

**CATCHING THE HEN**  
**AUGMENTED REALITY BASED ANDROID MOBILE GAME**



**By**  
**Adeel javed**  
**Supervised By**  
**Dr. Khalid saleem**

**Department of Computer Sciences**  
**Quaid-i-Azam University Islamabad**  
**2013-2017**

QUAID-I-AZAM UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE

Dated: \_\_\_\_\_

**Final Approval**

This is to certify that we have read the final report submitted by Adeel javed and it is our judgment that this report is of sufficient standard to warrant its acceptance by the Quaid-I-Azam University, Islamabad for the degree of the Bachelors of Science in Computer Science.

**Committee**

**1. EXTERNAL EXAMINER**

\_\_\_\_\_

**2. SUPERVISOR**

\_\_\_\_\_

Dr. Khalid saleem

Department of Computer Science

Quaid-I-Azam University, Islamabad.

**3. INCHARGE**

\_\_\_\_\_

Dr. Onaiza Maqbool

Department of Computer Science

Quaid-I-Azam University, Islamabad.

## ACKNOWLEDGMENT

---

In the name of ALLAH, Most Beneficent, Most Merciful.

First of all, I thank Allah at the completion of my project, as I completed this task only by His favour and grace. At this moment, this is due on me, to thank some personalities, because without their cooperation and supervision, I was unable to complete this work.

First of all, my respected teacher and supervisor, Dr. Khalid Saleem, whose door of kindness always remains open for me. His advices and instructions will guide me throughout my life. Completion of my task would not be possible without his help, encouragement, dynamic supervision and constructive criticism. I am highly indebted expressing my gratitude to him for his entire collaboration. Special thanks to Dr. M. Afzal Bhatti, Dr. Onaiza Maqbool, Dr. Rabeeh Ayaz Abbasi, Miss Ifrah Farrukh Khan, Madam Memona Afsheen, Dr. Mudassir Azam Sindhu, Dr. Shoaib Karim, Dr. Usman and Dr. Ghazanfar Farooq for their kind support and cooperation.

It is also necessary to me, to thank my parents whose prayers are treasure of my life. I have no words to pay gratitude to them whose affection, guidance and continuous encouragement did their best to shape my character.

Adeel Javed

2013-2017

## ABSTRACT

---

The world is advancing in technology with great pace and software systems add the computer generated data in our real living world. In this project we will use the concept of Augmented Reality to create a game named as Catching the Hen. The main feature of my game will be to target the low hardware resource devices and put the concept of augmented reality using them.

The game will use the simple camera of the android device and render the real environment images on the raw image of unity 3d in very quick time that gives the real time images. On top of it we added the virtual character designed and created in blender tool which will look like running walking and jumping hen in the real environment.

Player is provided with the three different levels with the objectives to complete with in the 60 seconds of timer. The player has to achieve all the three objectives in order to win the level. The player can control the game music from the settings panel which he/she can access through the main menu like every game. Player is also provided with the help instructions about how to play the game and finding hens in the real environment and at last there is game statistics panel where he/she can see the players play record.

# 1 TABLE OF CONTENTS

---

<b>1</b>	<b>Chapter 1: Software Project Management Plan.....</b>	<b>1</b>
1.1	Product overview .....	1
1.2	Deliverables.....	1
1.2.1	Project deliverables.....	1
1.2.2	Product deliverables .....	2
1.2.3	Software Deliverables .....	2
1.2.4	Document deliverables .....	2
1.2.5	Internal stake holders .....	2
1.2.6	External stake holders.....	2
1.3	Project Management Plan .....	2
1.3.1	Tasks.....	2
1.3.2	Writing the SPMP document .....	2
1.3.3	Drawing the animation frames .....	3
1.3.4	Writing the Software requirement specifications document .....	3
1.3.5	Writing The Software Design Description Document .....	4
1.3.6	Writing The Software Implementation .....	4
1.3.7	Writing The Software Test Document: .....	4
1.4	Project Organization .....	4
1.4.1	Software Process Model .....	4
1.4.2	Roles And Responsibilities .....	5
1.4.3	Tools and techniques .....	6
1.5	Assignments .....	6
<b>2</b>	<b>Chapter 2: Software Requirement Specification.....</b>	<b>8</b>
2.1	Purpose .....	8
2.2	Product Overview .....	8
2.2.1	Mission Statement .....	8
2.2.2	Product Motivations .....	8
2.2.3	Product Scope .....	9
2.3	Specific Requirement's .....	9
2.3.1	External Interface Requirement's .....	9
2.4	Product Features.....	9

2.4.1	Product Core Features .....	10
2.4.2	Additional Features .....	12
2.5	Operating Environment .....	12
2.6	Assumptions And Dependencies .....	12
2.6.1	Time Dependencies.....	12
2.6.2	Hardware Dependencies.....	12
2.7	System Features.....	12
2.8	System Core Features .....	13
2.8.1	Main Menu.....	13
2.8.2	Level Scene And Layout .....	13
2.8.3	Hen The Game Agent .....	14
2.8.4	Player Nets .....	15
2.8.5	Capture Hens.....	15
2.8.6	Gyro Camera Setup .....	16
2.8.7	Game Audio.....	16
2.8.8	Player Score.....	17
2.9	Additional Features.....	17
2.9.1	Game Stats Records .....	17
2.9.2	Settings Menu .....	18
2.9.3	Help Menu.....	18
2.10	Software System Attributes .....	19
2.10.1	Reliability.....	19
2.10.2	Availability.....	19
2.10.3	Security .....	19
2.10.4	Maintainability .....	19
2.10.5	Portability.....	19
2.10.6	Performance.....	19
2.11	use cases .....	20
2.11.1	Play game .....	20
2.11.2	Play next level .....	21
2.11.3	Change settings.....	22
2.11.4	Capture hen.....	23
2.11.5	Activate capture Animations.....	24

2.11.6	View game stats .....	24
2.11.7	Create game environment .....	25
2.11.8	Create game agent hen.....	26
2.11.9	Set hen movements .....	27
2.11.10	Activate hens idle state.....	28
2.11.11	Activate hens running state .....	29
2.11.12	Activate hens walking state .....	29
2.11.13	Activate hens jumping state .....	30
2.11.14	Create player nets.....	31
2.11.15	Activate initial net .....	32
2.11.16	Setup gyroscope.....	32
2.11.17	Create real environment.....	33
2.11.18	Play game audio .....	34
2.11.19	Display score .....	35
2.11.20	View help menu .....	36
2.12	Use case diagram .....	38
2.13	System Sequence Diagram.....	39
2.13.1	Play game .....	39
2.13.2	Create game environment .....	39
2.13.3	Create game agent hen.....	40
2.13.4	Create player nets.....	40
2.13.5	Calculate hen position.....	41
2.13.6	Play game audio .....	41
2.13.7	Setup camera .....	42
2.13.8	Display score .....	42
2.14	Summary .....	42
<b>3</b>	<b>Chapter 3: Software design.....</b>	<b>43</b>
3.1	Design Overview .....	44
3.2	Requirements Traceability Matrix .....	44
3.3	system architecture design .....	45
3.3.1	Chosen System Architecture .....	45
3.3.2	System Architecture Diagram .....	46
3.4	User Interface Design.....	46

3.4.1	Description of User Interface .....	46
3.4.2	Game prototype details .....	47
3.4.3	Screen Images .....	47
3.5	Domain Model .....	51
3.6	Sequence diagram.....	51
3.7	State diagrams .....	53
3.8	Class diagram .....	55
<b>4</b>	<b>Chapter 4: Software Implementation.....</b>	<b>59</b>
4.1	Introduction .....	59
4.1.1	Language Selection .....	59
4.1.2	Tools selection .....	59
4.2	Application development.....	59
4.3	Application Constraints.....	61
4.4	Application screenshots.....	62
<b>5</b>	<b>Chapter 5: Software test.....</b>	<b>64</b>
5.1	Test approach .....	65
5.2	Test plan.....	65
5.2.1	Features to be tested .....	65
5.2.2	Features not to be tested .....	65
5.2.3	Testing Tools and Environment .....	66
5.3	Test cases.....	66
5.3.1	Select Level.....	66
5.3.2	Game environment .....	67
5.3.3	Create game agent hen.....	67
5.3.4	Create game object net.....	68
5.3.5	Calculating gyroscope rotations.....	68
5.3.6	Play game music.....	69
5.4	Hardware testing on different devices .....	69
<b>6</b>	<b>Conclusion .....</b>	<b>70</b>
<b>7</b>	<b>Appendix's A Glossary .....</b>	<b>72</b>
<b>8</b>	<b>Appendix's B Project Proposal .....</b>	<b>73</b>
<b>9</b>	<b>References .....</b>	<b>74</b>



## **List of figures**

<b>1</b>	<b>Chapter 1: Software Project Management Plan.....</b>	<b>1</b>
1.	<i>Figure 1.1 the water fall model.....</i>	<i>6</i>
<b>2</b>	<b>Chapter 2: Software Requirement Specification.....</b>	<b>8</b>
1.	<i>Figure 2.1 Use case diagram.....</i>	<i>39</i>
2.	<i>Figure 2.2 System Sequence Diagram of Play game.....</i>	<i>40</i>
3.	<i>Figure 2.3 System Sequence Diagram of Game Environment.....</i>	<i>41</i>
4.	<i>Figure 2.4 System sequence Diagram for hen agent.....</i>	<i>42</i>
5.	<i>Figure 2.5 System Sequence Diagram of Create Player nets.....</i>	<i>42</i>
6.	<i>Figure 2.6 System Sequence Diagram of Hen Position.....</i>	<i>43</i>
7.	<i>Figure 2.7 System Sequence diagram of Play game audio.....</i>	<i>43</i>
8.	<i>Figure 2.8 System Sequence Diagram of Gyroscope Rotations.....</i>	<i>44</i>
9.	<i>Figure 2.9 System Sequence Diagram of Display Score.....</i>	<i>44</i>
<b>3</b>	<b>Chapter 3: Software Design Description.....</b>	<b>45</b>
1.	<i>Figure 3.1 System Architecture Diagram.....</i>	<i>47</i>
2.	<i>Figure 3.2 Main Menu Screen.....</i>	<i>48</i>
3.	<i>Figure 3.3 settings menu.....</i>	<i>49</i>
4.	<i>Figure 3.4 Help Menu.....</i>	<i>49</i>
5.	<i>Figure 3.5 Stats Panel.....</i>	<i>50</i>
6.	<i>Figure 3.6 Level List.....</i>	<i>50</i>
7.	<i>Figure 3.7 Game environment.....</i>	<i>51</i>
8.	<i>Figure 3.8 Score panel.....</i>	<i>51</i>
9.	<i>Figure 3.9 Domain model of catching the hen.....</i>	<i>52</i>
10.	<i>Figure 3.10 Sequence Diagram for main menu.....</i>	<i>53</i>
11.	<i>Figure 3.11 Sequence diagram for level and Game environment.....</i>	<i>53</i>
12.	<i>Figure 3.12 camera gyroscope and game time.....</i>	<i>54</i>
13.	<i>Figure 3.13 Class diagram of catching the hen game.....</i>	<i>55</i>

<b>4</b>	<b>Chapter 4: Software Implementation document .....</b>	<b>59</b>
1.	<i>Figure 4.1 The code sample of the Camera controller script.....</i>	<i>61</i>
2.	<i>Figure 4.2 The code sample of the hens random movements near player environment.....</i>	<i>61</i>
3.	<i>Figure 4.3 throwing net script code sample.....</i>	<i>62</i>
4.	<i>Figure 4.4 welcome screen.....</i>	<i>63</i>
5.	<i>Figure 4.5 Menu Screen.....</i>	<i>63</i>
6.	<i>Figure 4.6 Settings screen.....</i>	<i>63</i>
7.	<i>Figure 4.7 Help Screen.....</i>	<i>63</i>
8.	<i>Figure 4.8 Stats Screen.....</i>	<i>64</i>
9.	<i>Figure 4.9 Level Screen.....</i>	<i>64</i>
10.	<i>Figure 4.10 Level Objectives panel.....</i>	<i>64</i>
11.	<i>Figure 4.11 Game environment Screen.....</i>	<i>64</i>
<b>6</b>	<b>Chapter 6: Conclusion.....</b>	<b>70</b>
1.	<i>Figure 6.1 Tango Device Hardware details .....</i>	<i>71</i>

Chapter 1:

# Software Project Management Plan

# 1 CHAPTER 1: SOFTWARE PROJECT MANAGEMENT PLAN

---

This chapter introduces the project plan document for the augmented reality game catching the hen (CTH). It provides the project overview and the entire remaining requirement specification document features.

## 1.1 PRODUCT OVERVIEW

Considering the changing world perception more and more automation in gaming and real world environment in field of virtual reality and augmented reality. we have decided to work on project involving the augmented reality (AR). **AR** is a live direct or indirect view of a physical, real-world environment whose elements are augmented by computer-generated sensory input such as sound, video, graphics or GPS data. [11]

**Augmented reality (AR)** according to Ronald Azuma, Research Leader at Nokia Research Centre, is "an environment that includes both virtual reality and real-world elements. For instance, an AR user might wear translucent goggles; through these, he could see the real world, as well as computer-generated images projected on top of that world." (Azuma, 1997) [10]

The game will create graphical animated hen in real environment which is visible from the AR(Augmented reality) camera of the device. The main goal for the player is to catch the hen which is visible to him in his AR camera. Player will be provided by the number of nets which will be used for the purpose of catching the hen. Only thing the player had to do is throw the net onto the hen in correct direction to capture it. There will be three Different levels to complete the game.

### Problem statement

Our main goal is to create the android based augmented reality catching the hen game.

## 1.2 DELIVERABLES

### 1.2.1 Project deliverables

1. Software project plan document
2. Software requirements specification document
3. Software design description
4. Source code of the project
5. Software test documentation
6. Final product

### 1.2.2 Product deliverables

The game will be intangible product that will be played using the smart phones which will provide a real environment with the computer generated animated graphics.

### 1.2.3 Software Deliverables

Our target provides a working android based augmented reality game at the end of the project which will satisfy all the requirements of the game.

### 1.2.4 Document deliverables

A number of documents will be delivered during the course of the project. The documents to create the base design of the project.

1. **Documents for Product management:**
  1. Software project management plan
  2. Software design description
2. **Documents for designer:**
  1. Software requirement specification

### 1.2.5 Internal stake holders

The internal stake holder in this case will be myself.

### 1.2.6 External stake holders

The external stake holders will be the player who will play our game

## 1.3 PROJECT MANAGEMENT PLAN

The identified tasks for our product are as follows written in the. These tasks must be complete before the final evaluation of the project.

### 1.3.1 Tasks

There are many tasks related to the project which of them will be identified and enlisted below.

1. **Project discussion**

In this task I collect the information about what the project is about to do and what it requires to complete the project.
2. **Requirement analysis**

I have to gather all the data about game before writing the detailed SRS about the product these all requirements will have to be collect from the supervisor.

### 1.3.2 Writing the SPMP document

The SPMP software project management plan specifies the complete knowledge about how we plan the project tasks and gives complete details of the tools, roles and responsibilities till completing the project.

- 1. Write Introduction part**

In this task I will write the introduction part of the product what it's basically do what's its mission statement and product scope.

- 2. Identify project organization roles and responsibilities**

This task defines the process model, roles and responsibilities of the developer in completing the project.

- 3. Project management plan**

The project management plan will have to be written for the purpose to identify the tasks that are necessary to complete the project.

### **1.3.3 Drawing the animation frames**

In this task I have to draw the animated frame of the main component of the game that are going to give the details for me latter to make those animations.

### **1.3.4 Writing the Software requirement specifications document**

The SRS software requirement specification document gives complete knowledge about the product definitions, overview, scope, specific requirements and defines the complete product and system features.

- 1. Writing the introduction part**

In the introduction part of the SRS software requirement specification we will describe the product over view product scope and mission statements.

- 2. Writing the specific requirements**

In writing the specific part we will identify the user interfaces software interfaces operating environments.

- 3. Identify the product features**

This task has important part in the completion of our product because this part identifies all the features that are going to be implement in our final product.

- 4. Specifying assumptions and dependences**

I have to find the hardware, software and time dependencies of related to the product.

- 5. Writing the system features**

This task will define each and every product features every step performance how it will be and when that feature is going to be activate in our game.

## 6. Software system attributes

In this part of the SRS we have to describe the products attributes as follows

1. Reliability
2. Availability
3. Security
4. Maintainability
5. Portability
6. Performance

### 1.3.5 Writing the Software Design Description Document

The SDD is provided for the better understanding of the concepts explored in the software requirement specification document. We have made use case diagram there descriptions the system sequence diagrams, sequence diagram against the operations explored in the use case diagram with the final class diagram at the end of the document. We have to complete the entire above mentioned diagram from 15 January 2017 to 25 January 2017.

### 1.3.6 Writing the Software Implementation

In this phase of the documentation we will write about the detailed features about how we implemented the game using the specific tool we describe about all the functionalities how they are implemented at the back end and will describe some algorithms about the game.

### 1.3.7 Writing the Software Test Document

The last phase in our documentation part where we can describe the test cases and the test plan for our game. Which must be completed in the duration from 25 January 2017 to 2 February 2017.

## 1.4 PROJECT ORGANIZATION

A project organization is a structure that facilitates the coordination and implementation of project activities. Its main reason is to create an environment that fosters interactions among the team members with a minimum amount of disruptions, overlaps and conflict.

### 1.4.1 Software Process Model

Over all the software process model use will be the water fall model. It is referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

Following is a diagrammatic representation of different phases of waterfall model.[4]

## The water fall model

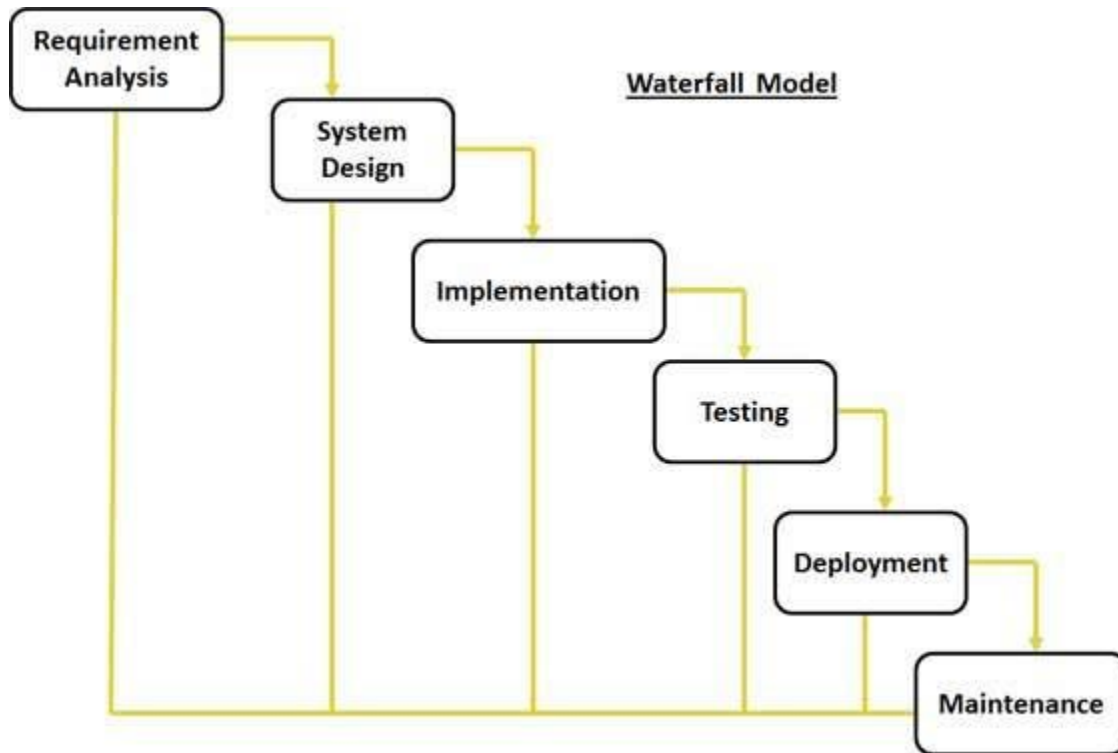


Figure 2.1 the water fall model

### 1.4.2 Roles And Responsibilities

As mentioned earlier in this document use the water fall model process model for developing the game so I will follow the same roles which were described in the process of the development of my game. [2]

**Requirement analysis:** The responsibility of gathering the requirements of the project is of person1. In this part person will be going to collect all the requirements about the project.

**Designing:** the responsibility of the designing of the documents is of person2 and the algorithms which will manipulate the game are of person3. Person2 will design the document by defining the use cases, class diagrams and state transition diagrams etc.

**Implementation:** the responsibility of implementing the project when documentation part is done will be going of person4 and person 5; one will script the game objects designed by the other person. Both persons must ensure the implementation of the whole game providing all the features that were described in the SRS document.

**Deployment:** after completing the implementation person1 will deploy the game to play store from where lots of other people can able to download and enjoy the game.

**Verification:** The final process of developing the game and responsibility is of person6, He had to test the game and check for all the errors and providing the complete software product to



the players. At the end he will have to verify the product and it will be allowed for everyone to run the product.

**Maintenance:** the maintenance part is very important for the game because with new software systems the game must run smoothly therefore all management team had to maintain and update the product.

### 1.4.3 Tools and techniques

For the game parts that are concerned with user interaction that are human computer Interfaces will be followed for the ease of the game play. An object oriented technique is applied for the scripting and writing of the SRS documents.

I will use unity 3d 32 bit for creating the game components which will provide the Augmented Environment. And for Scripting of different game components I will use C#, and for creating the characters of the game I will use blender tool which allow us to create the animated characters.

We will use blender to create the game object hen and nets.

For the purpose of creating use case diagrams, system diagrams, domain models and etc. I will use Argo UML and Microsoft Visio.

For writing the documents SRS, SPMP, SDD, STD I will use Microsoft word.

## 1.5 ASSIGNMENTS

- 1 Discussed the project proposal with the supervisor
2. Discussed the directions of completing the required feature.
3. Showed first product overview to the supervisor
4. Literature reviewed about the document making
5. Literature reviewed about augmented reality
6. Reconsidering changes made a new software requirement specification introduction part
7. Advised by supervisor to made the animated hens
8. Finalized the SPMP document
9. Finalized the SRS document.
10. Discussed the software design description document with the supervisor.
11. Literature reviewed about the design description document.
12. Created game prototype to check the main functionalities either they are implementable or not in the unity free version.
13. Made all media artefacts that are going to be used in the system
14. Implementation of the game

## Chapter 2

# Software Requirement Specification

## 2 CHAPTER 2: SOFTWARE REQUIREMENT SPECIFICATION

---

This chapter introduces the requirement specification document for augmented reality based android game catching the hen game. It provides the purpose and scope of the project. Any definitions and references are listed in this section as well as an overview of the remaining requirements specification document.

### 2.1 PURPOSE

The purpose of this Software Requirements Specification (SRS) document is to provide a detailed description of the functionalities of the “Catching the hen game”. This document will cover each of the products intended features.

### 2.2 PRODUCT OVERVIEW

Our game will create graphical animated hen in real environment which is visible from the AR (Augmented reality) camera of the device. The main goal for the player is to catch the hen. Player will be provided by the number of nets which will be used for the purpose of catching the hen. Only thing the player had to do is throw the net onto the hen in correct direction to capture it. There will be three different levels with some specific objectives to complete the level player must complete the level objectives.

#### 2.2.1 Mission Statement

Create the android mobile based augmented reality game Catching the Hen a game.

#### 2.2.2 Product Motivations

This game is designed for children who love to catch the hens as I have mentioned in the above overview my product will create the animated hen in the real environment so children will love to catch the animated hen created in the environment in there surrounding, children interact with the by rotating the device physically in their current environment.

Augmented reality and virtual reality techniques have revolutionized the world and it revolution is still ongoing some of major products related to the augmented reality are

##### 1. *Pokémon Go*:

It wouldn't be a list of the best AR apps without mentioning Niantic's *pokemon go*, a game that has quickly captured everyone's attention and given them a reason to go out into the world, walk around, and catch Pokémon.

**2. Ink Hunter:**

Ink Hunter is the app you should use when deciding on a tattoo and where to put it. The app lets you try out pre-made tattoos, as well as your own designs, and they can be oriented in whatever position you like and placed on any part of the body.

**3. Star chart:**

Star Chart may be an educational app, but it's a really cool one that's sure to appeal to people of all ages. When Star Chart is opened on your Android or iOS device and pointed at the sky, the app will inform you of what stars or planets you're currently facing, even during the day when the stars are at their hardest to see.

**2.2.3 Product Scope**

In this project i will develop Augmented Reality game catching the hen using Augmented Reality techniques. Goal of this project is to make a game which will be appealing to the children. And it will be designed for every mobile device that has gyroscope supported in it. Scope of this product is inside the boundaries of implementing suitable gaming techniques using augmented reality.

**2.3 SPECIFIC REQUIREMENT'S****2.3.1 External Interface Requirement's****1. User Interfaces**

The game interfaces are developed for Android. On Android the product is a native application. All devices have different screen layouts to fit user interface on different devices. All of those user interfaces contain main menu for five main functionality of this game. The prototypes of the user interfaces are given below.

**2. Hardware Interfaces**

This game needs an android phone that supports the Gyroscope.

**3. Software Interfaces**

Android, version 4 or later, is required for Android mobile devices to run error-free.

**2.4 PRODUCT FEATURES**

The following list offers a brief outline and description of the main features and functionalities of the catching the hen game. The features are divided into two main categories core features which are very necessary to implement without them the game will be void. Secondly there are some additional features which will be implemented to help user with more features.

### 2.4.1 Product Core Features

Following are the core features which are must for the completion of the game catching the hen. [6]

#### 1. Hen Agent

- This is the main feature of the game the agent which will be controlled by the game AI algorithm.
- This feature will appear in players screen every time he plays game.
- It will allow player to interact with the game.
- This feature allows player to capture the hen.
- This feature has more sub features
  - 1. Hens idle animation**
    - This feature is visible to the player whenever Hen is spawned in the game
    - This allows user to see the animated hen sitting idle in front of his AR camera.
  - 2. Hens Running animations**
    - This feature will be visible to the player when Hen feels player is going near him so it starts its running animation state.
    - This allow user to run behind the hen in order to capture it
  - 3. Hens walking animations**
    - This feature will be visible to the player when player throws and miss the net or player goes near the hens location.
    - This allow user to interact physically with the game agent hen.
    - available in the level layout.
  - 4. Hens jumping animations**
    - This animation will be visible to the player when player gets too close and hens AI algorithm detects that there is chance of capture.
    - User can interact with this animation mostly when he plays the game.

#### 2. Level Scene

- Level scene is the core features of the game which will visible to the player every time he plays the game.
- So far I have decided to implement the three different levels of the game
- These levels have different objectives to achieve in order to win the level
- Some sub features inside the levels layouts are.
  - 1. Level 1 layout**
    - Player had to capture 3 hens
    - The time given to user will be 60 seconds
    - The maximum number of used nets had to be less the 15
  - 2. Level 2 layout**

- Player had to capture 4 hens
- Player will also have to capture 1 time bonus
- The maximum number of used nets had to be less the 15

### **3. Level 3 layout**

- Player had to capture 3 hens
- Player will also have to capture 1 time bonus
- The maximum number of used nets had to be less the 7

### **3. Player Nets**

- This feature will be appearing to the player always when he starts playing the level.
- Allows user to throw the net onto the hen in order to capture it.
- Player had to press the reset button in order to re spawn thrown net

### **4. Capture Hen**

- This feature allows user to see the animations of the hen which will be captured by the player.
- Capture hen allows player to see the hen captured frame animations.

### **5. Main Menu**

- Appears on the player's mobile screen every day when he opens the game from his mobile screen.
- Allows player to go to select the level, settings, help menu, and quit the game.

### **6. Gyro Camera Setup**

- This feature in game allows player to see the real environment through is camera which will control the rotations of the camera.
- Allows player to see the animated game objects animated hen in the visible real environment.

### **7. Game Audio**

- This feature in game provides different sounds during the whole game play.
- This feature allows player to hear sounds like
- Game music
- Button sounds
- Net throw sounds

### **8. Player Score**

- Player score will be visible to the players whenever he completes the level
- It will be decided on the basis of time taken to capture the hen and number nets he used to capture the hen.

### 2.4.2 Additional Features

These features will groom the game appearance and interactions with the player. Following is the list of the additional features that I going to implement in my game.

#### 1. Game Stats Records

- It will be provide the details about the level completions stats and records the captured hen from that level.
- Allows user to view and reset the game stats and captured hens.

#### 2. Settings Menu

- Allows the user to customize his/her preferences
- Enables the user to modify certain features and functionalities

#### 3. Help Menu

- Lists and provides information to how to play game.
- Can be accessed at any time via the main menu.

## 2.5 OPERATING ENVIRONMENT

The main component of the project is game product. Which will be limited for the android operating system (specifically android 4.1 and higher). Beyond that the game is self-contained we will not rely on any other android related component.

## 2.6 ASSUMPTIONS AND DEPENDENCIES

### 2.6.1 Time Dependencies

As I have mentioned earlier that features are divided into two groups. Core features and additional features. Core features are must implement part of the game for the basic functionality to work. While additional features are not must they will be provided for the grooming of core features of the game.

### 2.6.2 Hardware Dependencies

As I have used the word Gyroscope in this document above. This is the main part of the hardware resource that will run the game.

## 2.7 SYSTEM FEATURES

Catching the hen game features are divided into two main categories: core features and additional features. Core features form the body of the game and include any features that are essential to the functionality of the order to have a fully-functioning game. Additional features,

however, are not required for the app to function. They include any features which, if time permits, will be added to the application in order to provide extra functionality. [6]

## 2.8 SYSTEM CORE FEATURES

### 2.8.1 Main Menu

When the application is installed and whenever player enters or launched the game he will see this menu from where he can go into the following [6]

- **play level** (get to the play level scene)
- **settings** (get to the settings panel)
- **help** (get to the help panel)
- **quit** (quit the game)
- **game stats** (get to the game stats)

Mentioned in the product features.

#### 1. Stimulus response sequence

**Step1:** catching the hen game launched from the android screen.

**Step2:** the user is displayed with this main menu to go to and perform his/her required operations which were provided in main menu. [6]

#### 2. User requirements

Main menu can be accessed at any time when player launches the game.

#### 3. System requirements

All the functionalities of the game menu will be implemented in such that player can get to the required operations successfully.

### 2.8.2 Level Scene And Layout

When the player select the play level option from the main menu this system feature will be invoked which allows user to select the open level from the listed levels in the level scene panel. This panel will provide the features to use.

- **Level 1**
- **Level 2**
- **Level 3**

#### 1. Stimulus response sequence

**Step1:** catching the hen game launched from the android home screen.

**Step2:** the player is allowed to tap the play level button on the main menu.



**Step3:** player will allowed here to select open level and start the game.

## 2. Player requirements

In order to open new level it's the requirement of the player to complete the previous and all previous levels.

## 3. System requirements

The level panel or scene must be implemented in the way which convey and provides all the functionalities that are mentioned above.

### 2.8.3 Hen The Game Agent

When player enters into the game level all the game objects will be created of which one of the main component is the hen game agent which has its own algorithms with which it can observe the environment and perform some animations as described briefly in the product features. This part of the game totally focuses on the game agent hen the hens animations are as follows.

- **Hen idle state** (animate the initial state of the hen)
- **Hen running state** (animate the running state of the hen)
- **Hen walking state** (animate the walking state of the hen)
- **Hen jumping state** (animate the jumping state of the hen)

## 1. Stimulus response sequence

**Step1:** catching the hen game launched from the android home screen.

**Step2:** the player is allowed to tap the play level button on the main menu.

**Step3:** the player is then allowed to select the level from the level scene.

**Step4:** the game environment with the hen capable of different states animations will be created for the player.

## 2. User requirements

The user will have to tap on the above mentioned steps to get to this stage of the game.

## 3. System requirements

The hen agent must be implemented in such a way whenever user enters to the game environment it will generate the hen and its states that I have explained in product features.

**Algorithm:** this object must uses the algorithm that defines the every frame when to happen in the game. Hen algorithm defines the change of different animation states of the hen that I have described above.

### 2.8.4 Player Nets

When player starts the game the player net will be provided to the player in order to capture it. This is the main core feature that will allow player to throw the net onto the hen there will be different state animations design with this part game of which are.

#### 1. Stimulus response sequence

**Step1:** catching the hen game lunched from the android main screen

**Step2:** catching the hen game allows player to taps the play level button and start game from the level panel.

**Step3:** then the system will animate the throw able net in front of the player's game screen which can be thrown by sliding fingers towards the game object hen.

#### 2. User requirements

Player was required to slide fingers on the hen in order to activate the throw net frames which will show that the net has been thrown the hen.

#### 3. System requirements

The player net is implemented in such a way that whenever player enters the game system must generate this player net object in order to capture the hen.

### 2.8.5 Capture Hens

When player throws the net onto the hen then this game object animation will be visible to the player which will show the hen captured inside the net trying to escape and then the additional feature that hen will be saved in the game stats and records in the local database of game.

#### 1. Stimulus response sequence

**Step1:** player is in the game environment and he throws the net onto the hen by sliding his finger on the net.

**Step2:** if net collides with the hen's collider then game will activate the hen captured animation to the player.

**Step3:** this information sent to the local database that will be used by the player latter to see which hen he captured.

#### 2. User requirements

There will be no such a player requirements in this case.

#### 3. System requirements

Capture the hen part of game will be implemented in such a way that allows player to view the captured animation of the hen.

### 2.8.6 Gyro Camera Setup

When player selects the play level option from the level panel catching the hen game will access the camera of mobile device and to calculate its rotations the gyroscope of mobile device will be used in order to follow the animated hen.

#### 1. Stimulus response sequence

**Step1:** the game will be launched from the androids home screen.

**Step2:** camera setup will show the real environment using the camera of mobile.

**Step3:** then gyro camera will calculate the cameras rotations and give a smooth view of environment to the player.

#### 2. User requirements

Only thing from the user will be the camera. This will be allowed to use by the game.

#### 3. System requirements

The gyroscope camera setup must be implemented in such a way that whenever user selects the play level it will create the real environment visible through his camera using the gyroscope rotations.

### 2.8.7 Game Audio

When player starts interacting with the hen the hen will going to make different sounds on the basis of its changing states. Those sounds will be provided by the system which is as follows.

- **Capture hen sound** ( system plays the capture hen sound)
- **Button sounds** ( sounds of buttons in menu and other game environment)
- **Net sound** ( sound of the net when thrown)
- **Game music** ( system plays the game music whenever player launches the game)

#### 1. Stimulus response sequence

**Step1:** player launches the game from android home screen and starts the game.

**Step2:** at first system will play the initial hen sound or the eating hen sound which if any food is there for hen.

**Step3:** when player moves towards hen it will detect the player getting near it so the system plays the walking sound, getting more nearer to the hen plays the running hen sound from the system, similarly jumping sound is played when player throws and misses the net.

#### 2. User requirements

Player is not required to do anything in this part implementation of game all operations will be performed by the system.

### 3. System requirements

All the above mentioned operations will be implemented in such way that it interacts the player with the game. System must uses algorithm that defines the changing of the above state sounds of the hen.

#### 2.8.8 Player Score

Whenever user completes the level system will provide the scores for the player. Player scores are defined in stars.

##### 1. Stimulus response sequence

**Step1:** catching the hen game is launched from the android home screen and user will be in the play level stage.

**Step2:** player uses the nets in order to capture the hen.

**Step3:** player is rewarded with the level score and game time nets used and other details of the game.

##### 2. User requirements

Player is required to slide the net and try to use the minimum nets to get good scores.

##### 3. System requirements

Catching the hen game score will be implemented in such a way that it will calculate the number of nets used and displays the stars in a push menu displayed at the end of the level which displays the stars.

## 2.9 ADDITIONAL FEATURES

### 2.9.1 Game Stats Records

This panel allows user to see the previous level game scores and catcher hens. For the storing of the game scores, stars and hens system will use small database.

##### 1. Stimulus response sequence

**Step1:** from main menu player selects the game stats and records option

**Step2:** the system gets the game scores, level received stars and hens from the database.

**Step3:** it also allows player to go back to the main menu screen.

##### 2. User requirements

There is no as such requirement from the player in this case.

### 3. System requirements

System will send request to the database to get the saved records of game scores, level received stars and hens from the database.

#### 2.9.2 Settings Menu

This menu allows the user to modify more advanced settings within the game. The menu is accessed from the main menu where we place the setting button, Modifiable settings include:

- **Game music slider**

##### 1. Stimulus response sequence

**Step1:** from the main menu player can select the settings option.

**Step2:** this will display the settings panel to the user can modify or customize the above mentioned settings.

**Step3:** the user is then allowed to go back to the main menu and can select another option.

##### 2. User requirements

There is no any player requirement as such player can access the settings form the main menu any time he want to change his settings.

##### 3. System requirements

System will implement the settings in such way that all the modifiable settings will be applied successfully to the player.

#### 2.9.3 Help Menu

The Help menu is meant to answer any questions the user may have while using the game. The Menu displays a list of topics related to features, menus, and the game in general. The user can Select any of these topics to access the further information.

##### 1. Stimulus response sequence

**Step1:** from the main menu player can select the help menu.

**Step2:** this displays the help menu panel to the player.

**Step3:** the user is presented with the list of help topics

**Step4:** the user can go back to the main menu and can play game.

##### 2. User requirements

There will be no player requirements in this case all the operations will be done by the system to display the interactive help menu for player.

### 3. System requirements

The system must contain small data base to containing the detail information's of the help menu topics.

## 2.10 SOFTWARE SYSTEM ATTRIBUTES

### 2.10.1 Reliability

The game must be reliable because it is dealing with the lots of graphics so it must be design with the best animations rules to not get slow down. It will not crash when user wants to perform desired operation it will be reliable enough to integrate the player with game. I have to consider the memory managements also to make my product more reliable so it can be used with low memory mobile phone easily.

### 2.10.2 Availability

The defined features in the SRS must be available and working properly to make the basic game runnable. GPS data calculation must be available in our product to similarly creation or making of game hen will be available to the player whenever he plays the game. It also works fine on the operating systems or software's I have mentioned earlier.

### 2.10.3 Security

None game will not use any personal data of the player therefore this product does not concern with any security attribute.

### 2.10.4 Maintainability

"The ease with which a software system or component can be modified to correct faults, improve performance or other attributes, or adapt to a changed environment."

Product must stay in its original state and condition like it is when it was first install it ensures no memory leakages, which causes whole mobile system to be slow down so it's necessary to remove memory flaws.

### 2.10.5 Portability

"Portability in high-level computer programming is the usability of the same software in different environments."

Catching the hen is game product and it must run on all the devices of android whose version is higher the 4.0

### 2.10.6 Performance

Game will not compromise on the performance issues every features of the game must be implemented in a way that player will face not any problem playing the game.

The main component of the game agent hen is made up of lots of animations these animation states will have to be implemented in very detailed algorithm to make the game features more expressive there will be several other features like playing sound

throwing nets on the all of them will work properly with crashing will ensure the products performance. [3]

## 2.11 USE CASES

As I have completed my requirement phase here I will explain the design phase of my project about creating the augmented reality game catching the hen. The use cases I explored in my requirement analysis are as follows. [3][4]

1. Play game
2. Play next level
3. Change settings
4. Capture hen
5. Activate captured Animations
6. View game stats
7. Create game environment
8. Create game agent hen
9. Set hen movements
10. Activate hen's idle animation.
11. Activate hen's running animation
12. Activate hen's walking animation
13. Activate hen's jumping animation
14. Create player nets
15. Activate throw net
16. Setup gyro camera
17. Create real environment
18. Play game audio
19. Display score
20. View help menu

### 2.11.1 Play game

<b>I. use case name</b>	Play game
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.

<b>V. Pre-conditions</b>	Game must be launched from android mobile device
<b>VI. post conditions</b>	Game runs successfully on the device and welcome player to the main menu screen
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player launches the game from the android home screen</li> <li>2. Game successfully runs on the players device</li> <li>3. Game will display the main menu as welcome screen for the player</li> <li>4. Game will now allow player to select any option from the main menu.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. if game crashes and unable to run on the android device</li> <li>2. Game unable to display the main menu instead it crashes or displays some other content</li> <li>3. Game fails to accept the inputs from the player</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can play game on daily basis

### 2.11.2 Play next level

<b>I. use case name</b>	Play next level
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre-conditions</b>	Player must open play levels scene so he will able to select a level
<b>VI. post conditions</b>	Level scene will be opened for the player where he will able to select the desired level.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player launches the game from the android home screen</li> <li>2. Game successfully runs on the players device</li> <li>3. Game will display the main menu as welcome screen for</li> </ol>



	<p>the player</p> <ol style="list-style-type: none"> <li>4. Player clicks on select level option and game will take him to the level panel</li> <li>5. Player can now select level from all three different level options.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. if game crashes and unable to run on the android device</li> <li>2. Game unable to display the main menu instead it crashes or displays some other content</li> <li>3. Game fails to accept the inputs from the player</li> <li>4. Game will not take the player into the level scene</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

### 2.11.3 Change settings

<b>I. use case name</b>	Change settings
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre-conditions</b>	Player must be in the main menu in order to go into the settings.
<b>VI. post conditions</b>	Settings menu runs successfully and runs all the changes layer made.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player selects the change settings option from the main menu.</li> <li>2. Settings menu will appear on the players mobile device</li> <li>3. Player will able to change the audio settings.</li> <li>4. Made changes must be operational to the player.</li> </ol>

<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player was not able to select the settings option.</li> <li>2. Settings menu will not be displayed on the screen.</li> <li>3. Player will not be able to change the settings inside the Settings menu.</li> <li>4. Made changes doesn't work in the game.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

#### 2.11.4 Capture hen

<b>I. use case name</b>	Capture hen
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will go to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer-generated hens. A player net is provided in the game which will allow the player to capture the game agent hen.
<b>V. Pre-conditions</b>	Player must throw the net in order to capture the hen.
<b>VI. post conditions</b>	Player captured the hen and completes the level.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player throws the net in the direction of the hen</li> <li>2. If player net collides with the game agent collider.</li> <li>3. Activate the hen captured state</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player didn't throw the net.</li> <li>2. If it does not collide with the hen collider</li> <li>3. Activation of the hen captured state was unsuccessful.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

**2.11.5 Activate capture Animations**

<b>I. use case name</b>	Activate captured net
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre-conditions</b>	Net collider collides with the hens collider
<b>VI. post conditions</b>	Hens captured animation will be displayed on the player's mobile screen.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player throws the net in the direction of the hen.</li> <li>2. Game detects the net collider collides with the hen.</li> <li>3. Game activates the captured net animation in the player's mobile device.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player didn't throw the net in the direction of the hen.</li> <li>2. Game wasn't able to detect the collider of net and the hen.</li> <li>3. Game wasn't able to activate the captured net animation in the player's mobile device.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

**2.11.6 View game stats**

<b>I. use case name</b>	View game stats
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player

<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre-conditions</b>	Player must be in the main menu screen. And selects the game stats
<b>VI. post conditions</b>	Game will display the player played statistics i.e high scores in the level.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player selects the game stats form the main menu of the game.</li> <li>2. Game successfully takes player to the game stats panel</li> <li>3. Player successfully sees the records of the high scores he made.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player will not able to select the game stats from the main menu of the game.</li> <li>2. Game failed to take the player into the game stats panel</li> <li>3. Player wasn't was no provided the correct high score data he needed.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

### 2.11.7 Create game environment

<b>I. use case name</b>	Create game environment
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.

<b>V. Pre-conditions</b>	Player selects next level from the level panel.
<b>VI. post conditions</b>	Proper game environment will be created to the player where he can play his game.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player selects the level from the level panel.</li> <li>2. Player press the play game button.</li> <li>3. Game will take player into the real game environment.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player wasn't able to select the level from the level panel</li> <li>2. Player wasn't able to press the play game button</li> <li>3. Real game environment wasn't displayed in the player's mobile screen.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

#### 2.11.8 Create game agent hen

<b>I. use case name</b>	Create game agent hen
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre-conditions</b>	Player must be inside the game environment.
<b>VI. post conditions</b>	Game will create the hen near the location of the player which can able to walk, run and jump inside the game environment of the player.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player play the game from the level panel it brings him to the game environment.</li> <li>2. Game will create the hen near the player.</li> <li>3. Player rotates the camera in order to see the created hen.</li> </ol>

<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player wasn't able to play the game from the level panel.</li> <li>2. Game wasn't able to create the hen near the player.</li> <li>3. The rotation of player camera doesn't worked.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

### 2.11.9 Set hen movements

<b>I. use case name</b>	Set hen movements
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre-conditions</b>	In the game environment of the current level the game hen agent was created.
<b>VI. post conditions</b>	Game will able to provide the hens movements left right.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player is provided by the animated hen in his real game environment.</li> <li>2. Game will randomly select the hen's right or left changing in the latitude and longitudes of the hen.</li> <li>3. If game selects the right changing latitudes then hen will move right of the screen</li> <li>4. If game selects the left changing latitudes then hen will move left of the screen.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player was not provided the animated hen in his real game environment.</li> <li>2. Game wasn't able to select the hen's right or left changing positions.</li> <li>3. Game wasn't able to change the position of the hen</li> </ol>

	according to the latitudes.
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

### 2.11.10 Activate hens idle state

<b>I. use case name</b>	Activate hens idle state
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre conditions</b>	Hen must be created in the game environment.
<b>VI. post conditions</b>	Initial hen movements like moving head and sitting, standing animations are activated.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player must open game environment</li> <li>2. Game environment creates hen in the real environment</li> <li>3. Activate the hen's idle animation.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player failed to open the game environment</li> <li>2. Game environment failed to create the hen in the real environment.</li> <li>3. Activation of hen's idle animation failed to load.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

**2.11.11 Activate hens running state**

<b>I. use case name</b>	Activate hens running state
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre-conditions</b>	Hen must be created in the game environment.
<b>VI. post conditions</b>	Hen's movements like running states are enabled and hen starts to run in the real environment.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player must open the game environment</li> <li>2. Game environment creates hen in the real environment</li> <li>3. Activate the hen's running animation.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player failed to open the game environment</li> <li>2. Game environment failed to create the hen in the real environment.</li> <li>3. Activation of hen's running animation failed to load.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

**2.11.12 Activate hens walking state**

<b>I. use case name</b>	Activate hens walking state
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player



<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre-conditions</b>	Hen must be created in the game environment.
<b>VI. post conditions</b>	Hen's movements like running states are enabled and hen starts to run in the real environment.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player must open the game environment</li> <li>2. Game environment creates hen in the real environment</li> <li>3. Activate the hen's walking animation.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player failed to open the game environment</li> <li>2. Game environment failed to create the hen in the real environment.</li> <li>3. Activation of hen's walking animation failed to load.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

### 2.11.13 Activate hens jumping state

<b>I. use case name</b>	Activate hens jumping state
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre-conditions</b>	Hen must be created in the game environment.

<b>VI. post conditions</b>	Hen's movements like running states are enabled and hen starts to run in the real environment.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player must open the game environment</li> <li>2. Game environment creates hen in the real environment</li> <li>3. Activate the hen's jumping animation.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player failed to open the game environment</li> <li>2. Game environment failed to create the hen in the real environment.</li> <li>3. Activation of hen's jumping animation failed to load.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

#### 2.11.14 Create player nets

<b>I. use case name</b>	Create Player nets
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre-conditions</b>	Player must be inside the game environment.
<b>VI. post conditions</b>	Game will create the player nets at the bottom of the player game environment screen, which player can able to throw in order to capture the hen.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player play the game from the level panel it brings him to the game environment.</li> <li>2. Game will create the hen net.</li> <li>3. It will allow player to project the net on to the hen</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player wasn't able to play the game from the level panel.</li> <li>2. Game wasn't able to create the net.</li> <li>3. It will not allow player to project the net onto the hen.</li> </ol>

<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can perform this task when he plays the game environment.

#### 2.11.15 Activate initial net

<b>I. use case name</b>	Activate initial net
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre-conditions</b>	Net must be created in the game environment.
<b>VI. post conditions</b>	Nets initial net state where net is displayed for the player in the screen will be created.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player must open the game environment</li> <li>2. Game environment creates net in the real environment</li> <li>3. Activate the nets initial state where net moves a little in order to show that this net can be thrown.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player failed to open the game environment</li> <li>2. Game environment failed to create the net in the real environment.</li> <li>3. Activation of nets initial state failed to load.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can interact with this state whenever he plays the game.

#### 2.11.16 Setup gyroscope

<b>I. use case name</b>	Setup gyroscope
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player

<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. This case is especially in interest because it allows user to capture the real images of environment through camera and display them on our mobile screen.
<b>V. Pre-conditions</b>	Player selects the play game option and will must be in the game environment.
<b>VI. post conditions</b>	Game environment will first access the camera and starts displaying the real environment in the player's mobile screen.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player selects the play game option in the level list.</li> <li>2. Game will access the gyroscope the mobile.</li> <li>3. Game will display the real environment in the player's mobile screen.</li> <li>4. It will calculate the left, right, top, bottom rotations of the camera.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player wasn't able to select the play game option</li> <li>2. Game failed to access the gyroscope.</li> <li>3. Game wasn't able to display the real environment in the player's mobile screen.</li> <li>4. It will not rotate according the payer's camera movements.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game. It also needs the access of the gyroscope of the mobile device enabled in order to get the rotations of the camera.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

#### 2.11.17 Create real environment

<b>I. use case name</b>	Create real environment
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player

<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre-conditions</b>	Gyroscope must be accessed and set to calculate the rotations.
<b>VI. post conditions</b>	Game will create the camera accessed images on to the surface of the mobile screen.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player selects the play game option in the level list.</li> <li>2. Game will access the gyroscope of the mobile.</li> <li>3. Game will display the real environment in the player's mobile screen.</li> <li>4. The images were placed on the Quad object where our game camera is looking at right now.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player wasn't able to play the game.</li> <li>2. Game fails to access the gyroscope of the mobile.</li> <li>3. Game failed to display the real environment.</li> <li>4. The images were not placed on the quad object where our game camera is looking at right now.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game. It also needs the Gyroscope of the mobile enabled to get the access of the camera rotations.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

#### 2.11.18 Play game audio

<b>I. use case name</b>	Play game audio
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player

<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.
<b>V. Pre-conditions</b>	Player must launch the game from the android home menu.
<b>VI. post conditions</b>	Game will play the game audio in order to give more attraction to the player.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player launches the game from the android home screen</li> <li>2. Game successfully runs on the players device</li> <li>3. Game will play the game audio.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. if game crashes and unable to run on the android device</li> <li>2. Game unable to display the main menu instead it crashes or displays some other content</li> <li>3. Game will failed to play the game audio.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game. It needs the mobile set of speaker's to listen the audio of the game.
<b>X. frequency of use</b>	Player can listen the game music or he can mute it from the setting panel of the game.

### 2.11.19 Display score

<b>I. use case name</b>	Display score
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level.

<b>V. Pre-conditions</b>	Player captures the hen or he misses all the three nets.
<b>VI. post conditions</b>	Game will display the achieved stars according to the nets he consume to capture the hen at the end.
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player throws the net on to the hen in order to capture the hen</li> <li>2. If player catches the hen then the score will be displayed to the player.</li> <li>3. If he catches the hen consuming one net then he will get the three stars displayed on the screen</li> <li>4. If catches the hen with 2 nets consumed then he will get the two stars displayed on the screen</li> <li>5. If catches the hen with consuming all the stars then he will get only one star.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. Player wasn't able to throw the net on to the hen.</li> <li>2. If player catches the hen the score panel wasn't displayed in the player's screen</li> <li>3. System fails if score of stars displayed in wrong manners.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game

#### 2.11.20 View help menu

<b>I. use case name</b>	View Help menu
<b>II. scope</b>	An augmented reality catching the hen game
<b>III. primary actor</b>	Player
<b>IV. stake holders and interests</b>	<p>Player will be any type of user who will going to play this game. He had to achieve the best scores in different levels to get to the other level. Players were provided the real augmented reality environment where he can interact with the computer generated hens. A player net is provided in game which will allow player to capture the game agent hen.</p> <p>Will have to provide the help screen in order to teach the player how to play the game.</p>

<b>V. Pre conditions</b>	Player must be in the game menu in order to switch into the help menu screen.
<b>VI. post conditions</b>	Game will take player into the help menu screen where he can see the
<b>VII. success scenarios</b>	<ol style="list-style-type: none"> <li>1. Player launches the game from the android home screen</li> <li>2. Game successfully runs on the players device</li> <li>3. Game will display the main menu as welcome screen for the player</li> <li>4. Player selects the help menu from the main menu.</li> <li>5. The required help will displayed for the player.</li> </ol>
<b>VIII. failure scenarios</b>	<ol style="list-style-type: none"> <li>1. if game crashes and unable to run on the android device</li> <li>2. Game unable to display the main menu instead it crashes or displays some other content</li> <li>3. Game fails to accept the inputs from the player</li> <li>4. Game will not take the player into the help menu screen</li> <li>5. The required help was not helpful for the player.</li> </ol>
<b>IX. Special requirements</b>	Touch screen mobile device which is used for user interface requirements is needed for running the game.
<b>X. frequency of use</b>	Player can do this task on daily basis whenever he plays game



2.12 USE CASE DIAGRAM



Figure 2.1 Use case diagram

## 2.13 SYSTEM SEQUENCE DIAGRAM

In software engineering, a system sequence diagram (SSD) is a sequence diagram that shows, for a particular scenario of a use case, the events that external actors generate their order, and possible inter-system events. [3] System sequence diagrams are visual summaries of the individual use cases. A system sequence diagram should be done for the main success scenario of the use case, and frequent or complex alternative scenarios.

### 2.13.1 Play game

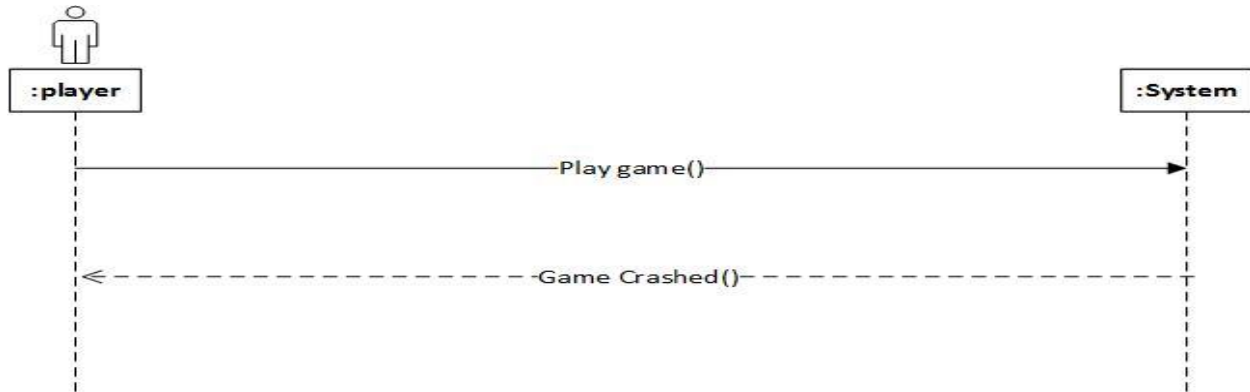


Figure 2.2 System Sequence Diagram of Play game

The above diagram explains the scenario when player clicks on the play game to play a level it crashes because of the failure in the system we always mention the alternate scenarios in system sequence diagrams.

### 2.13.2 Create game environment

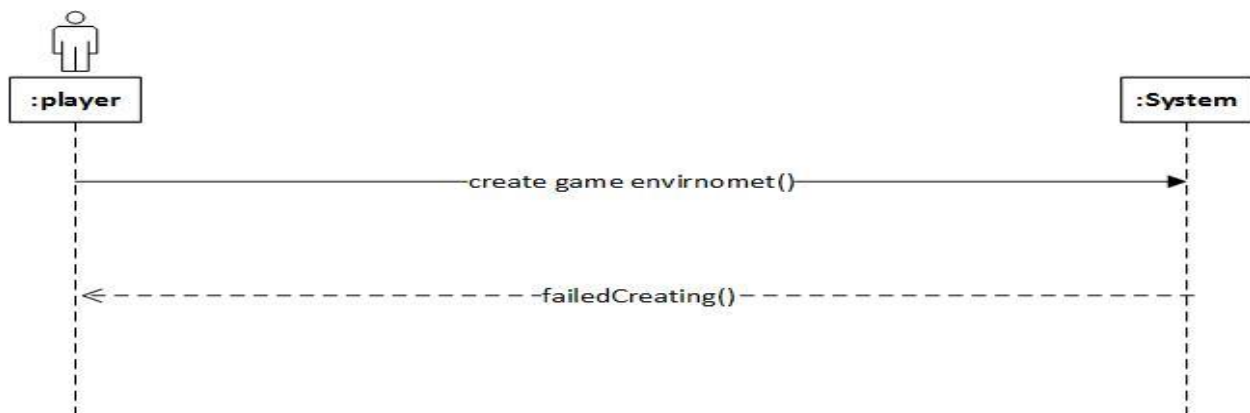


Figure 2.3 System Sequence Diagram of Game Environment

The above diagram explains the scenario when player clicks on the specific level it displays blank screen because of the failure in the system to access the camera.

### 2.13.3 Create game agent hen

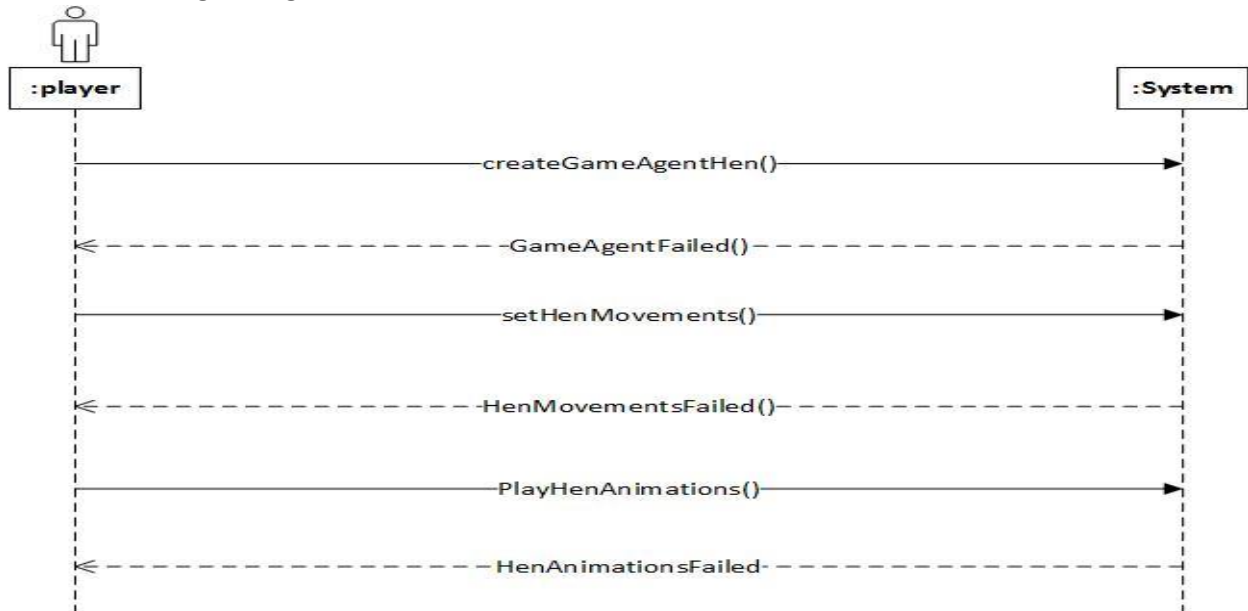


Figure 2.4 System sequence Diagram for hen agent

The above diagram explains the scenario of the hen game object when system is loads the level it failed to spawn the hen into environment of the animations of the hen failed to load by the system.

### 2.13.4 Create player nets

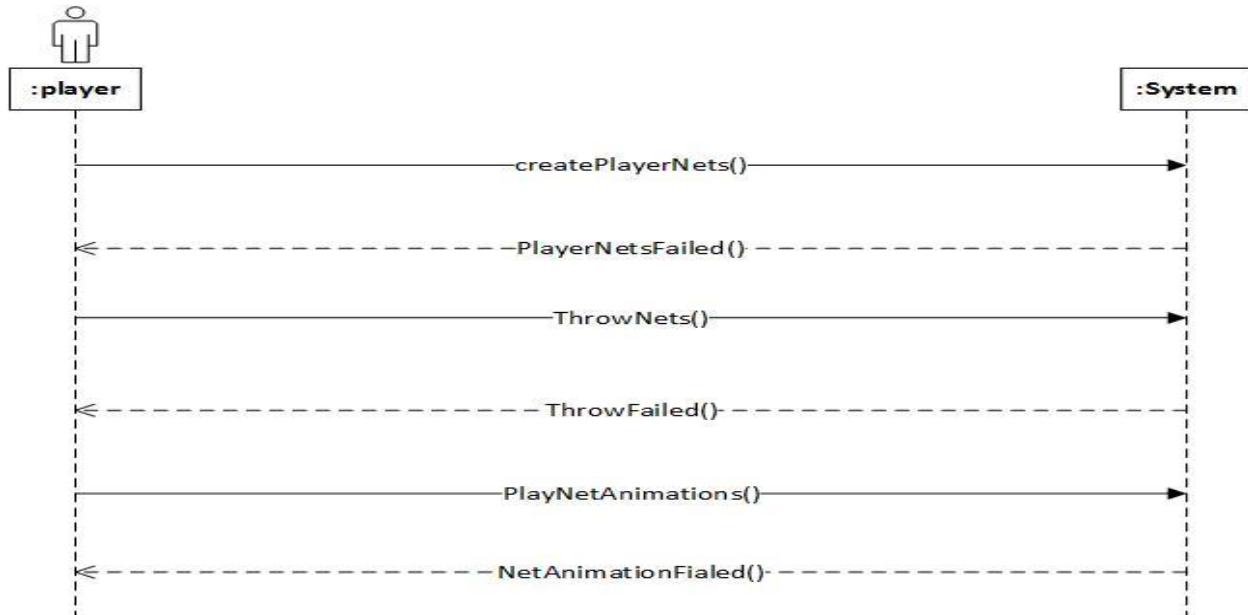


Figure 2.5 System Sequence Diagram of Create Player nets

The above diagram explains the scenario of the game net user flip the finger on the mobile screen and it failed to throw the net in the proper direction. The resetting net position also failed to reset the net in proper place.

### 2.13.5 Calculate hen position

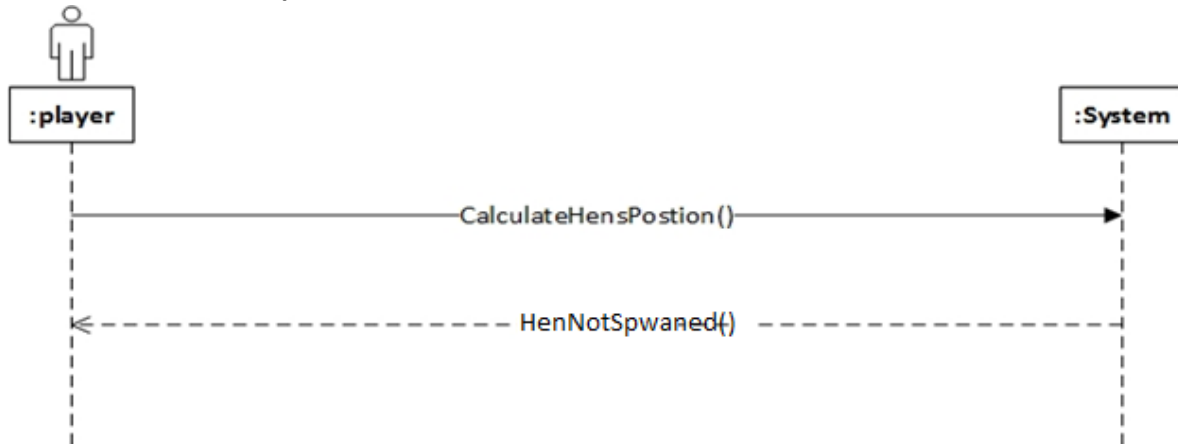


Figure 2.6 System Sequence Diagram of Hen Position

The above diagram explains the scenario of the calculating the hens positions it will be failed if the hen is not spawned in the game environment there will be no calculation of the hens position if the hen is not spawned.

### 2.13.6 Play game audio

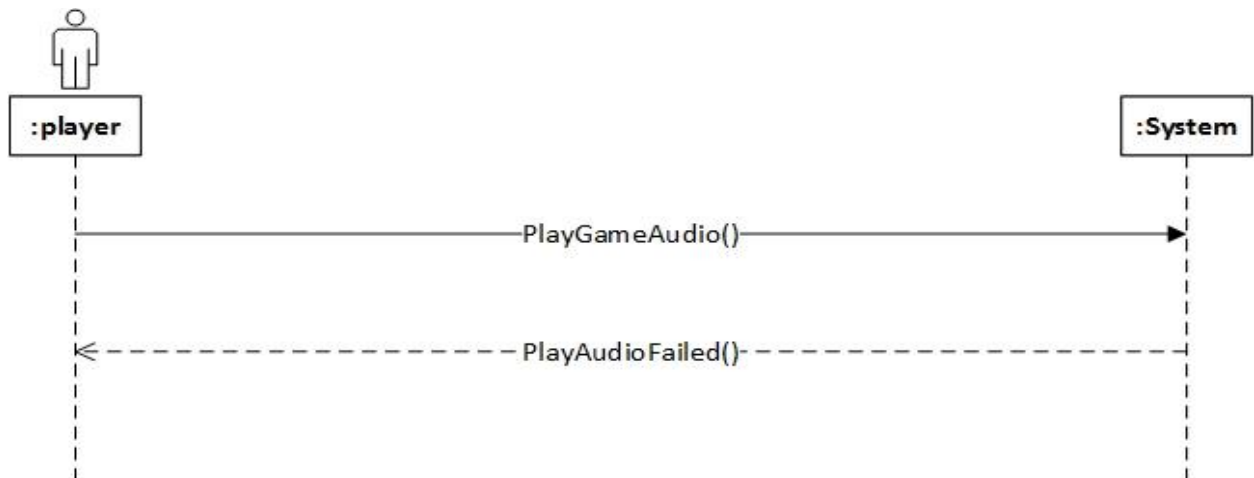
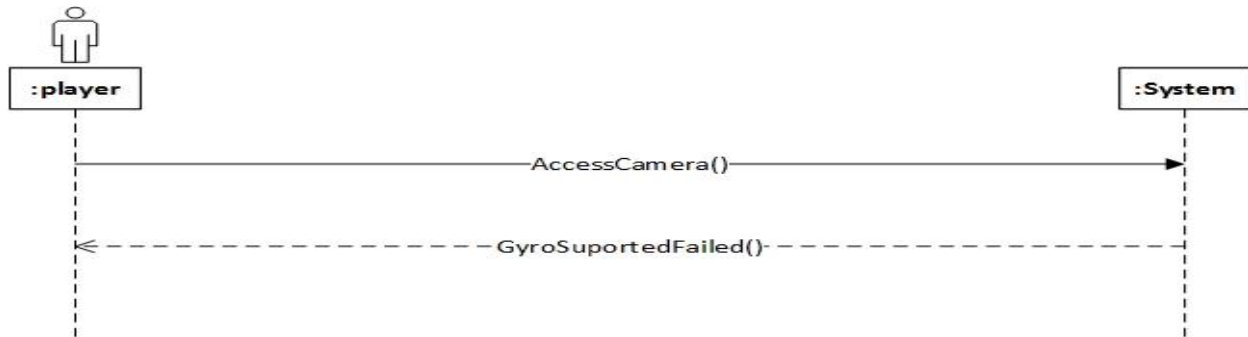


Figure 2.7 System Sequence diagram of Play game audio

The above diagram explains the scenario of the game audio whenever the game audio had to play by the system it failed to play because of the wrong reference to the game music.

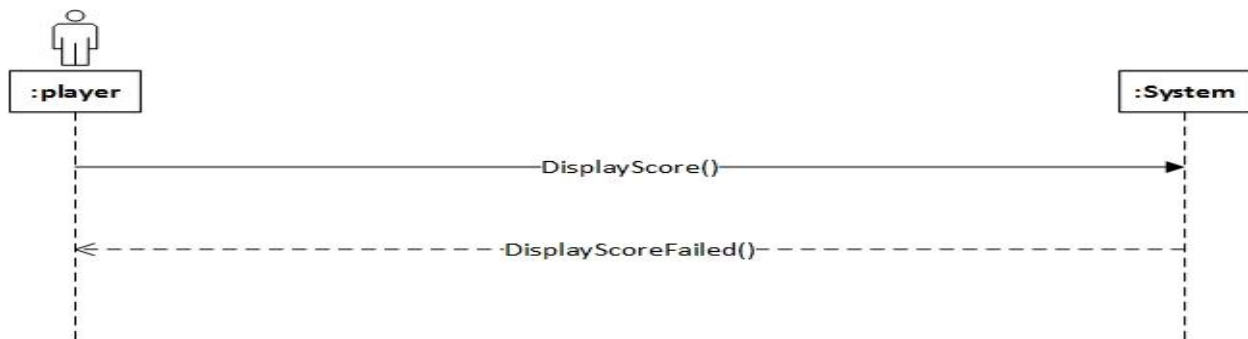
### 2.13.7 Setup camera



*Figure 2.8 System Sequence Diagram of Gyroscope Rotations*

The above diagram explains the scenario of the Gyroscope rotations the system fail to access the camera when there will be no gyroscope supported and it will not calculate any rotations for the game environment.

### 2.13.8 Display score



*Figure 2.9 System Sequence Diagram of Display Score*

## 2.14 SUMMARY

In this chapter we analyzed all the software requirements of our project and write them in an proper way first with the stimulus responses for the particular functionalities then we explore all the use cases associated with the project we also described the each use case in detail. The use case diagram is also provided in this chapter for good explanation of the project functionalities with the system sequence diagrams that also illustrate the alternate scenarios with the system.

## Chapter 3

# Software design

### 3 CHAPTER 3: SOFTWARE DESIGN

In this chapter, we are discussing about the design part of the game. I have discussed about the software specification part and the basic requirement part in above two chapters. Now it's time to write the design part of the system in this part I will explain about the user interfaces, system sequence diagrams related to their use cases. State transition diagram, sequence diagram, class diagram.

#### 3.1 DESIGN OVERVIEW

Systems design is the process of defining the architecture, components, modules, interfaces, and data for a game to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development.

#### 3.2 REQUIREMENTS TRACEABILITY MATRIX

The Requirements Traceability Matrix (RTM) is a document that links requirements throughout the validation process. The purpose of the Requirements Traceability Matrix is to ensure that all requirements defined for a system are tested in the test protocols. [3]

Requirement id	Requirement name	Sequence Diagram	Test case	Class Diagram	Interface
UC-1	Play game	Figure 3.10	Test1	Main menu	Figure 3.2
UC-2	Play next level	Figure 3.10	n/a	Level	Figure 3.6
UC-3	Change settings	Figure 3.10	n/a	Settings Control	Figure 3.3
UC-4	Throw nets	Figure 3.11	Test4	Net Throw	Figure 3.7
UC-5	Capture hen	Figure 3.11	Test4	Net Throw	n/a
UC-6	Activate captured net	Figure 3.11	Test4	Net Throw	n/a
UC-7	View game stats	Figure 3.10	n/a	Stats	Figure 3.5
UC-8	Create game environment	Figure 3.12	Test2	Level	Figure 3.6
UC-9	Create game agent hen	Figure 3.12	Test3	Hen Spawn	Figure 3.7
UC-10	Set hen movements	Figure 3.11	Test3	Hen Controller	Figure 3.7
UC-11	Activate hens idle frame	Figure 3.11	Test3	Hen controller	Figure 3.7
UC-12	Activate hens running frame	Figure 3.11	Test3	Hen controller	Figure 3.7
UC-13	Activate hens walking frame	Figure 3.11	Test3	Hen controller	Figure 3.7
UC-14	Activate hens	Figure 3.11	Test3	Hen	Figure 3.7

	jumping frame			controller	
UC-15	Create player nets	Figure 3.11	Test4	Net Throw	Figure 3.7
UC-16	Activate initial net	Figure 3.11	Test4	Net Throw	Figure 3.7
UC-17	Activate thrown net	Figure 3.11	Test4	Net Throw	Figure 3.7
UC-18	Setup gyro camera	Figure 3.12	Test5	Camera Gyroscope	n/a
UC-19	Create real environment	Figure 3.12	Test2	Level	Figure 3.7
UC-20	Play game audio	Figure 3.10	Test6	Music	n/a
UC-21	Display score	Figure 3.11	n/a	n/a	n/a
UC-22	View help menu	Figure 3.10	n/a	n/a	Figure 3.4

### 3.3 SYSTEM ARCHITECTURE DESIGN

Ever since the first program was divided into modules, software systems have had architectures, and programmers have been responsible for the interactions among the modules and the global properties of the assemblage. Historically, architectures have been implicit accidents of implementation, or legacy systems of the past. Good software developers have often adopted one or several architectural patterns as strategies for system organization, but they use these patterns informally and have no means to make them explicit in the resulting system.

In software engineering, multitier architecture often referred to as n-tier architecture or multilayered architecture in which presentation, application processing, and data management functions are physically separated. The most widespread use of multitier architecture is the three-tier architecture. [3]

#### 3.3.1 Chosen System Architecture

The Chosen system architecture will be **three-tier architecture**.

Three-tier architecture is typically composed of a *presentation tier*, a *domain logic tier*, and a *data storage tier*.

**Presentation tier:** it is layer which user directly accesses and can interact to the main logic using the presentation layer.

**Application tier:** The logical tier is pulled out from the presentation tier and, as its own layer; it controls an application's functionality by performing detailed processing.

**Data tier:** the data tier would be in little touch with our project it is only used to store the game high scores that were latterly displayed to the user correctly and safely.



### 3.3.2 System Architecture Diagram

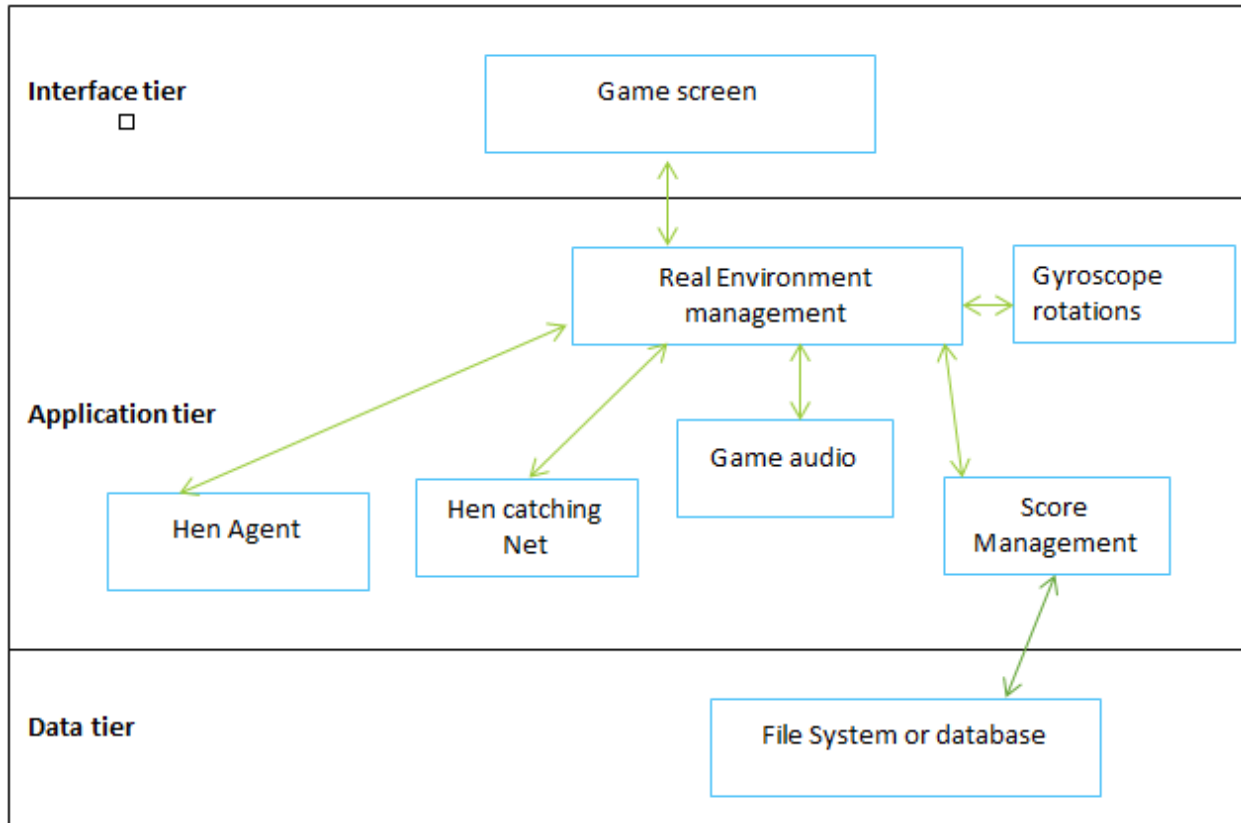


Figure 3.1 System Architecture Diagram

#### Simple and appealing:

User can easily interact with the systems buttons and the gameplay because they were simple and the game graphics will be very attractive so it felt appealing for the players.

#### Responsiveness:

We have lots of smart phones with different types of sizes and screens so our game will adjust its screen sizes in every android smart phone.

## 3.4 USER INTERFACE DESIGN

### 3.4.1 Description of User Interface

The interfaces used in this game are very simple so an ordinary user can also use this game very easily. The details about the system are defined and made clearly there will be also the help panel for the player in case if he didn't understand anything. The elements such as: buttons and colors, animations of the hen game object and the net throwing are designed very well for the player who is going to play this game. [3]

### 3.4.2 Game prototype details

I have developed prototype of some main operations which are mentioned below.

- Developed a functional main menu to check the required results.
- Developed the camera accessed image as real environment
- Developed the inside game object displayed inside that environment.
- Calculated the rotations from the gyroscope which are used to set the rotations and transformations of the game camera where the real world images are rendered on the unity raw image game object.

### 3.4.3 Screen Images

#### 1. Main menu

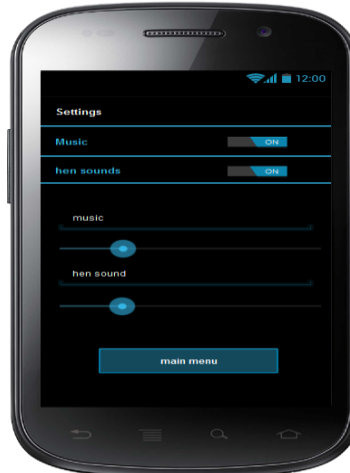
Main menu provides player to go to the game Environment using the play button it provides Other facilities as well like settings help menu Stats and quit option of the game. For further details about the main menu we explained It in the stimulus response part of system attributes In chapter 2 already.



Figure 3.2 Main Menu Screen

## 2. Settings menu

Settings menu of the game provides user to change the settings of the game. In which Player is only allowed to up and down the volume of the game music with the help of slider the detailed description of this functionality is written in the system attributes part in chapter 2.



*Figure 3.3 settings menu*

## 3. Help menu

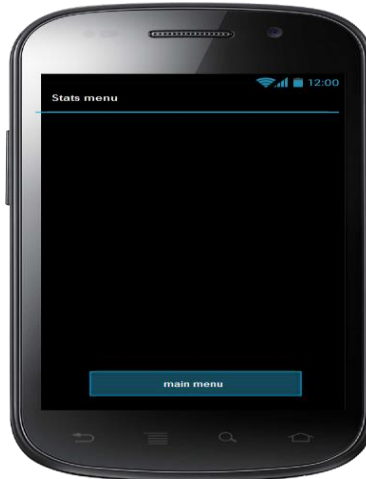
In the help menu there will be instructions for the player of how to play game. The stimulus responses written in the system core features will illustrate what will be in the Help menu.



*Figure 3.4 Help Menu*

#### 4. Stats panel

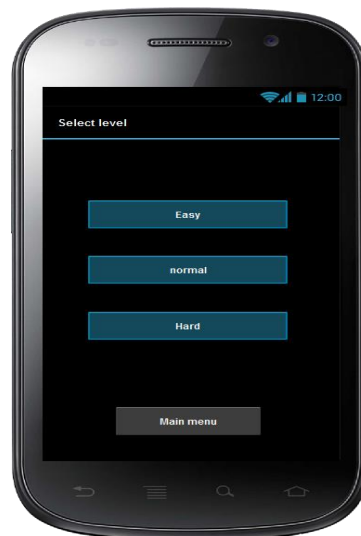
in the stats panel there will be record of the total hens captured during the whole game play there will be three individual best scores of the three levels. The stimulus responses written in the system core features will illustrate what will be in the stats panel.



*Figure 3.5 Stats Panel*

#### 5. Level list

In the level list screen there will be list of three levels on players choice on of which will be selected for the gameplay the stimulus responses written in the system core features will illustrate how levels will work.



*Figure 3.6 Level List*

## 6. Game environment

The game environment has real environment images displayed on the device. Where player can see an animated hen agent walking, running or jumping in a real environment. Game environment also holds the hen catching net.



Figure 3.7 Game environment

## 7. Score display

In the score display this is basically the winning panel which will display the result of the level when time finishes this will be displayed as level successful or level failed. The stimulus responses written in the system core features will illustrate how score will be displayed



Figure 3.8 Score panel

### 3.5 DOMAIN MODEL

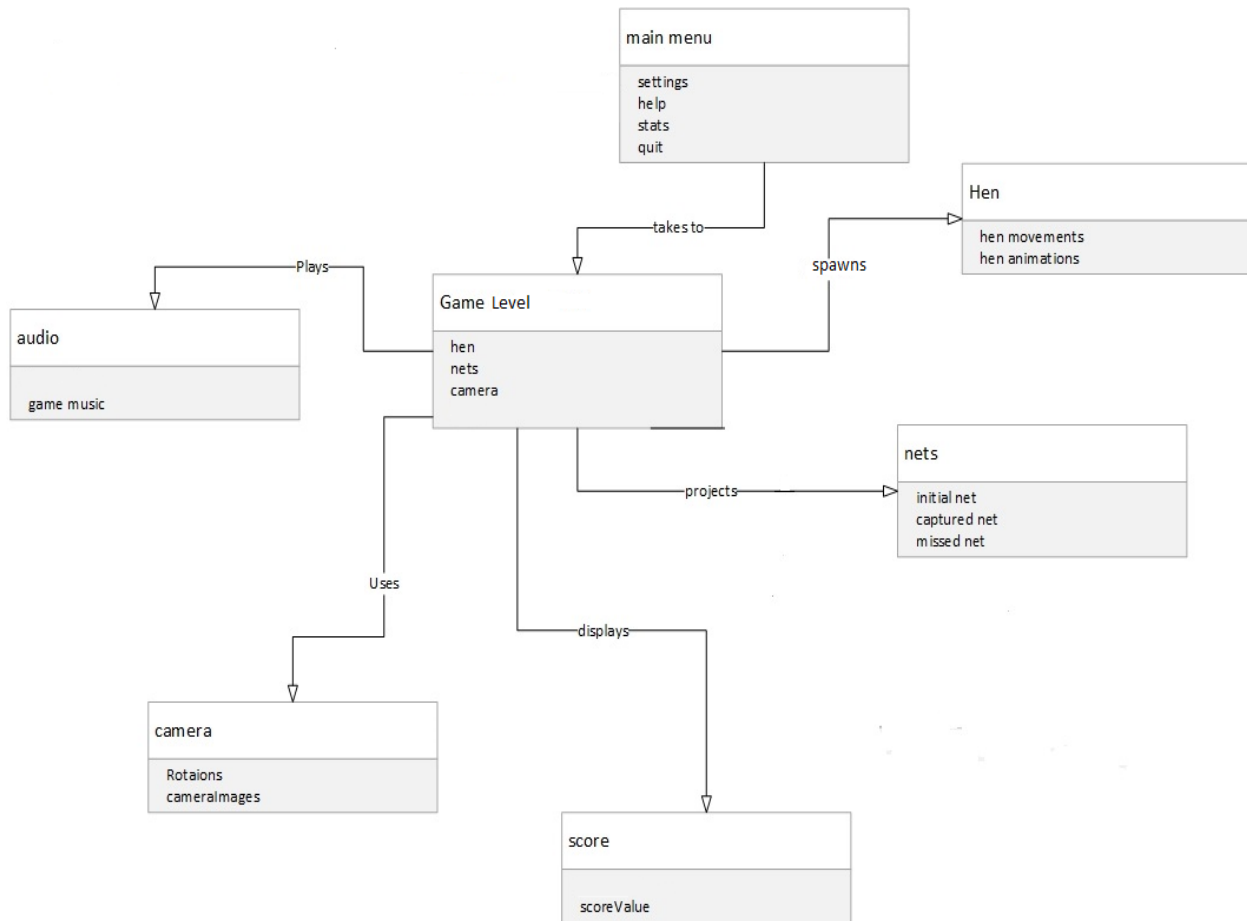


Figure 3.9 Domain model of catching the hen

### 3.6 SEQUENCE DIAGRAM

A Sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios. A sequence diagram shows, as parallel vertical lines (lifelines), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner. [9]

#### 1. Sequence diagram for Main menu

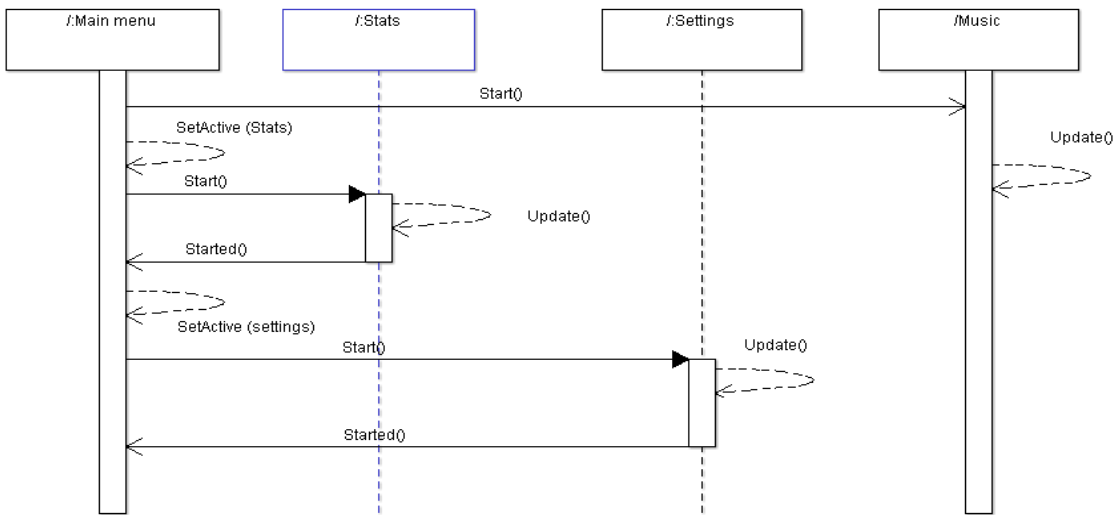


Figure 3.10 Sequence Diagram for main menu of catching the hen game

**Description:**

The sequence diagram of the main menu describes the interaction of the functions between the classes as we have described that the first screen is going to be the main menu it starts all the functions associated with the game menu.

**2. Sequence diagram for level and Game Environment**

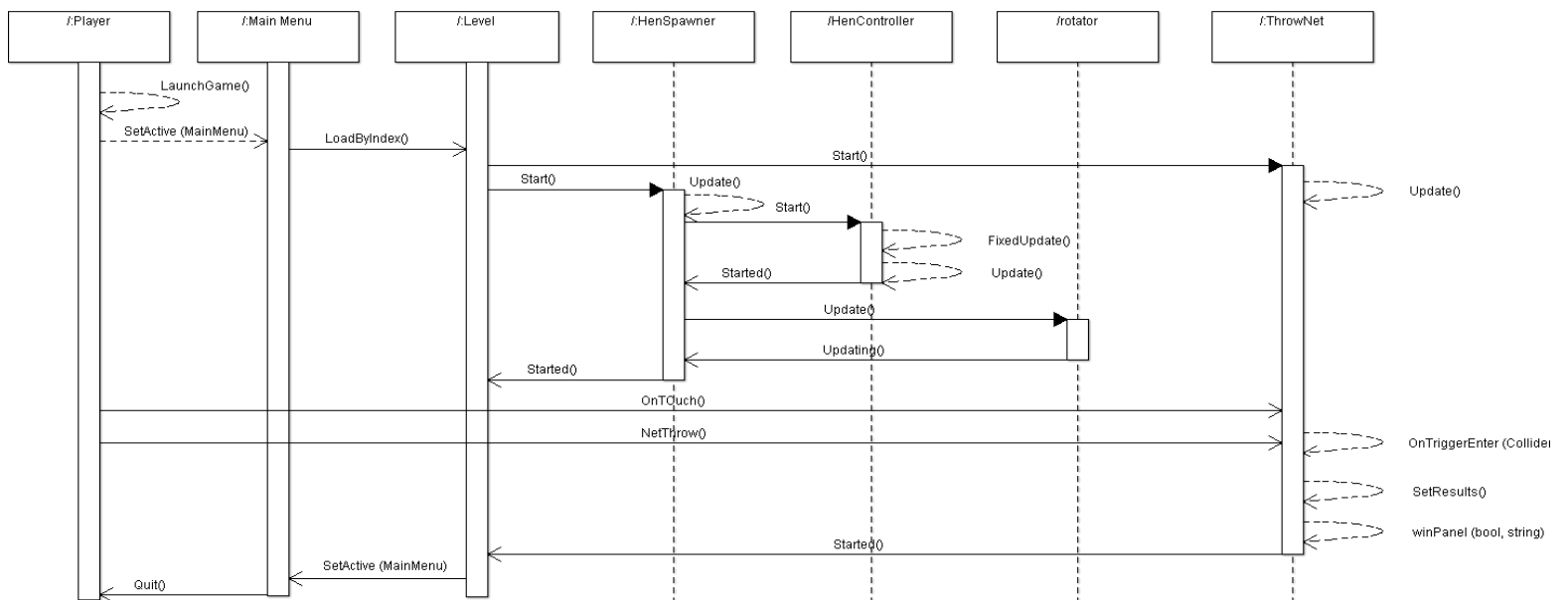


Figure 3.11 Sequence diagram for level and Game environment

**Description:**

The sequence diagram of the level and game environment shows the interactions of the functions associated with the level objects like loading a level that is selected, spawn hen in random position add bonus point to the level rotating the bonus point. Provide net for the player in order to catch the hen.

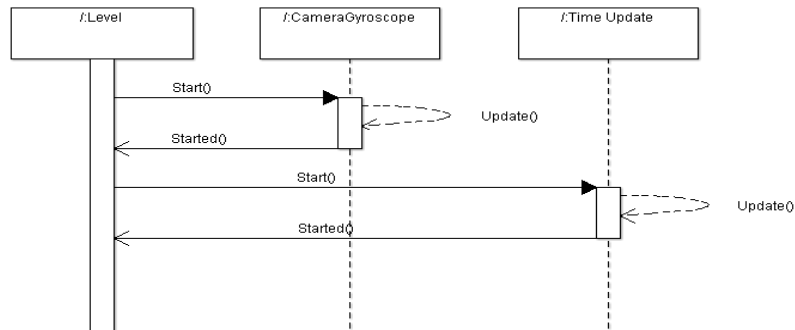
**3. Sequence diagram for camera gyroscope and game time**

Figure 3.12 Sequence diagram for camera gyroscope and game time

**Description:**

The sequence diagram of the camera gyroscope and game time is used to add the real world environment images into the mobile screen and time update script is used to update the game time.

**3.7 STATE DIAGRAMS****State diagram:**

The state diagram shows how the objects and actors need to interact to allow the junction to the function. What now is a picture of how the internal logic of these objects works to deliver this behaviour. In particular, we need to know that all the possibilities are correctly handled. The following figures will show the state chart for game exaction, changings of the hens animation state and changings in the net animation states. [4]



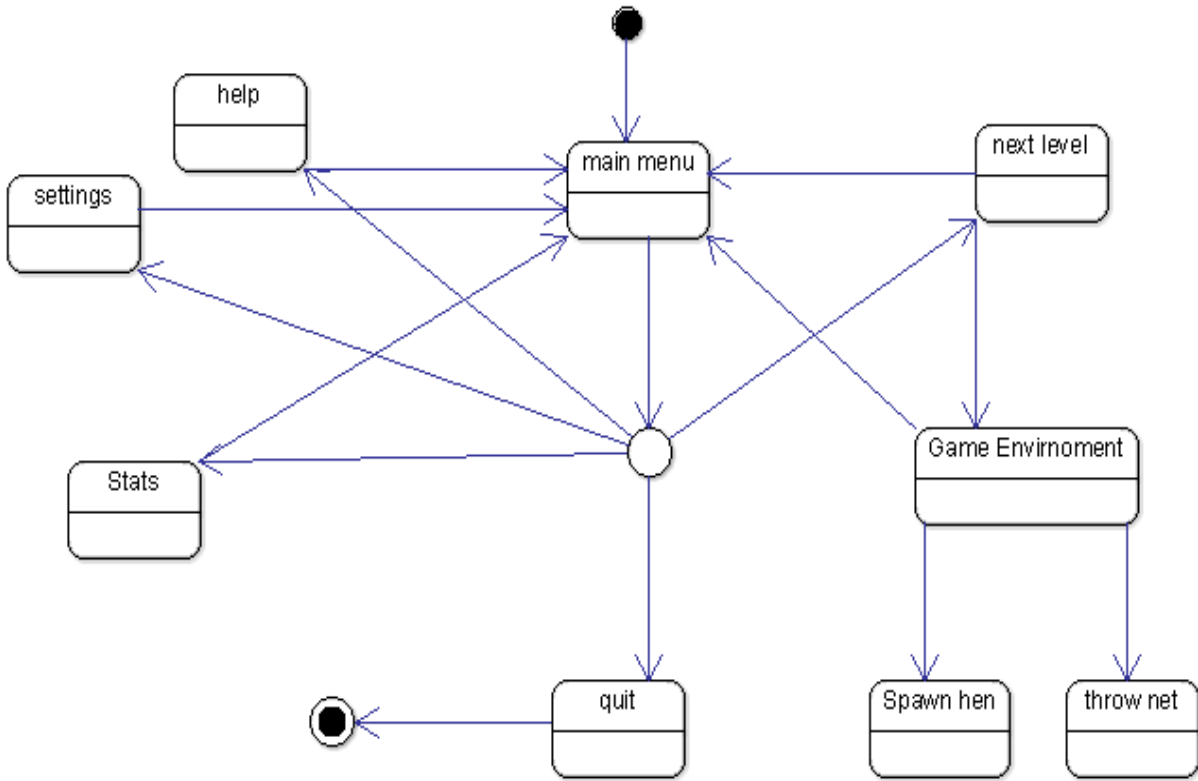
**Game execution state chart diagram:**

Figure 3.13 State diagram of catching the hen game

### 3.8 CLASS DIAGRAM

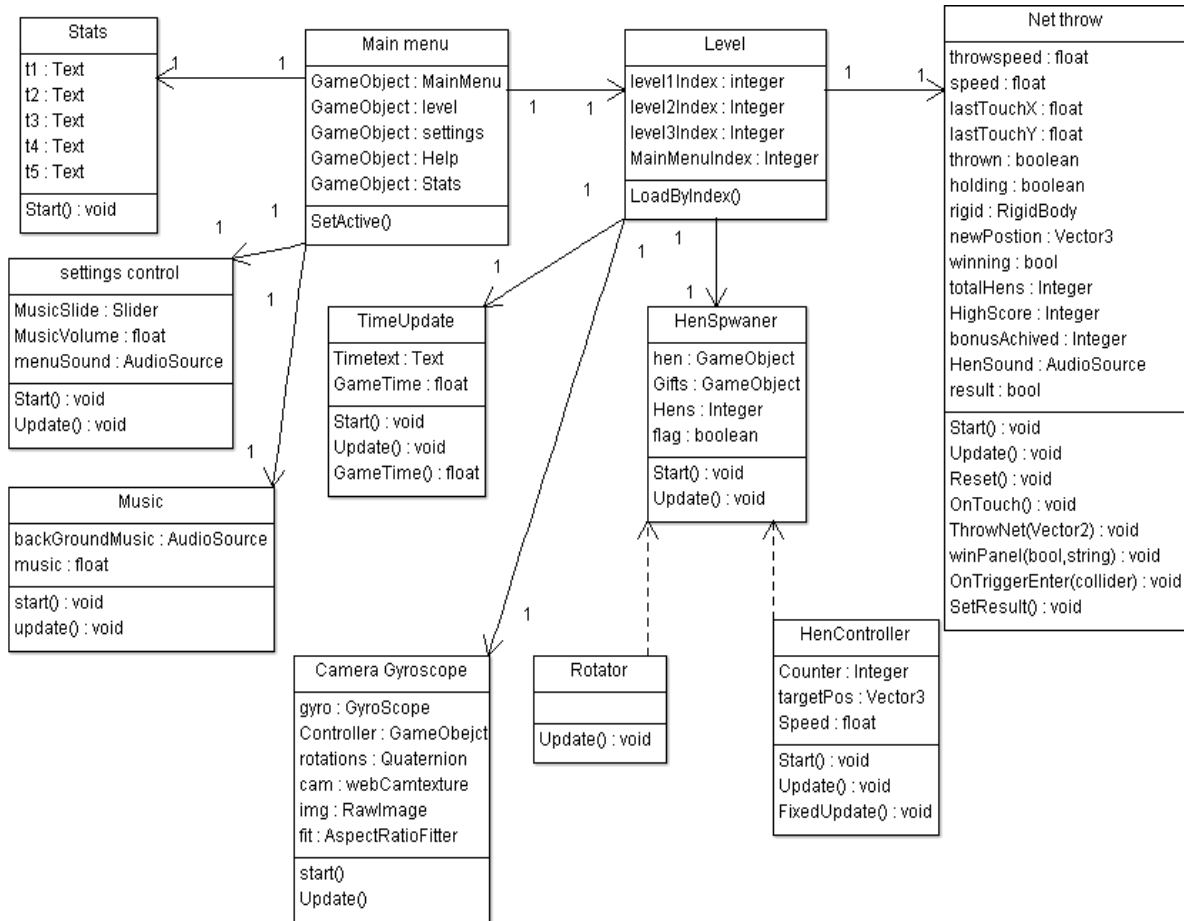


Figure 3.13 Class diagram of catching the hen game

#### 1. Main menu

This class is activated whenever the game is played for the first time the main menu class holds the reference to other game objects like level panel, settings, stats, help panel and quitting game. It holds the reference of other classes in order to activate or deactivate them whenever it also activates the game background music.

#### 2. Level

This class used to load the specific level of the game for which it takes an integer level value and load the specific level accordingly. We have implemented a function which is attached to the buttons to load new level whenever a player clicks on specific button it loads the level index associated with that game object.

### **3. Net Throw**

This class is used to control the net throw physics it is also important in way that it detects the collision of the net with the hen and shows the winning or losing panel accordingly.

### **4. Hen spawner**

The hen spawner class is used to spawn the net in the current environment of the player. It takes the hen game object as the public variable and places the hen in random positions in the current environment of the user.

### **5. Hen Controller**

The Hen controller class is used to update the animations of the hen the walking animation, running jumping. It also used to determine the new position for the hen and move hen towards to that position by doing its own animations

### **6. Time update**

This class is used to show the timer of the game during the level it takes the text game object and update the time as text on that text game object this class also provides its instance of time so the other classes like net throw uses it to determine the winning or losing of the game level.

### **7. Stats**

The stats class is used to display the game records played by the user during his whole game play. It holds the record of total hens captured by the user. The total number of nets used by the user and the best score of all three levels.

### **8. Settings control**

This class is used to control the level of music on which volume the music going to play at the background of game. For that we have used the player preferences which takes a key and stores the value of that key we can access the data part latter by submitting that key. In the settings control class we set the value of the player preference variable according to the value that is set by user by dragging slider up or down.

### **9. Music**

This class is used to play the background music of the game. It take the reference of the music as game object and play the music by checking the volume level and play the music with the same level which is selected by the player.

### **10. Rotator**

The rotator class is used to rotate the game bonus point.

### **11. Camera gyroscope**

This class is used to display the real environment images on the mobile screen using the camera we have implemented the entire camera images inside update function so that every image will update in 60 fps that will display the current world scenario like moving movie.

Chapter 4:

Software

Implementation

## 4 CHAPTER 4: SOFTWARE IMPLEMENTATION

---

### 4.1 INTRODUCTION

This document describes the project implementation of the game catching the hen.

#### 4.1.1 Language Selection

The project is implemented in

- C#

Used for the scripting purpose of the game objects

In unity it compiles two types of languages to control the game objects one is the C# and other is Java script we have choose the c# because errors were in c# are well defined by the visual studio itself and we do not need to go to the unity editor to check for the errors. The observed errors are well explained by visual studio so find the solution very easily.

#### 4.1.2 Tools selection

- Visual studio for scripting
- Unity 3d to control game Objects
- Blender to create the game models hen and the throwing net

### 4.2 APPLICATION DEVELOPMENT

In this phase of application development we will attach some algorithms about how we controlled the real camera rotations with the game camera at first then the randomly hen movements in real environment also I will explain how able to make the net throw able by swiping the hand on the mobile screen.

#### Camera controller

The camera controller code sample will basically access the current environment images from the mobile camera and place them on the unity game object Raw image as texture and updating those texture images coming from the camera inside the update function in 60 fps gives a live view of real environment in a mobile screen by this we achieve the real environment images inside our game, secondly we have to control the rotations of physical device and apply them onto the Main game camera which is used to view the augmented objects laid over in the real environment this achieved with help of hardware resource gyroscope in C# we have function Gyro.Attitued which calculates the real device rotations so we placed them with the transform rotations of our game camera. The following figure shows the code sample of the above described scenario in C#.

```

//both services are available now..
cameraController = new GameObject("Camera container");
cameraController.transform.position = transform.position;
transform.SetParent (cameraController.transform);
gyro = Input.gyro;
gyro.enabled = true;
cameraController.transform.rotation = Quaternion.Euler (90,0,0);
rotation = new Quaternion (0, 0, 1, 0);
cam.Play();
backGorund.texture = cam;
ArReady = true;
}

private void Update(){
    if (ArReady) {
        //update camera
        float ratio = (float)cam.width / (float)cam.height;
        fit.aspectRatio = ratio;
        float scaleY = cam.videoVerticallyMirrored ? -1.0f : 1.0f;
        backGorund.rectTransform.localScale = new Vector3 (1f, scaleY, 1f);
        int orient = -cam.videoRotationAngle;
        backGorund.rectTransform.localEulerAngles = new Vector3 (0, 0, orient);
        transform.localRotation = gyro.attitude * rotation;
    }
}
}

```

Figure 4.1 The code sample of the Camera controller script

### Hen Controller

The hen controller class is used in this project to control and play the animations of the hen we have used random target place near player environment in players 360 degree angle the hen will make target position and start walking and jumping around the player in his real environment the code sample for this brief explanation is as follows.

```

// Update is called once per frame
void FixedUpdate()
{
    if (counter == 50)
    {
        counter = 0;
        targetpos = new Vector3(Random.Range(-35.0f, 35.0f), -3.0f, Random.Range(-35f, 35f));
    }
    else
    {
        counter++;
    }
}

void Update()
{
    transform.LookAt(targetpos);
    transform.position += transform.forward * speed * Time.deltaTime;
}
}

```

Figure 4.2 The code sample of the hens random movements near player environment

## Throwing net

Keeping the positions of last touched finger x and y position and the new swiped positions of the screen x and y location there difference in the time to reach the point calculates the speed force at which speed the net will be thrown on to the hen and also the x and y positions of the mobile screen describes in which direction the net is will be thrown. The code sample for the above explanation in c# is given below.

```
void ThrowBall(Vector2 mousePos){
    rigid.useGravity = true;
    float differenceY = (mousePos.y - LastTouchY) / Screen.height * 100;
    speed = throwSpeed * differenceY;
    float x = (mousePos.x / Screen.width) - (LastTouchX / Screen.width);
    x = Mathf.Abs(Input.GetTouch(0).position.x - LastTouchX) / Screen.width * 100 * x;
    Vector3 direction = new Vector3(x, 0f, 1f);
    direction = Camera.main.transform.TransformDirection(direction);
    rigid.AddForce((direction * speed / 2f) + (Vector3.up * speed));
    holding = false;
    thrown = true;
    netCount++;
}
```

Figure 4.3 throwing net script code sample

## 4.3 APPLICATION CONSTRAINTS

As the application is in augmented reality so there are lots of hardware constraints and few software restrictions the game will not run on the devices below 4.0 android API versions of level 19.

Gyroscope must be available in the target device to run this game.



#### 4.4 APPLICATION SCREENSHOTS

The following images are the actual screenshots of the running implemented game in an android device.

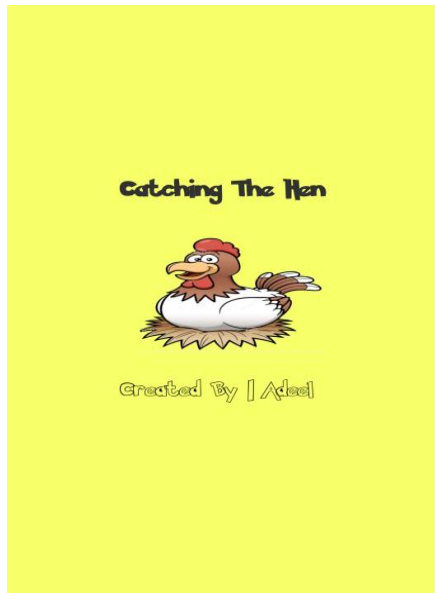


Figure 4.4 welcome screen



Figure 4.5 Menu Screen



Figure 4.6 Settings screen



Figure 4.7 Help Screen

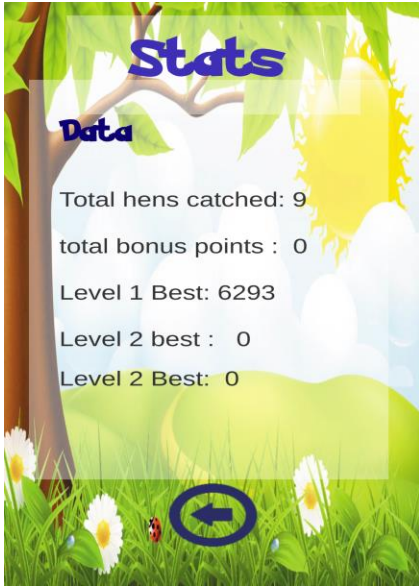


Figure 4.8 Stats Screen



Figure 4.9 Level Screen

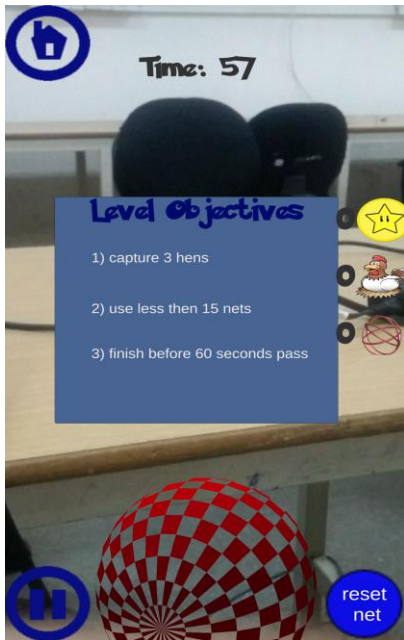


Figure 4.10 Level Objectives panel

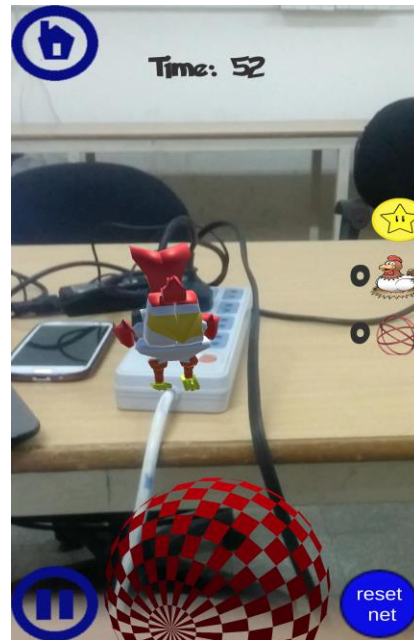


Figure 4.11 Game environment Screen

# Chapter 5: Software test

## 5 CHAPTER 5: SOFTWARE TEST

---

This section of the document specifying the test approach which i use, testing tools and environment and the test cases.

### 5.1 TEST APPROACH

A test approach is the test strategy implementation of a project, defines how testing would be carried out. Test approach has two techniques:

**Proactive:** An approach in which the test design process is initiated as early as possible in order to find and fix the defects before the build is created.

**Reactive:** An approach in which the testing is not started until after design and coding are completed.

We use the reactive approach because by this approach we can analyses the field or tool expert's knowledge extremity. We can analyses various risks linked to the project. Consider people, environment, and the company. Understand your project's nature and the business setup.

### 5.2 TEST PLAN

#### 5.2.1 Features to be tested

The following features of the system will be tested which are necessary for the completion of the game catching the hen.

1. Game object hen operations
2. Net for the players operations
3. Checking of the gyroscope operation which is must for the real environment
4. Getting the GPS detail of the player
5. Playing game music

#### 5.2.2 Features not to be tested

The following features in our project are not that much necessary to check but the working is would increase the appealing behavior of the game.

1. Changes in settings
2. Viewing help menu
3. Viewing game stats

### 5.2.3 Testing Tools and Environment

The testing environment which i will use for the project is black box testing in which the internal structure design implementation of the item being tested is not known to the tester. These tests can be functional or non-functional, mostly they are functional. This method is named black box because the software program in the eyes of the tester is like a black box inside which one cannot see what the details of the implementation and structure design is. This method attempts to find errors in the following categories

1. Incorrect or missing functions
2. Interface errors in data structures
3. Behaviour or performance errors.
4. Initialization and termination errors.

## 5.3 TEST CASES

The following test cases are very necessary to be complete for the completion of the project

1. Select Level
2. Game environment
3. Creating game agent hen
4. Creating the net object
5. Calculating gyroscope rotations
6. Playing game music

### 5.3.1 Select Level

ID	Test1
Description	User can go to game environment by selecting the play level from the level scene.
Tester	Player
Setup	Already implemented the game environment logic at the back end player needs to select the option of the level game from the level menu.
Instructions:	<ol style="list-style-type: none"> <li>1. launch game from your androids home screen</li> <li>2. Select the play level option from the main menu</li> <li>3. select level</li> </ol>
Expected Results	Player will successfully lead into the game environment.
Observed output	Player successfully see the real environment in his game screen
Verdict	Pass

### 5.3.2 Game environment

ID	Test2
Description	User can view the game environment.
Tester	Player
Setup	Already implemented the game environment logic at the back end player needs to select the option of the play game from the level menu.
Instructions:	<ol style="list-style-type: none"> <li>1. launch game from your androids home screen</li> <li>2. Select the play level option from the main menu</li> <li>3. select level difficulty from the level panel</li> <li>4. press play game button</li> <li>5. Player will see the real game environment of his surroundings is created inside of his mobile screen.</li> </ol>
Expected Results	Player will successfully see the real game environment as the background of his game screen.
Observed output	Player successfully see the real environment in his game screen
Verdict	Pass

### 5.3.3 Create game agent hen

ID	Test3
Description	User can go to game environment by selecting the play game from the level scene.
Tester	Player
Setup	Already implemented the game environment logic at the back end player needs to select the option of the play game from the level menu.
Instructions:	<ol style="list-style-type: none"> <li>1. launch game from your androids home screen</li> <li>2. Select the play level option from the main menu</li> <li>3. select level difficulty from the level panel</li> <li>4. press play game button</li> <li>5. Player needs to rotate camera in direction to see the hen object.</li> </ol>
Expected Results	Player will successfully see the real game environment as the background of his game screen. And also his target game agent will be also created somewhere near him.
Observed output	Player successfully see the environment and the game object hen walking in his surroundings
Verdict	Pass

### 5.3.4 Create game object net

ID	Test4
Description	User can throw the net in order to capture the hen
Tester	Player
Setup	Already implemented the game environment logic at the back end player needs to select the option of the play game from the level menu.
Instructions:	<ol style="list-style-type: none"> <li>1. launch game from your androids home screen</li> <li>2. Select the play level option from the main menu</li> <li>3. select level difficulty from the level panel</li> <li>4. press play game button</li> <li>5. player throws the net onto the hen</li> </ol>
Expected Results	Player can throw net in all directions in order to capture the hen
Observed output	Player throws the net where ever he want to capture the hen
Verdict	Pass

### 5.3.5 Calculating gyroscope rotations

ID	Test5
Description	User can go to game environment by selecting the play game from the level scene. The rotations of the gyroscope will update the real environment in real time of the game.
Tester	Player
Setup	Already implemented the game environment logic at the back end player needs to select the option of the play game from the level menu.
Instructions:	<ol style="list-style-type: none"> <li>1. launch game from your androids home screen</li> <li>2. Select the play level option from the main menu</li> <li>3. select level difficulty from the level panel</li> <li>4. press play game button</li> </ol>
Expected Results	Player will successfully see the real game environment as the background of his game screen. And the gyroscope will update the real environment with its rotations count.
Observed output	Player can rotate in 360 degrees in game environment by physically rotating the camera
Verdict	Pass

### 5.3.6 Play game music

ID	Test6
Description	User can go to game environment by selecting the play game from the level scene. And the game music will enhance the appealing nature of the game
Tester	Player
Setup	Already implemented the game environment logic at the back end player needs to select the option of the play game from the level menu.
Instructions:	<ol style="list-style-type: none"> <li>1. launch game from your androids home screen</li> <li>2. Select the play level option from the main menu</li> <li>3. select level difficulty from the level panel</li> <li>4. press play game button</li> </ol>
Expected Results	Player will successfully see the real game environment as the background of his game screen. Back ground game music will be played.
Observed output	Player successfully see the environment and game environment music played
Verdict	Pass

## 5.4 HARDWARE TESTING ON DIFFERENT DEVICES

We have tested the game on different devices we created the APK file of the game and install the application on different devices and tests all the functions either are working properly or not we also consider the different sizes of screen and tested how the game behaved. There are few list of devices on which the game has been tested and performs all the functionalities well

1. LG g3 (supports Gyroscope performs all functionalities properly)
2. Lenovo A 7000 (supports Gyroscope performs all functionalities properly)
3. LG g2 (supports Gyroscope performs all functionalities properly)
4. Samsung s3 (supports Gyroscope performs all functionalities properly)
5. Samsung A5 2017 (supports Gyroscope performs all functionalities properly)
6. Haier I 6 (does not supports Gyroscope game will not lead to main menu instead will say system not supports gyroscope)



## 6 CONCLUSION

---

The augmented reality apps require more hardware resources to be more appealing. Our target is to achieve the concept of augmented reality in small devices there for we have designed and implemented this game for less hardware supported mobiles and can be run on more devices then the typical android augmented reality apps on basis of which we can conclude few features from this game.

### The achieved features of our project

1. Consumes less power while using battery because of not using many hardware resources like fish eye cameras, inertial measurement unit (IMU) and also did not put device into more processing of images that perceive with the 60 frames per second and calculating there corners and edges at that rate.
2. While talking about less hardware resources means we can run the game on more devices our game uses only gyroscope and simple camera of the android device to display the real world and look into the computer generated animated hen character in current live environment.
3. We have provided attractive level objectives for three levels in our game which users love to achieve and complete the game.
4. Added a player interactive object a game net whose input is given by the touch and sliding the finger on a mobile screen.

### What we cannot achieve using a simple android phone

1. Area learning which needs 180 degree fish eye camera to view the larger viewing angle to learn maximum number of area and place the colliders on them. Secondly the learning algorithm will take lots of processing efforts which we cannot implement on phones with 1 GB RAM to 2 GB of RAM because it needs to learn every single frame image and when the frame rate is 60 fps in a game, it will have to take 60 textured images and calculate their corners simultaneously.
2. Motion tracking with simple device is not achievable this can be detected using the GPS but the distance between the two points of GPS is much more larger therefore it is not possible to create the smooth movements of our animated hen this can be achieved if our character is static in one fixed place.
3. The following is the image reference of new tango android device “the tango devices are new android supported phones designed especially for the purpose to achieve better results in augmented reality applications these devices are populated with the required hardware’s which are necessary for an appealing augmented reality application ”.

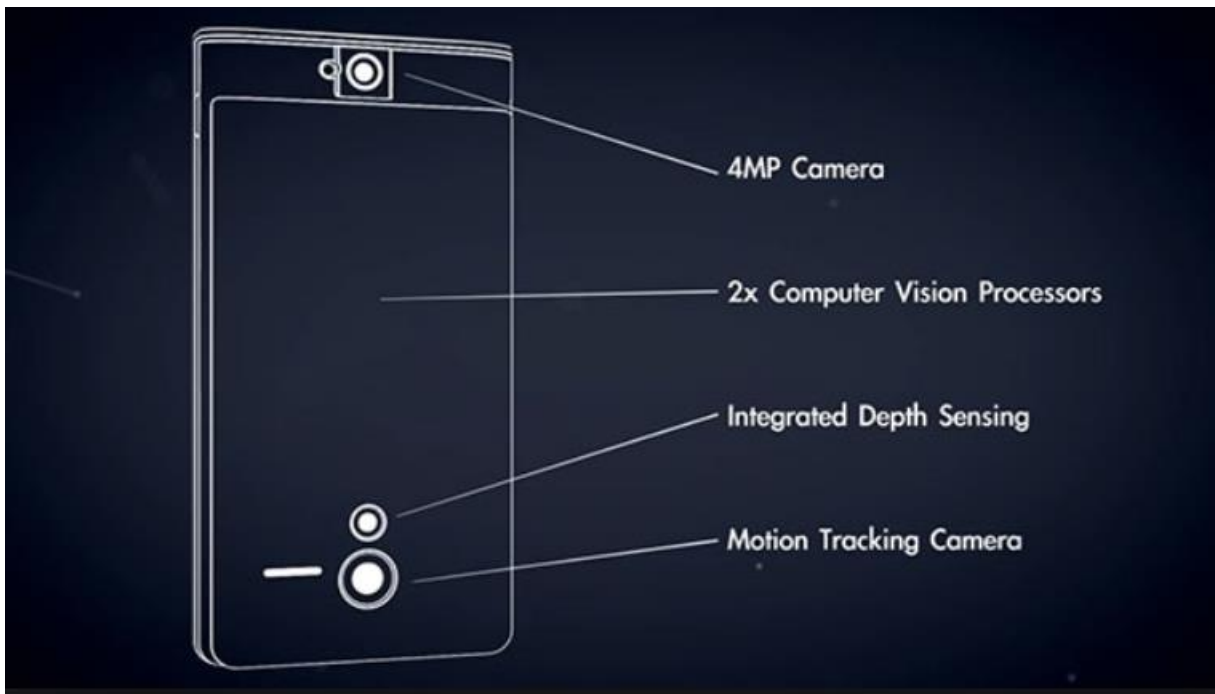


Figure 6.1 Tango Device Hardware details

### Future Enhancements

I have described earlier that with the passage of time there are lots of new devices in market which presents augmented reality in much more detailed way. Our future enhancement of this project will be to made this game for such devices like ODG glasses and the Holo lens of Microsoft and the new Tango android devices by Google supporting all the required hardware resources for augmented reality applications. In these devices we will get features like the following.

1. We can touch the virtual characters that are augmented by the system.
2. We can see the generated object from all directions.
3. We can be with the computer and real world at the same time.

### Future functionalities in my game to achieve are

1. We will implement this game for Holo lens and ODG glasses in which we can touch the hen by hand gestures in real world and throw it on to the hens.
2. With the hen in Holo we can run with it by using the motion tracking and hit the by going near to it.

## 7 APPENDIX'S A GLOSSARY

---

- **Player:** player is the one who is going to play the game he can be any one child, young or old this game will be appealing for all of them as it attracts the people because of the real environment nature.
- **Game object hen:** game agent hen is basically behaving as the game agent. Which want to get away from the players screen in order to fail the user before completing his or her level. These game objects are associated with their own animations so whenever player comes to the state where an animation is going to change its state it can be played e.g. the game agent hen has four different animations in this game the agent can play its idle state animation.
- **Game object net:** game object net is the available net for the player which will be provided for the player in order to capture the game agent hen. These game objects are associated with their own animations so whenever player comes to the state where an animation is going to change its state it can be played e.g. the game object net has four animations initial net frame animation, throw net frame animation, missed net frame animation and at last the captured net frame animation.
- **Augmented environment:** Augmented reality (AR) is a live direct or indirect view of a physical, real-world environment whose elements are augmented (or supplemented) by computer-generated sensory input such as sound, video, graphics or GPS data.
- **Main menu:** That provides or platform for the player to go through all the game features.
- **Settings:** Player can change the game music or hen sounds whenever he likes to by go into the settings option.

## 8 APPENDIX'S B PROJECT PROPOSAL

---

**Working title:** Catching the hen augmented reality game.

**Platform:** Android 4.0 jelly bean or higher.

**Motivations:** This game is designed for children who love to catch the hens as I have mentioned in the above overview our product will create the animated hen in the real environment so children will enjoy the existence of real world scenario, They will definitely love to catch the animated hen created in the environment in there surrounding, children interact with the hen game object by rotating physically in their current environment.

**Problem statement:** the problem of statement of the project says.

- Create augmented reality android based catching the hen game.

## 9 REFERENCES

---

- [1] Pratap Mohapatra, Software Engineering A Lifecycle Approach, New Age Publications, 2010
- [2] Roger S. Pressman, Software Engineering - A Practitioner's Approach, McGraw Hill, 7th Edition, 2010
- [3] Ian Somerville, Software Engineering, Addison Wesley, Ninth Edition, 2010
- [4] Applying UML Advanced Application, Rob pooley, pauline wilcox.
- [5] Relevant IEEE standards: IEEE-1058 [8], IEEE-1540 [9]
- [6] Relevant IEEE standards: IEEE-830 [4]
- [7] Relevant IEEE standards: IEEE-1016 [7]
- [9] Article about making the definition of sequence diagram  
[https://en.wikipedia.org/wiki/Sequence\\_diagram](https://en.wikipedia.org/wiki/Sequence_diagram)
- [10] Book of augmented reality in games and education  
<http://augmented-reality-in-education.wikispaces.com/home>
- [11] Definition of Augmented reality on Wikipedia  
[https://en.wikipedia.org/wiki/Augmented\\_reality](https://en.wikipedia.org/wiki/Augmented_reality)