



GAME DEVELOPMENT USING UNREAL ENGINE 4

Dr. Rohini G
dept. of CSE, Reva University
Bangalore, India.
e-mail: Rohini.g@reva.edu.in

Hemanga Borpatra Gohain
dept. of CSE, Reva University
Bangalore, India.
e-mail: R17CS150@cit.reva.edu.in

Chilaka Syam Chand
dept. of CSE, Reva University. Bangalore, India.
email: R17CS095@cit.reva.edu.in

Shehan J Silva
dept. of CSE, Reva University
Bangalore, India
e-mail: R17CS157@cit.reva.edu.in

Pooja KN
dept. of CSE, Reva University
Bangalore, India.
e-mail: R18CS820@cit.reva.edu.in

Abstract: Our project is primarily focused on educating the people who are willing to learn and have interest in coding C++ is great start and showing its application on unreal engine is a great interactive way of doing so. The project will help those in such needs and the perfect platform for learning is by making it game which is fun. Flexibility is another aspect that can be applied here so learning at your preferred curve of learning to match your learning pace so anyone can learn this at any pace and enjoy the learning process. This project can be used as a demonstration for big corporates in the gaming industry where u will be able to explain the game how it works in depth with the coding magic behind it .It also uses decision making algorithms where NPC bots interact with the player.

Keywords: C++, NPC, Unreal Engine, UPROPERTY.VTT, Getters.

I. INTRODUCTION

Unreal Engine is the world's most generally accessible open and progressed ongoing 3D creation device. Ceaselessly developing to fill not just its unique need as a best in class game motor, today it gives makers across businesses the opportunity and control to convey front line content, intelligent encounters, and vivid virtual universes. Pair that with C++ coding language. Since C++ is an undeniable level language that will show you the rudiments of item arranged writing computer programs, it's a smart thought to learn it. supplemented by C in these games, and low level computing constructs for making low-level motor module. In the undertaking that our group is doing will be a game utilizing both these applications unbelievable motor and C++ to achieve out getting the hang of coding in the best time way imaginable. Games, for example, BULLS AND COWS GAME and Escape room which executes all the C++ ideas into one major program. This program is flexible and can be tested step by step to see various results so learning how to code is our main goal for this project its educating beginners as well as intermediate programmers into understanding the concepts like switch statements for example or how connecting function to its specific task like in a games we have observed how "hints" help us progress in the game

further so a function statement is established depending on the input the player/user has used. The second part of our project is combining C++ code and unreal gaming application platform to create a game using "NPC" in which we will understand how AI of game objects work according to the interaction with the objects and obstacles around. This game is called testing grounds where NPC interact with surroundings and players so many concepts will be applied so we can see how C++ benefits along with unreal engine to see results that could even help us into understanding coding and get jobs in well reputed gaming companies as a whole.

II. RECOMMENDED SYSTEM

It is defined as a technique that can be used to offer people their area of interest topics or in other words what they are likely looking for. The recommendations is known to analyze aspects and help with the best option.

Recommendation Systems are generally used to provide personalized experience to users for any given product.

A Recommendation System is a relatively new technique used to provide these personalized experiences.

This particular system is relatively modern towards game development using platforms like unreal engine.

The unreal engine platform is very user friendly and which makes it even more interesting to develop game of your choice.

With Unreal Engine you can do a lot of things simultaneously as it allows parallel work environments which speeds up the production process.

III. PROBLEM FORMULATION

Flexibility of the program and the use of the unreal editor is where the users of these two games where we see a interesting issue we are able to witness the weakness in certain parts of code by the user when testing various application of the program and game.

The effort delivers the outcome and be able to improve and compromise according to the needs of the user where he/she falls under where the weakness are and strengths are in understanding the concepts of the code.

Complexity of the code is confine within the knowledge we acquire in our academic years.

IV. Methodology

Combination of C++ syntax, #include and namespaces variables for cin for input, using getline functions, simplifying with functions ,iteration with for and while loops, Boolean and comparison ,do and while loops, classes, getter methods, constructors. Pseudo programming where we can implement our own function to control and develop our own purpose of the parts of the code with enumerations and error checking code where the code can deliver a precise position where the code might show errors and be visible and easily debugged.

Using visual code studio 2017 is the application for the game. Unreal engine on the other hand "gitgnore" files are used initially for Unreal. VTT.

Using the unreal editor interface with pointer primers Unreal Class system with runtime messages for feedback, accessing object names ,getting transforms in C++ ,moving the objects and laying out the geometry while applying the materials on these objects.

Macros starting with UPROPERTY. VTT is also used for implementation with additional trigger and collision volumes and using player controller and inherit the Game mode blueprints while getting to know the viewpoints with memory and accessors layout.

V. Application of the Recommended System

Use of this project is mainly educating the user on concepts of C++ and using its application of coding in a platform like unreal engine which is an industry standard application

which even big corporations use we can see its potential. Which is very beneficial for internships as well when applying our coding skills and demonstrating the code and its application.

Unreal-Engine is a finished set-up of improvement devices for anybody working with ongoing innovation. From plan representations and true to life encounters to excellent games across PC, support, versatile, VR, and AR, Unreal

Engine gives you all you require to begin, transport, develop, and stand apart from the group.

Each Unreal Engine permit accompanies free admittance to the whole different library for use in Unreal Engine. In view of certifiable outputs, this exceptional quality library highlights a huge number of 3D and 2D PBR resources with enhanced geography, UVs and LODs, and steady scale and resolution.

To help you track down the correct beginning stage for your project and accomplish your expected outcome in the most brief conceivable time.

Unreal Engine empowers you to browse an assortment of valuable formats, including ones for cooperative multi-client configuration audits on work area and VR gadgets, studio lighting with a HDRI scenery for item plan, and profoundly practical sun and sky environments for structural representation.

VI. Motivation

For developers just starting in the industry, the task of choosing the best game engine can be daunting so our project will could be really Helpful at the time of decision for a developer to choose which platform to use as the project is build using very basic C++ knowledge.

For many years, the process of developing a game was sluggish and, cumbersome after placing things into the engine.

With Unreal engine you can do a lot of things simultaneously which at the same time enhance the speed of production. So, we decided that unreal engine would be our primary platform for this project.

Programming will consistently be a fundamental piece of game turn of events, however for somebody simply beginning, you might not have a great deal of programming experience, Unreal Engine utilizes a progressive new simplified programming framework called Blueprint Visual Scripting. Without composing a solitary line of code

VII. Requirement to run the recommended system

To be able to use the program user needs to have operating system such as windows 8,10 64 bit.

The user also needs to have this minimum processor quad-core intel or AMD,2.5ghz.

Memory required for the program to be able to run is 4GB.

System requirements are as follows:

Unreal Engine 4- Windows 7

Visual studio version- vs 2017 v15.6.

VIII. Proposed system

Effective use of all the available C++ facilities.

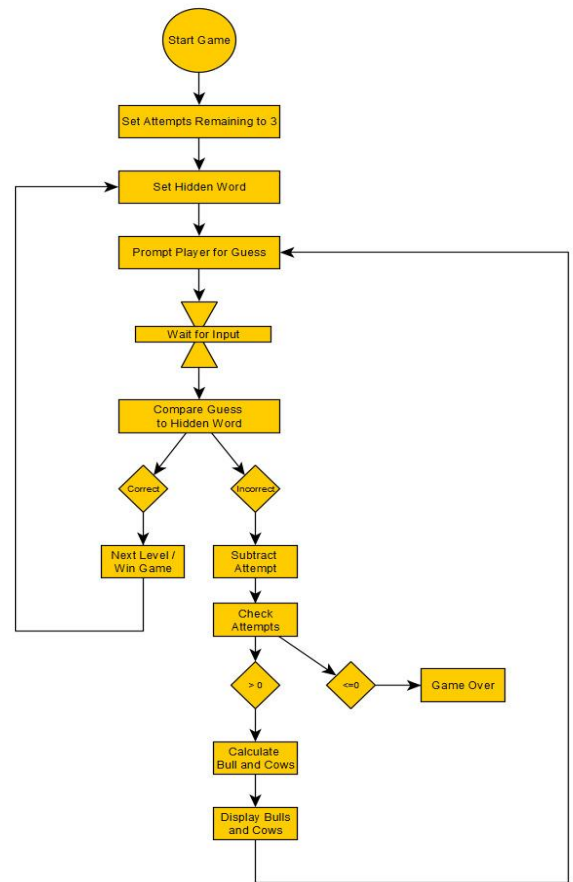
Pseudo programming where we can implement our own function to control and develop our own purpose of the parts of the code with enumerations and error checking code.

Where the code can deliver a precise position where the code might show errors and be visible and easily debugged.

Using the unreal editor interface with pointer primers ,Unreal Class system with runtime messages for feedback, accessing object names ,getting transforms in C++ ,moving the objects and laying out the geometry while applying the materials on these objects.

Macros starting with UPROPERTY. VTT is also used for implementation with additional trigger and collision volumes.

For more enhanced explanation please refer to our flow chart made below.



IX. Snippets in C++

Below is the snippets of our Implementation using C++

```

BullCowGame > Source > BullCowGame > BullCowCartridge.cpp
25 }
26 void UBullCowCartridge::SetupGame()
27 {
28     // Welcoming The Player
29     PrintLine(TEXT("Welcome to Bull Cows!"));
30
31     HiddenWord = GetValIdWords(words)[FMath::RandRange(0, GetValIdWords(words).Num() - 1)];
32     Lives = HiddenWord.Len() * 2;
33     BGameOver = false;
34     PrintLine(TEXT("Guess the N1 letter word!", HiddenWord.Len()));
35     PrintLine(TEXT("You have N1 lives.", Lives));
36     PrintLine(TEXT("Type in your guess and impress enter to continue...")); // Prompt Player For Guess
37     PrintLine(TEXT("The Hiddenword is: N1-"), HiddenWord); // Debug Line
38 }
39 void UBullCowCartridge::EndGame()
40 {
41     BGameOver = true;
42     PrintLine(TEXT("toPress enter to play again.));
43 }
44 void UBullCowCartridge::ProcessGuess(const FString& Guess)
45 {
46     if (Guess == HiddenWord)
47     {
48         PrintLine(TEXT("You have won!"));
49         EndGame();
50     }
51 }
    
```

Fig- 1.1

```

BullCowGame > Source > BullCowGame > @ BullCowCartridge.cpp > @ UBullCowCartridge::GetBullCowCount(FString& Guess)
44 void UBullCowCartridge::ProcessGuess(const FString& Guess)
45 {
46     if (Guess == HiddenWord)
47     {
48         PrintLine(TEXT("You have Won!"));
49         EndGame();
50         return;
51     }
52     if (Guess.Len() != HiddenWord.Len())
53     {
54         PrintLine(TEXT("The hidden word is %i letters long.", HiddenWord.Len()));
55         PrintLine(TEXT("Sorry, try guessing again! (you have %i lives remaining.)", Lives));
56         return;
57     }
58     // Check if Isogram
59     if (!IsIsogram(Guess))
60     {
61         /* code */
62         PrintLine(TEXT("No repeating letters, guess again!"));
63         return;
64     }
65     // Remove Life
66     PrintLine(TEXT("Lost a life!"));
67     --Lives;
68     if (Lives <= 0)
69     {
70         // ...
71     }
72 }

```

Fig-1.2

```

BullCowGame > Source > BullCowGame > @ BullCowCartridge.cpp > @ UBullCowCartridge::IsValidWord(const FString& Word) const
82 }
83 bool UBullCowCartridge::IsValidWord(const FString& Word) const
84 {
85     for (int32 Index = 0; Index < Word.Len(); Index++)
86     {
87         for (int32 Comparison = Index + 1; Comparison < Word.Len(); Comparison++)
88         {
89             if (Word[Index] == Word[Comparison])
90             {
91                 return false;
92             }
93         }
94     }
95     return true;
96 }
97 TArray(FString) UBullCowCartridge::GetValidWords(const TArray(FString& WordList) const
98 {
99     TArray(FString) ValidWords;
100     for (FString Word : WordList)
101     {
102         if (Word.Len() >= 4 && Word.Len() <= 8 && !IsValidWord(Word))
103         {
104             ValidWords.Emplace(Word);
105         }
106     }
107 }

```

Fig-1.3

X. Conclusion

This project now concludes the concepts of C++ covered and its application covered in the unreal engine as we can see how useful it is for students and teachers to see this project as way of educating the people who have viewed this project in a very effective manner in a short span of time.

We have touched a huge number of its systems, through this project we have obtained how to architect good C++ coding skills.

This project is not only used for game development but its transferable across a wide range of programming discipline like Web development or integrated system development

Any tool is great in the right hands, but games with great visual quality depends more on an artistic vision, so Unreal just is the right tool for them. That’s why the change, that’s why the preference

With Unreal Engine always updating with even faster interfaces for the users makes it easier for user to understand the interface better with the help of the community on how to improve the application .

We can see the success in the usage of the application as many companies which definitely help in resume of students who are into coding and aspire to be programmers.

XI. References

In 2011 [1] Was published by Steven G LeMay, Dwayne R nelson, Robert E Brecker, Greg A Schlottmann “Game development architecture that decouples the game logic from the graphics logics”.

In 2010 [2] Was published by Erik bethke “Game development and production”. It covers various aspects of game development.

In 2009 [3] Was published by Will Goldstone “unity Game development essentials”. This follows an informal, demystifying approach to the world of game development with unity Game engine.

In 2009 [4] Was published by Mayra Frans “An introduction to game studies, Game in culture”.

[5] Carlos Mauricio Torres- Ferreyros; Matthew Alexander Festini- Wendorff; Pedro Nelson Shiguihara -Juárez2016 IEEE ANDESCON

[6] Oswald Comber ;Renate Motschnig ; Hubert Mayer ; David Haselberger2019 IEEE Global Engineering Education Conference (EDUCON)

[7] Xiaowei Chen ;Meihong Wang Qingfeng Wu 2017 4th International Conference on Systems and Informatics (ICSAI)

[8] Adrian Drozina;Tihomir Orehovacki2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)

[9] Carlos Mauricio Torres-Ferreyros;Matthew AlexanderFestini-Wendorff;Pedro Nelson Shiguihara-Juárez2016 IEEE ANDESCON

[10] Review of methodologies and Games engines for their developmentJennifer Tomalá-González;José Guamán-Quinche;Edwin Guamán-Quinche;Wilman Chamba-Zaragocin;Silvana Mendoza-Betancourt2020 15th Iberian Conference on Information Systems and Technologies (CISTI)

[11]Ashwinkumar.U.M and Dr. Anandakumar K.R, "Predicting Early Detection of cardiac and Diabetes symptoms using Data mining techniques", International conference on computer Design and Engineering, vol.49, 2012

[12] KongdeeNop;JintapitakManissaward;Tangto Oattarapon2019 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT-NCON)