

Table of Contents

<u>ACKNOWLEDGEMENT</u>	<u>4</u>
<u>ABSTRACT</u>	<u>5</u>
<u>NOTATION, NAMING AND CONVENTIONS</u>	<u>6</u>
<u>PROJECT CHARTER</u>	<u>8</u>
➤ PROJECT DEFINITION	8
➤ COMPANY PROFILE	9
➤ PURPOSE	10
➤ SCOPE OF PROJECT	11
3-DIMENSIONAL GAMES:	11
AUGMENTED REALITY GAMES:	12
VIRTUAL REALITY GAMES:	14
➤ AUGMENTED REALITY:	16
➤ IMAGE-TARGET:	17
➤ VIRTUAL REALITY:	18
➤ AUGMENTED REALITY VS. VIRTUAL REALITY	19
<u>PROJECT PLAN</u>	<u>20</u>
➤ OBJECTIVE	20
➤ SOFTWARE MODEL-TRADITIONAL (PROTOTYPE MODEL)	21
➤ WORK DIVISION (W.R.T TIME)	22
<u>SYSTEM REQUIREMENTS</u>	<u>24</u>
<u>HARDWARE AND SOFTWARE REQUIREMENTS</u>	<u>25</u>
➤ HARDWARE REQUIREMENTS:	25
➤ SOFTWARE REQUIREMENTS:	25

TOOLS AND TECHNOLOGIES	26
➤ TOOLS:	26
➤ TECHNOLOGIES:	29
SYSTEM ANALYSIS	30
➤ SYSTEM FLOW DIAGRAMS	30
GPR-UI	30
RUNNER (3D)	31
CUBBER (3D)	32
DUCK STRIKE (AR)	33
FIGHT MASTER (AR)	34
BALL MAZE (AR)	35
CAR RACE (VR)	36
BOWLING (VR)	37
HURDLES (VR)	38
SYSTEM DESIGN	39
➤ 3D-RUNNER	39
➤ AR-DUCK STRIKE	40
➤ AR-FIGHT MASTER	41
➤ AR-BALL MAZE	42
➤ VR-CAR RACE	43
➤ VR-BOWLING	44
➤ VR-HURDLES	45
SCREENSHOTS	46
APPLICATION USER INTERFACE	46
➤ RUNNER	51
➤ CUBBER	53
➤ DUCK STRIKE	56
➤ FIGHT MASTER	59
➤ BALL MAZE	63
➤ CAR RACE	65
➤ BOWLING	69
➤ HURDLES	71

<u>TESTING</u>	73
➤ TESTING PLAN:	73
➤ TESTING METHODS:	74
➤ TEST SCENARIOS:	75
3D-RUNNER	75
3D-CUBBER	75
AR-DUCK STRIKE	76
AR-FIGHT MASTER	77
AR-BALL MAZE	77
VR-CAR RACE	78
VR-BOWLING	78
HURDLES	78
MAIN USER INTERFACE	79
➤ BUG FIXING /TEST REPORT	80
<u>CODING STANDARD FOLLOWED</u>	81
<u>LIMITATIONS AND FUTURE ENHANCEMENTS</u>	84
<u>EXPERIENCE AND LEARNING</u>	85
<u>REFERENCE AND BIBLIOGRAPHY</u>	86

Acknowledgement

Game Pack Reality is created using Prototype methodology. It has been a great experience for our team. Practical application of the studies as well as exploring new tools and technologies like Unity3D, Visual Studio, Vuforia, Google Cardboard, Android Studio and Adobe Photoshop was undertaken. This knowledge will help us forever.

The results received from this Final project (M.Sc.IT. Semester IV) are due to the active guidance and assistance received from many individuals. We are grateful towards them. We could never have successfully completed our work if they hadn't contributed their part.

We are thankful to **Prof. Rakesh S. Shrivastava**(Head, Department of Computer Application) for providing us with this platform and encouragement for trying new technologies.

Very special thank you to our project guide **Mrs. Khushbu Raval** who portrayed extreme efforts towards providing appropriate guidance at all times.

We also want to thank other faculties for helping us in the doubts aroused during project documentation.

We would also like to extend our heartfelt thanks to the staff of Department of Computer Applications helped us in completing our project.

Abstract

In the last few years, there has been a growing interest in the application of the game elements to the real-life goals and tasks.

Game Pack Reality is an Android gaming application that could be played only by a single-player.

Game Pack Reality is all about impressing and elevating the gaming experience which could possibly be a revolution in the gaming world by combining the best of gaming technologies together on a single platform.

The aim of the Application is to provide multiple games on a single platform. This is achieved by merging various technologies on a single platform.

Game Pack Reality combines multiple gaming platform like 3-Dimensional, Augmented Reality, Virtual Reality on one platform.

3-Dimensional: provides traditional type of gaming experience which all the users are acquainted with.

Augmented Reality: provides a real - physical view in which elements are enhanced by computer-generated input, so this would enhance user's gaming experience.

Virtual Reality: provides total virtual gaming environment so that user could feel his presence in the game for best gaming experience.

Game Pack Reality is a Platform that involves eight major games namely:

- 1) Runner(3-Dimensional)
- 2) Cubber (3-Dimensional)
- 3) Duck Strike (Augmented Reality)
- 4) Fight Master(Augmented Reality)
- 5) Ball Maze(Augmented Reality)
- 6) Car Race (Virtual Reality)
- 7) Bowling (Virtual Reality)
- 8) Hurdles (Virtual Reality)

Notation, Naming and Conventions

States:

❖ Start

Indicates the beginning and ending of a program or sub-process. Represented as a stadium oval or rounded (fillet) rectangle. They usually contain the word "Start" or "End".



❖ Process

Represents a set of operations that changes value, form, or location of data. Represented as a rectangle.



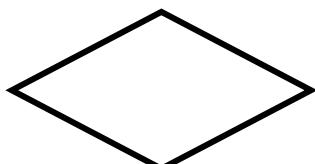
❖ Transition

Shows the process's order of operation. A line coming from one symbol and pointing at another. Arrowheads are added if the flow is not the standard top-to-bottom, left-to right.



❖ Decision

Shows a conditional operation that determines which one of the two paths the program will take. The operation is commonly a yes/no question or true/false test. Represented as a diamond (rhombus).



❖ **Predefined Process**

Shows named process which is defined elsewhere. Represented as a rectangle with double-struck vertical edges.



❖ **Input/output**

Indicates the process of inputting and outputting data, as in entering data or displaying results. Represented as a parallelogram.



Full forms:

- ❖ GPR - Game Pack Reality
- ❖ 3D - 3 Dimensional
- ❖ AR - Augmented Reality
- ❖ VR - Virtual Reality
- ❖ AI - Artificial Intelligence
- ❖ App - Application
- ❖ GUI - Graphical User Interface
- ❖ SDK - software development kit
- ❖ APK - Android application package
- ❖ UWP - Universal Windows Platform

Project charter

➤ Project Definition

Game Pack Reality is 3-Dimensional, Augmented and Virtual reality based software application that provides variety of games on a single platform, so any game user does not need to download different apps.

The platform would also enhance the user experience while playing the game by giving him a composed and virtual view (real-world and super-imposed objects).

Apart from this, the application also allows the integration of game visual and audio content with the user's environment.

Game Pack Reality is all about impressing and elevating the gaming experience which could possibly be a revolution in the gaming world by combining the best of gaming technologies together on a single platform.

➤ Company Profile

The Syntax Studio was established to help the business to get the visibility of ideas to optimize their profit.

They are experts that specialize in providing a comprehensive, fast and affordable services and support in Apps and Website building.

They design and manage infrastructure, custom API and other digital products which leaves a good impression on the end-users.

➤ Purpose

In the last few years, there has been a growing interest in the application of the game elements to the real-life goals and tasks. Game would allow the user to get a break from the tedious day to day life and could provide entertainment in the leisure time.

The main purpose of developing a game was the keenness to learn new software and technologies like Unity, Augmented Reality and Virtual Reality respectively and gain experience from it.

Game Pack Reality combines multiple gaming platforms like 3-Dimensional, Augmented Reality, Virtual Reality on one platform.

Game Pack Reality is all about impressing and elevating the gaming experience which could possibly be a revolution in the gaming world by combining the best of gaming technologies together on a single platform.

So that user need not have to download different games, this would provide all in one gaming platform with exciting experience of new technologies.

Till now there is no such other application that would provide a combination of these technologies, so any user would be attracted towards this application.

➤ Scope of Project

Game Pack Reality is all about impressing and elevating the gaming experience which could possibly be a revolution in the gaming world by combining the best of gaming technologies together on a single platform.

Game Pack Reality is 3-Dimensional, Augmented and Virtual reality based software application that provides variety of games on a single platform. The application would provide a combination of these technologies, so any user would be attracted towards this application. Game Pack Reality application involves eight major games with innovative gaming technology namely:

❖ 3-Dimensional Games:

1. Runner

- The goal of this game is to hop over the obstacles and beat your own high-score.
- Here the user is provided with a horizontal game view where the player model moves dynamically on a platform with various obstacles (Models: Bat and Ghost).
- The more you are accurate in hopping, the more you survive.
- As soon as you collide with an obstacle, the player will be deactivated and the game can be played again by selecting the Replay option.
- At any point of time, player is allowed to move back to the Main Menu or Quit.

2. Cubber

- The aim of the player in this game is to avoid crashing into the obstacles or rolling off the platform while moving forward to win the game and reach the next level.
- Here the camera follows the player while the player moves dynamically on a platform.
- Once you finish a level you would be redirected to another level.
- As soon as you collide with an obstacle, the same game level will be reloaded.
- At any point of time, player is allowed to move back to the Main Menu or Quit.

❖ Augmented Reality Games:

3. Duck Strike

- The purpose of this game is to shoot as many ducks as possible with a shotgun.
- Player will require a specific image-target to be detected by the device camera to load the game. The game is overlaid on the real-world environment.
- The player is provided with 2 lives and 3 bullets initially.
- The rounds are generated dynamically which informs about the target ducks to be shot.
- As soon as the target is fulfilled, a new round is spawned.
- On each misfire, a bullet is decreased, while on three consequent misfire, a life is lost.
- On every successful fire, bullet count remains the same and the score will be incremented.
- The highest number of the ducks hit by the player is stored as a high-score.
- Appropriate sounds and effects are added to enhance user's experience.
- The player can also restart the game when the game is over.
- At any point of time, player is allowed to move back to the Main Menu or Quit.

4. Fight Master

- The intent of this game is to smash down the Artificial Intelligence bot with punches and kicks.
- The game is superimposed battle ground environment on top of a user's actual environment where the user would place the specific image-target.
- The Artificial Intelligence bot also attacks the player, which in turn can reduce the player's health.
- While the player hits the AI bot on his damage area, the health of the enemy would be reduced.
- When any of the fighter's health is reduced to zero, he is knocked out and the opponent wins.
- Appropriate sounds and animations are added to enhance user's experience.
- The player can also restart the game when the game is over.
- At any point of time, player is allowed to move back to the Main Menu or Quit.

5. Ball Maze

- The scheme of this game is to reach to the other end of the maze.
- This game is designed to recall childhood memories of playing the maze game.
- The ball is re-spawned when it falls off the maze.
- The game is overlaid by sensing the image target of maze, so the user could interact with the mixed environment.
- The lowest time taken by the player to complete is stored as the high-score.
- The ongoing time of the game is shown at the top of the screen to keep track of it.
- The player can also restart the game when the game is over.
- At any point of time, player is allowed to move back to the Main Menu or Quit.

❖ Virtual Reality Games:

6. Car Race

- The aim of this game is to reach the finish line before Artificial Intelligence Car(Competitor).
- The game is build in virtual environment so the user can feel a new world while sitting anywhere in the world.
- The player moves according to the joystick movement.
- Sound effects are added to enhance the user's experience.
- The game would restart automatically as soon as the player reaches the finish line.
- The winner of the game is the one who reaches the finish line first.
- User needs a VR-Box and a joystick (controller) in order to play the game.
- User can go back to main menu at any point of time, by clicking on escape("X") on VR screen.

7. Bowling

- The purpose of this game is to hit as many pins as you can.
- The game is built in virtual environment so the user can feel a new world while seating anywhere in the world.
- The score is the pins hit per round.
- Ball follows camera movement, for setting the angle of the throw.
- Ball is thrown in the direction of camera on button press from the VR-Controller.
- User needs a VR-Box and a joystick (controller) in order to play the game.
- The game would restart automatically after a span of time, until the ball is thrown.
- User can go back to main menu at any point of time, by clicking on escape("X") on VR screen.

8. Hurdles

- The objective of this game is to hop over all possible the hurdles in minimum time.
- The game is built in virtual environment so the user can feel a new world while seating anywhere in the world.
- Time and score of missed hurdles can be seen on the screen.
- Once the player misses any hurdle, the speed will be decreased.
- Camera follows player in order to keep moving.
- Player can jump through button press from the VR-Controller.
- User needs a VR-Box and a joystick (controller) in order to play the game.
- The game would restart automatically after the player reaches the Finish Line.
- User can go back to main menu at any point of time, by clicking on escape("X") on VR screen.

➤ **Augmented Reality:**

- A technology that superimposes a computer-generated image on a user's view of the real world.
- Augmented reality is all about viewing synthetic-environment.
- Augmented reality is an interactive experience of a real-world environment.
- The objects that reside in the real-world are "augmented" by computer-generated perceptual information.
- Types of Augmented Reality:
 - Marker Based Augmented Reality-Some also call it to image recognition, as it requires a special visual object and a camera to scan it. It may be anything, from a printed QR code to special signs. The AR device also calculates the position and orientation of a marker to position the content, in some cases. Thus, a marker initiates digital animations for users to view, and so images in a magazine may turn into 3D models.



- Marker-less Augmented Reality- location-based or position-based augmented reality, that utilizes a GPS, a compass, a gyroscope, and an accelerometer to provide data based on user's location. This data then determines what AR content you find or get in a certain area. With the availability of smart phones this type of AR typically produces maps and directions, nearby businesses info. Applications include events and information, business ads pop-ups, navigation support.



- Famous Augmented Reality examples:

- Pokémon-Go
- Snapchat
- Google SkyMap

➤ **Image-Target:**

- Image Targets represent images that Vuforia Engine can detect and track.
- Unlike traditional markers, data matrix codes, and QR codes, Image Targets do not need special black and white regions or codes to be recognized.
- The Engine detects and tracks the features that are naturally found in the image itself by comparing these natural features against a known target resource database.
- Once the Image Target is detected, Vuforia Engine will track the image as long as it is at least partially in the camera's field of view.

Image-Target for Game Pack Reality:



➤ **Virtual Reality:**

- The definition of 'virtual' is near and reality is what we experience as human beings. So the term 'virtual reality' basically means 'near-reality'
- Virtual reality is the creation of a virtual environment presented to our senses in such a way that we experience it as if we were really there.
- A person using virtual reality equipment is able to "look around" the artificial world, move around in it, and interact with virtual features or items.
- Virtual Reality's most immediately-recognizable component is the head-mounted display (HMD).



VR-Box allows you to see thought virtual world, while placing the headset on you head.

VR-Controller allows the user to control the virtual environment (example: throw a ball while playing a game or switching between virtual body parts)

➤ Augmented Reality vs. Virtual Reality

- Virtual Reality and Augmented Reality are two sides of the same coin. You could think of Augmented Reality as VR with one foot in the real world: Augmented Reality simulates artificial objects in the real environment; Virtual Reality creates an artificial environment to inhabit.
- In Augmented Reality, the computer uses sensors and algorithms to determine the position and orientation of a camera. AR technology then renders the 3D graphics as they would appear from the viewpoint of the camera, superimposing the computer-generated images over a user's view of the real world.
- In Virtual Reality, the computer uses similar sensors and math. However, rather than locating a real camera within a physical environment, the position of the user's eyes are located within the simulated environment. If the user's head turns, the graphics react accordingly. Rather than compositing virtual objects and a real scene, VR technology creates a convincing, interactive world for the user.
- Unlike virtual reality, which requires you to inhabit an entirely virtual environment, augmented reality uses your existing natural environment and simply overlays virtual information on top of it. As both virtual and real worlds harmoniously coexist, users of augmented reality experience a new and improved natural world where virtual information is used as a tool to provide assistance in everyday activities.
- Unlike Virtual Reality (VR), AR does not create the whole artificial environments to replace real with a virtual one. AR appears in direct view of an existing environment and adds sounds, videos, and graphics to it.



Project Plan

➤ Objective

The world is moving towards a digital life and due to hard-working life, people don't get much time for entertainment and so aim is to provide a little fun and giving best experience in playing the game with different technologies.

There aroused a requirement of a system for amusement from the daily chore.

Game Pack Reality gives access to multiple games on single platform to users.

This would save users time in downloading different gaming applications.

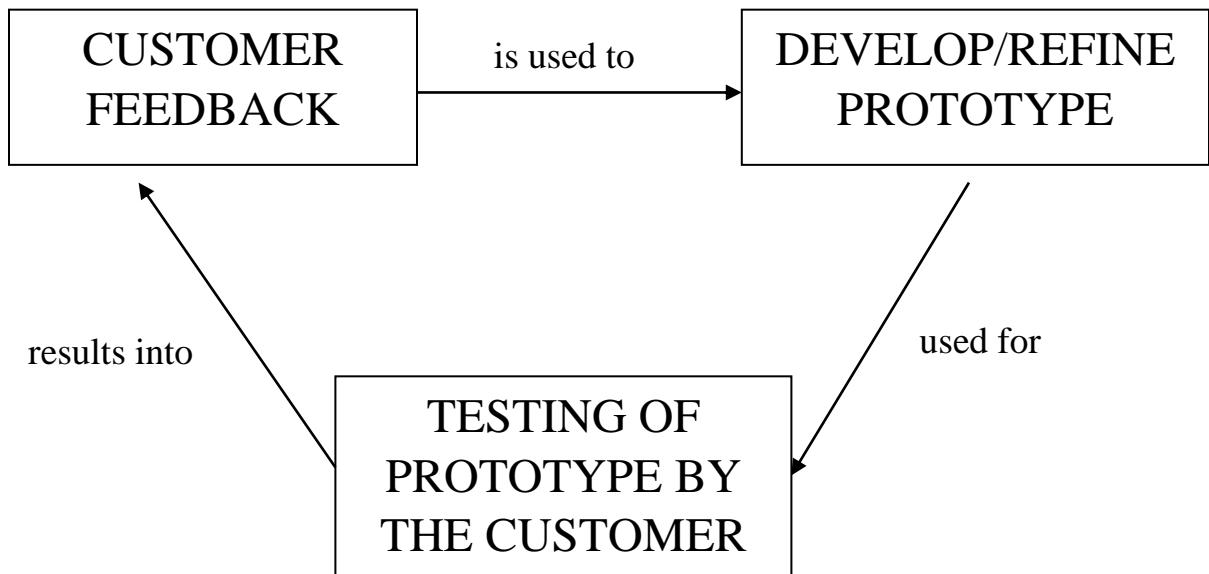
Moreover GPR provides a special platform by merging emerging gaming technologies to enhance the feature.

The main objective of GPR is to bring revolution in gaming world by providing a single platform for different types of games in terms of technology as well as category.

“Until now no industry/firm has even tried to merge these emerging technologies together on a single platform but GPR is the first application to merge different concepts (like: 3-Dimensional, Augmented Reality and Virtual Reality) and unite them to make a successful application.”

Game Pack Reality is all about impressing and elevating the gaming experience which could possibly be a revolution in the gaming world by combining the best of gaming technologies together on a single platform.

➤ **Software Model-Traditional (Prototype Model)**



❖ **Reason for Selecting the Model:**

- Increased user involvement in the product even before its implementation.
- Since a working model of the system is displayed, the users get a better understanding of the system being developed.
- Reduces time and cost as the defects can be detected much earlier.
- Quicker user feedback is available leading to better solutions.
- Missing functionality can be identified easily.
- Confusing or difficult functions can be identified.
- The customers get to see the partial product early in the life cycle. This ensures a greater level of customer satisfaction and comfort.
- New requirements can be easily accommodated as there is scope for refinement.
- Flexibility in design.

➤ **Work division (w.r.t time)**

Month	Work Division
January	<ul style="list-style-type: none"> • Defining the Scope • Deciding software and software methodology • Learning of Augmented reality • Designing the flow for Runner and Application UI • Building a 3D game runner for practice • Deciding models for runner • Working on high score of runner • Making runner work properly for android device • Completion of runner
February	<ul style="list-style-type: none"> • Started building a learning app for AR like image target detection, Placing model on image target • Designing the flow for Duck Strike, Fight Master • Started building Duck Strike • Deciding models for Duck Strike • Features and functions for Duck Strike • Fight Master Development Started • Finalizing models for Fight Master • Learning AI BOT for Fight Master and developing it • Animation for Kick, punch, knockout, reaction in Fight Master • Designing the flow for Ball Maze and Cubber • Learning different materials to use in Cubber and Ball Maze • Started building Cubber
March	<ul style="list-style-type: none"> • Designing levels for Cubber • Learning VR Technology and how to use Google cardboard • Designing the flow for the Hurdles and Car Race • Started Building Hurdles • Finalized models for hurdles • Started Building Ball Maze • Building maze in unity for Ball Maze • Started Building Car race • Learning AI car working and developing it • Learning self-driving car controls and developing it

	<ul style="list-style-type: none"> • Solving Duck Strike Bugs • Solving Fight Master Bugs • Completion of Fight Master
April	<ul style="list-style-type: none"> • Learning Joystick controls for VR Games • Completion of Car Race and Hurdles • Bug solving of Car Race • Started development of Bowling • Finalizing models for Bowling • Completion of Bowling • Designing final flow for Main Application • Building application GUI for GPR • Integrating games in GUI • Solving a lot of issues while integrating AR and VR • Testing of each and every thing in whole project after integrating

System Requirements

Device OS	
Android Operating System	5.0.x
Windows	10 UWP

Development OS	
Windows	7+
iOS(OS)	10.11

Unity Version	
Windows	2017.4+
iOS(OS)	2017.4+

Hardware and Software Requirements

➤ **Hardware Requirements:**

Android Mobile
Touch Screen
Minimum 1GB Storage
Device Camera
Minimum 2GB RAM
Virtual Reality Box
Virtual Reality Controller

➤ **Software Requirements:**

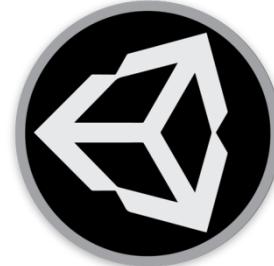
Android Mobile
Android Operating System (5.0 and above)
Bluetooth 2.0+

Tools and Technologies

➤ Tools:

❖ **Unity3D 2018.2:** Unity is so much more than the world's best real-time development platform. Used to create more than half the world's mobile games, Unity empowers creators with the most accessible and powerful real-time 3D content-creation platform available. Unity allowed us:

- Importing Models from Asset Store
- Importing Audio from Asset Store
- Importing SDK's from Asset Store
- Unity includes powerful and well-documented APIs with access to the complete range of Unity systems, including physics, rendering, and communications to enable a rich interaction model and integration with other systems.



❖ **Adobe Photoshop CS6:** Photoshop, the world's best imaging and graphic design software. Create and enhance photographs, illustrations, and 3D artwork. Design websites and mobile apps. Edit videos, simulate real-life paintings, and more. It's everything you need to make any idea real. Photoshop helped us to edit Models, its materials and also design game icons.



❖ **Vuforia Engine v8.1.10:** Vuforia is an augmented reality(AR) software development kit (SDK) for mobile devices that enables the creation of augmented reality applications. It uses computer vision technology to recognize and track planar images and simple 3D objects, such as boxes, in real time. Vuforia lets us:

- Create a database
- Store image-targets
- Import its packages to Unity



Vuforia License Manager:

The screenshot shows the Vuforia License Manager interface. At the top, there's a green header bar with the Vuforia logo and "vuforia™ Developer Portal". Below the header, a navigation bar has tabs for Home, Pricing, Downloads, Library, Develop (which is selected), and Support. Under the Develop tab, there are two sub-tabs: License Manager (selected) and Target Manager. In the main content area, the path "License Manager > AR_App" is shown. Below the path, the title "AR_App" is displayed with options to "Edit Name" and "Delete License Key". There are two tabs: "License Key" (selected) and "Usage". A large text area contains the license key: "AawN+4D////AAAABmWMKMJeznEd1iJCnJUAnBr4LbS69gExcFVqD5nOyRsWqWxb/VS3Zcwc9SjTDqbu/yn7tvTog4cZ40vk4WQiFeAaB7d0hbpX71Cz4+4bCmvIvUOQyoAkVmNec1a79rGbPUfV8bv3VTUgVrigLfosc4/WUwQxqXOKSA1T64a7QqNY+12AktT3N3L/4AnHkgpYp1/DR8oNIRnNpP/ngCc8Cg6dfE3k+kWOJzFUp3W4/+v/uc+SpevTa0gbZQWUPV/qAXX/mx1XXBkTPydvCwnrXtR8HSTwp6Ivu1W4hZjbJz98DJRzev8/XUYuIujB9udy45otndctOs09QAQurGGVLVDs3G8iY+PSLawBitwIeZwj".

Vuforia Database for Image Target and Image Recognition:

The screenshot shows the Vuforia Target Manager interface. At the top, there's a dark header bar with the Vuforia engine™ developer portal logo, "Home", "Pricing", "Downloads", "Library", "Develop" (selected), "Support", and user information "Hello marmeeek". Below the header, a green navigation bar has tabs for "License Manager" (selected) and "Target Manager". In the main content area, the path "Target Manager > Ar_Game > image_target_Maze" is shown. The title "image_target_Maze" is displayed with "Edit Name" and "Remove" options. To the right, detailed target information is shown: Type: Single Image, Status: Active, Target ID: 82865043675b/dcd863e5a707aa57a28, Augmentable: ★★★★★, Added: Apr 16, 2019 15:54, Modified: Apr 16, 2019 15:54. Below this, there's a preview image of the target, which is a yellow and black patterned square with a central character and text "Start to discover". At the bottom, there are buttons for "Update Target" and "Hide Features".

- ❖ **Android Studio v3.2.1:** Android Studio is the official integrated development environment for Google's Android operating system, built on Jet Brains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems. Android studio helped us building our application into our android phones in terms of APK.



- ❖ **Google Cardboard v1.9:** The Cardboard SDK for Unity allows you to easily bring your Unity 3D projects to VR on Android and iOS. It provides SDKs for many popular development environments.

- These SDKs provide native APIs for key VR features like user input, controller support, and rendering. We used Google cardboard to develop our VR applications by importing its package and using its controllers along with project settings.



- ❖ **Visual Studio 2017:** Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.



➤ **Technologies:**

- ❖ **C#** :It is a general-purpose, multi-paradigm programming language encompassing strong typing, lexically scoped, imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines. Coding in Unity is done through C#. Hence we used C# to write scripts for the game-objects.



System Analysis

➤ System Flow Diagrams

❖ GPR-UI

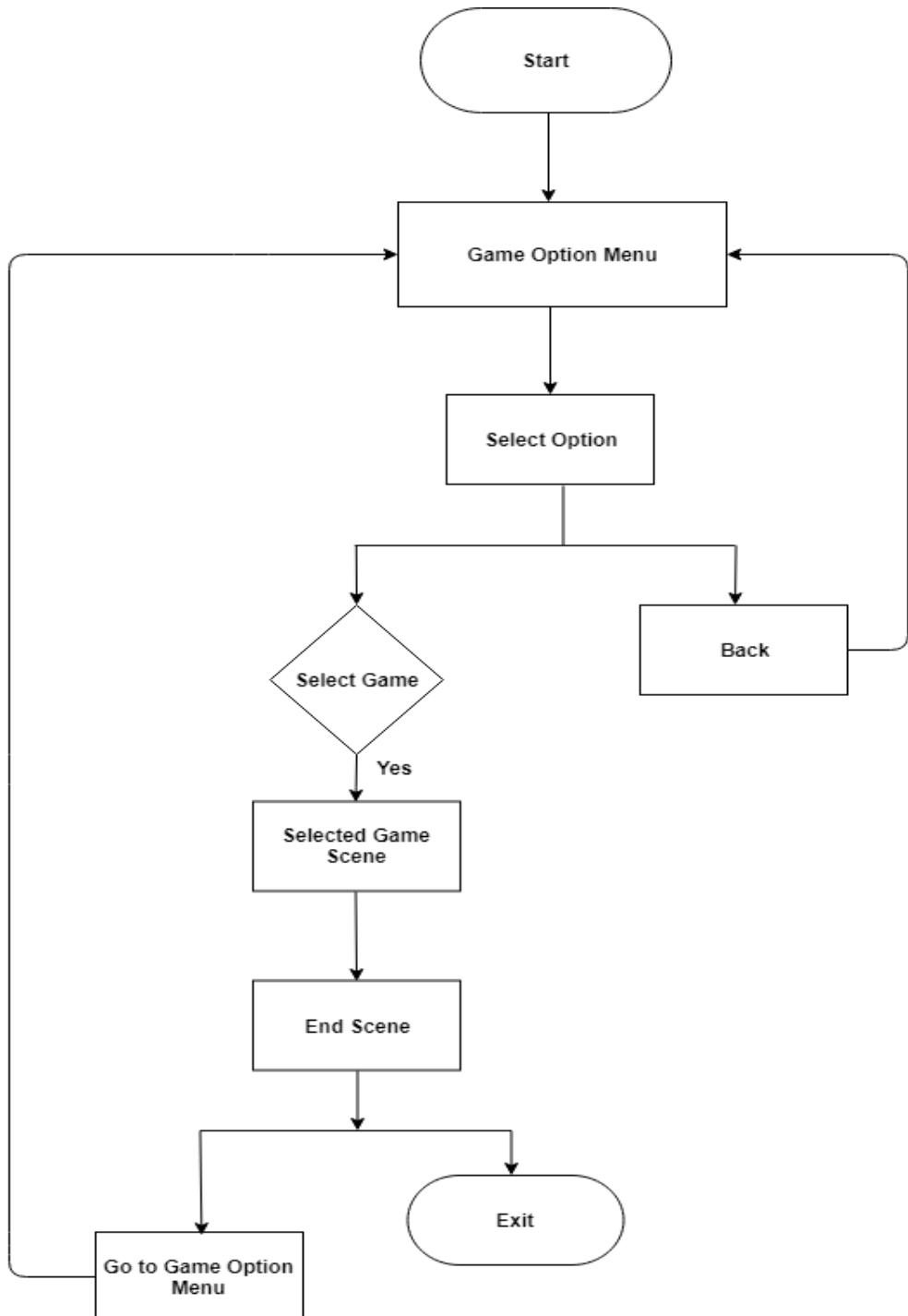


Figure 1 Game Pack Reality UI flow

❖ Runner (3D)

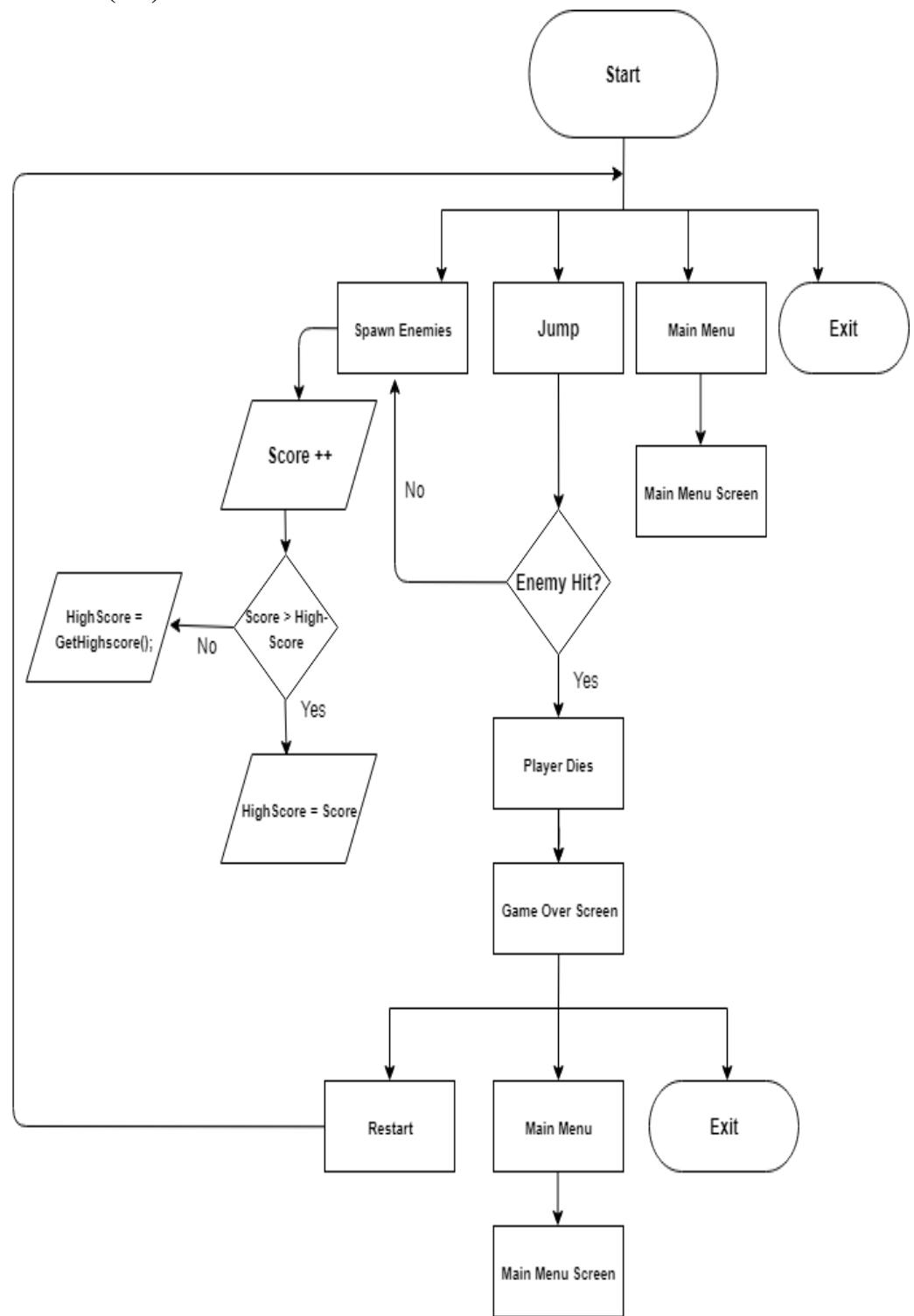


Figure 2Runner3D Flow

❖ Cubber (3D)

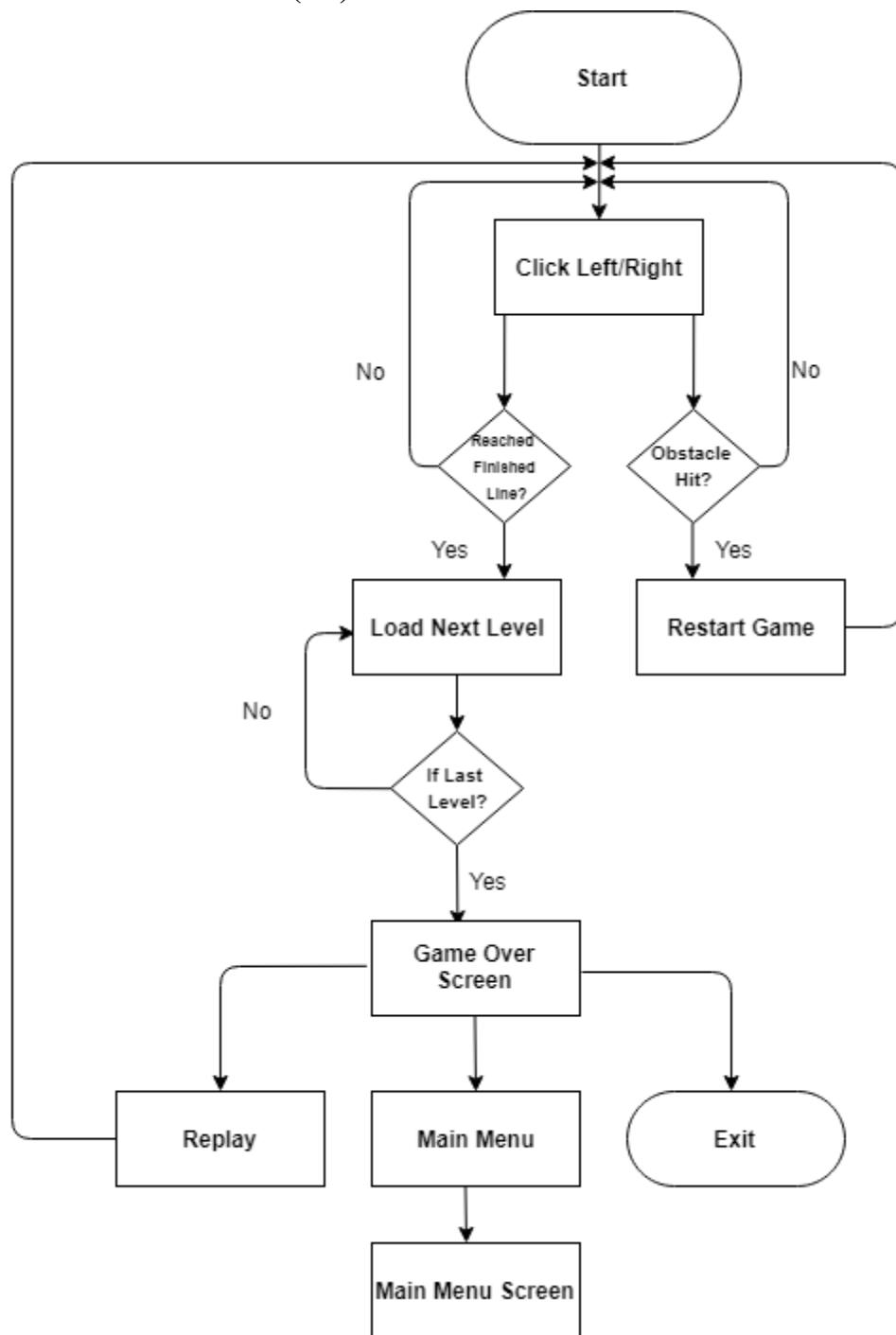


Figure 3 Cubber 3-D Flow

❖ Duck Strike (AR)

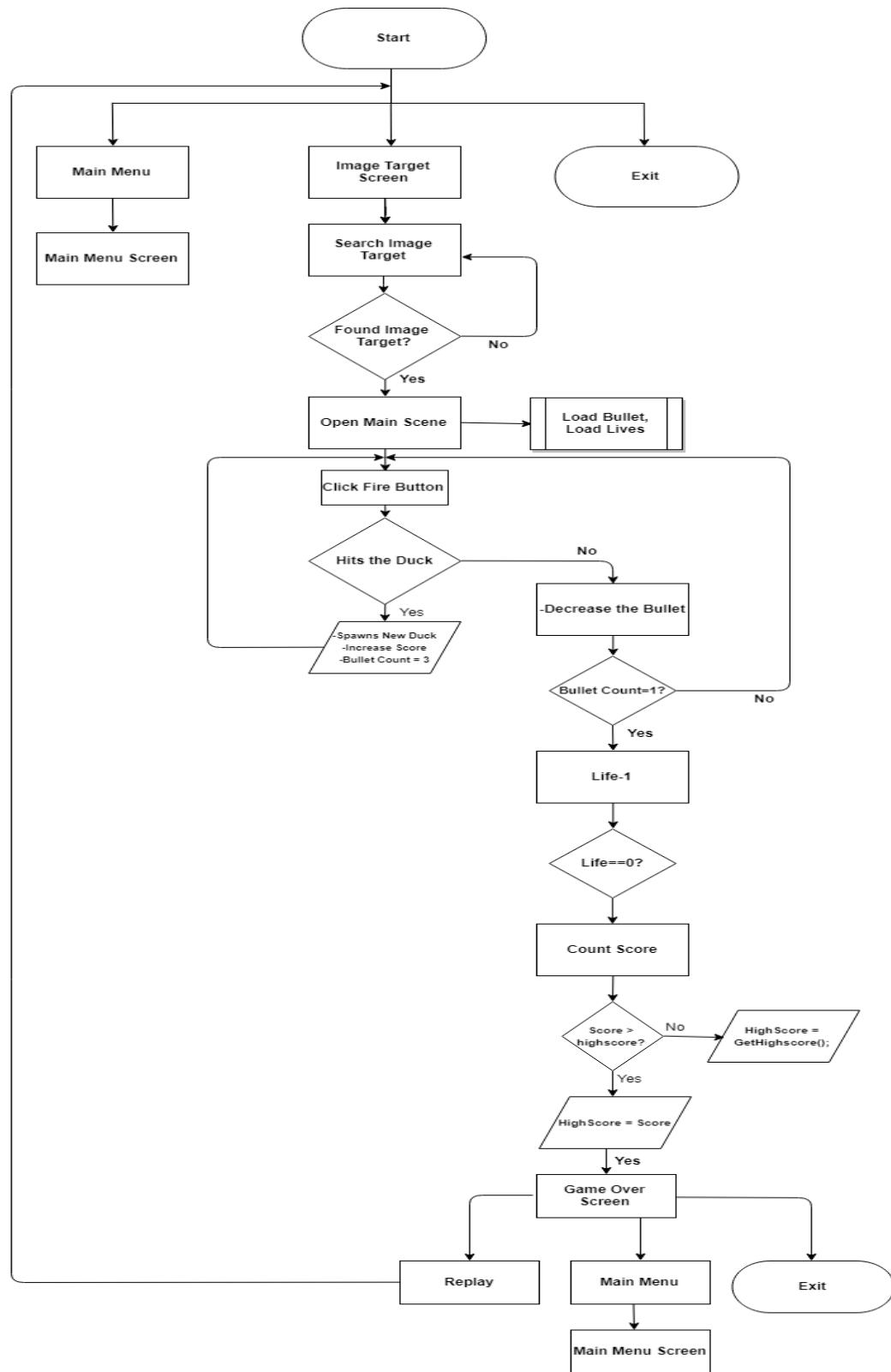


Figure 4 AR-Duck Strike Flow

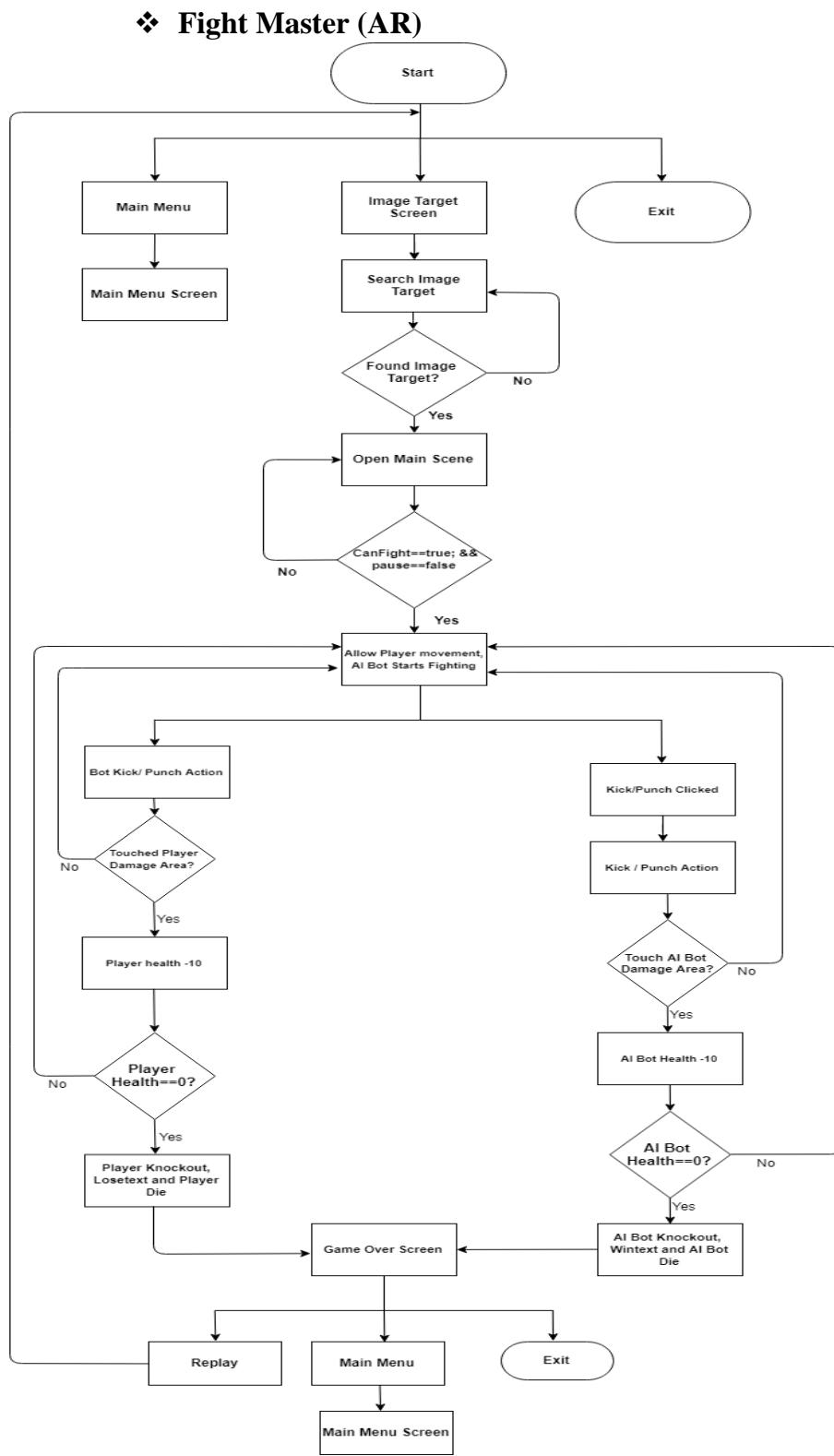


Figure 5 AR-Fight Master Flow

❖ Ball Maze (AR)

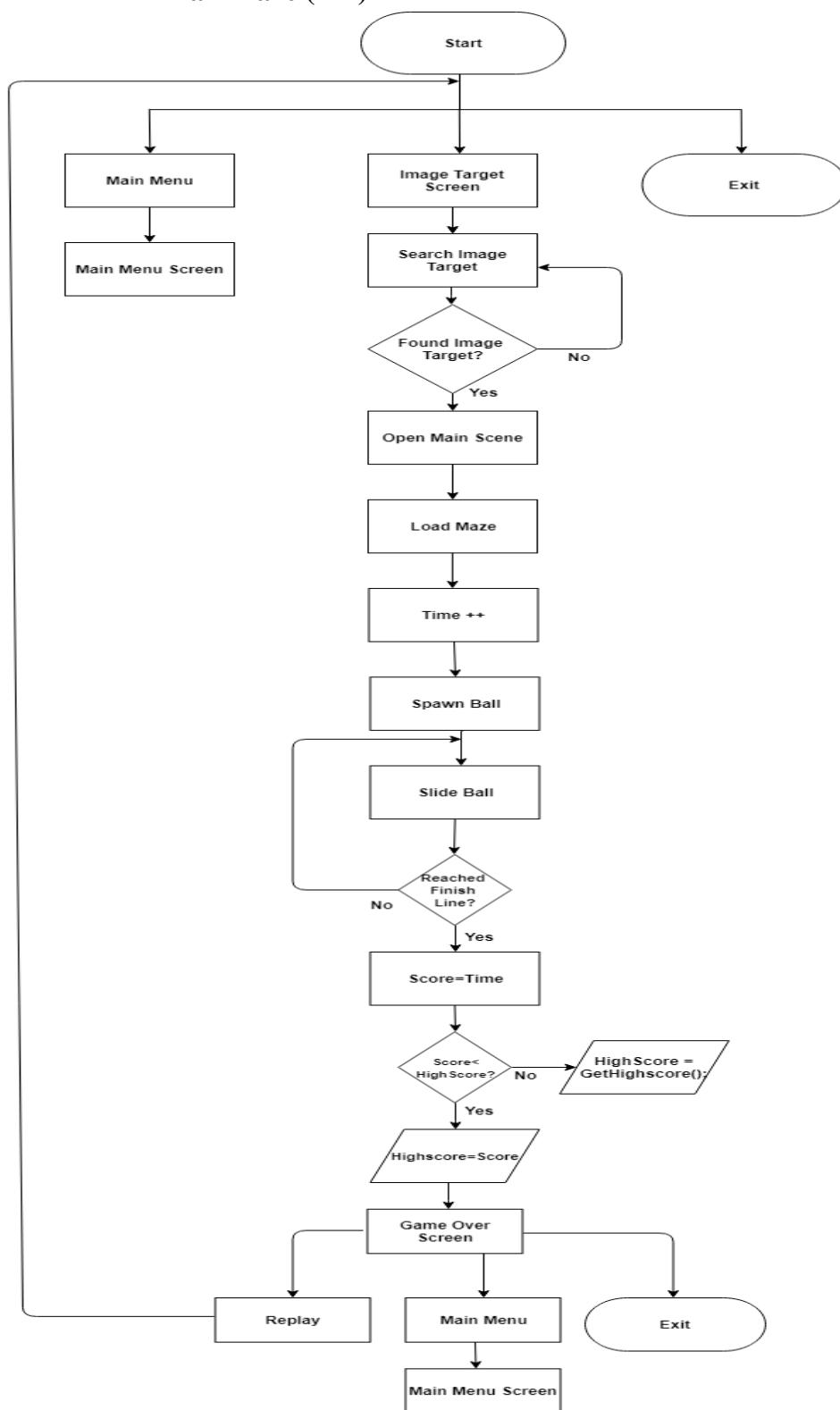


Figure 6 AR-Ball Maze Flow

❖ Car Race (VR)

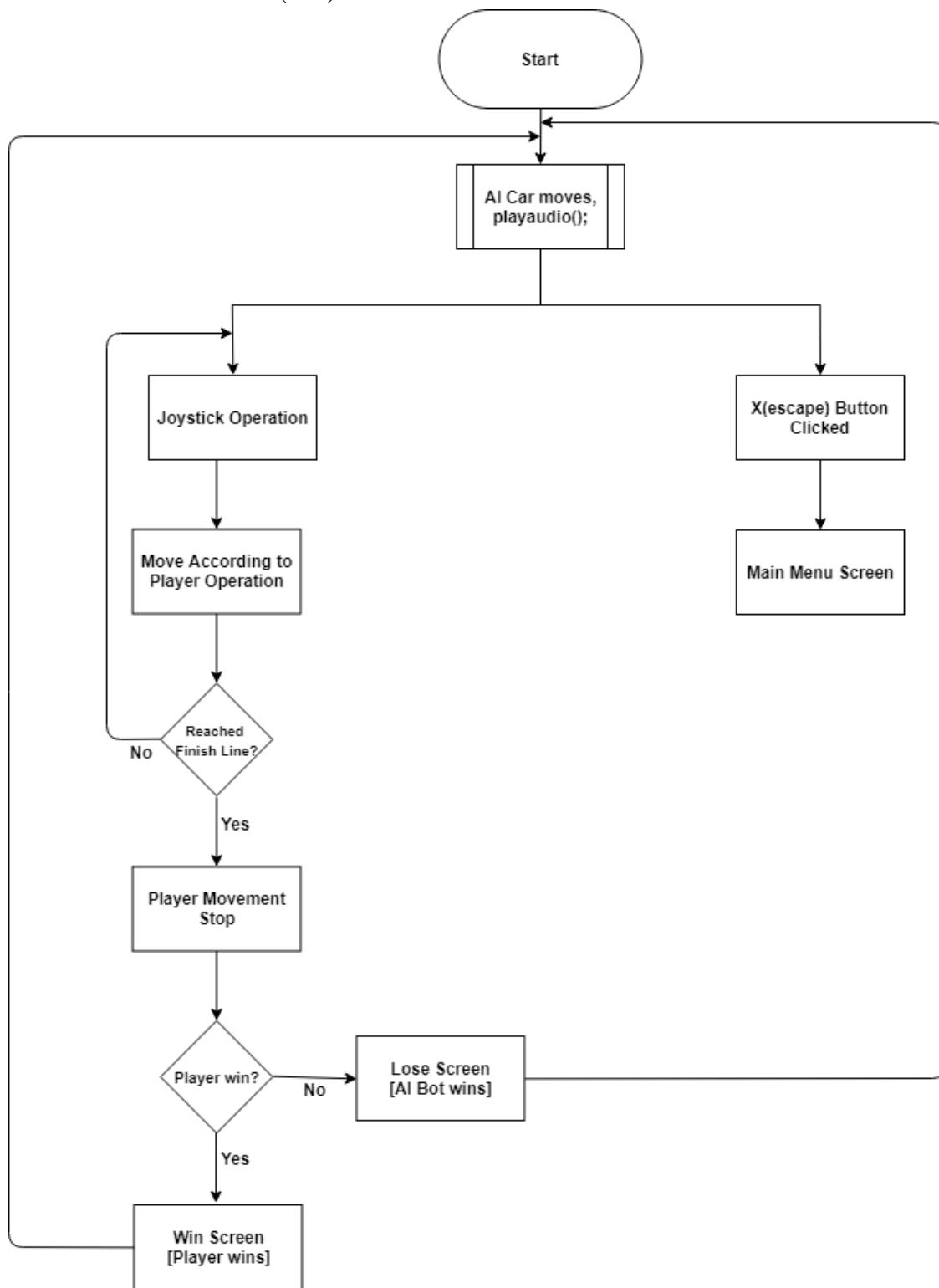


Figure 7 VR-Car Race Flow

❖ Bowling (VR)

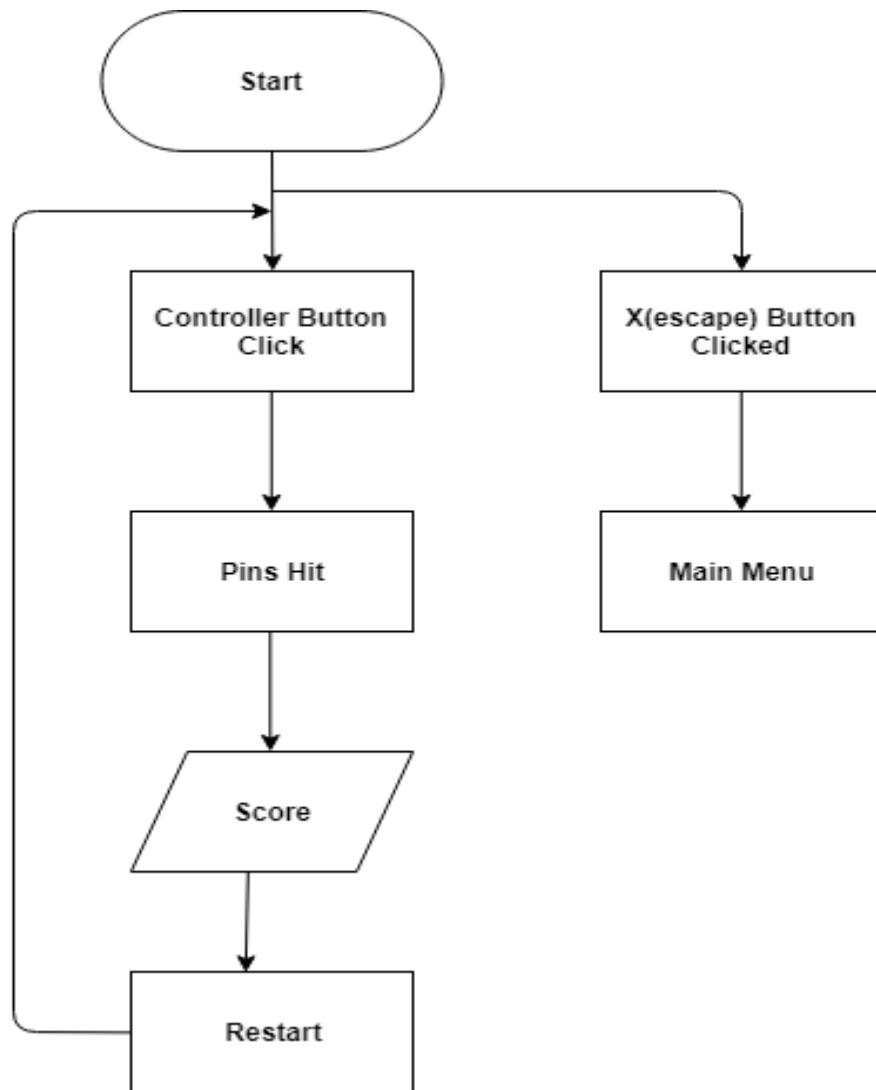


Figure 8 VR-Bowling Flow

❖ Hurdles (VR)

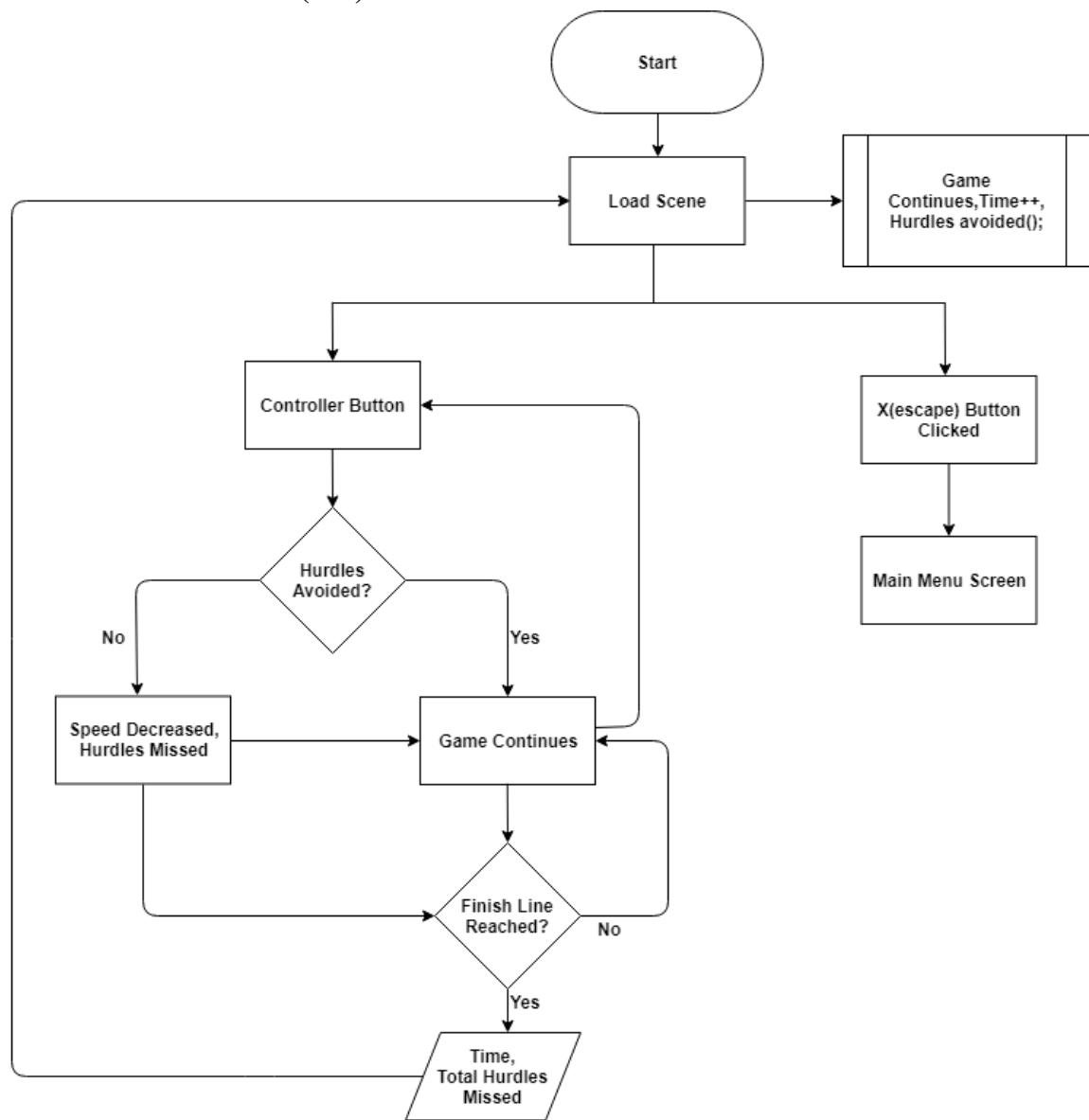


Figure 9 VR- Hurdles Flow

System Design

➤ 3D-Runner



Figure 10 3D- Runner: Player Model



Figure 11 3D- Runner: Obstacle Model



Figure 12 3D- Runner: Obstacle Model



Figure 13 3D- Runner: Obstacle Model

➤ AR-Duck Strike



Figure 14 AR-Duck Strike Duck Model



Figure 15 AR-Duck Strike Gun Model



Figure 16 AR-Duck Strike Park Model

➤ AR-Fight Master



Figure 17 AR-Fight Master Player Model



Figure 18 AR-Fight Master AI-BOT Model

➤ AR-Ball Maze

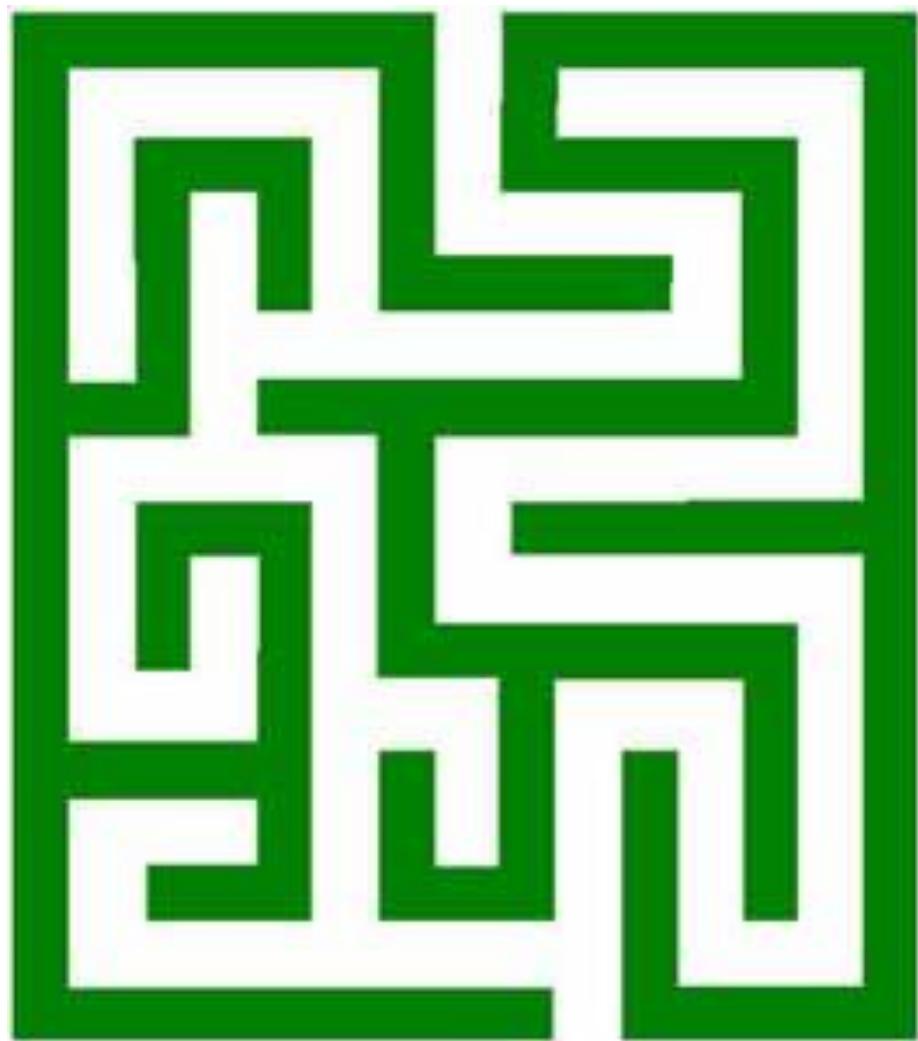


Figure 19 AR-Maze Maze Model



Figure 20 AR-Maze Ball Model

➤ VR-Car Race

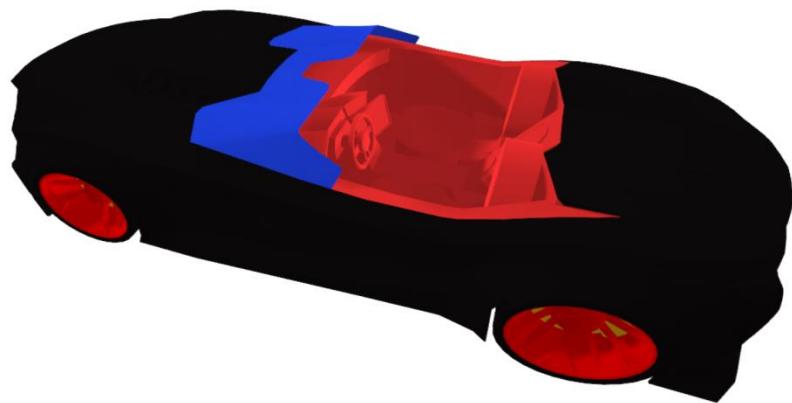


Figure 21 VR-Car Race Player-Car Model



Figure 22 VR-Car Race AI-Car Model

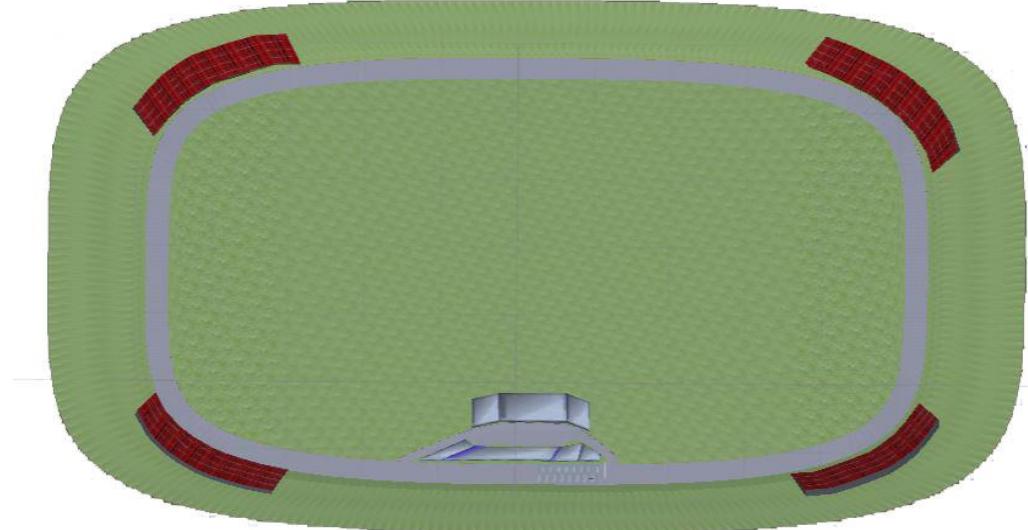


Figure 23 VR-Car Race Track Model

➤ **VR-Bowling**



Figure 24 VR-Bowling Pins Model



Figure 25 VR-Bowling Ball Model



Figure 26 VR-Bowling Room(Alley) Model

➤ VR-Hurdles

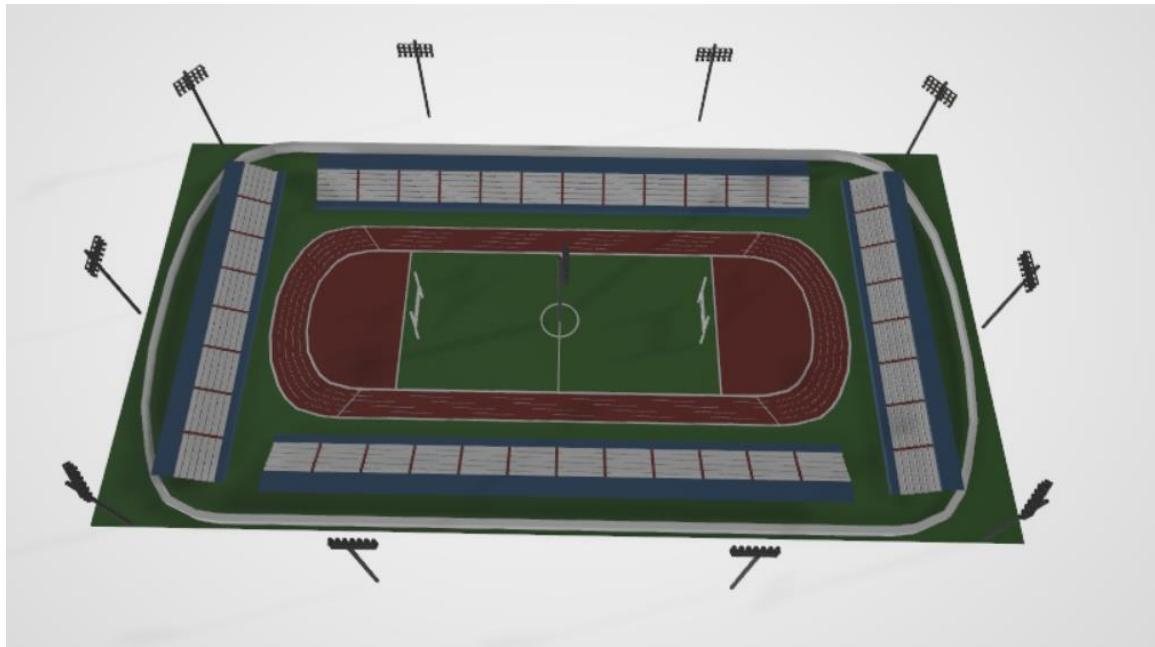


Figure 27VR-Hurdles Track Model



Figure 28VR-Hurdles Hurdle Model

Screenshots

Application User Interface



Figure 29 GPR Main Screen

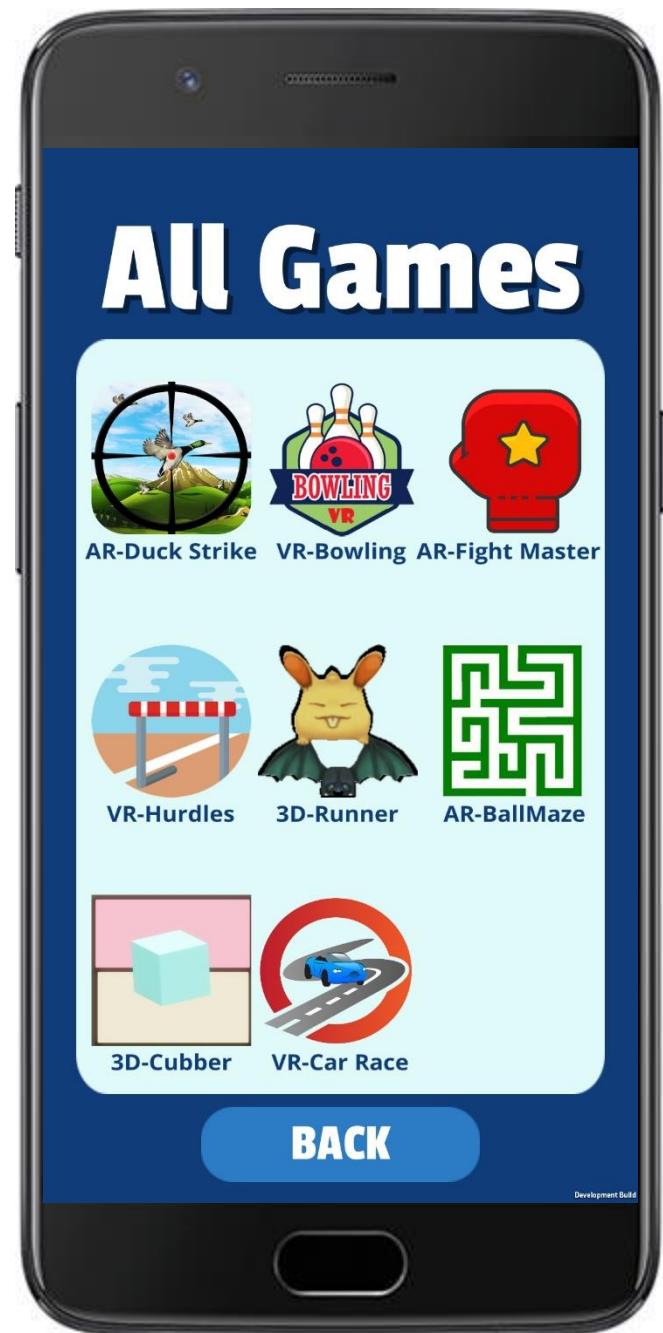


Figure 30 GPR All Games Screen

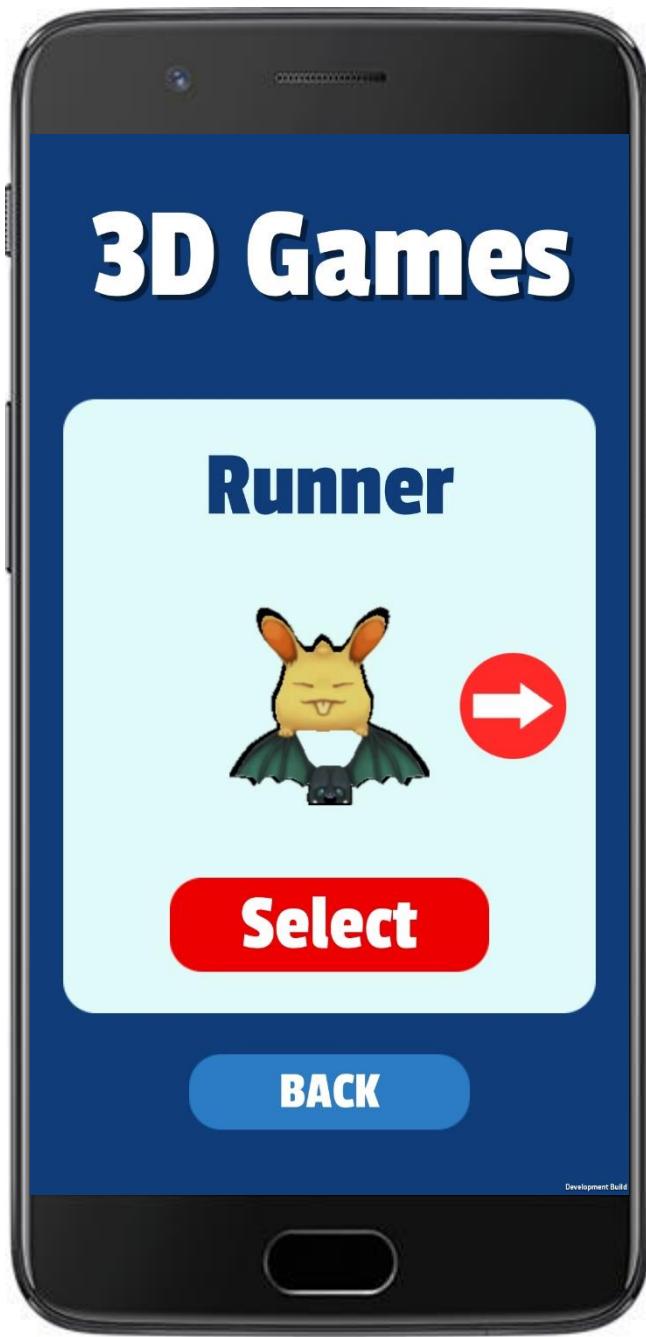


Figure 31 GPR 3D Game Main Menu Screen



Figure 32 GPR 3D Game Main Menu Screen



Figure 33 GPR AR Game Main Menu Screen



Figure 34 GPR AR Game Main Menu Screen



Figure 35 GPR AR Game Main Menu Screen

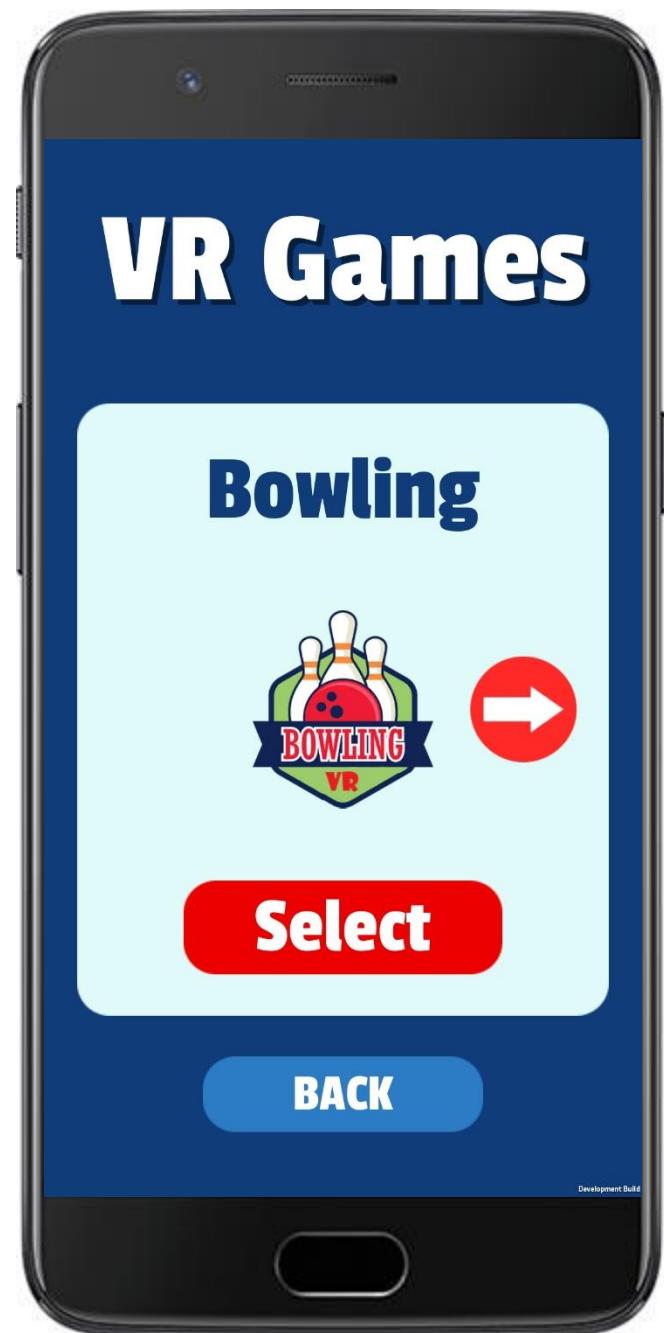


Figure 36 GPR VR Game Main Menu Screen



Figure 37 GPR VR Game Main Menu Screen

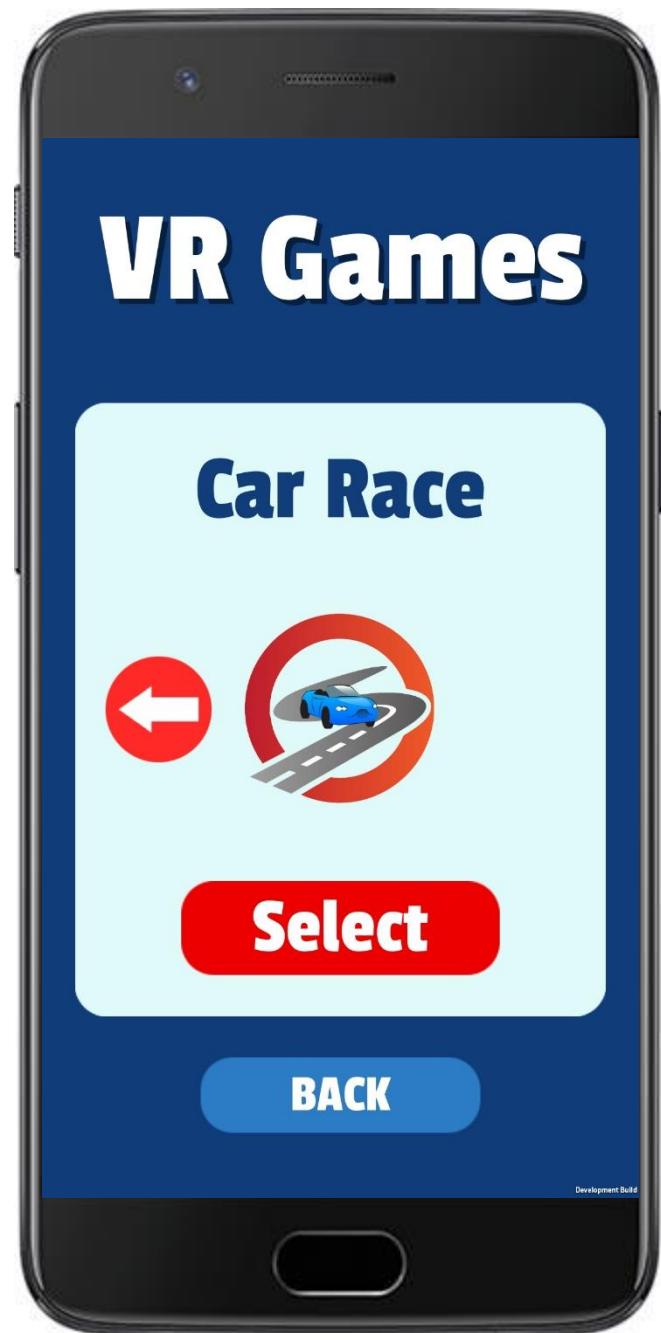


Figure 38 GPR VR Game Main Menu Screen

➤ **Runner**

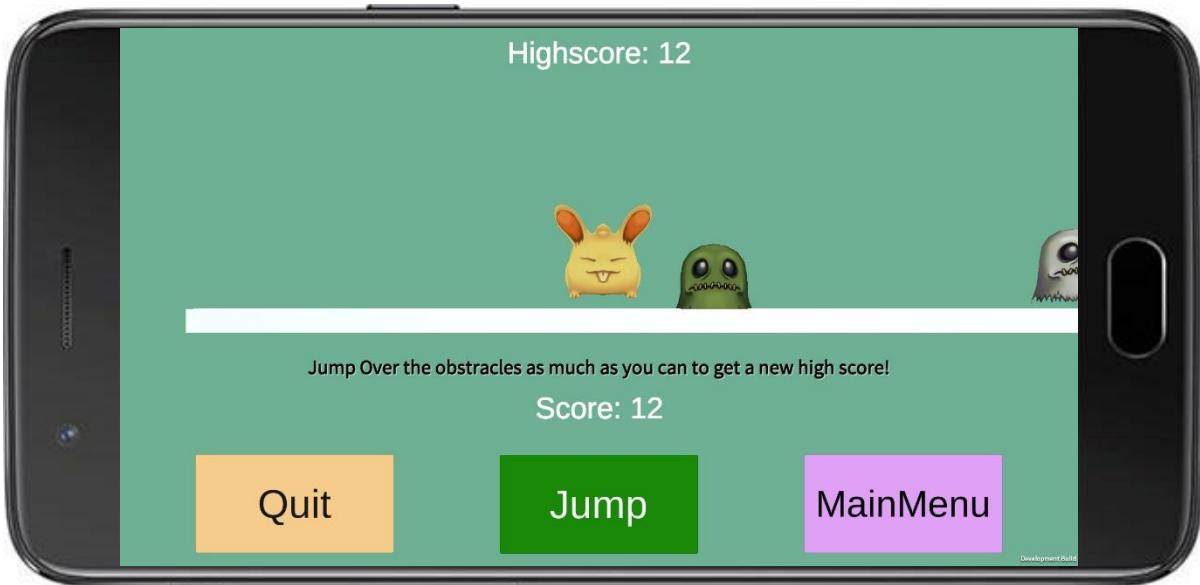


Figure 39 Runner Start Screen

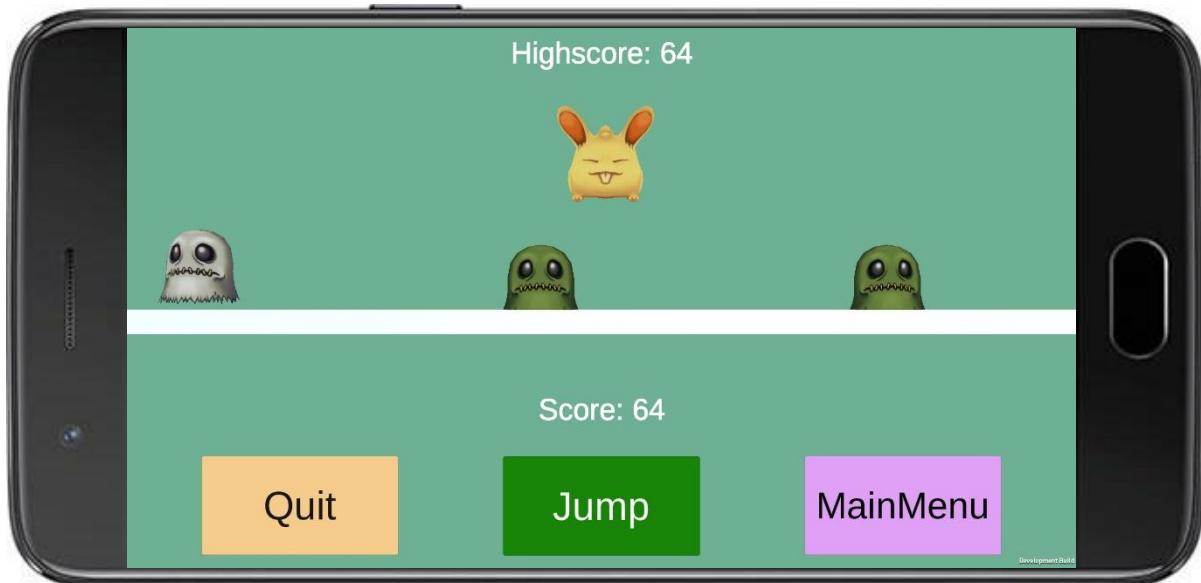


Figure 40 Runner Jump Screen

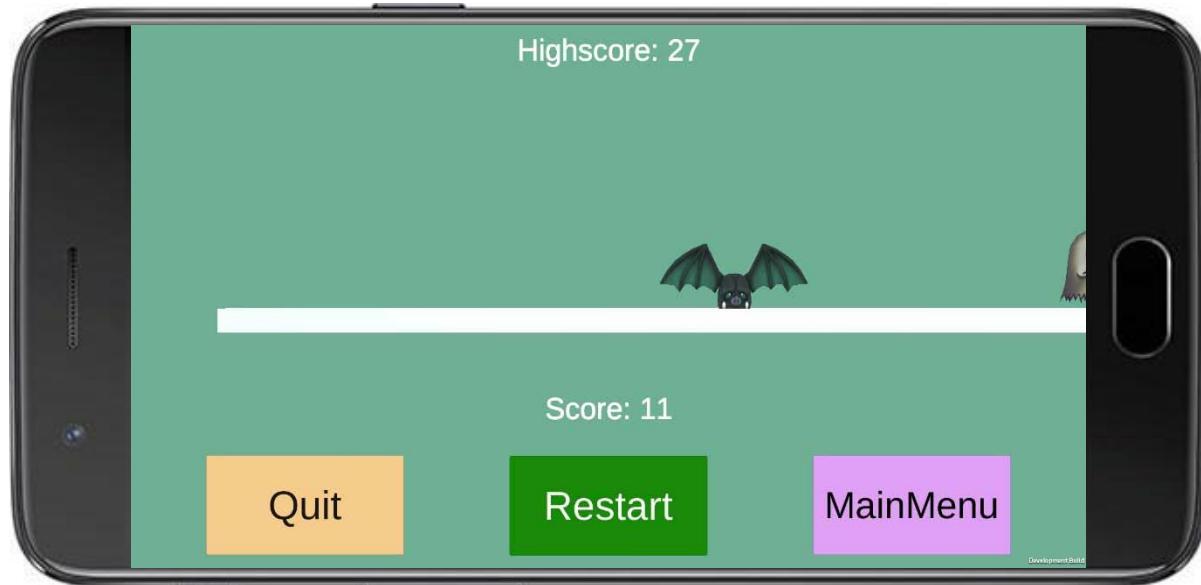


Figure 41 Runner End Screen

➤ **Cubber**

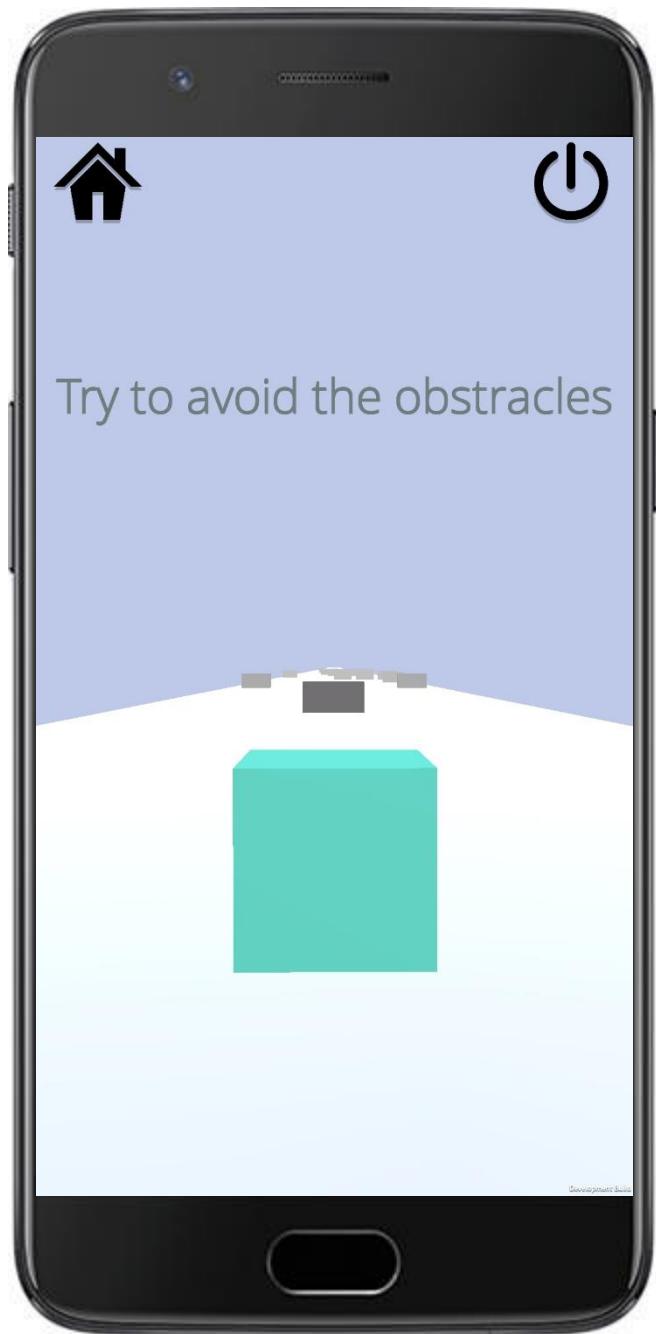


Figure 42Cubber Start Screen



Figure 43Cubber Level 1 Screen

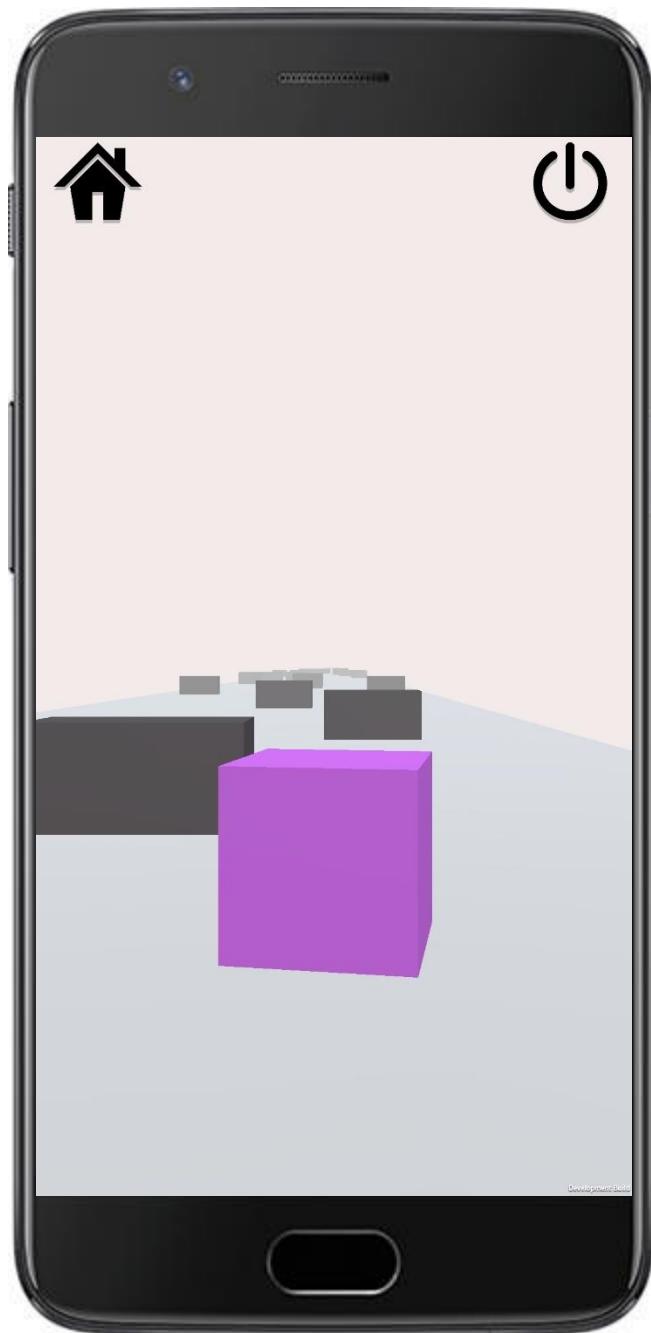


Figure 44 Cubber Hits the Obstacles Screen

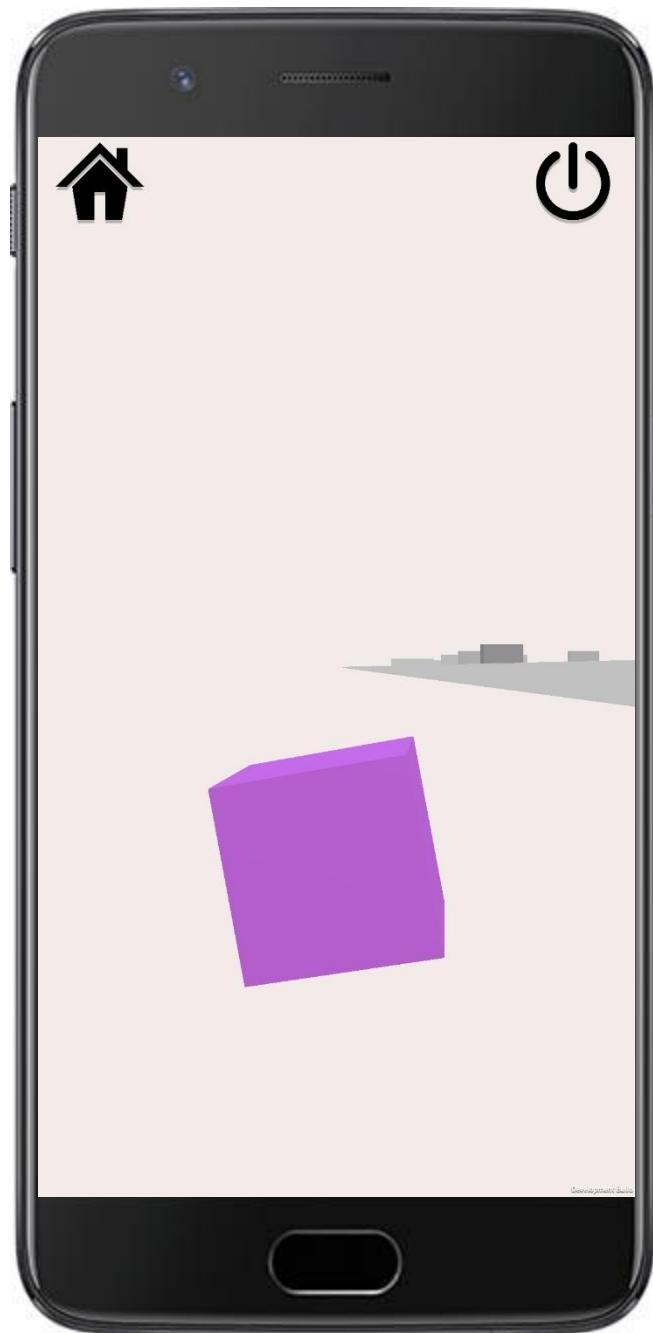


Figure 45 Cubber Hits the Obstacles Screen

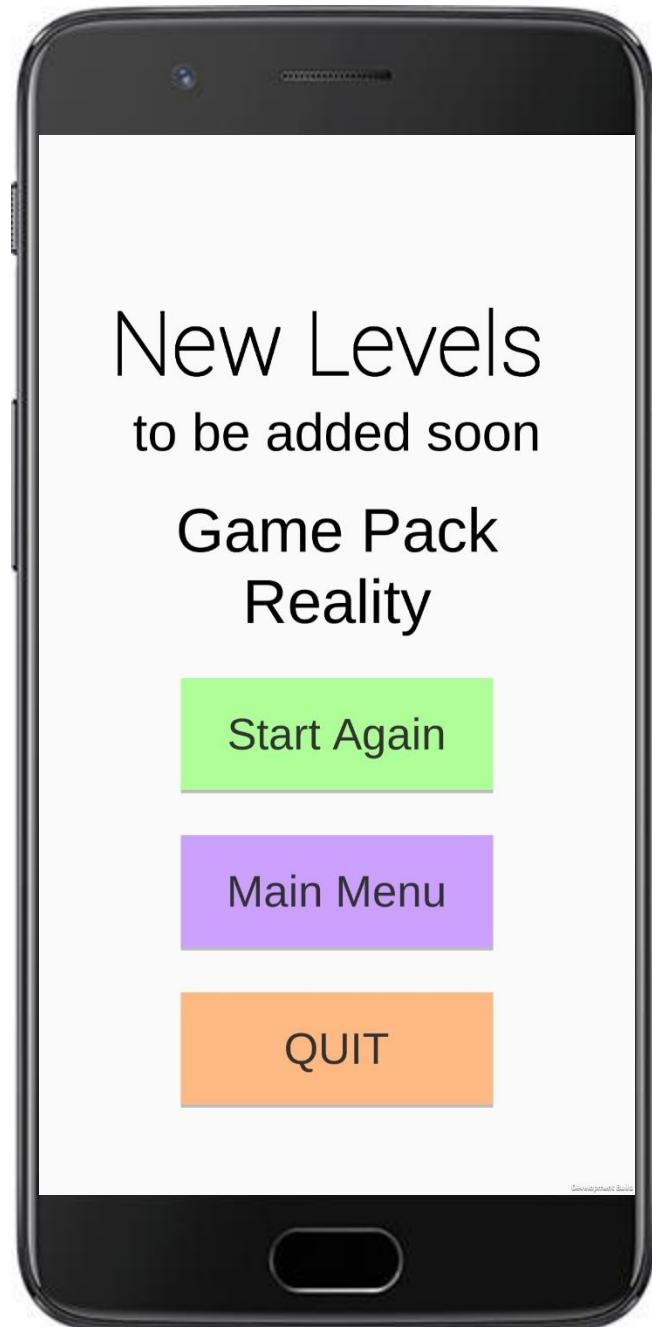


Figure 46Cubber End Screen

➤ Duck Strike



Figure 47 Duck Strike Image Target Screen



Figure 48 Duck Strike Bird Spawn Screen



Figure 49 Duck Strike Environment Screen



Figure 50 Duck Strike Hit Screen



Figure 51 Duck Strike End Screen

➤ **Fight Master**



Figure 52 Fight Master Image Target Screen

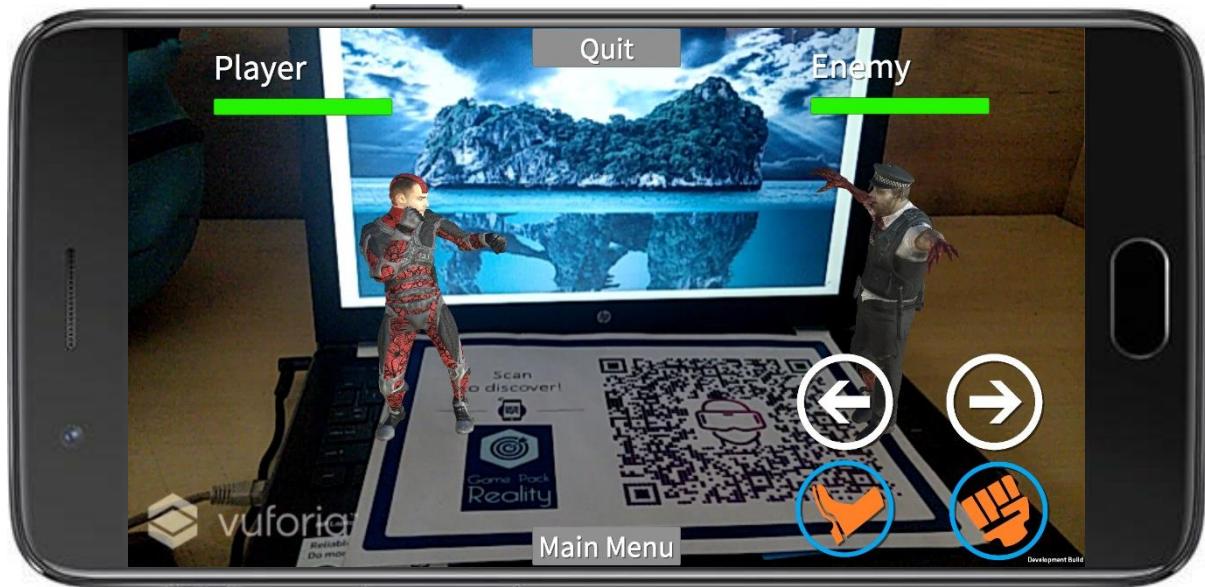


Figure 53 Fight Master Main Screen

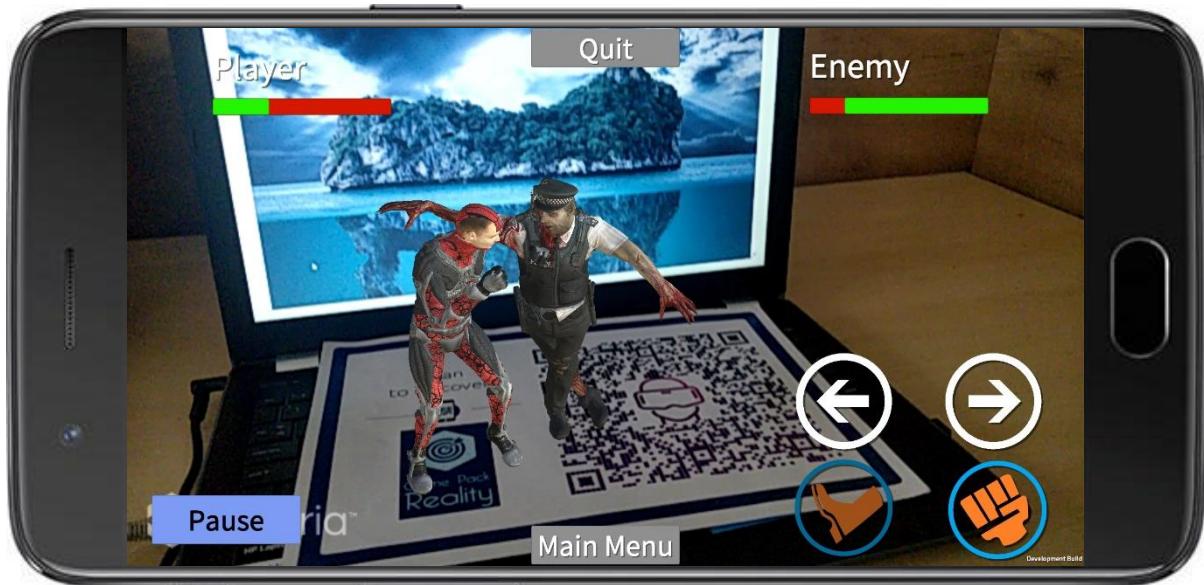


Figure 54 Fight Master Zombie Punch Screen



Figure 55 Fight Master Player Punch Screen

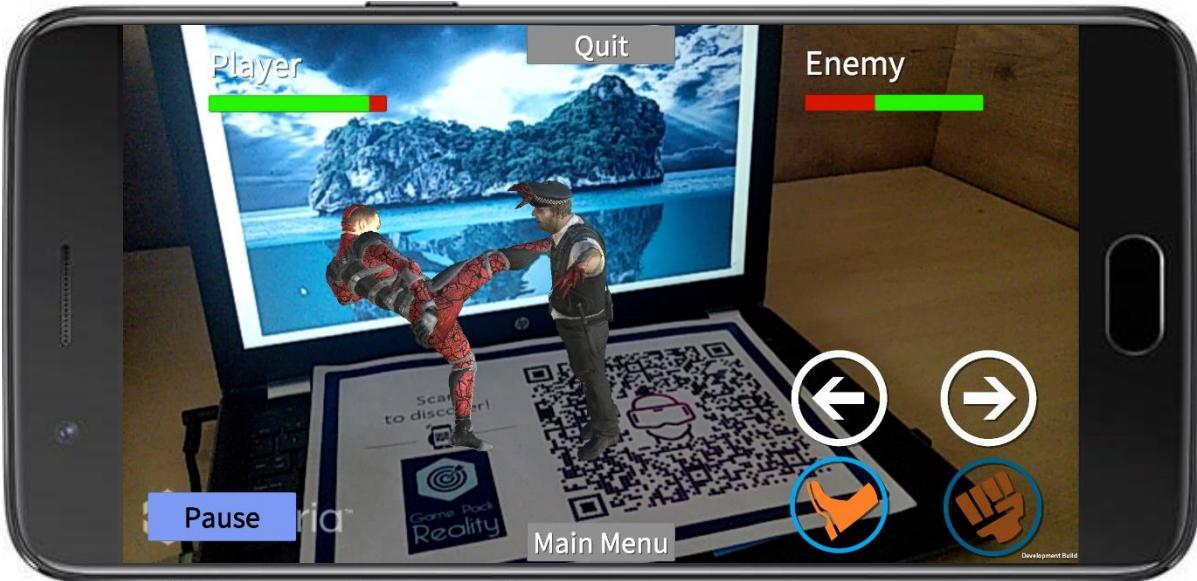


Figure 56 Fight Master Player Kick Screen

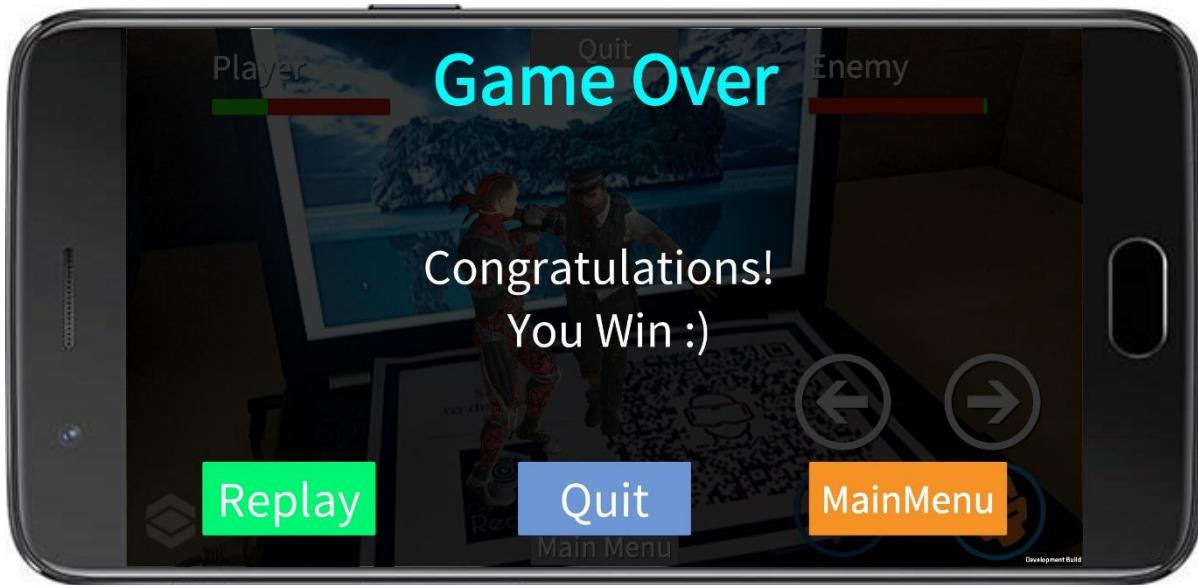


Figure 57 Fight Master Win Screen

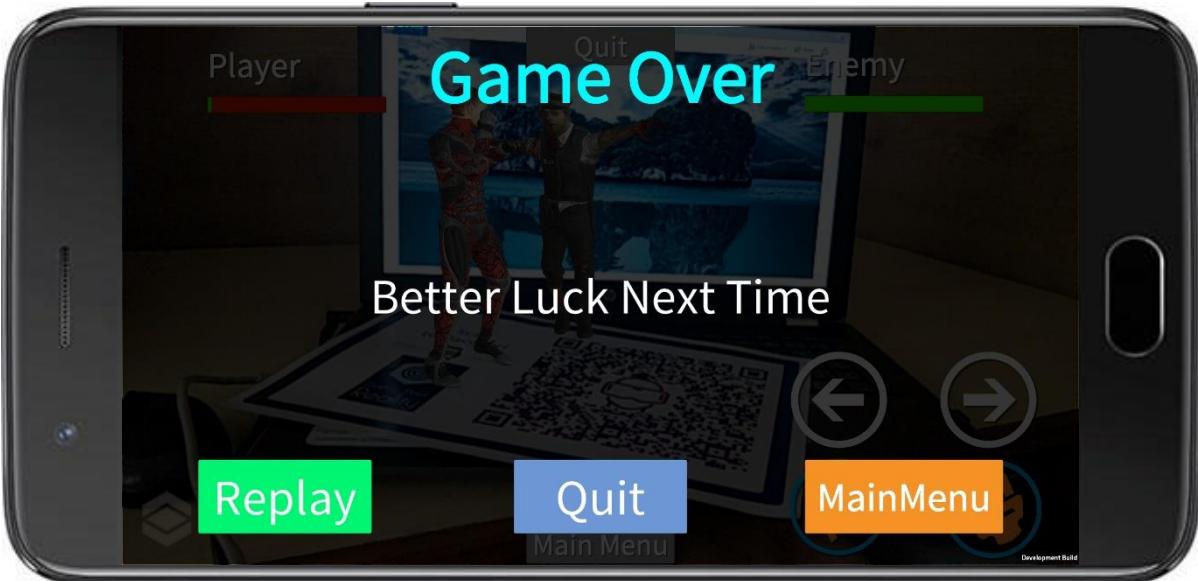


Figure 58 Fight Master Lose Screen

➤ **Ball Maze**



Figure 59 Maze Image Target Screen



Figure 60 Maze ball Spawn Screen

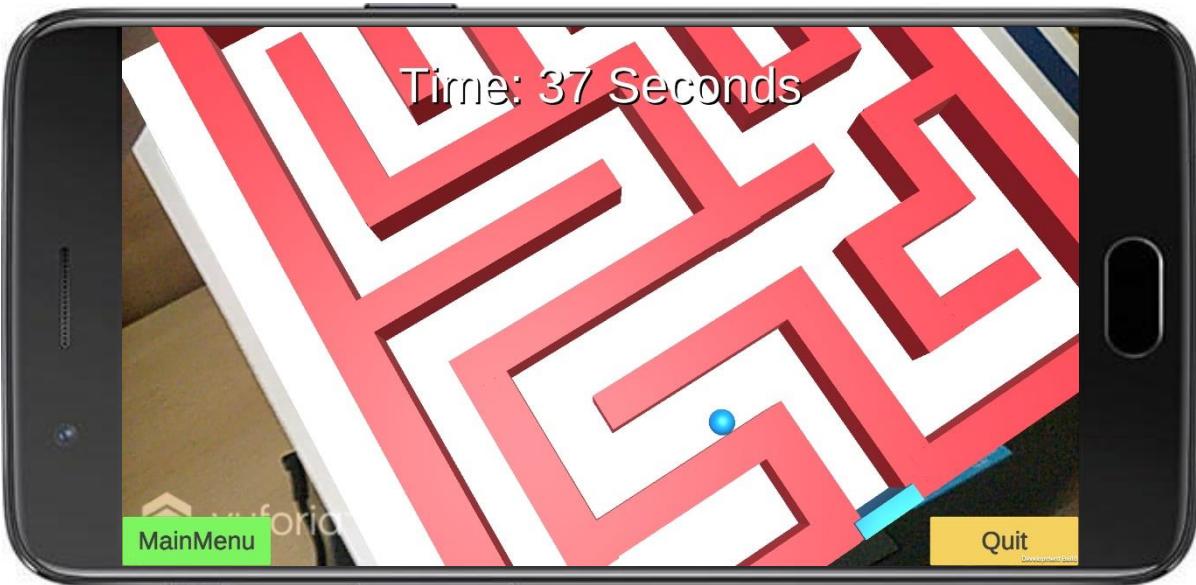


Figure 61 Maze ball Finish line Screen



Figure 62 Maze End Screen

➤ **Car Race**

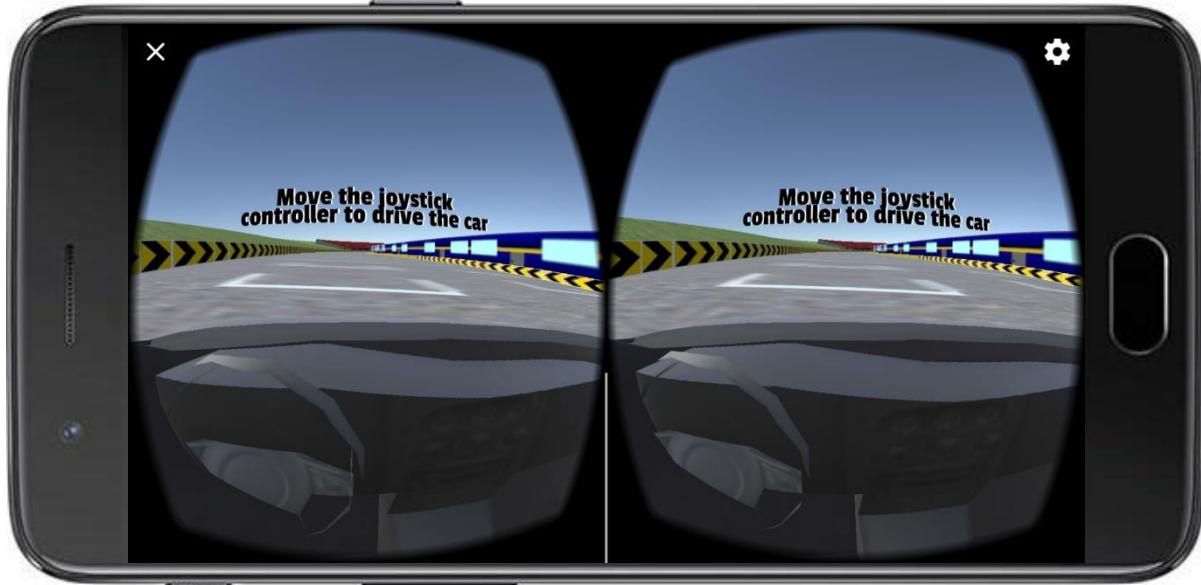


Figure 63 Car Race Starting Screen

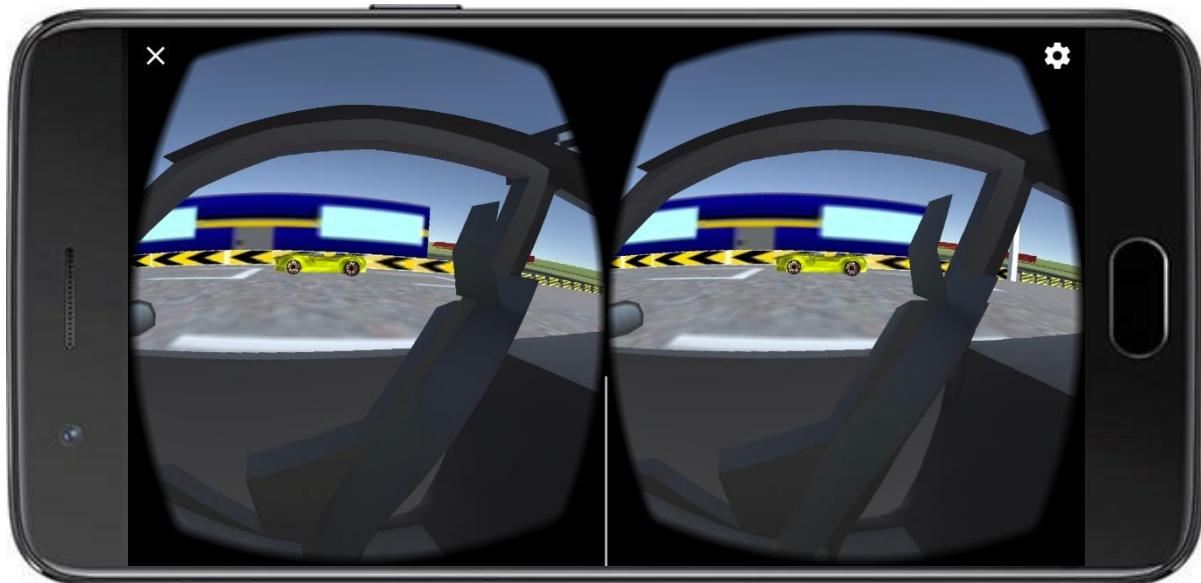


Figure 64 Car Race AI Car Starting Screen

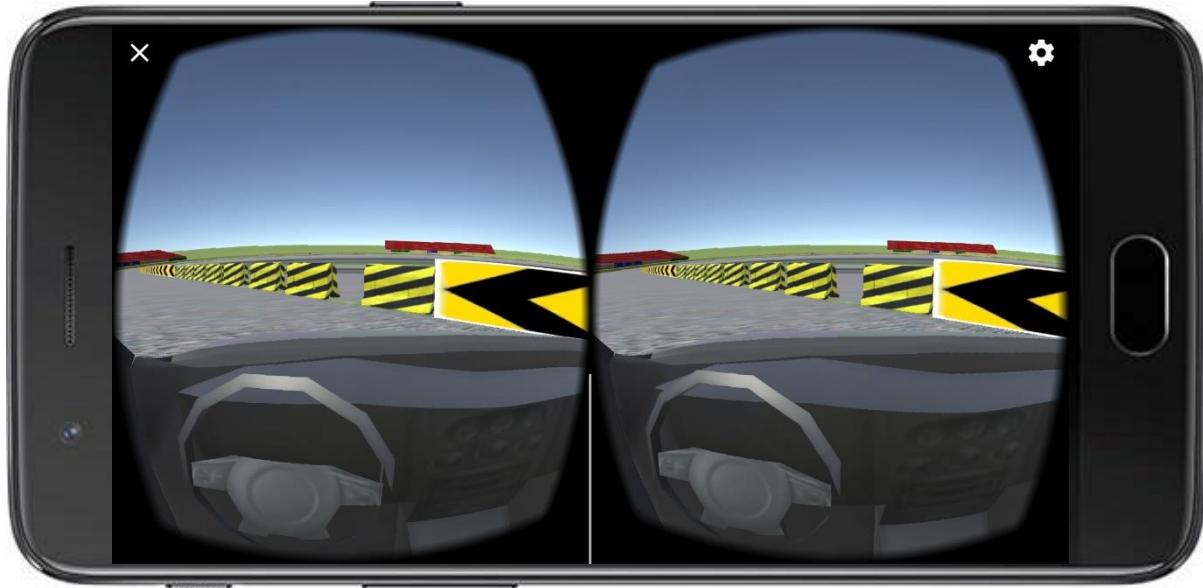


Figure 65 Car Race Obstacles Screen

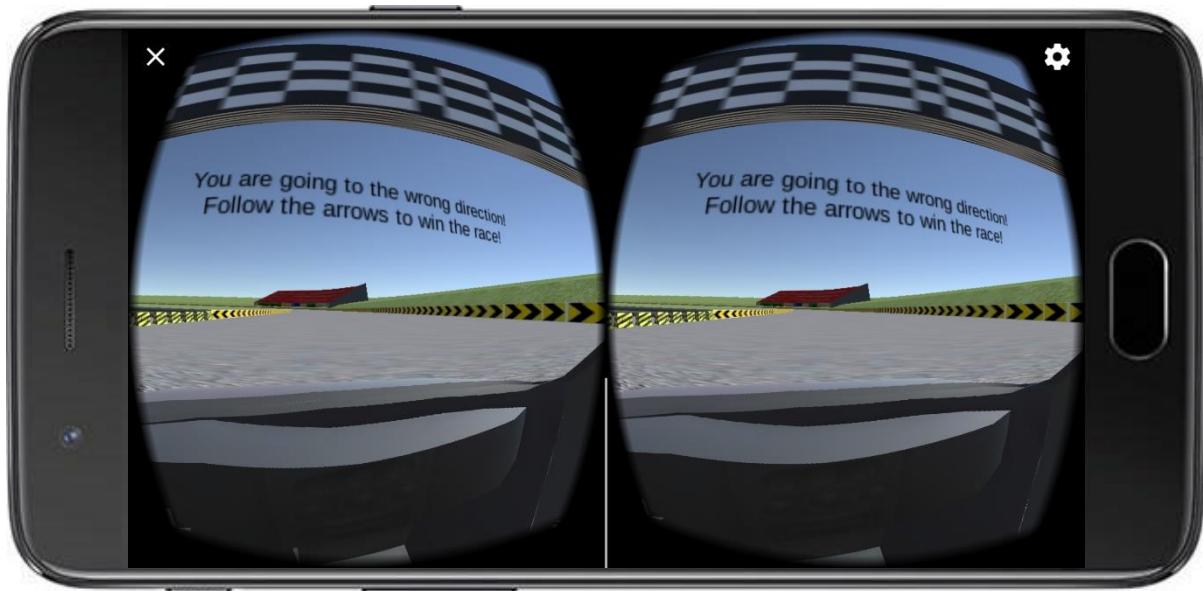


Figure 66 Car Race Wrong Way Screen

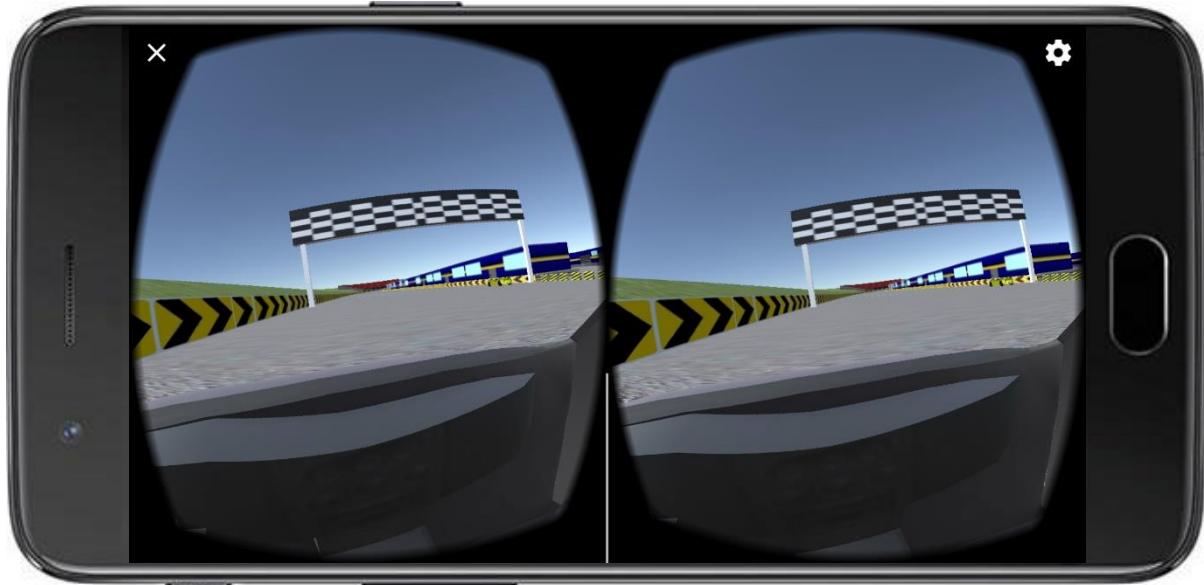


Figure 67 Car Race AI Finish Line Screen

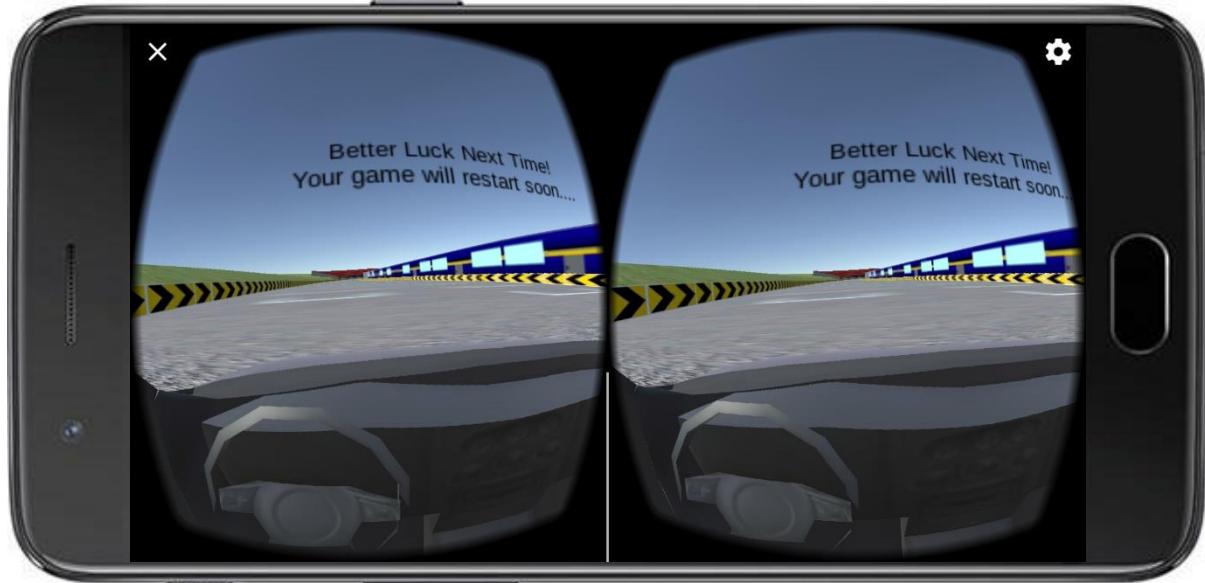


Figure 68 Car Race Lose Screen

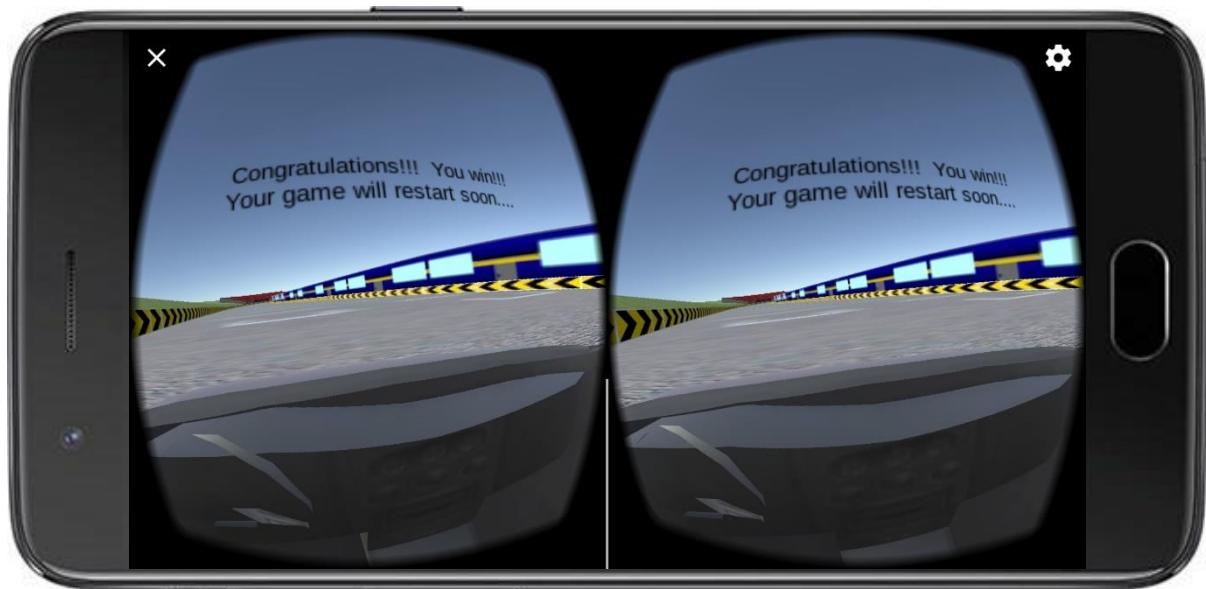


Figure 69 Car Race Winning Screen

➤ **Bowling**

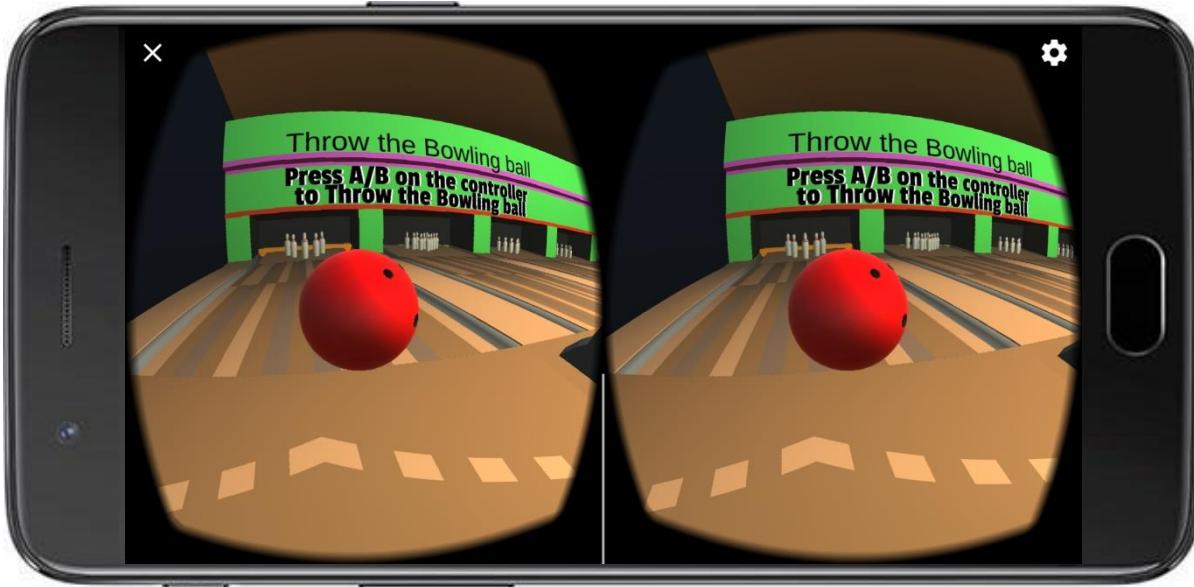


Figure 70Bowling Starting Screen

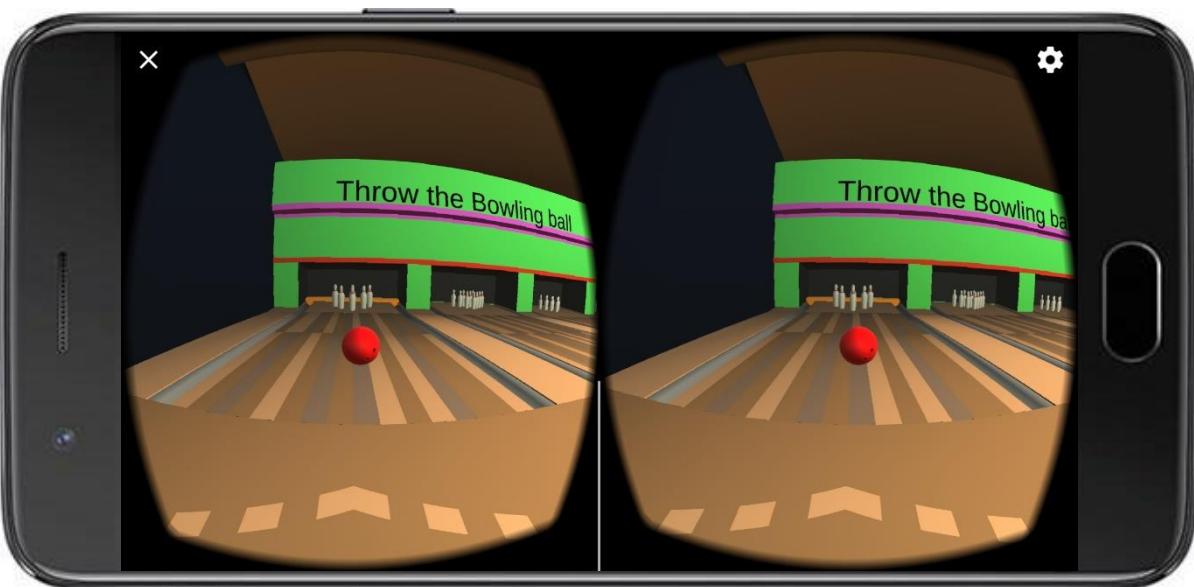


Figure 71Bowling Ball Drag Screen

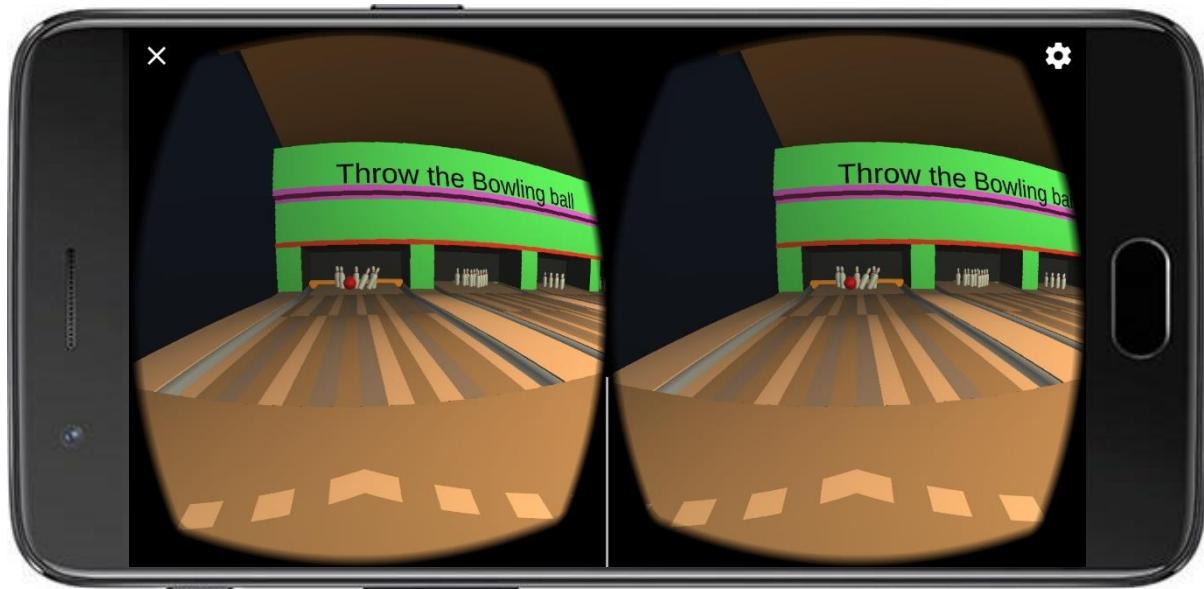


Figure 72Bowling Ball Hits the Pin Screen



Figure 73Bowling Ball Score Screen

➤ **Hurdles**

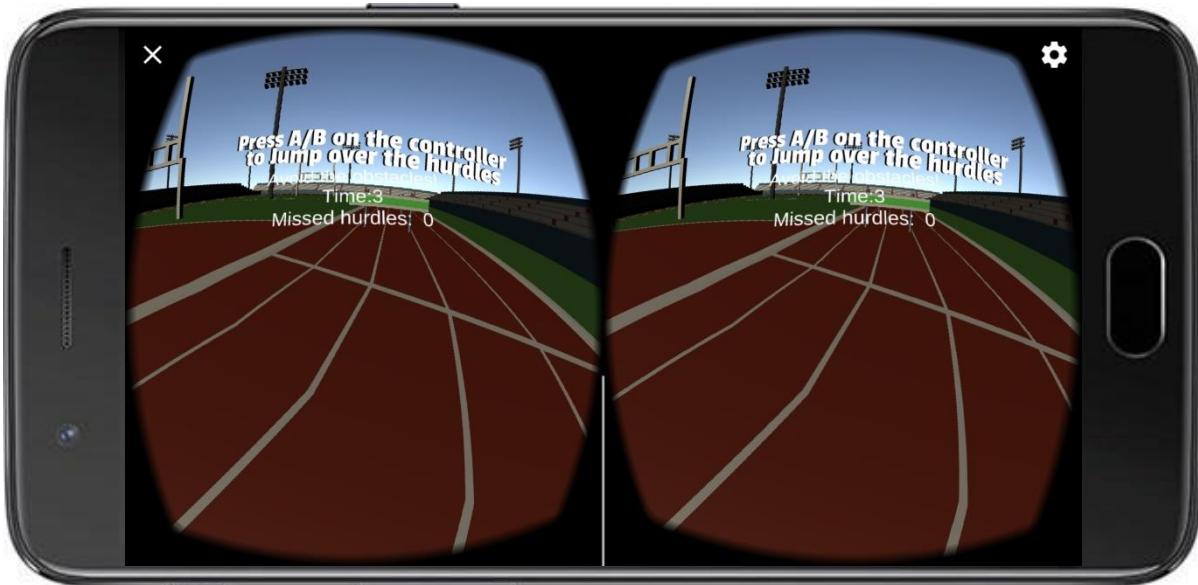


Figure 74Hurdles Starting Screen

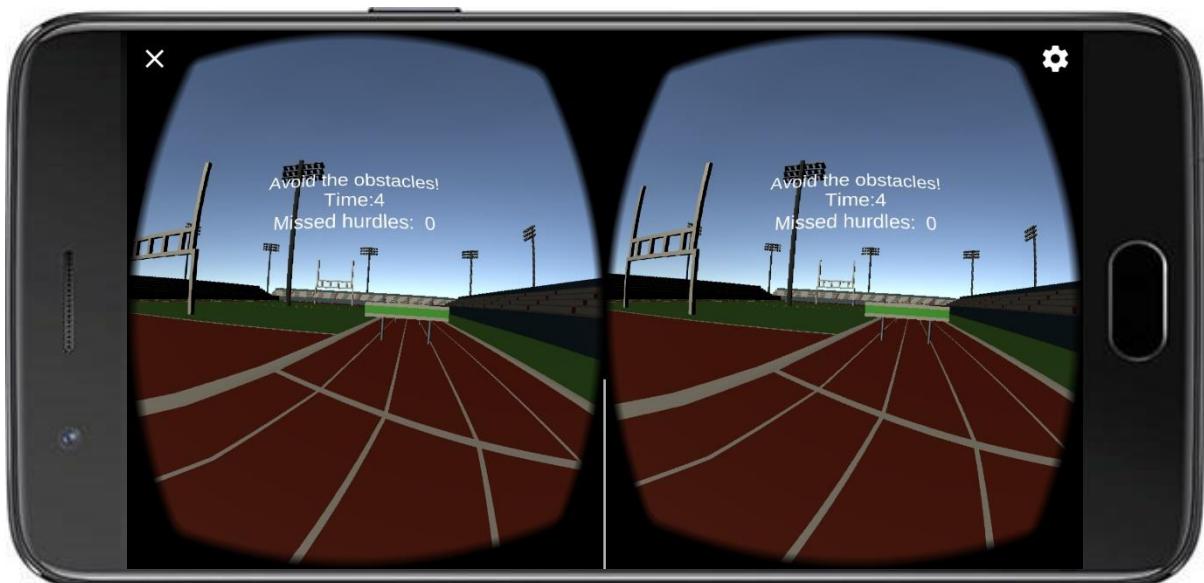


Figure 75Hurdles Timing Screen

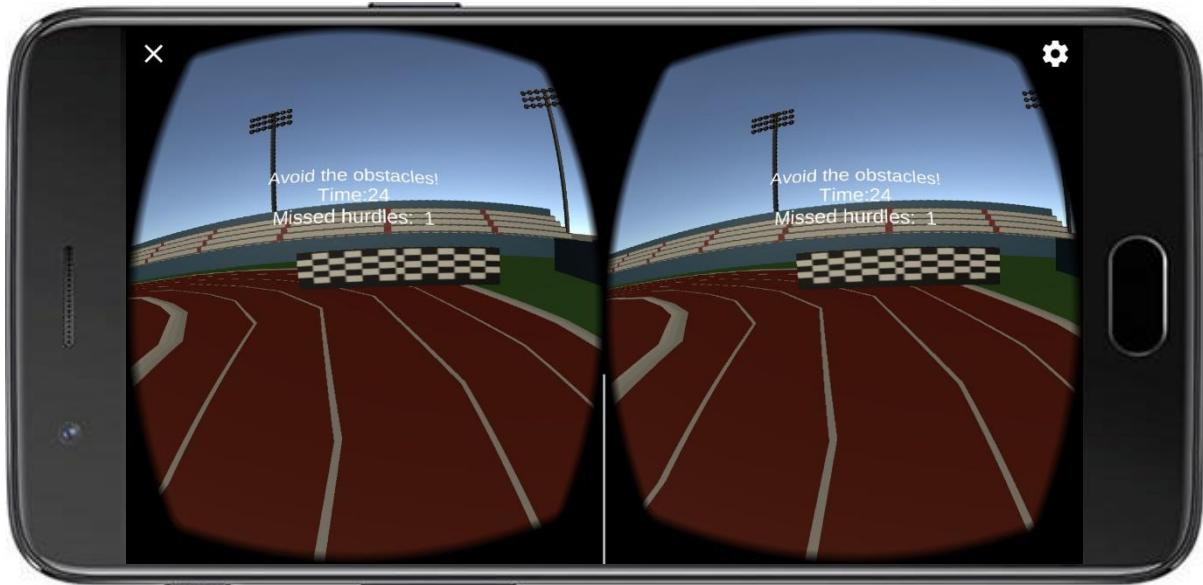


Figure 76 Hurdles Finish Line Screen

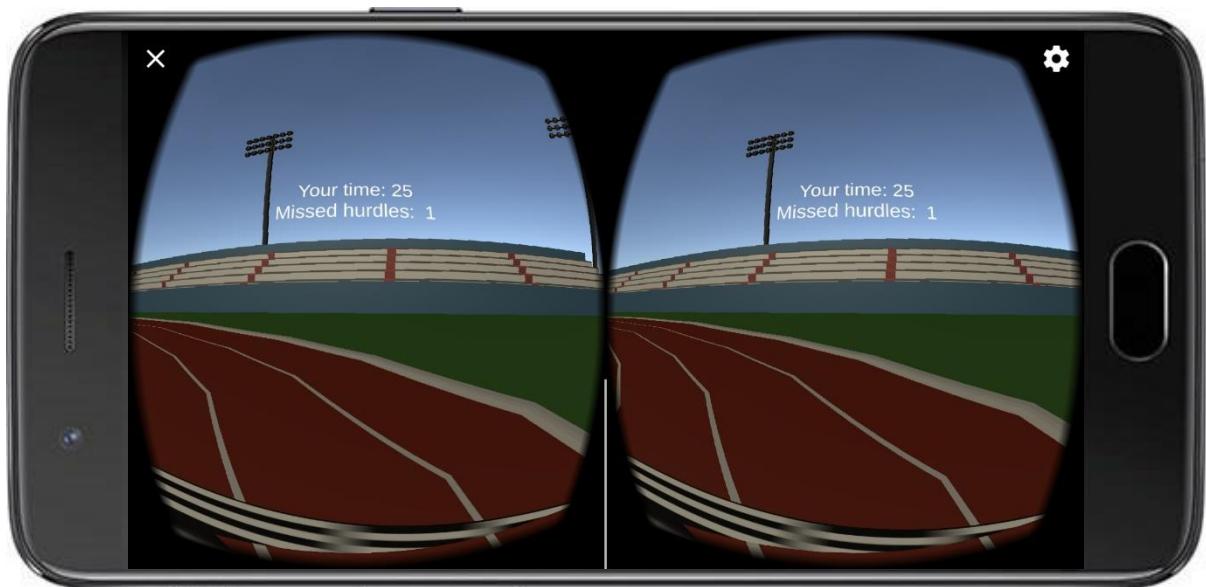


Figure 77 Hurdles End Screen

Testing

➤ Testing Plan:

Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not.

Test planning, the most important activity to ensure that there is initially a list of tasks and milestones in a baseline plan to track the progress of the project. It also defines the size of the test effort.

➤ **Testing Methods:**

Testing methodologies are the strategies and approaches used to test a particular product to ensure it is fit for purpose. Software testing methodologies are the different approaches and ways of ensuring that a software application in particular is fully tested. Software testing methodology such as Test Scenarios was prepared so as to improve product quality.

➤ **Test Scenarios:**

Some of the test scenarios are listed below:

❖ **3D-Runner**

Sr. No.	Scenarios	Comments
1	Player movement	When jump clicked more than once player used to fly in the air and arrive after long time on platform
2	Spawning of platform	Two platform overlapped and so enemies distance was decreased and resulted to game over early
3	Score and High-score	Perfect at once
4	Adding audio(Sounds)	Perfect at once

❖ **3D-Cubber**

Sr. No.	Scenarios	Comments
1	Player movement	Optimizing it for mobile was difficult
2	Level designing	Designing of level was not that difficult
3	Loading next level	Switching between levels was difficult
4	Colliding player with obstacle	Perfect at once

❖ AR-Duck Strike

Sr. No.	Scenarios	Comments
1	Duck placement/movement	Bird was not spawning on defined area it used to fly above the scene and made many problems
2	Park object placement	Placement was very time consuming as it took to find the perfect models for the park. While some where not suitable for Augmented reality.
3	Fire	Making fire working properly needed much research and testing.
4	Adjusting lives and high score	Adjusting live took time and high score was not getting scored permanently and took time to figure out solution
5	Adding audio(Sounds)	Perfect at once but some needed some corrections
6	Setting image target screen	Needed a slight changes in the libraries of Vuforia

❖ AR-Fight Master

Sr. No.	Scenarios	Comments
1	Player	Walking forward and backward was not proper at first Player used to fly in air when game restarted
2	Artificial Intelligence BOT	This stage was the toughest. It took many hours of research and hard work to make it possible as our need. Zombie was not moving towards the player properly
3	Health bar	Decreasing health at correct time while the any hit was a bit challenging
4	Knockout and other behaviors	Adding behaviors at correct time in the scene was a time consuming
5	Setting image target screen	Needed a slight changes in the libraries of Vuforia and it was easier this time as it was done earlier
6	Adding audio(Sounds)	Perfect at once but some needed some corrections
7	Deciding the winner	Perfect at once

❖ AR-Ball Maze

Sr. No.	Scenarios	Comments
1	Designing of maze model	Perfect at once
2	Re-spawning of ball	Ball couldn't detect surface properly
3	Finishing game	Finishing game made many issues like high score storage detecting finish cube
4	Adding audio(Sounds)	Perfect at once
5	Giving look and feel like in realty	Needed some research for assigning the perfect materials

❖ VR-Car Race

Sr. No.	Scenarios	Comments
1	Race Track	First track made many issues So used second one
2	Artificial Intelligence Bot	This stage was the toughest. It took many hours of research and hard work to make it possible as our need. Bot was not running properly on path given to it Due to speed it used to skid many times
3	User Car	Wheel collider was a new concept and so took more time to understand and apply
4	Adding audio(Sounds)	Perfect at once
5	Win Screen / Lose Screen	Due to collider properties every time only win screen was displayed and took more time to get proper logic for it

❖ VR-Bowling

Sr. No.	Scenarios	Comments
1	Adjusting bowling room and bowling alley models	Perfect at once
2	Camera movement according to player movement	Perfect at once
3	Ball throwing	Due to high speed all pins were hit by the ball and lead to no excitement in game
4	Displaying score according to pins hit	Was done perfectly after some trial and errors

❖ Hurdles

Sr. No.	Scenarios	Comments
1	Track	Perfect at once
2	Player movement	Perfect at once
3	Hurdles placement	Firstly hurdles were placed much near and so user was not able to pass them
4	Displaying score and other details	Perfect at once

❖ Main User Interface

Sr. No.	Scenarios	Comments
1	Designing the User Interface	Was perfect after some efforts
2	Integration 3d games in the application	Perfect at once
3	Integration Augmented Reality games in the application	Perfect at once after some research
4	Integration Virtual Reality games in the application	Was the toughest challenge in the entire developing phase. As Virtual reality had totally different project setting and SDKs than that of Augmented and 3d games, it was almost impossible to get it done after many expert advices and research. No one had yet tried this challenge to merge different technologies on the same platform. But at the end we were successful in integrating the technologies on the same platform
5	Switching between the scenes	Perfect at once for few, while some needed corrections.

➤ Bug fixing /Test report

The main principal of Software testing of an application is to find important bugs in the software application & try to make software application bug free.

Once a bug is found, we should know conditions under which this bug is occurring, how many times it occurs & the expected result of the bug. The bug report should be accurate & complete, so that we can get the exact failure reason. Based on this we can get exact idea of problem faced by user & it helps to resolve the problem accurately.

After rectifying the bugs, it has to be solved and system was re-tested until the bugs were fixed.

Few bug fixings are listed below:

- Player movement in Runner
- Spawning of platform in Runner
- Level designing in Cubber
- Duck flying at random position in Duck strike
- Park placement on terrain in Duck strike
- Adding high score in Duck strike
- Playing audio at desired time in games such as Duck Strike, Fight Master and Car Race
- Setting image target screen for AR games needed correction in library of Vuforia
- Creating desired movement for AI Bots in games like Fight Master and Car Race
- Animations for different models
- Excavating desired materials for Ball Maze
- Driving car though controller
- Implementing score and Health-bar
- Implementing Play/Pause controls
- Some arrangements in scenes
- Giving text in VR screen
- Setting proper camera angle to give better look and feel for Hurdles and Bowling
- Adding sounds on button click
- Clashing packages while integration of games
- Integrating AR and VR on same platform

Coding Standard Followed

- Different scripts are made for proper separation of the codes
- Using appropriate conditions/ functions/ methods
- Comments have been included where ever necessary
- Built-in functions are used where needed
- Declaring objects and giving reference
- Required objects are referred to scripts

Classes and Variables are given proper names

```
public class GameController : MonoBehaviour
{
    public static GameController instance;
    public static bool allowMovement = false;

    public GameObject GUIbackbutton;
    public GameObject GUIfwdbutton;
    public GameObject GUIkickbutton;
    public GameObject GUIpunchbutton;
    public GameObject GUIplayerScore;
    public GameObject GUIenemyScore;
    public GameObject GUIStartPanel;
    public AudioClip[] audioClips;
    public GameObject gameOverPanel;
    public Text winText;

    public GameObject pausebtn;
    public Text pausetxt;

    AudioSource audio1;

    public static int playerScore=0;
    public static int enemyScore=0;

    private bool played321=false;

    private void Awake()
    {
        if (instance == null)
        {
            instance = this;
        }
    }
}
```

Loops are properly nested

```
        }
        if (gameController.allowMovement == true)
        {
            if (isAttacking == false)
            {
                if (moveBack == true)
                {
                    anim.SetTrigger("walkBack");
                    anim.ResetTrigger("Idle");
                    setAllBoxColliders(false);
                }
                else
                {
                    anim.SetTrigger("Idle");
                    anim.ResetTrigger("walkBack");
                }

                if (moveFwd == true)
                {
                    anim.SetTrigger("walkForward");
                    anim.ResetTrigger("Idle");
                    setAllBoxColliders(false);

                }
                else if (moveBack == false)
                {
                    anim.SetTrigger("Idle");
                    anim.ResetTrigger("walkForward");
                }
            }
            else if (isAttacking == true)
            {
                setAllBoxColliders(true);
            }
        }
```

Methods are assigned properly and called

```
public void Kick()
{
    isAttacking = true;
    anim.ResetTrigger("Idle");
    anim.SetTrigger("kick");
    playAudio(1);
}

public void React()
{
    isAttacking = true;
    health = health - 10;
    PlayerHB.value = health;
    if (health < 10)
    {
        knockout();
    }
    else
    {
        anim.ResetTrigger("Idle");
        anim.SetTrigger("react");
        playAudio(2);
    }
}

public void knockout()
{
    playAudio(3);
    anim.SetTrigger("knockout");
    gameController.allowMovement = false;
    gameController.instance.scoreEnemy();
    gameController.instance.pausebtn.SetActive(false);
}
```

Limitations and Future Enhancements

- Defining levels in Cubber
- Let user choose difficulty levels (from beginner to Pro)
- Integrate more games
- Add marker-less AR games
- Allow user to choose different models in games
- Defining rounds in AR-Fight Master
- IOS Application Development
- Allow player to interchange his choice of technology (3D, AR,VR) for any of games.
- As AR and VR are still developing technologies, there are many regular changes and updates in Vuforia and Cardboard.

Experience and Learning

We still remember the day, when we decided to develop Game Pack Reality game project. We had faced many difficulties. But thanks to teachers and colleagues, who helped us in every way to overcome the problems, this project has given us immense knowledge and valuable experience throughout. We as team members had greater understanding and coordination that made us stronger and hence there was also a social bonding with the project. Even though we were following a new technology the team took it as a challenge and worked very smoothly.

Throughout the project there were ups and downs but overcoming them was our aim and hence finally we got a fruitful result. During the span of time of this project we learned many lessons like developing convincing power and realized working in a team with feeling of togetherness is so important in such projects.

We are all very satisfied working as a team, undergoing the challenges and facing them with the right spirit. We have learned to be positive on our way and to never give up. This project has given us opportunity to enhance our development, project management and communication skills.

We choose this field of project just to excavate new technologies, which are under-development. We preferred learning Augmented reality and Virtual reality as they are future in every field. The keen interest in developing games and attraction towards these technologies ended up in creating something which is very new to the world today. We are very proud of ourselves today as we are successful today in developing such applications.

Reference and Bibliography

1. <https://unity.com/education>
2. <https://stackoverflow.com/>
3. <https://gamedev.stackexchange.com>
4. <https://www.c-sharpcorner.com/>
5. <https://www.mixamo.com/>
6. <https://assetstore.unity.com/>
7. [https://www.youtube.com/watch?v=cCOLdX1JMo4 \(Vuforia\)](https://www.youtube.com/watch?v=cCOLdX1JMo4)
8. [https://www.youtube.com/watch?v=fx7eGUCNSzI\(Google cardboard\)](https://www.youtube.com/watch?v=fx7eGUCNSzI)
9. <https://www.sparkar.com/ar-studio/learn/documentation/>
10. <https://youtu.be/FqoT-lCANlY>
11. https://youtu.be/MtiUx_szKbI
12. <https://youtu.be/23HEMKfMKKk>
13. <https://youtu.be/gyOOf25321M>
14. <https://youtu.be/HGWZwXVWLbk>
15. <https://unity.com/solutions/ar-and-vr-games>
16. <https://youtu.be/8pFlnyfRfRc>
17. <https://youtu.be/o1XOUkYUDZU>
18. <https://youtu.be/5ozKp5qC4aA>
19. <https://youtu.be/FxedJgTsFyQ>