

A Synopsis on

Video-game development in Unreal Engine

Submitted in partial fulfillment of the requirements
of the degree of

Bachelor of Engineering

in

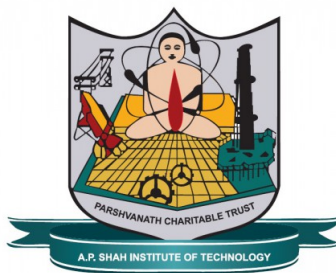
Computer Engineering

by

Ajinkya Darshane (16102001)
Falguni Tailor (16102031)

under the guidance of

Prof. Sachin Malave



Department of Computer Engineering
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI
2018-2019

CERTIFICATE

This is to certify that the project Synopsis entitled “***Title of project***” Submitted by “***Name of students (Student ID)***” for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Branch Name***.to the University of Mumbai,is a bonafide work carried out during academic year 2018-2019

(Name)
Co-Guide

(Name)
Guide

Prof. Kiran Deshpande
Head Department of Information Technology

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place:A.P.Shah Institute of Technology, Thane

Date:

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

(Name of Student and Student ID)

Date:

1 Contents At A Glance

1 Introduction.....	2
2.1 Overview.....	2
2.2 History.....	2
2.3 Evolution.....	2
2 Abstract.....	1
3 Objectives.....	3
4 Literature Review.....	4
4.1 Advancements in Computer Graphics.....	4
5 Problem Definition.....	5
6 Scope.....	6
7 Technology Stack.....	7
8 Benefits to the environment.....	8
9 Benefits to the society.....	9
10 Applications.....	10
Project Description.....	11
Project Design.....	12
Design.....	13
Activity Diagram.....	14
Use Case Diagram.....	15
Modules.....	16
References.....	17
Publication.....	18

2 Introduction

2.1 Overview

Video game development is the process of creating a video game. Game development is a software development process, as a video game is software with art, audio and gameplay. Planning is important for individual and group projects alike. One method employed for game development is agile development. It is based on iterative prototyping, a subset of software prototyping. This method is effective because most projects do not start with a clear requirement outline. A popular method of agile software development is Scrum [1].

2.2 History

During the 1940s and 1950s, computers took up entire rooms and were so expensive that only universities and large companies could afford them. Games like tic-tac-toe were excellent ways to attract public interest and support. Computer programmers were able to learn from the creation of games as well because it allowed them to break away from the usual subroutines and challenge the computer's capabilities. It was this mindset that led a group of MIT students during the 1960s to create one of the first and most ground-breaking computer games [2].

2.3 Evolution

Later, sometime in the late 1970s, came the arcade machines (also called coin operated machines). The first popular “arcade games” included early amusement-park midway games such as shooting galleries, ball-toss games, etc. In 1966, Sega introduced an electro-mechanical game called Periscope – an early submarine simulator and light gun shooter which used lights and plastic waves to simulate sinking ships from a submarine.

3 Abstract

The goal of this project is to develop a simple game using Unreal Engine based on an agile methodology that is economic, sustainable and practical. This methodology comprises of four stages, viz. preproduction, production, testing and post-production. We achieve to prove the applicability of the four-stage methodology to make a simple game in a short period of time, using limited resources. The three major game development engines available to free-lance programmers are the Crytek engine, the Unity Engine and the Unreal engine, of which Crytek Engine is proprietary. The code generated in the backend of Unity Engine is in C, while that in Unreal Engine is Visual C++. UE4 also makes usage of Blueprints and Environment Query System to program Artificial Intelligence and game mechanics. The Agile framework shall involve updating the game post release for newer versions, as well as removal of bugs in existing versions via beta testing

4 Objectives

4.1 Why : Purpose of this videogame The prime aim of this project is to make its users aware of the current deterioration of the environment via air pollution, poaching and felling of trees, how they affect us and what steps we can and should take to solve these major issues in the form of a simulation. 4.2 Who: Market Conditions The project also highlights the newly emerging videogame industry as a sub-set of the entertainment industry and focuses mostly towards youngsters. 4.3 What: The Point of creating the game Until the early 10s creating a videogame was considered costly and outside the scope of a small production firm. With the introduction of video-game development engines by Epic Games and Unity, it has become possible for anyone to develop one and craft a source of income as videogame industry is hugely profit-based.

5 Literature Review

5.1 Advancements in Computer Graphics Video games have long been a part of entertainment well early since the 1970s. But they were very expensive, to say the least. Even the resources for developing such games were very limited, and mostly closed source. With the advent of recent computers (mostly due to their computing power, hardware, memory usage, resource allocation, and being open source), it is now possible to develop simple games using minimum expense and efforts.

Recent developments in computer graphics have concentrated on advancements in hardware and software equally. Greater computing power equals greater yield.

One of the latest research areas in computer science and computer graphics is augmented reality and virtual reality. Virtual reality is an interactive computer-generated experience taking place within a simulated environment. It incorporates mainly auditory and visual feedback but may also allow other types of sensory feedback. On the other hand, augmented reality is a technology that superimposes a computer-generated image on a user's view of the real world, thus providing a composite view.

6 Problem Definition

A videogame that receives overwhelmingly positive reviews on steam is a one that has the perfect gameplay, a well-balanced story between main missions and side-quests, proper graphics optimization for all platforms including PC, XBOX ONE and PS4, a reasonable price and no needless post release DLC. This actually was the case when EA ruled the gaming market in the early 2000s, while as of today, it is rare for a production company to release a complete game that does not involve micro-transactions. This has annoyed all gamers alike as they demand a standalone without pay-to-win DLC. This project intends to solve a major fraction of this gamer dilemma whilst deploying the finished product on the Epic Games Store.

7 Scope

The Unreal game engine makes use of C++ and Blueprints to efficiently render skeletal meshes, materials, assets and integrate them into a simulated environment. The infractions caused by human beings have been visually demonstrated by Artificial Intelligence (BOTS). The AI has been rationalized using behaviour trees, environmental query system to replicate the behaviour of humans and animals. The AI plays the role of the antagonists which deteriorate the environment around an industrial ecosystem. The player has a set of objectives to accomplish to restore balance in the habitat and perform some side quests parallel to go with the mission objective. The AI have a clearly defined range of interaction logically designed using the BTs. The interactive cases get execute one at a time. Certain nodes in the BTs have an embedded Environmental Query System which acts as a separate flow of execution. Once the ecosystem gets restored to its initial state, the flora and fauna return to initial status. The Unreal Engine version used for this project is 4.22; it's the first IDE that supports ray tracing and offers advanced AI scripts. The particle effects rendered in the engine emphasize the present and foreseeable effects on the environment. Updated lighting effects act as visual boost.

8 Technology Stack

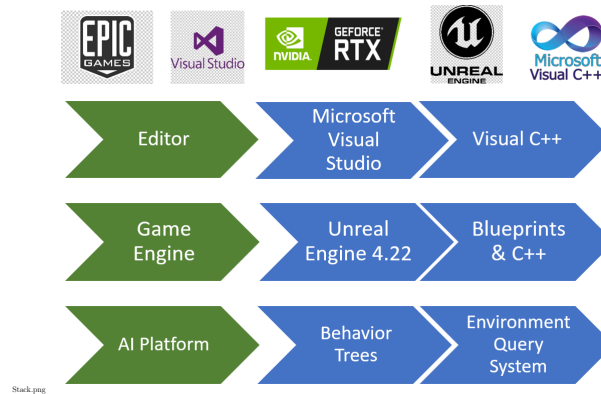


Figure 1: Technology Stack

Unreal Engine is a part of Epic games and every finished game is deployed on the Epic Games Store. The game uses assets that are created within the material editor or those available from the Epic Games Store. The logic for the game is designed using the Microsoft Visual Studio IDE in the Visual C++ programming language. The major difference between C++ and Visual C++ is that in the later, for an unreal project, the compiler ignores `UFUNCTIONS()` and `UPROPERTIES()` declared in the header files and makes use of engine functions in the editor. An important aspect of creating video games is rendering. Stronger the GPU, smoother and faster is the rendering. The GPU used in this project is the EVGA Nvidia GeForce RTX 2070.

9 Benefits to the environment and society

Environmental Aspects

This project aims to bring into notice the importance of environmental aspects in our everyday life and the deterioration caused by humanity to our surroundings including the atmosphere, flora and fauna, as well as aquatic bodies and make the users of this application aware of their role to the environment and the society. The user can exhaust the in-game mechanics to alter the outcomes of each successive mission on the game environment in the form of particle effects rendered in game and the change in the behaviour of the AI. This acts as a direct simulation of how it works in the real world, the only difference being, out there, the changes are irreversible, but this simulation make an impact on users towards their responsibility, especially now that the Amazon rainforest has been adversely depleted, and species are on the verge of extinction.

Society Aspects

Video games have long been a part of entertainment well early since the 1970s. But they were very expensive, to say the least. Even the resources for developing such games were very limited, and mostly closed source. With the advent of recent computers (mostly due to their computing power, hardware, memory usage, resource allocation, and being open source), it is now possible to develop simple games using minimum expense and efforts.

Recent developments in computer graphics have concentrated on advancements in hardware and software equally. Greater computing power equals greater yield. Since UE4 is now free, videogames can be created by individual users and generate an easy source of income whilst house-sitting. Epic Games offers programmers to put up not just games but also individual assets on the marketplace, thus flourishing the videogame industry.

10 Applications

One of the latest research areas in computer science and computer graphics is augmented reality and virtual reality. Virtual reality is an interactive computer-generated experience taking place within a simulated environment. It incorporates mainly auditory and visual feedback but may also allow other types of sensory feedback. On the other hand, augmented reality is a technology that superimposes a computer-generated image on a user's view of the real world, thus providing a composite view. Recent developments in computer graphics have concentrated on advancements in hardware and software equally. The latest advancements in videogame industry have been made in the healthcare sector. The in-focus videogame highlights creation and usage of artificial intelligence using behaviour trees, based on Environmental Query System that runs in a separate blueprint in the form of a stack. The Behaviour Trees follow a flow of logic that executes sequentially in pre-order traversal, such that the first node has the highest priority and last one has the lowest, showing us how the AI thinks. AI can also be hardcoded to perform a specific task. A heavily rendered environment is aesthetically pleasing to all the viewers and attracts a whole generation quite easily. Sending a message to a wider audience through the means of a game is relatively easier as its objectives subconsciously tend to affect the mindset of a youth better than the currently available means.

Project Design

Proposed System Architecture/Working

3.1 The Four Stage Methodology Before starting out, it was necessary to plot the development cycle in terms of stages. For this purpose, we used a four-stage methodology [3] that consisted of four different stages:

- Pre-production
- Production
- Testing
- Post-production

Details for the same are given below.

Pre-production

In the initial stage, the requirements and information about the project were collected and met. Hardware Requirements According to wiki.unrealengine.com,

- Quad-core Intel or AMD processor, 2.5GHz or Faster.
- Nvidia GeForce 470 GTX or AMD Radeon G870 HD Series card or higher.
- 8 GB of RAM(Recommended).

After verifying the critical requirements, three platforms of succeeding generations of Intel Processors and Graphics Cards were chosen to implement the project.

Software used:

- Unreal Engine 4
- Visual Studio 2017 Community Edition
- Autodesk Maya 2018

Hardware used:

Platform 1:

- Intel Core i5 8600K (8th Gen)
- EVGA Nvidia GeForce RTX 2070 (8 GB DDR6)
- 16 GB DDR4 3000MHz RAM
- 22" FHD Screen

Platform 2:

- Intel Core i7 6700HQ (6th Gen)
- Nvidia GeForce GTX 960M
- 16 GB DDR3L 1666MHz RAM
- 15.6" 4K Screen (HDR)

After gathering feedback about the type of game to develop, it was decided to go with a simpler approach and then build on it further. 3.1.2 Production During the production stage, all the necessary asset files were created within Unreal Engine itself, while external assets, such as audio files, were imported.

Unreal Engine contains 2 different types of 3D models: a static mesh and a skeletal mesh. Static meshes are not used for movement, while skeletal meshes use vertices to connect their joints, hence can be used for movement.

Point lights are used for luminosity/irradiance/particle effects.

Design

C++ Header and Source Files

Each class and its constituent function are declared in a header(.h) file and defined in the .cpp file of the same name. If declared in some other .cpp file, the header file is included in the beginning. There is no main function for the flow of execution of the code, but rather the “UnrealBuildTool”, which uses C for building the project using a string of Game Modules.



Figure 2: Visualization of C++ Classes in the UE Editor

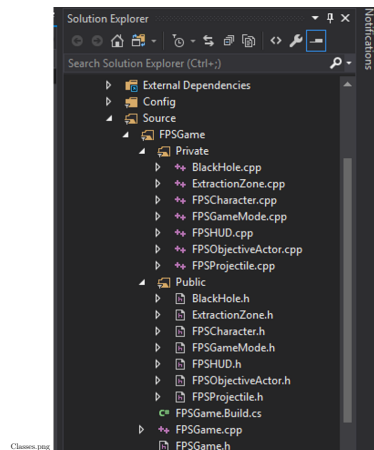


Figure 3: C++ Header and Source files

Activity Diagram

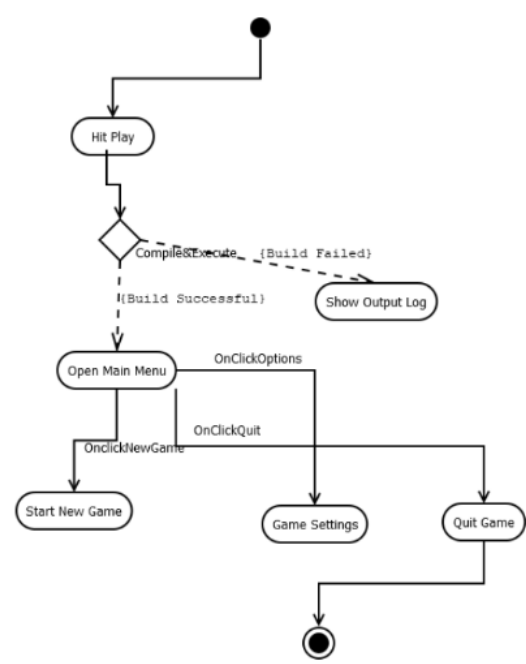


Diagram.png

Figure 4: Activity Diagram

Use Case Diagram

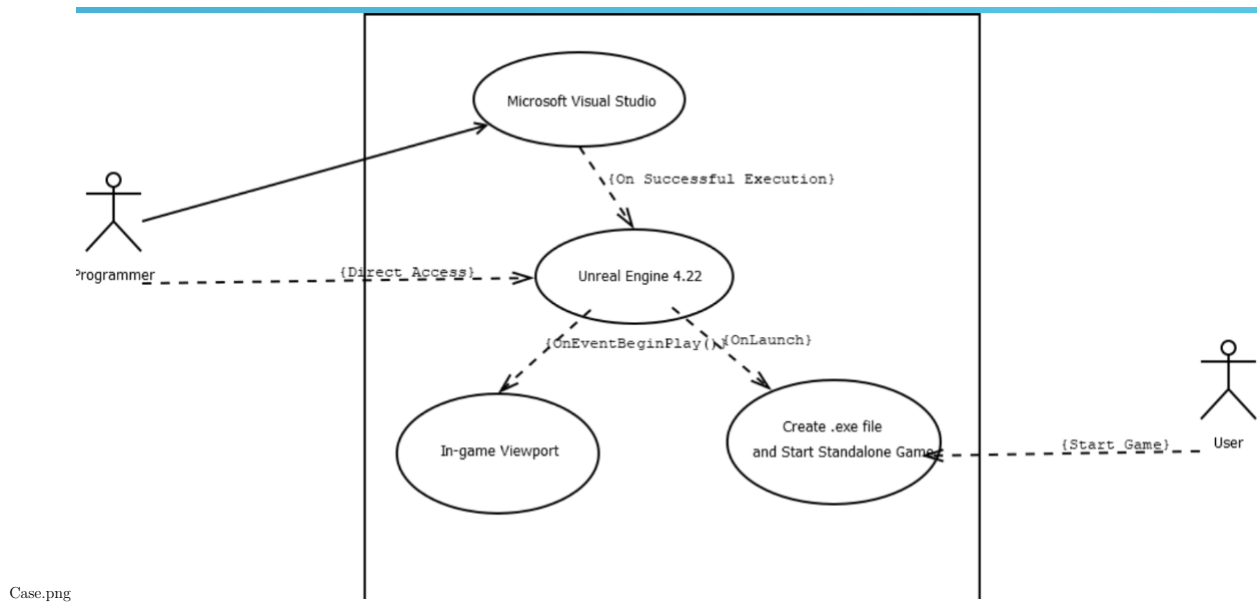


Figure 5: Use Case Diagram

Modules

Introduction to Blueprints and Visual C++ The Blueprints Visual Scripting system in Unreal Engine is a complete gameplay scripting system based on the concept of using a node-based interface to create gameplay elements from within Unreal Editor. As with many common scripting languages, it is used to define object-oriented (OO) classes or objects in the engine. As you use UE4, you'll often find that objects defined using Blueprint are colloquially referred to as just "Blueprints." This system is extremely flexible and powerful as it provides the ability

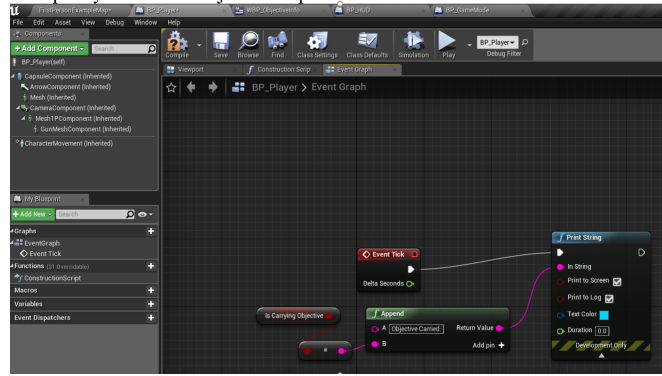


Figure 6: Blueprint Event Graph

for designers to use virtually the full range of concepts and tools generally only available to programmers. In addition, Blueprint-specific mark-up available in Unreal Engine's C++ implementation enables programmers to create baseline systems that can be extended by designers. Functions Function Calls are actions that can be formed within Blueprints that correspond to functions belonging to a targeted Actor or Object. In the case of Level Blueprints, the associated Actor in many cases is the Level Blueprint itself. Function Calls are displayed as boxes with titles that show the name of the function. Different types of function calls have different colour titles.

There are basically two possible function types used in blueprints, i.e. the getter function and the setter functions. Getter functions are used to get parameters or inputs from the LOC, or the user and the setter functions set certain values to parameters. The above blueprint function represents a basic Health Bar Heads-up Display (HUD). The values of key-bindings created in the Character's blueprint are returned and Casted to The Player's blueprint. The CurrentHealth MaxHealth, converted to percentage is returned to the Heads-Up Display.

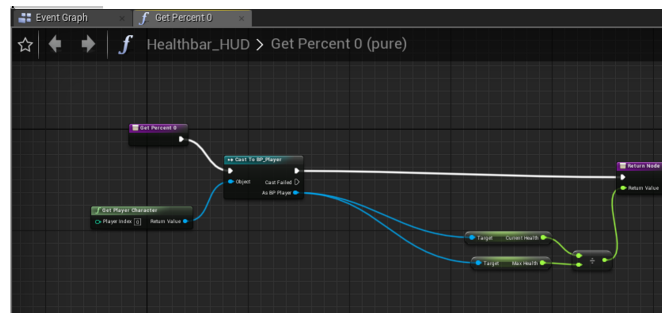


Figure 7: Blueprint Function

References

- [1] <https://www.unrealengine.com/en-US/blog/a-new-look-for-the-unreal-engine-documentation>
- [2] <https://docs.unrealengine.com/en-US/GettingStarted/index.html>
- [3] <https://www.unrealengine.com/marketplace/en-US/store>

11 Publication

Paper entitled “**Paper Title**” is presented at “**International Conference/Journal Name**” by “**Author Name**”.