

A
Major project
On
Designing an Intellective Gaming Using Unity User Interface
(Submitted in partial fulfillment of requirements for reward of Degree)
BACHELOR OF TECHNOLOGY
In
COMPUTER SCIENCE AND ENGINEERING
BY
Shaik Saifuddin (177R1A05H4)
M Nishanth Reddy (177R1A05F6)
T Abhilash(177R1A05N3)
Under the Guidance of
G Latha
(Associate professor)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
CMR TECHNICAL CAMPUS
UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi) Recognized Under Section 2(f) & 12(B) of the UGC Act.1956,
Kandlakoya (V), Medchal Road, Hyderabad-501401.

2017-2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



CERTIFICATE

This is to certify that the project entitled "**Designing an Intellective Gaming Using Unity User Interface**" being submitted by **SK.SAIFUDDIN(177R1A05H40), M.NISHANTHREDDY(177R1A05F6),T.ABHILASH(177R1A05N3)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering of the Jawaharlal Nehru Technological University Hyderabad, during the year 2021-2022. It is certified that they have completed the project satisfactorily.

INTERNAL GUIDE

G LATHA

(Associate professor)

DIRECTOR

Dr.A.Raji Reddy

HOD

Dr.K.Srujan Raju

EXTERNAL EXAMINER

Submitted for viva voce Examination held on _____

ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **Mrs. G.Latha** Associate Professor for her exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by her shall carry us a long way in the journey of life on which we are about to embark. We also take this opportunity to express a deep sense of gratitude to Project Review Committee (PRC) **Mr. J. Narasimha Rao, Dr. T. S. Mastan Rao, Mr. A. Uday Kiran, Mr. A. Kiran Kumar, Mrs. G. Latha** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. K. Srujan Raju**, Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy**, Director for being cooperative throughout the course of this project. We also express our sincere gratitude to Sri. **Ch. Gopal Reddy**, Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

SHAIK SAIFUDDIN(177R1A05H4)

M NISHANTHREDDY(177R1A05F6)

T ABHILASH(177R1A05N3)

ABSTRACT

The word cognition comes from the Latin root *cognoscere*, which means “to know”. By cognition, we are usually referring to everything that is related to knowledge. In other words, the accumulation of information that we have acquired through learning or experience.

The most accepted definition is the ability to process information through perception (stimuli that we receive through our different senses), knowledge acquired through experience, and our personal characteristics that allow us to integrate all of this information to evaluate and interpret our world. It is the ability that we have to assimilate and process the information that we receive from different sources (perception, experience, beliefs, etc.) and convert them into knowledge. It includes different cognitive processes, like learning, attention, memory, language, reasoning, decision making, etc., which are part of our intellectual development and experience.

This project focuses on three core cognitive abilities. We used unity game engine to create a game to improve cognitive skills of the players without any special immersive techniques.

Unity is a game engine created by unity technologies. It is ranked in the top three game engines of the decade. Unity is popular for 2d graphics. We used unity’s 2d system to create this project.

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
Figure 3.1	Project Architecture	21
Figure 3.2	Use case Diagram	23
Figure 3.3	Class Diagram	24
Figure 3.4	Sequence Diagram	25
Figure 3.5	Activity Diagram	26

LIST OF SCREENSHOTS

SCREENSHOTS NO.	SCREENSHOT NAME	PAGE NO.
Screenshots 5.1	Home Screen initial	35
Screenshots 5.2	Main home Scene	35
Screenshots 5.3	ReflexioneOne initial	36
Screenshots 5.4	ReflexioneOne main	36
Screenshots 5.5	Reflexione Two initial	37
Screenshots 5.6	Reflexione Two Main	37
Screenshots 5.7	Words Wizard Initial	38
Screenshots 5.8	Words Wizard Main	38

TABLE OF CONTENTS

	PAGE NO'S
ABSTRACT	i
LIST OF FIGURES	ii
LIST OF SCREENS	iii
1. INTRODUCTION	1
1.1. PROJECT SCOPE	2
1.2. PROJECT PURPOSE	2
1.3. PROJECT FEATURES	2
2. SYSTEM ANALYSIS	3
2.1. PROBLEM STATEMENT	4
2.2. EXISTING SYSTEM	4
2.2.1. LIMITATIONS	6
2.3. PROPOSED SYSTEM	6
2.3.1. ADVANTAGES OF PROPOSED SYSTEM	7
2.4. FEASIBILITY STUDY	9
2.4.1. ECONOMIC FEASIBILITY	10
2.4.2. TECHNICAL FEASIBILITY	10
2.4.3. BEHAVIOURAL FEASIBILITY	10
2.5. HARDWARE AND SOFTWARE REQUIREMENTS	1
2.5.1. HARDWARE REQUIREMENTS	11
2.5.2. SOFTWARE REQUIREMENTS	11
3. ARCHITECTURE	12
3.1. PROJECT ARCHITECTURE	13
3.2. DESCRIPTION	13
3.3. USE CASE DIAGRAM	15
3.4. CLASS DIAGRAM	16
3.5. SEQUENCE DIAGRAM	17
3.6. ACTIVITY DIAGRAM	18
4. IMPLEMENTATION	19

4.1.	SAMPLE CODE	20
5.	SCREENSHOTS	32
6.	TESTING	36
7.	CONCLUSION	40
8.	BIBLIOGRAPHY	41
8.1.	GITHUB REPOSITORY LINK	42
8.2.	REFERENCES	42
8.3.	WEBSITES	42

INTRODUCTION

1. INTRODUCTION

1.1. PROJECT SCOPE

This project is titled as “Cognitive game using unity”. This application provides interactive entertainment based cognitive ability training to the users. This project is developed using the Unity game engine. There are 3 different types of brain areas we focus on, they are memory, reflex and language.

1.2. PROJECT PURPOSE

This application(interactive entertainment application) has been developed to improve the brain functionality of the users by playing cognitive games. Developed in unity, a top game engine for 2d graphics.

1.3. PROJECT FEATURES

Cognition is the ability to process information through perception (stimuli that we receive through our different senses), knowledge acquired through experience, and our personal characteristics that allow us to integrate all of this information to evaluate and interpret our world. In the Interactive application we focused on and developed Interactive application for three core cognitive abilities

Memory improvement is part of this memory game, the process can be done with a few steps that can be easy and performed effortlessly.

Brain reflection improves the reaction time of the user.

Language : A language-game (German: *Sprachspiel*) is a philosophical concept developed by Ludwig Wittgenstein, referring to simple examples of language use and the actions into which the language is woven.

SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

SYSTEM ANALYSIS

System Analysis is the important phase in the system development process. The System is studied to the minute details and analysed. The system analyst plays an important role of an interrogator and dwells deep into the working of the present system. In analysis, a detailed study of these operations performed by the system and their relationships within and outside the system is done. A key question considered here is, "what must be done to solve the problem?" The system is viewed as a whole and the inputs to the system are identified. Once analysis is completed the analyst has a firm understanding of what is to be done.

2.1. PROBLEM STATEMENT

The gaming industry is one of the branches of the entertainment industry, which generates a lot of revenue. Games use some techniques to make players feel more immersive, these techniques are also being used in cognitive games lately. For example by using music developers make players more manipulatable and feel immersive. But the use of these kinds of techniques in cognitive games makes the whole cognition's purpose different. Cognitive games are meant to evolve brain activities. So we developed a game using unity engine without any special immersive techniques to create a true cognitive Interaction.

2.2. EXISTING SYSTEM

There are many cognitive games in the play store and online. Some of them focus on all around cognitive skill sets. While others focus on some particular skills that are required for some selected audience(players). Some of the services focusing on them are:

LUMINOSITY

Lumosity is a leading brain training program, brought to you by Lumos Labs, Inc. Lumosity's web and mobile games are designed by scientists to challenge core cognitive abilities.

Lumosity's scientists take neuropsychological and cognitive research tasks, or design new ones, and work with game designers to transform them into 50+ cognitive games. Designed to be both fun and adaptively challenging, Lumosity's training program is accessible to people of all ages — helping them stay challenged to the full extent of their abilities.

Lumos Labs also has a collaborative research initiative, the Human Cognition Project, which currently partners with over 90 university collaborators. Through the HCP, we grant qualified researchers free access to Lumosity's cognitive training tasks, assessments, research tools, and, in some cases, limited access to data on cognitive task performance — helping them conduct larger, faster, and more efficient studies. For more information, please visit lumosity.com/hcp

ELEVATE

Elevate Labs is on a mission to improve people's minds. Since 2014, they've helped over 40 million people improve their communication and math skills with Elevate, which won Apple's App of Year award. With Balance, they've created an innovative, personalized meditation experience that helps people improve their stress, sleep, and more.

They're a fully distributed company of voracious learners and passionate doers, driven by our vision to help billions of people lead healthy, joyful, and productive lives.

BRAINGLE

Braingle is a place to solve puzzles, brush up on your trivia, play games and give your brain a workout. Get ready to have your brain tangled.

Braingle at a glance have brain teasers, optical illusion, puzzle experiences, codes & ciphers, puzzlepedia...and many other different types of games.

2.2.1. LIMITATIONS

- We know video games can generate addiction issues. With cognitive games the chances of addiction are more than any normal games.
- The excessive use of video games can cause rejection towards other didactic means such as books, CD's, etc.
- It can cause violent behaviour if the resource used is based on this type of behaviour.
- It can be expensive for the educational center, not only the acquisition of video games, but also the equipment necessary to execute them.
- Video games often act as a replacement for real-time human connections. People who find themselves playing hours upon hours of video games may find themselves losing touch with the relationships they've built with other people. People argue that with internet-connected games, they can easily hang out with friends and families without ever leaving home, but this type of virtual get together is no replacement for actual face-to-face interaction.
- New studies show that the risk of being overweight increases with every hour people spend on virtual play. Lack of motion and overplaying lead to muscle pain as well. Moreover, the addiction to these games leads to sleep deprivation which contributes to obesity

2.3. PROPOSED SYSTEM

We divide our games and activities into 3 critical brain areas: Memory, attention, language. We keep track of your progress and the games are based on scientific research.

Memory improvement is part of this memory game, the process can be done with a few steps that can be easy and performed effortlessly. Many people have trouble remembering faces or names. How to remember things is only a technique that you need to utilize, for example to **remember a face** you just need to examine a person's face discretely when you are introduced.

Try to find an unusual feature, ears, hairline, forehead, eyebrows, eyes, nose, mouth, chin, complexion, etc . This memory game is going to activate some areas of your brain responsible for memory acquisition which therefore can help your **memory improve**. Enjoy it and try to finish before 1:30 min.

Brain reflection This is a simple exercise to test the reflection of your brain, the faster the better, try to get a score of 0.20 in this reflex test. When you manage to reach that score then that means you're very fast, if you can't achieve that score then keep practicing. Practice makes perfect. Human reflection is the capacity of humans to exercise a willingness to learn more about our fundamental nature, purpose and essence. Brain reflection invariably leads to inquiry into the human condition and the essence of humankind as a whole. Humans often consider themselves to be the dominant species on Earth, and the most advanced in intelligence and ability to manage their environment, thanks to the mind reflection.

Language : A language-game (German: *Sprachspiel*) is a philosophical concept developed by Ludwig Wittgenstein, referring to simple examples of language use and the actions into which the language is woven. Wittgenstein argued that a word or even a sentence has meaning only as a result of the "rule" of the "game" being played. Depending on the context, for example, the utterance "Water!" could be an order, the answer to a question, or some other form of communication.

2.3.1. ADVANTAGES

Improves critical thinking skills :

Sharpening a child's thinking skills means that he or she will have the mental aptitude to process information, make decisions, create new ideas, ask questions, try to make sense of things, organize information and so forth. Memory games are also good for visual memory aid. A child's ability to store and retrieve memories, when the stimuli that first evoked them no longer exists, and without help is a critical aspect of reading, writing, spelling, including developing mathematical skills. Kid's whose visual memory hasn't fully developed can be trained with various interactive memory games.

Exercise for the brain:

Fun android and iPhone games are like having a personal trainer for a child's mind that is developing at a rapid rate. They can help strengthen individual skills like communication, good eye contact, social skills, reaction time, pattern recognition, and much more. Cognitive skill development is critical as a child matures, because he or she is able to not only learn new information, but to also remember, think more critically, and process what they learn and apply it in a progressive manner. But since all kids are different, some may fail to hit age-appropriate cognitive skill milestones, and this is where memory building activities come in handy because a child needs a stimulating environment that can help develop their brain, boost their confidence, focus and attention.

Playing online games can make learning easier, however, there are specific fun online games for adults and kids where parents can play with their children to promote active learning and encourage their cognitive development in the areas of attention, memory, concentration, visualization, and perception.

Sharpen problem-solving skills:

Memory matching games that involve strategy help sharpen a child's problem-solving skills, memory, analytical thinking skills, communication, general knowledge, concentration, social skills, even patience. Imagine you as an adult trying to solve the Rubik's cube or a Tangram puzzle, or assemble flat-pack furniture in a matter of minutes without losing your top? With the right memory game, children can significantly improve their problem-solving skills, consequently enhancing their brain power, concentration, communication, and social skills, even strengthening their numeric skills, as well as visual skills.

Improve spatial skills:

Excellent spatial reasoning skills in children are directly linked to great mathematical skills; in fact, such kids are more likely to pursue careers in fields such as engineering and architectural designs. But the good news is that even those without these skills can gain them through playing structured online games that involve a whole lot of rotating, moving, and positioning pieces.

Enhance visual perception:

Visual acuity plays a vital role in a child's ability to identify, understand, assess, and interpret one set of different objects from others including symbols, letters, colors, forms, shapes, patterns, size, etc. Memorization games can help improve visual perceptual skills in a child, to a point where they are able to mentally visualize outcomes with little to no trial and error. Such games can also help improve language and vocabulary.

Even where English is the first language, some kids do struggle with mastering it, and while strong vocabulary skills can be developed and polished through consistent reading and writing, children often get bored with books. There are fantastic interactive memory games that can help a child improve both their language and word association, and unlike homework sheets, fun games can keep a child focused for extended periods, and have tons of fun while at it.

2.4. FEASIBILITY STUDY

The feasibility of the project is analysed in this phase and the business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. Three key considerations involved in the feasibility analysis are

- Economic Feasibility
- Technical Feasibility
- Social Feasibility

2.4.1. ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on a project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also, all the resources are already available, it gives an indication that the system is economically possible for development.

2.4.2. TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

2.4.3. BEHAVIOURAL FEASIBILITY

This includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm? The project would be beneficial because it satisfies the objectives when developed and installed. All behavioural aspects are considered carefully and conclude that the project is behaviourally feasible.

2.5. HARDWARE AND SOFTWARE REQUIREMENTS

2.5.1. HARDWARE REQUIREMENTS

Hardware interfaces specifies the logical characteristics of each interface between the software product and the hardware components of the system, the following are some hardware requirements

- | | |
|-------------|-----------------------------|
| ● Processor | Intel i5 core @ CPU 2.4GHz |
| ● RAM | 8GB or More |
| ● Storage | 25GB or More |
| ● GPU | Nvidia geforce 970 or Above |

2.5.2. SOFTWARE REQUIREMENTS

Software requirements specifies the legal characteristics of each interface and software components of the system. The following are some software requirements

- | | |
|--------------------|------------------------------|
| ● Operating System | Windows 10 |
| ● Languages | C# |
| ● Engine | Unity |
| ● IDE | Visual studio community 2019 |
| ● Browser | Google Chrome, Edge |

ARCHITECTURE

3. ARCHITECTURE

3.1. SYSTEM ARCHITECTURE

This project architecture shows the procedure followed in the Unity Engine to make this IE Application

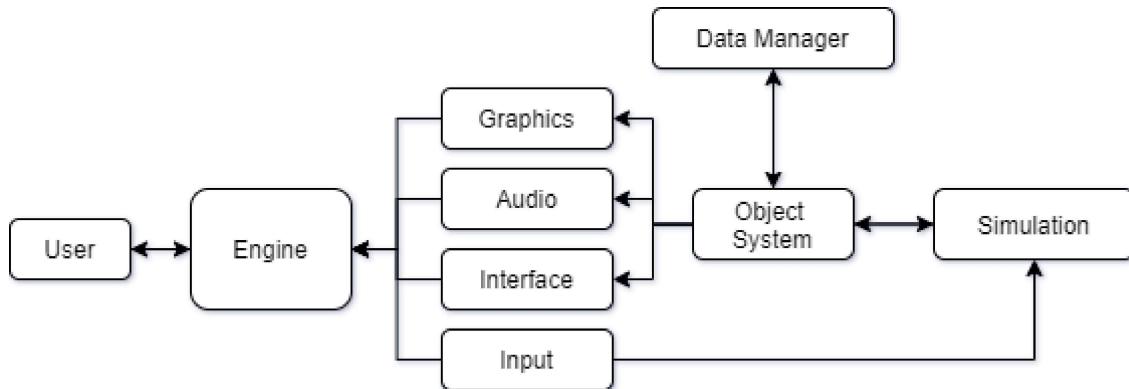


Figure 3.1 System architecture

3.2. DESCRIPTION

Engine : The game engine used in this project is Unity 5 v2019.04 . Unity is a cross-platform game engine developed by Unity Technologies, first announced and released in June 2005 at Apple Inc.'s Worldwide Developers Conference as a Mac OS X-exclusive game engine. The engine has since been gradually extended to support a variety of desktop, mobile, console and virtual reality platforms.

Graphics: A variety of computer graphic techniques have been used to display video game content throughout the history of video games. The predominance of individual techniques have evolved over time, primarily due to hardware advances and restrictions such as the processing power of central or graphics processing units.

Audio: In video games, music can be streamed, where the audio is pre-recorded and played back when required. While early video games were restricted to sequenced music, streaming music has become a more viable option as technology has improved.

Interface: a user interface (UI) is the space where interactions between humans and machines occur. The goal of this interaction is to allow effective operation and control of the machine from the human end, whilst the machine simultaneously feeds back information that aids the operators' decision-making process.

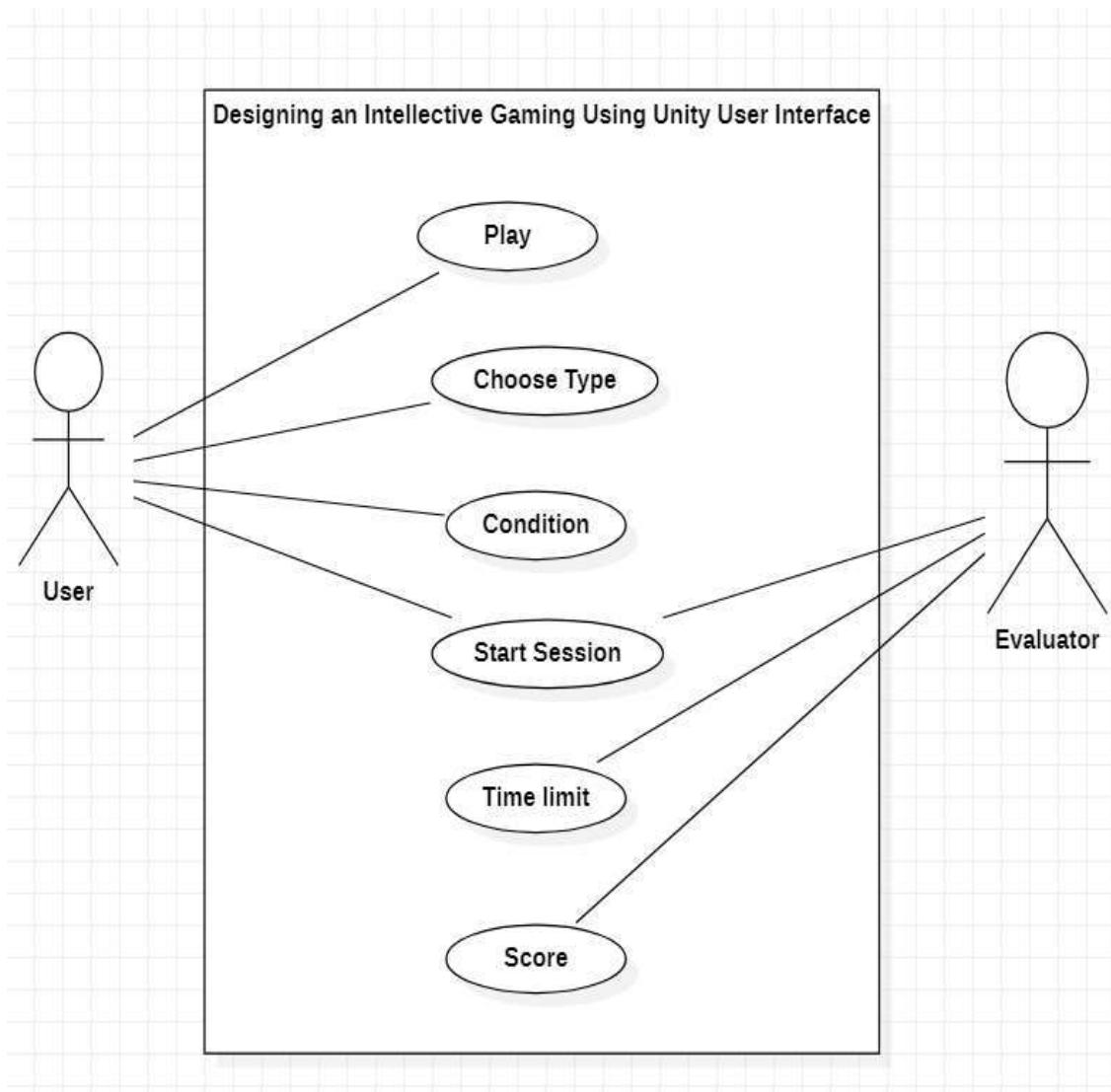
Input: Pointer events and buttons are created on the Input window, and end users can configure Keyboard input in a nice screen configuration dialog.

Data Manager: comprises all disciplines related to managing data as a valuable resource.

Object System: Unity engine's core object system and C#'s object system is used in the project

Simulation: A simulation is the imitation of the operation of a real-world process or system over time.^[1] Simulations require the use of models; the model represents the key characteristics or behaviors of the selected system or process, whereas the simulation represents the evolution of the model over time. Often, computers are used to execute the simulation.

3.3. USE CASE DIAGRAM



3.4. CLASS DIAGRAM

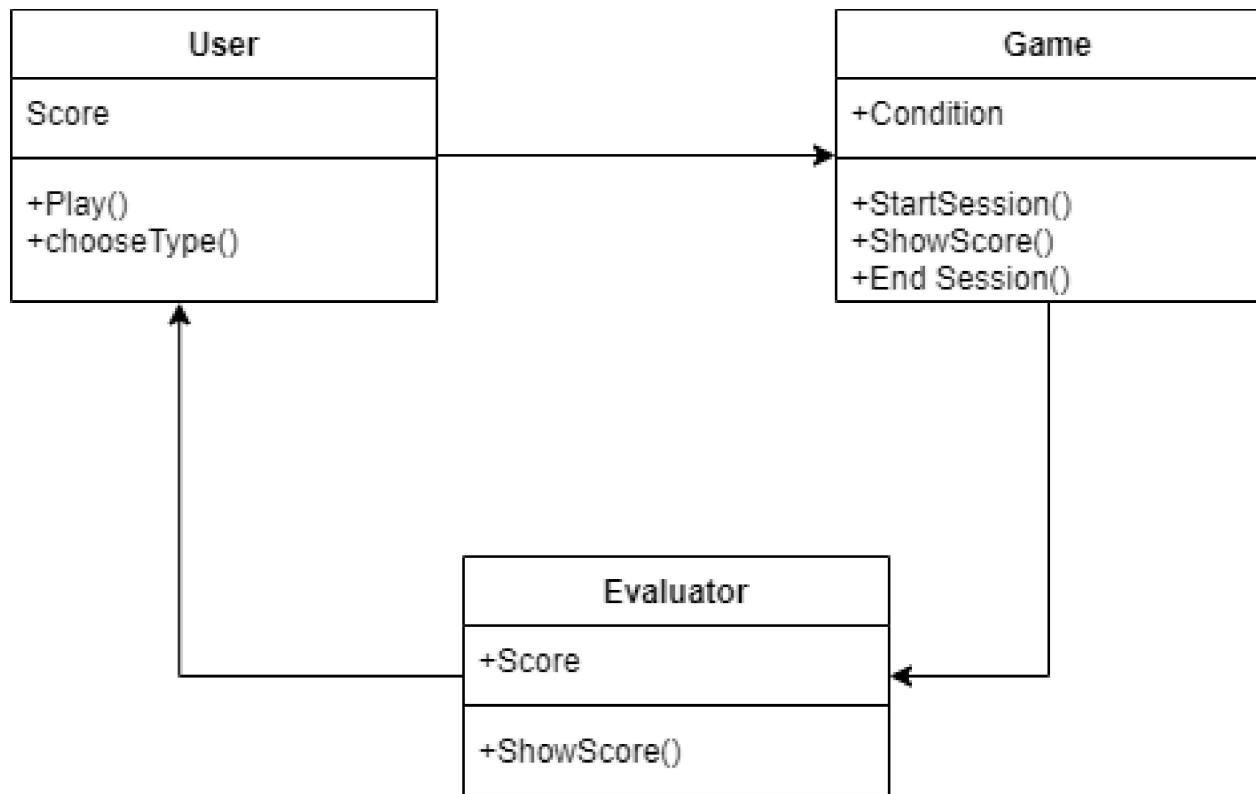


Figure 3.4 Class Diagram

3.5. SEQUENCE DIAGRAM

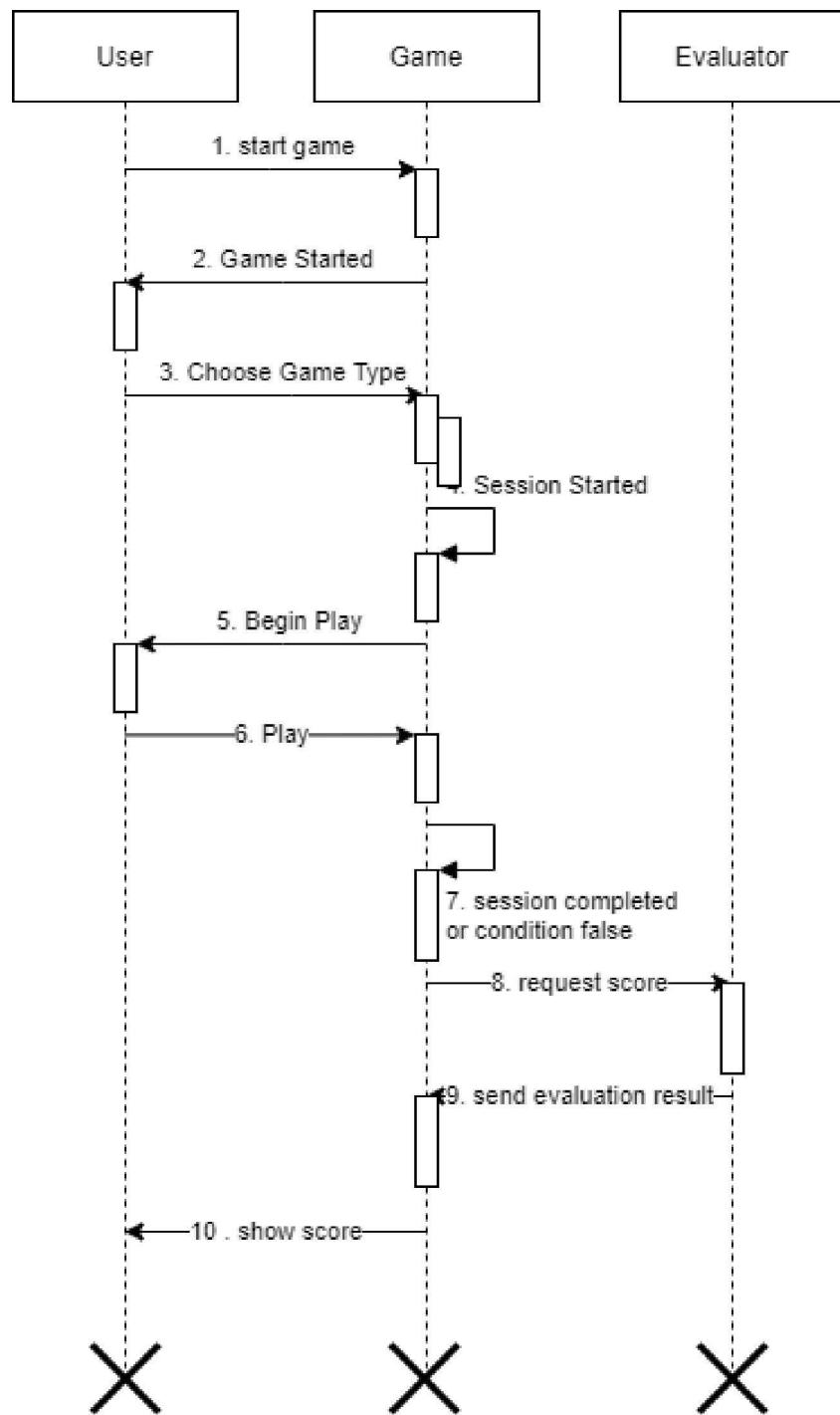


Figure 3.5 Sequence Diagram

3.6. ACTIVITY DIAGRAM

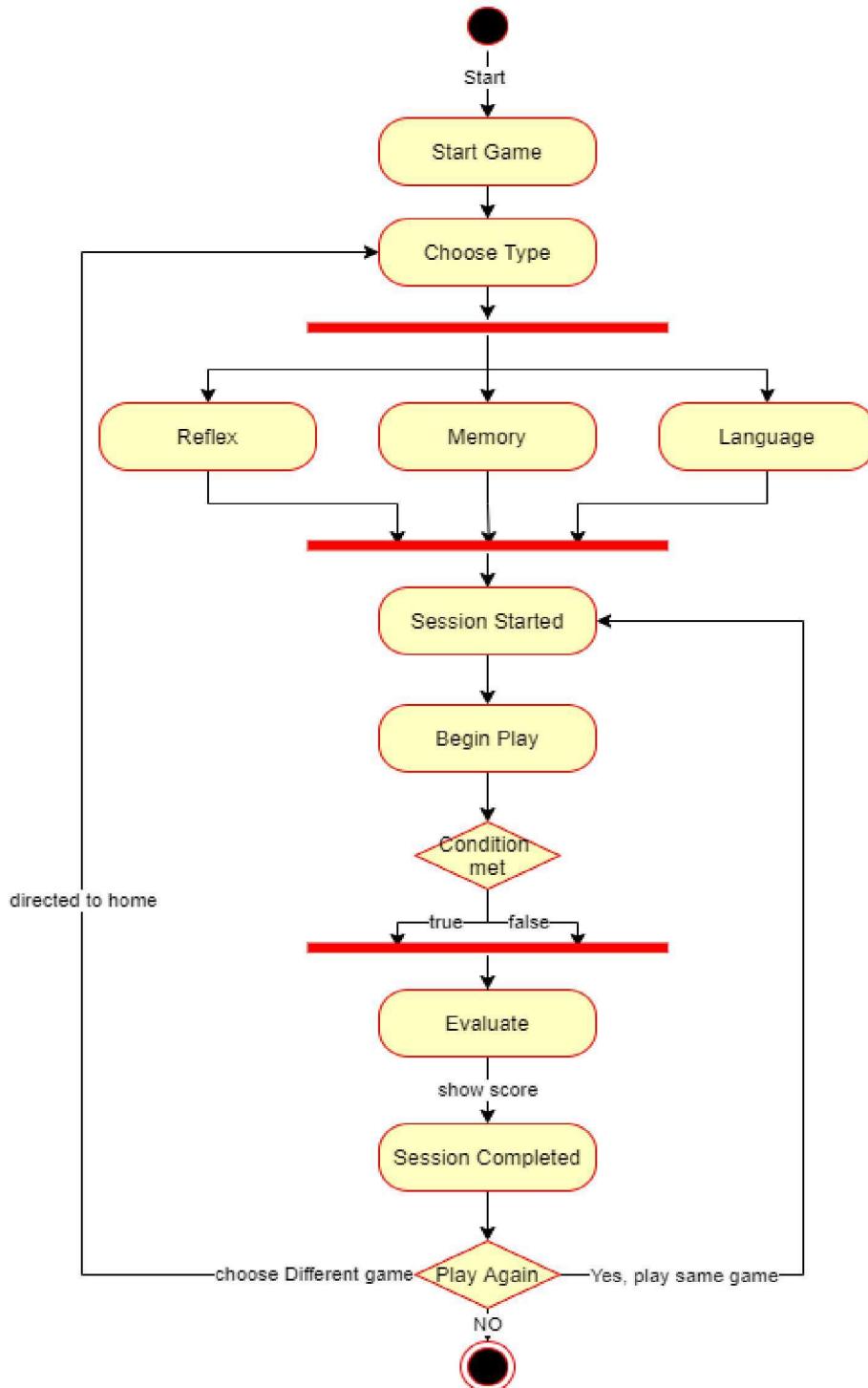


Figure 3.6 Activity Diagram

IMPLEMENTATION

4. IMPLEMENTATION

4.1. SAMPLE CODE

HomeScreen_Manager.cs

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.SceneManagement;

public class HomeScreen_GameManager : MonoBehaviour
{
    public GameObject homeScreenTopPart;
    public Text t;
    public Animator[] anims;
    public void slide()
    {
        homeScreenTopPart.GetComponent<Image>().enabled = false;
        homeScreenTopPart.GetComponent<Button>().enabled = false;
        anims[0].SetTrigger("EntryButtonClicked");
        anims[1].SetTrigger("one");
        anims[2].SetTrigger("two");
        anims[3].SetTrigger("three");
        anims[4].SetTrigger("four");
        t.enabled = (false);
    }
}
```

```
    }

    public void loadScene(int sceneIndex)

    {

        SceneManager.LoadScene(sceneIndex);

    }

    public void exitButtonClicked()

    {

        Application.Quit();

    }

}
```

Reflexione_GameManager.cs

```
using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.UI;

using UnityEngine.SceneManagement;

using System.IO;

public class ReflexioneOne_GameManager : MonoBehaviour

{

    public GameObject objectToDisable;

    public void disableObject()

    {

        objectToDisable.SetActive(false);

    }

}
```

```
    }

//Timer

private float time

{
    get;    set;
}

public Text timerText;

private int int_time;

private void Update()

{
    if(time > 0)

    {
        time -= Time.deltaTime;

    }

    int_time = (int)time;

    timerText.text = int_time.ToString();

}

//Initialize Board

private int difficulty

{
    get;    set;
}

public void getDifficulty(int _difficulty)
```

```

{
    difficulty = _difficulty;
}

public GameObject environment;

public GameObject prefab;

public void start Setting Up Environment()

{
    time = 60;

    while(environment.transform.childCount > 0)

    {
        DestroyImmediate(environment.transform.GetChild(0).gameObject);
    }

    environment.GetComponent<GridLayoutGroup>().constraintCount = difficulty;

    for (int i = 0; i < difficulty * difficulty; i++)

    {
        GameObject g = Instantiate(prefab, environment.transform);

        g.name = i.ToString();

    }

    StartCoroutine(chooseObjects());
}

private IEnumerator chooseObjects()

{
    int objectCount = environment.transform.childCount;
}

```

```

        if(time > 0)

    {

        enviroment.transform.GetChild(Random.Range(0, enviroment.transform.childCount -
1)).GetComponent<box>().time = 1f;

        yield return new WaitForSeconds(1f);

        StartCoroutine(chooseObjects());

    }

}

public Text scoreText;

public Text highScoreText;

public void increaseScore()

{

    int hscore = int.Parse(highScoreText.text);

    int sscore = int.Parse(scoreText.text) + 1;

    if(sscore > hscore)

    {

        highScoreText.text = sscore.ToString();

        string newline = "ReflexioneOne$" + highScoreText.text;

        File.AppendAllText("Assets/Resources/ReflexioneOneResources/SaveData.txt", newline
+ System.Environment.NewLine);

    }

    else

    {

        scoreText.text = sscore.ToString();
    }
}

```

```

        }

    }

public void decreaseScore()

{
    scoreText.text = ((int.Parse(scoreText.text)) - 1).ToString();

}

public void ResetGame()

{
    SceneManager.LoadScene(2);

}

private void Start()

{
    string[] lines = File.ReadAllLines("Assets/Resources/Reflexione Resources/SaveData.txt");

    string[] splitted = lines[lines.Length - 1].Split('$');

    highScoreText.text = splitted[1];

}

public void ExitToHomeScreen()

{
    SceneManager.LoadScene(0);

}

}

Shader "Custom/Tree_Billboard"

{

```

Properties

```
{  
    [PerRendererData] _MainTex("Sprite Texture", 2D) = "white" {}  
    _Color("Tint", Color) = (1,1,1,1)  
    //_Time ("Time", Float) = 0  
    [MaterialToggle] PixelSnap("Pixel snap", Float) = 0  
}
```

SubShader

```
{  
    Tags  
    {  
        "Queue" = "Transparent"  
        "DisableBatching" = "True"  
        "SortingLayer" = "Resources_Sprites"  
        "IgnoreProjector" = "True"  
        "RenderType" = "Transparent"  
        "PreviewType" = "Plane"  
        "CanUseSpriteAtlas" = "True"  
    }  
}
```

Cull Off

Lighting Off

```

ZWrite Off
Blend One OneMinusSrcAlpha

Pass
{
    CGPROGRAM
    #pragma vertex vert
    #pragma fragment frag
    #pragma target 2.0
    #pragma multi_compile _ PIXELSNAP_ON
    #pragma multi_compile _ ETC1_EXTERNAL_ALPHA
    #include "UnityCG.cginc"

    // uniform Float _Time;

    struct appdata_t
    {
        float4 vertex : POSITION;
        float4 color : COLOR;
        float2 texcoord : TEXCOORD0;
        UNITY_VERTEX_INPUT_INSTANCE_ID
    };
}

```

```

struct v2f
{
    float4 vertex : SV_POSITION;
    fixed4 color : COLOR;
    float2 texcoord : TEXCOORD0;
    UNITY_VERTEX_OUTPUT_STEREO
};

```

```
fixed4 _Color;
```

```

v2f vert(appdata_t IN)
{
    v2f OUT;
    UNITY_SETUP_INSTANCE_ID(IN);
    UNITY_INITIALIZE_VERTEX_OUTPUT_STEREO(OUT);
    //        OUT.vertex = UnityObjectToClipPos(IN.vertex);
    OUT.texcoord = IN.texcoord;
    OUT.color = IN.color * _Color;
    // #ifdef PIXELSNAP_ON

    OUT.vertex = mul(UNITY_MATRIX_P,
                    mul(UNITY_MATRIX_MV, float4(0.0, 4.1, 0.0, 1.0))
                    - float4(IN.vertex.x, -IN.vertex.y, 0.0, 0.0)

```

```

* float4(6.0, 8.0, 1.0, 1.0));

//      OUT.vertex = UnityPixelSnap (OUT.vertex);

// #endif

return OUT;

}

sampler2D _MainTex;
sampler2D _AlphaTex;

fixed4 SampleSpriteTexture(float2 uv)

{
    fixed4 color = tex2D(_MainTex, uv);

#if ETC1_EXTERNAL_ALPHA
    // get the color from an external texture (usecase: Alpha support for      ETC1 on android)
    color.a = tex2D(_AlphaTex, uv).r;
#endif //ETC1_EXTERNAL_ALPHA

    return color;
}

```

```
fixed4 frag(v2f IN) : SV_Target
{
    fixed4 c = SampleSpriteTexture(IN.texcoord) * IN.color;
    c.rgb *= c.a;
    return c;
}

ENDCG

}
}
}
```

SCREENSHOTS

5. SCREENSHOTS

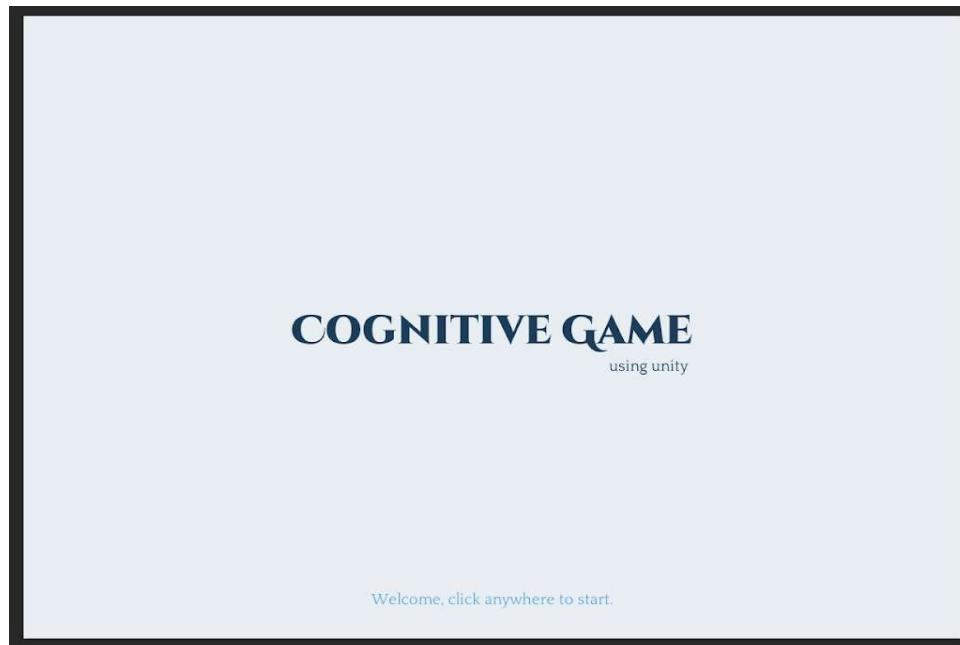


Figure 5.1 Initial HomeScreen

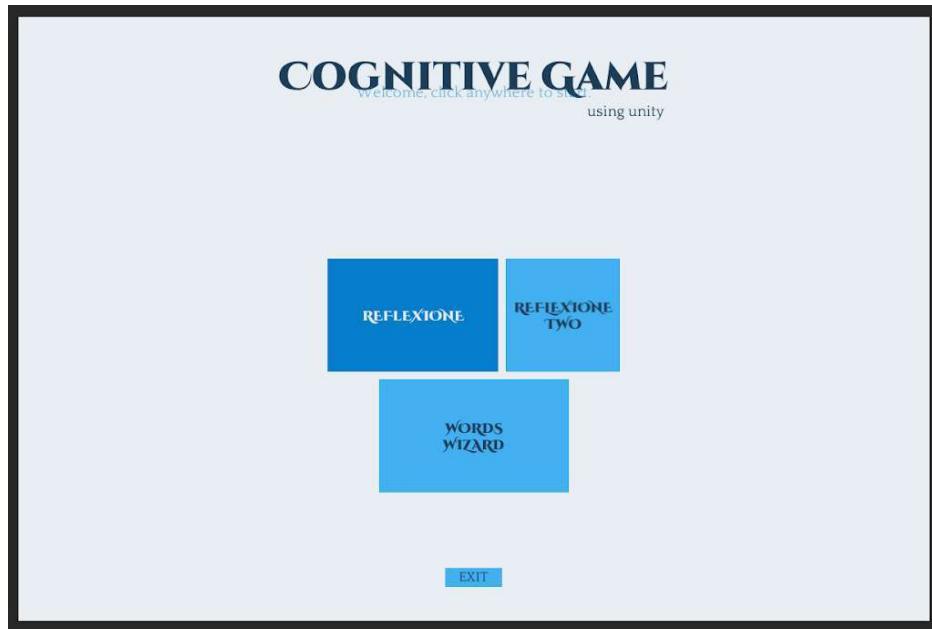


Figure 5.2 Main Home Screen

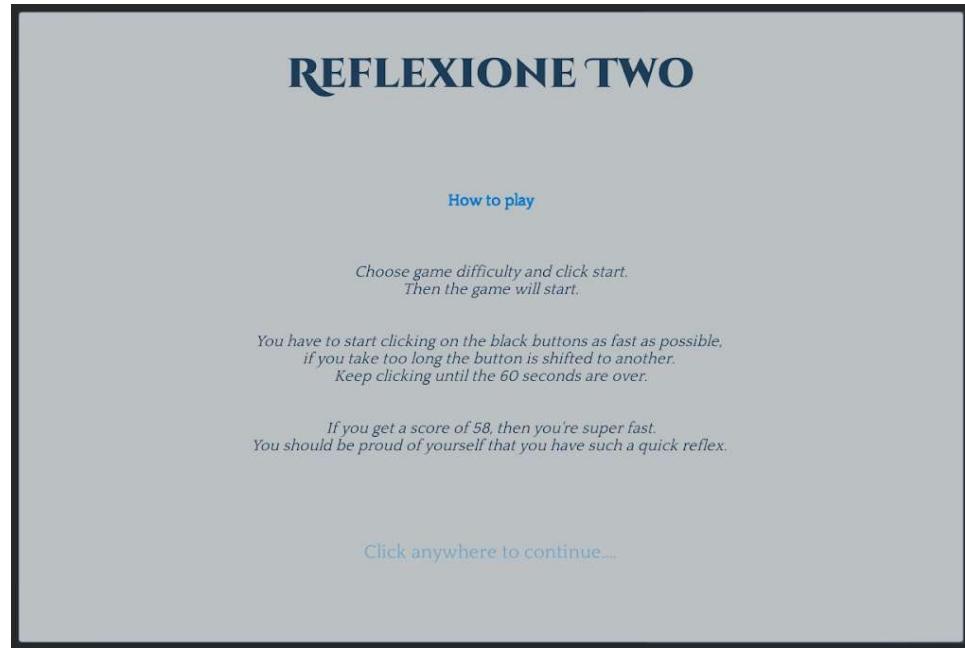


Figure 5.3 Reflexione One Initial Screen

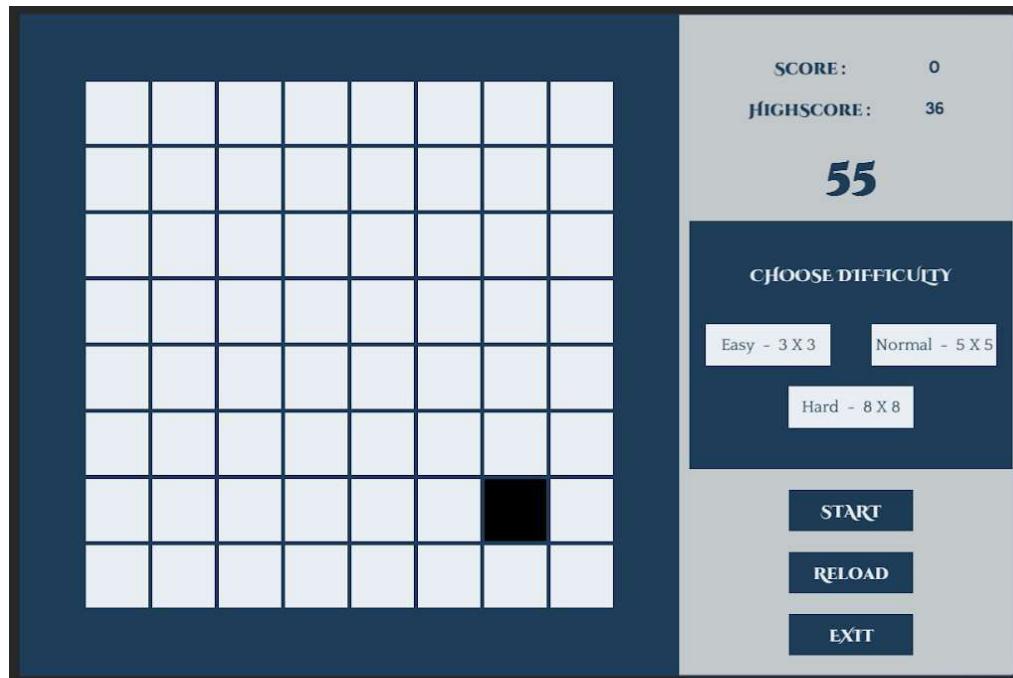


Figure 5.4 Reflexione One Game Scene

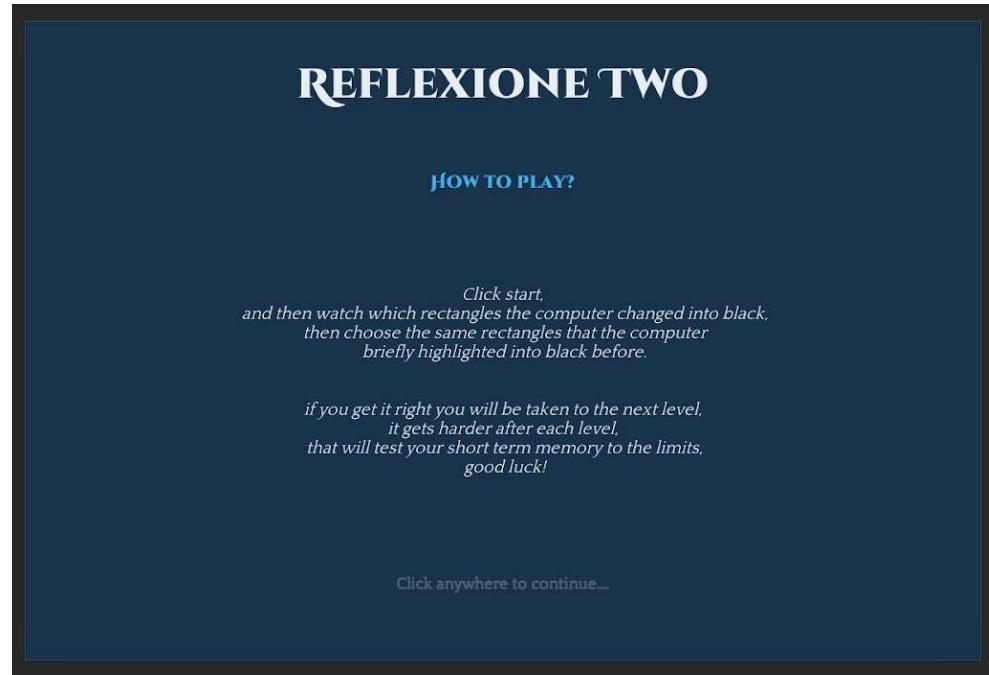


Figure 5.5 Reflexione Two Initial Screen



Figure 5.6 Reflexione Two Game Scene

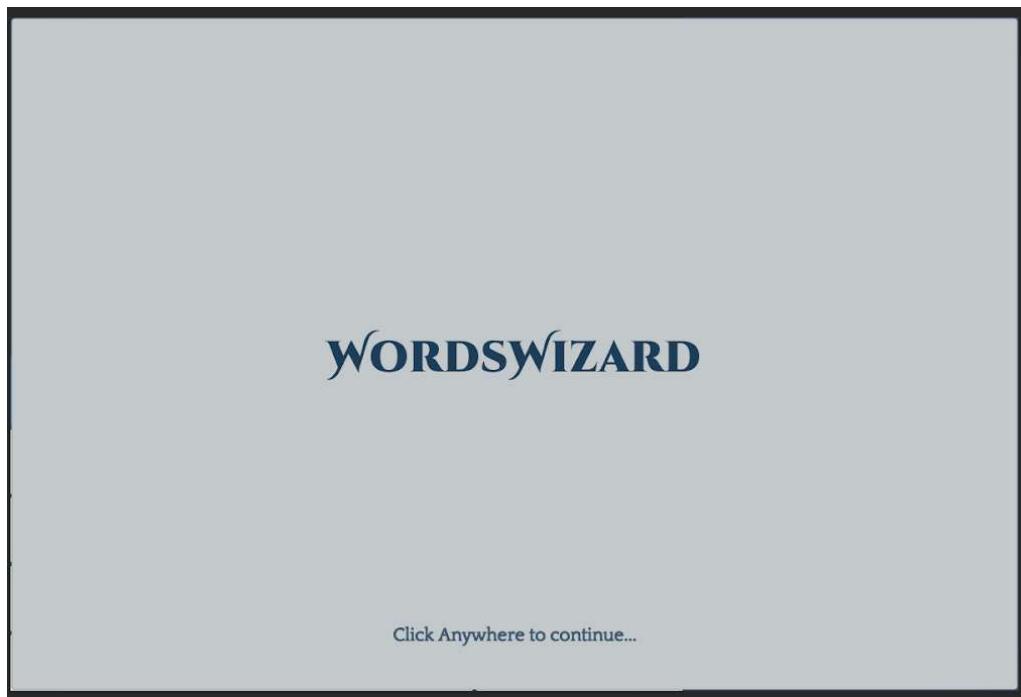


Figure 5.7 Words Wizard Initial Screen

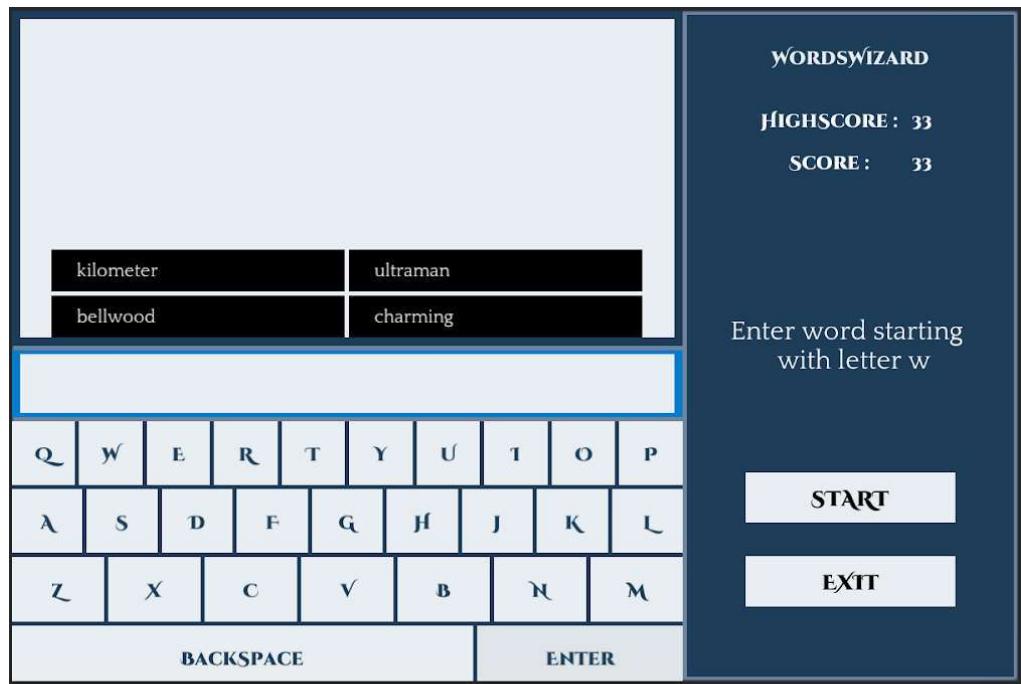


Figure 5.8 Words Wizard Game Scene

TESTING

6. TESTING

6.1. INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

6.2. TYPES OF TESTING

6.2.1. UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .It is done after the completion of an individual unit before integration. This is a structural testing that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.2.2. INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.3. TEST CASES

Test Case ID	Game	Correct Input	User Input	Result
1	Reflexione _One	Sequence Random(0,5,8)	Incorrect Sequence	Sound, Incorrect Input Reset
2	Reflexione _One	Sequence Random(3,5,8)	Correct Sequence	Sound, Correct Input Advance to next
3	Reflexione _Two	Random Container (id = 5)	Container (Id = 5)	Sound, Correct Input Advance to next
4	Words Wizard	Letter random (V)	Word starting with V	Updated in layout, next random letter
5	Words Wizard	Letter random (D)	Word starting with G	No updation in layout, next random letter

CONCLUSION

7. CONCLUSION

7.1. PROJECT CONCLUSION

The project is titled as “Cognitive game using unity” ia an interactive entertainment based cognitive ability improvement application. This Interactive Application provides three different types of cognitive ability training environments. This Interactive Application is developed using the Unity game engine.

7.2. FUTURE SCOPE

Unity releases its yearly LTS versions with new features every year. Because this engine is used solely for game development we can use its latest features to develop more immersive Interactive Applications.

BIBLIOGRAPHY

8. BIBLIOGRAPHY

8.1. GITHUB REPOSITORY LINK

https://github.com/Saiteja1705c8/CSE_CMRTC

8.2. REFERENCES

1. Anderson, C. A., & Dill, K. E. (2000). Video games and aggressive thoughts, feelings and behavior in the laboratory and life. *Journal of personality and social psychology*, 78, 772-790.
2. Bandura, A. (1986). Social foundations of thought and action: a social cognitive theory. Englewood cliffs: Prentice hall.
3. Dempsey, J. V., Rasmussen, K., & Lucassen, B. (1996). The instructional gaming literature: implications and 99 sources. Mobile: University of South Alabama.
4. Durkin, K., & Barber, B. (2002). Not so doomed: computer game play and positive adolescent development. *Applied developmental psychology*, 23, 373-392.
5. Egenfeldt-Nielsen, S. (2005). Beyond edutainment: exploring the educational potential of computer games. Unpublished PhD thesis, IT-University of Copenhagen, Copenhagen.
6. Unity Technologies. 2017a. LearningtheInterface. Available:<https://docs.unity3d.com/Manual/LearningtheInterface.html>. Accessed 14 September 2017.
7. Unity Technologies 2017d. Game View. Available : <https://docs.unity3d.com/Manual/GameView.html>. Accessed 14 September 2017
8. Unity Technologies .2017e .Hierarchy. Available:<https://docs.unity3d.com/Manual/Hierarchy.html>. Accessed 14September 2017.
9. Unity Technologies. 2017f. Inspector Window. Available:<https://docs.unity3d.com/Manual/UsingTheInspector.html>. Accessed 14 September 2017.

8.2. WEBSITES

1. Unity - Scripting API: (unity3d.com)
2. Unity - UnityLearn : (Unity.learn)