GRIERSON, MATTHEW t7094911

Games engine construction report

User Guide

How to Play

In this assignement I have attempted to create a top-down Action Adventure in a similar style to the original legend of zelda game. Unfortunatly due to time constraints some of the features I intended to implement have not been incorporated into this game demo/engine. The current list of features is the ability to move around the world space on screen, A variety of menu screens, inventory, and pause menu’s while playing the game, there are 3 different types of enemy; ranged that patrol around the screen and when the player is within sight will fire projectiles at the player, melee wich also patrol, but when the player is too close they will chase the player untill next to them when they will attack the player and finally a randomly moving behaviour that justs roams randomly around the screen. Ranged and melee attacks have been implemented for the player. I also made a few black boxed systems; Graphics, World, Artificial Intelligence and Sound. There are collectable gold scatered around the screen, and other key items that can be picked up and added to the inventory. There are a couple of user interface bars and a display of the amount of gold upon the screen. So the small demo that I have been able to produce is a top-down perspective rooms where a random amount of coins and enemies are spawned within the room the aim is then to kill all of the enemies and collect all of the coins, but you are also able to leave by any of the four doors leading off of the room to regenerate the room.

Control Guide

WASD buttons are used for controlling movement of the player. 3 to heal the player

1 and 2 are used for firing projectile fireballs

F to pick up gems, keys and potions and to leave the inventory, map and arsenal screens

C To open the arsenal screen

I to open the inventory

M to open the map

Melee attack

Melee attack and menu button pressing

Block

Magic 1

Inventory

Blocking



Arsenal



Healing

Magic 2

Movement

Interaction Button

Map screen

Issues/Bugs

There are a few issues that are known mostly due to bad planning as the deadline approached. Firstly the left and right wall sliding is a bit catchy due to the use of multiple individual sprites and not as one continuous wall and so the player gets caught upon the tops and bottoms of the other wall sprites. If you enter one of the doors facing any direction than at the door you wish to go to you may end up at an unintended door. Unimplemented map and arsenal screens. Not fully implemented controls to be able to start playing the game solely with the Xbox controller. Finally an unknown memory leak.

Implementation Check list

|  |  |
| --- | --- |
|  | Adequate Rated Items (D to C) |
|  | Graphics |
|  | A ‘black boxed’ graphic system is in place |
|  | Textures can be efficiently drawn to arbitrary positions on and partially off the screen (clipped) |
|  | Animation is implemented and working correctly |
|  | World State |
|  | A player entity exists |
|  | Input is recognised and can be used to alter the world state e.g. move the player entity |
|  | The Xbox controller is supported and ‘hot pluggable’ |
|  | Code Quality |
|  | Class interfaces are minimal and complete. Class function and member variable visibility is correct |
|  | Code can be built and executed without compiler errors or warnings in debug and release |
|  | Code is well commented |
|  | There are no memory leaks |
|  | There is good error handling throughout |
|  | You have followed all the submission requirements e.g. made a video, submitted the correct files etc. |
|  | Report |
|  | All requested sections have been attempted adequately and the report is professionally presented |
|  | Good Rated Items (C to B) |
|  | World State |
|  | A world model system is in place. It is separate from other code and black boxed |
|  | There is a game loop handling input, world update and rendering |
|  | Bounding rectangle collisions are detected |
|  | There are multiple world entity types |
|  | Code Quality |
|  | Good use of object-oriented techniques e.g. polymorphism, member variable visibility |
|  | Memory is only allocated / deallocated outside of the game loop |
|  | Const is used correctly |
|  | AI |
|  | Some AI routines are in place e.g. enemy entities move around the world following paths,  use state machines etc. |
|  | Report |
|  | This report would allow another programmer to work with your code systems |
|  | Other |
|  | Some sound effects are in place |
|  | Excellent Rated Items (B to A) |
|  | Graphics |
|  | Interpolation is used to smooth entity movement |
|  | World State |
|  | The player entity can shoot projectiles (or equivalent functionality) |
|  | Explosion and bullet management |
|  | Game play is independent of platform capabilities (i.e. uses a model tick approach) |
|  | Game cycling e.g. detection of win / lose conditions and restarting the game |
|  | There is a scoring system with the score shown on screen |
|  | AI |
|  | Several different enemies with differing behaviours |
|  | Report |
|  | This report has insightful and balanced reflection |
|  | Extra Marks (Examples) |
|  | Mapping of world space on to screen space |
|  | Other graphics techniques have been implemented e.g. background scrolling, blending modes etc. |
|  | Level data is loaded from a file |
|  | A difficulty level |
|  | More advanced C++ e.g. use of namespaces, STL, C++ 11 and patterns |
|  | ‘Intelligent’ enemy behaviour |
|  | there are sound effects for collisions, explosions and firing |
|  | Additional black box systems have been implemented e.g. for AI, Sound |
|  | HUD features beyond simple text e.g. health bars, mini maps etc. |
|  | Other features, please list below: |
|  | Sliding along diagonals |
|  | complicated animation state system |
|  | pixel perfect collisions |

The items with green boxes next to them I believe I have accomplished.

Maintenance Guide

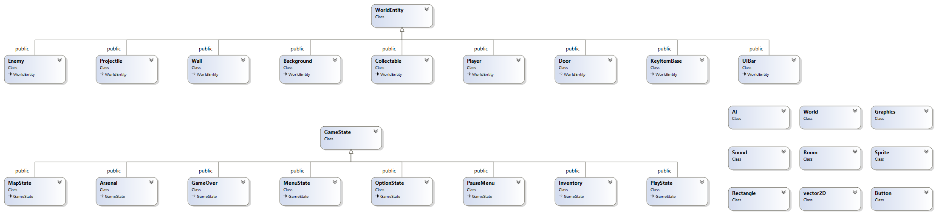


Figure 1 A UML diagram of the classes that I have made for my game engine

There are many different black boxed systems that you could implement as systems for a game engine; World, AI, Graphics, Sound, UI and a network. In my own game engine I implemented the World, Graphics and partially AI and Sound systems. The purpose of the World system is to simulate the World environment of the games physics and mechanics and essentially to make the game as a whole work and fun. The purpose of the graphics system is to show a graphical representation of the game world onto the computer screen of the user for the 2D engine that I have made it is the rendering and clipping of pixel sprites, but in a 3D engine the graphics system would need to display 3D models onto the 2D screen. In my engine the graphics is only linked to the World by a single pointer held by the world which is passed to the many different functions that have use of passing or receiving information from this system. You would be able to replace the visualisation system used as long as the system that you replace it with has the same functions to be able to get the same data and achieve the same functionality. The advantage of minimising the interlinking of the different systems means that it is easier to debug if you have a specific bug with the graphics system or other system it also means that if you have need of a graphics system in the future then it would presumably be easier to just re-use the system in another game without having to re-invent the wheel. The disadvantages of this minimal linking is that if you have need of the different parts of the graphics system is that you need to pass a pointer or reference to the relevant function. In the Graphics system there is a sprite class that looks after all of the data for the images that are going to be used to display the game and the graphics class itself which has control of the screen itself and getting information on the sprites outside of this system. In the world system you have the world class that hold the visualisation class and holds the game loop and in my engine controls which of the scene states is currently being drawn and updated. There are also world entities which are the different actor in the system ranging from the background which doesn’t have much functionality to the player and enemies that have movement, combat, collision etc.

Conclusions

Firstly I have to say that I really enjoyed the challenges that this project was able to give me and I feel that I learnt a lot more about the use of classes and pointers and C++ as a whole I feel that I could have used my time on this project and the others I had at the same time more effectively at times. I learnt how to use shared pointers for the first time, I was able to solve some complicated logical problems in my own time mostly under my own steam. For example I found a way to animate a sprite sheet with eight directions and multiple different actions upon the same sheet with the use of a state machine. I feel that I made a few simple mistakes when using data structures and forgetting to clear them by not commenting my code effectively or at all in most cases. Going forward I intend to try to make sure that I comment my code as I go and to use less ambiguous throwaway symbols within my code for when I am testing some of my logic. I also feel like I am a little too ambitious at times striving to accomplish more than I have the time to accomplish. In the future I would like to fix all of the small bits that I rushed to complete at the end and the bits I never really started, like the Map screen and the ability to switch to a different weapon and spell type during play as you play more of a complete gaming experience. I would really like to track down the mysterious memory leak that appeared upon deadline day. Overall I feel I made a good start on a simple 2D engine that was fun as well as a good challenge to my coding skills that helped me learn a lot along the way.

Art Credits

Bellanger, C (2018) *various art* [Art] Available at: <https://opengameart.org/users/clint-bellanger>

Matei, S (2017) *various art* [Art] Available at: <https://sergiumatei.artstation.com/>

Anonymous *Fireball* [Sound] Available at: <https://www.freesoundeffects.com/free-sounds/fireball-10079/>