Interface

Use Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass or Fail
IT01	Assure if the player go to main menu by clicking play button	1. Go to main menu 1. Click Play	Press "S","W","A","D", enter, LMB, RMB	Main screen switches to play menu	As expected	Pass
IT02	Assure if the player go to settings from main menu	1. Go to main menu 1. Press setting button	RMB,LMB	Screen switches to settings	As expected	Pass
IT03	Assure if the player exits the game	1. Go to main menu 1. Click on exit game	RMB,LMB	Game exits	As expected	Pass

6.3 System Testing

System testing is a type of testing that involves evaluating the overall functioning of the project, including its hardware and software components. It is important to identify and fix any issues or bugs found during system testing to ensure that the project is of high quality and enjoyable for players. [34]

To ensure the successful development of the proposed game, the development team required system testing to determine if the project was functioning correctly. Given the nature of the project, the overall performance analysis was critically dependent on the results obtained from system testing. In the Hold to Ransom project, the sustainability and functionality of its features were evaluated by playing the game from start to end, which is an integral part of system testing.

6.3.1 Game Play

Table 6.10 represents the most comprehensive form of system testing for a game. It encompasses the full testing of the game from the loading stage until the gameplay screen is displayed. This testing method is utilized to check for any possible malfunctions that might occur when the game is fully loaded by customers. The process involves evaluating the game's response to ensure that it is functioning correctly. During gameplay testing, the game's features, mechanics, controls, and other aspects are tested to ensure that they work together seamlessly. This is done to guarantee that the players' experience is not hindered by any bugs or technical issues. The results obtained from gameplay testing are essential in ensuring the overall quality and reliability of the game.

Table 6.10: Game Play

Use Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass or Fail
ST01	Selecting the desired level	1. Move to main menu 2. Click on play 3. Click on level 4. Click on desired level	Press "S","W","A","D"	Level selection complete	As expected	Pass
ST02	Level selection within the map	1. Press ESC to open menu 2. Click back to main menu 3. Click play 4. Select desired level again	Press "S","W","A","D"	Level selection complete	As expected	Pass
ST03	Assure if the character moves correctly in selected level	1. Move to main menu 2. Click on play 3. Click on Select level 4. Click on desired level 5. Press controls to move and stick player	Press "S","W","A","D", LMB,RMB	Level selection complete	As expected	Pass

Chapter 7

Conclusion and Future Work

7.1 Conclusion

The proposed game is a delightful and entertaining physics-based puzzle game that offers players a unique and engaging gaming experience. The game's intuitive controls, colorful graphics, and variety of levels combine to create an immersive and enjoyable world for players to explore and solve puzzles in. The sticky ability of the player character add s an extra layer of strategy and creativity to the gameplay, allowing players to approach challenges in a range of different ways. Whether playing alone or with friends, the game is sure to provide hours of fun and challenging entertainment. In addition to its engaging gameplay, the game also boasts a high level of replay value. With a variety of levels to explore and challenges to overcome, players will find themselves returning to the game again and again to test their skills and uncover all of its secrets. This, coupled with the game's cute and charming art style, make the game a must-play for fans of puzzle games and anyone looking for a fun and unique gaming experience.

Overall, the game is a well-crafted and enjoyable game that is sure to provide players with a satisfying and memorable gaming experience. Its creative gameplay mechanics and charming art style make it a standout in the puzzle game genre, and we highly recommend giving it a try.

7.2 Future Work

There are many exciting directions in which the product could be developed in the future. Some possibilities for future work include: adding new levels, expanding the character roster, and incorporating user-generated content. One way to keep users engaged and offer additional replay value would be to add new levels to the product. These could include new environments and challenges to keep users on their toes. Another option would be to expand the character roster by adding new playable characters to the game, each with their own unique abilities and playstyles. This could offer users additional ways to approach the product's challenges and add more variety to the gameplay. Another possibility for future work would be to allow users to create and share their own levels and challenges. This could add an extra layer of replay value to the product and encourage user creativity. Overall, the future of the product is full of exciting possibilities, and we look forward to seeing how it continues to evolve and grow in the future.

References

[1] OnlineLibrary."Product Scope. Business Analysis Techniques: 72 Essential Toolsfor Success, 2nd ed. John Wiley Sons, Ltd., 2012, pp. 195-201.(Accessed: 2022).

[2] Steam. Human Fall Flat. Available:https://store.steampowered.com/Human_Fall_Flat.(Accessed:2022).

[3] Steam.Gang Beasts. Available:https://store.steampowered.com/Gang_Beasts/ (Accessed: 2022).

[4] Steam. Crab Game. Available:https://store.steampowered.com/Crab_Game/ (Accessed: 2022).

[5] Fall Guys. Fall Guys. Available: https://www.fallguys.com/enUS(Accessed: 2022).

[6] Steam. The Academy. Available:https://store.steampowered.com/The_Academy_The_First_Riddle(Accessed: 2022).

[7] Steam. Omno. Available:https://store.steampowered.com/Omno/ (Accessed: 2022).

[8] Steam. Risk of Rain. Available: https://store.steampowered.com/Risk_of_Rain_2 (Accessed: 2022).

[9] Steam. Etherborn. Available:https://store.steampowered.com/Etherborn/ (Accessed: 2022).

[10] Steam. Paws and Soul. Available: https://store.steampowered.com/Paws_and_Soul/ (Accessed: 2022).

[11] Scribbr.Problem-Statement.Available:https://www.scribbr.com/research-process/problem-statement/(Accessed: 2022).

[12] Middle_East_Technical_University.Software-Process-Model.Available: https://senior.ceng.metu.edu.tr/2011/eriksoft/documents/srs.pdf (Accessed: 2022).

[13] Middle_East_Technical_University. Software Requirements Specification (SRS) Template. Available: https://senior.ceng.metu.edu.tr/2011/eriksoft/documents/srs.pdf (Accessed: 2022).

[14] Middle_East_Technical_University.IEEE_Recommended_Practice_for_Software_Requirements_Specifications. Available: https://senior.ceng.metu.edu.tr/2011/eriksoft /documents/srs.pdf(Accessed: 2022).

[15] Senior.ceng.metu.edu. (n.d.). Writing a Requirements Document. Retrieved from https://senior.ceng.metu.edu.tr/2011/eriksoft/documents/srs.pdf(Accessed: 2022).

[16] Senior.ceng.metu.edu. (n.d.). Functional Requirements Specification. Retrieved from https://senior.ceng.metu.edu.tr/2011/eriksoft/documents/srs.pdf(Accessed: 2022).

[17] Senior.ceng.metu.edu. (2016). Software Process Model. Retrieved from https://senior.ceng.metu.edu.tr/2011/eriksoft/documents/srs.pdf(Accessed: 2022).

[18] Perforce(2021).How to Write a Software Requirements Specification (SRS) Document. Retrieved from https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document(Accessed: 2022).

[19] Perforce.(2021).Functional_Requirements_Specification.Available:https://www.perfo rce.com/blog/alm/how-write-functionalrequirements-specification-srs-document(Accessed: 2022).

[20] Perforce(2021).Requirements_Document.Available:https://www.perforce.com/blog/ alm/how-write-requirements-srs-document(Accessed: 2022).

[21] Perforce(2021).Software_Requirements_Specification.Available: https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document (Accessed: 2022).

[22] Perforce(2021).Software_Requirements_Specification_Audience.Available: https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document(Accessed: 2022).

[23] Altexsoft(2021)._Functional_Requirements.Available:https://www.altexsoft.com blog business functional and nonfunctional requirements specification and types(Accessed: 2022).

[24] Altexsoft(2021).Non_Functional_Requirements.Available: https://www.altexsoft.com/blog/business-functional-and-non-functional requirements-specification-and-types(Accessed: 2022).

[25] Altexsoft(2021).Functional_Requirements.Available: https://www.altexsoft.com/bl og/business functional-and-non-functional-requirements-specification-and-types(Accessed: 2022).

[26] Altexsoft(2021).Software_Quality_Attributes_[Online].Available: https://www.altexsoft.com/blog/business/softwarespecification-and-types/(Accessed: April 5, 2023).

[27] Visual-paradigm(2021).Sequence diagram [Online].Available: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/ (Accessed:April 5,2023)

[28] IGI-global(2021). Interface and physical design[Online].Available:https://www.igi-global.com/dictionary/physical-user-interface-design/22709/(Accessed:April 3,2021)

[29] Altexsoft(2021).Function_Quality_Attributes_[Online].Available: https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types (Accessed: April 5, 2023).

[30] Altexsoft(2021).Safety_Requirements_[Online].Available: https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types (Accessed: April 5, 2023).

[31] TutorialsPoint(2022). Requirements Gathering Techniques Used [Online]. Available: https://www.tutorialspoint.com/business_analysis/business_analysis requirement_gathering_techniques.html (Accessed: April 5, 2023).

[32] Techtarget(2022).Unit_Testing_[Online]. Available: https://www.techtarget.com/searchsoftwarequality/definition/unit-testing (Accessed: April 5, 2023).

[33] Techtarget(2022).Unit_Testing_[Online]. Available: https:http://tryqa.com/what-is-integration-testing (Accessed: April 5, 2023).

[34] Techtarget(2022)._System_Testing_[Online].Available: https://www.techtarget.com/searchsoftwarequality/definition/system-testing (Accessed: April 5, 2023).