User Stories – Alex/David

As a player I want to move so that I can get somewhere.

As a player I want to look so that I can see my surroundings.

As a player I want to throw a fireball so that I can kill enemies.

As a player I want to pick up mana so that I can throw more fireballs.

As a player I want to lose health so that I can die and lose the game, posing a challenge to me.

As a player I want to reach the end of the level so that I can win the game.

As a player I want to jump so that I can dodge enemy fireballs.

As a player I want to quit so that I can exit the game.

As a player I want to respawn so that I can try again if I die.

As an enemy I want to move so that I pose a challenging target to the player.

As an enemy I want to attack so that I can kill the player.

* As a **player** I want to **move** so that **I can get to a new location**
* As a **player** I want to **look** so that **I can see where I am going and aim the fireballs**
* As a **player** I want to **throw fireballs** so that **I can kill demons**
* As a **player** I want to **pick up mana** so that **I can recharge my mana meter**
* As a **player** I want to **recharge my mana** so that **I can throw more fireballs**
* As a **player** I want to **lose health** so that **I can die**
* As a **player** I want to **reach the end of the level** so that **I can win**
* As a **player** I want to **jump** so that **I can dodge fireballs**
* As a **player** I want to **quit the game** so that **I can stop playing**
* As a **player** I want to **respawn** so that **I can try again**

Class Diagram – David



Software Design – Flowcharts & Pseudocode

## Player Movement (David Mincer)



Start

If ‘W’ button down

Add walking force to Direction vector z

If ‘S’ button down

Subtract walking force to Direction vector z

If ‘A’ button down

Subtract walking force to Direction vector x

If ‘D’ button down

Add walking force to Direction vector x

Multiply Direction vector by y Rotation matrix

Set Index as 0

While index is less than number of assets in level

If Player + direction does not collide

Add Direction vector to Player position

Add 1 to index

Reset Direction vector

End

## Jumping (David Mincer)



Start

If jumper counter is more than 0

Add jump force to Direction vector y

Subtract 1 from jump counter

If on contact with floor and ‘SPACE’ button pressed

Set jump counter to maximum

End

## Player Looking (David Mincer)



Start

Gets 2D vector from mouse axis input

Multiply x axis by pitch rotation matrix

Multiply y axis by yaw rotation matrix

Multiply Player’s rotation by yaw

Multiply Camera’s rotation by full rotation matrix

End

## Throw Fireballs (David Mincer)



Start

If ‘throw’ button inputted

Instantiate new instance of fireball at fireball spawn point

Set fireball direction as normalised distance from camera

Add direction to fireball direction vector

End

## Pick Up Mana (David Mincer)



Start

While index is less than the size of array of mana

If player and mana[index] are colliding

Add mana to player’s mana

Remove mana from array of mana

Add 1 to index

End

## Lose Health (David Mincer)



Start

If fireball collides with player

Subtract fireball damage from player health

If health is less than 0

Set health to 0

End

## End Level (David Mincer)



Start

If player is within range of end

Deconstruct level

Load menu

End

## Respawn (David Mincer)



Start

If health is 0

Set health to maximum

Set player position as start

End

## Quit the Game (David Mincer)



Start

If ‘Escape’ button inputted

Deconstruct game

Close game

End

Demon – Move/Attack - Alex

currentAttackCooldown -= frameDuration //frameDuration = 1/60 second

Move towards target

If (position == target)

If (target == location A) target = location B

Else target = location A

If ((player->getPosition() – this.position).magnitude < attackRange)

If (currentAttackCooldown <= 0)

throwFireball()

currentAttackCooldown = maxAttackCooldown



Mana Bar - Alex

Int manaRemaining = player->getMana()

Foreach (manaSegment in manaBar)

If (manaRemaining > 0)

manaSegment->setActive(true)

manaRemaining -= 1

else manaSegment->setActive(false)



Particles - Alex

Start{

Create particles

Add to vector

Set inactive location to off screen position (under the map)

Move to inactive location

}

Activate{  
Move particles to desired location

Apply transformations to particles (Mainly movement/translations)

Count duration of particles

If (duration >= maxDuration)

Return particles to inactive location

Reset transforms

}



Menus - Alex

Main menu – Start, Quit

Start

Load screen

Load level

Spawn enemies

Set player

Remove load screen

Begin level

Quit

Destroy Enemies

Destroy player

Unload level

Shutdown Direct3D

Pause Menu – Resume, Restart, Quit to Menu

Resume

Resume update calls

Restart

Load screen

Unload entire level

Reload entire level

Set player

Remove load screen

Begin level

Quit to Menu

Load screen

Unload entire level

Remove load screen

Display main menu

White Box Testing - Alex

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Related Feature** | **Expected Result** | **Actual Result** | **Action** |
| 1 | Enemy attack | Attack cooldown is counted down every frame |  |  |
| 2 | Enemy movement | Enemy’s pathing targets are correctly set |  |  |
| 3 | Enemy attack | The range to the player is correctly determined |  |  |
| 4 | Mana bar | The player’s remaining mana is correctly returned to the mana bar |  |  |
| 5 | Mana bar | The mana segments are set active when appropriate |  |  |
| 6 | Mana bar | The right number of segments are set to represent remaining mana |  |  |
| 7 | Particle system | The set number of particles are correctly created |  |  |
| 8 | Particle system | The particles are set to an inactive location that cannot be accessed by the player |  |  |
| 9 | Particle system | The particles are transformed as per the defined parameters |  |  |
| 10 | Particle system | The particles are set to inactive once their duration has maxed out |  |  |
| 11 | Player movement | The movement direction matches that of the input |  |  |
| 12 | Player movement | Collision detection accurately detects collisions |  |  |
| 13 | Player movement | The direction vector is reset after moving/at the end of the frame |  |  |
| 14 | Player jumping | The jump counter counts when jump is pressed |  |  |
| 15 | Player jumping | The resulting jump vector is in an upwards direction |  |  |
| 16 | Player attack | The direction of the fireball is equal to the facing of the player |  |  |
| 17 | Player health | The player’s health value decreases by the determined amount |  |  |
| 18 | Player respawn | The player respawns in the correct position |  |  |

Black Box Testing - Alex

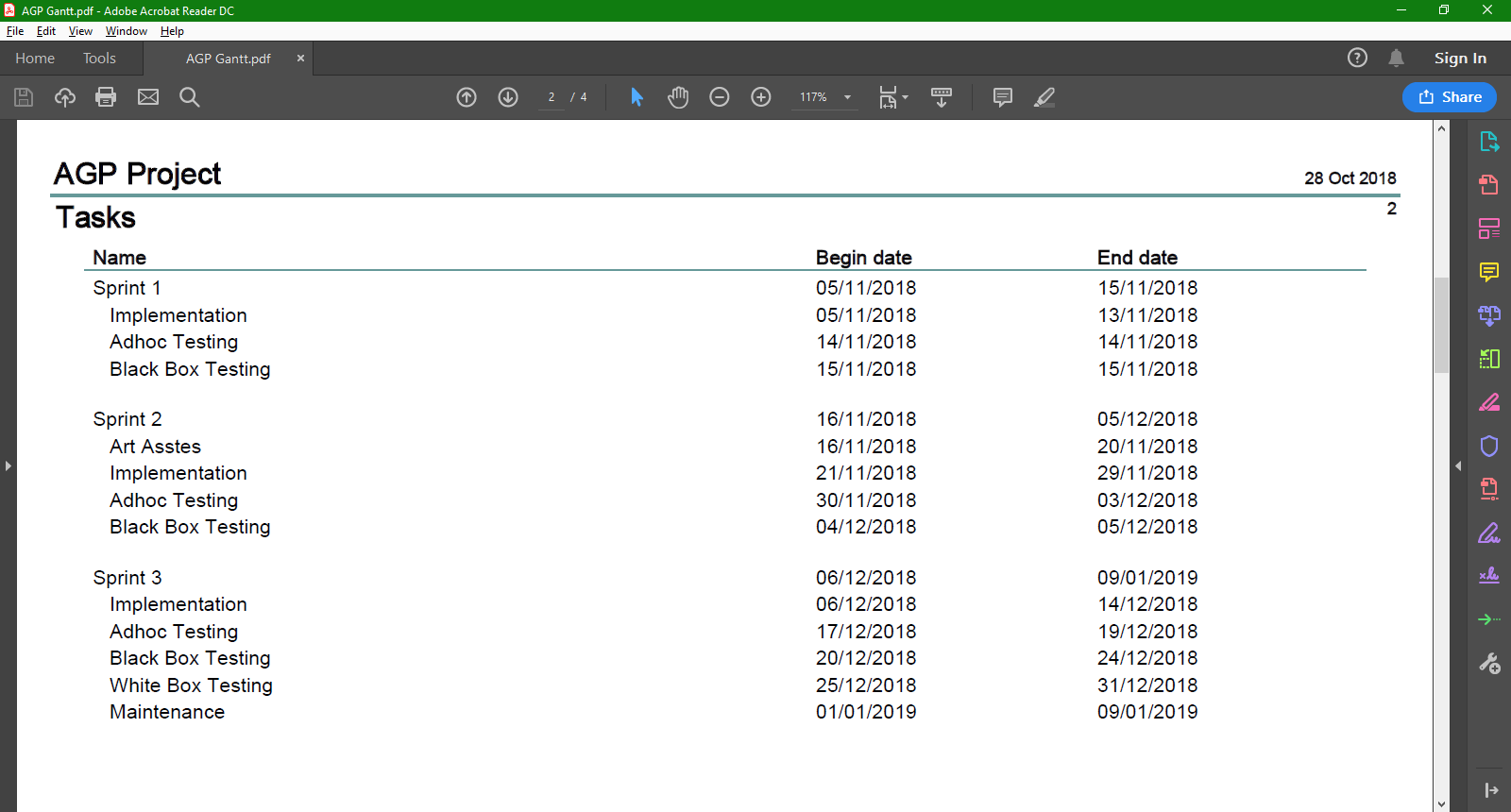
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Feature** | **Test** | **Expected Result** | **Actual Result** | **Action** |
| 1 | Player movement | Using the “wasd” keys to move the player | The player moves around relative the used input |  |  |
| 2 | Player looking | Using the mouse to look around | The player looks around relative to the mouse input |  |  |
| 3 | Player throwing fireball | Using the action button to shoot a fireball | A fireball is thrown forwards and mana is spent |  |  |
| 4 | Player picking up mana | Walking over a mana pickup | Mana is replenished if it is missing |  |  |
| 5 | Player losing health | Getting hit by an enemy fireball | The player will lose health |  |  |
| 6 | Player beats the level | Reaching the end of the level | The level will end, and the player will have won |  |  |
| 7 | Player jumping | Pressing the jump button | The player will jump up and then fall back down, due to gravity |  |  |
| 8 | Player respawning | Dying | The player will respawn at the beginning of the level |  |  |
| 9 | Enemy movement | Watch the enemies | The enemies will follow appropriate patrol paths |  |  |
| 10 | Enemy attacks | Approach the enemies to make them attack | The enemies will throw fireballs at the player when in range |  |  |
| 11 | Enemy dying | Throw fireballs at the enemies | The enemies will die |  |  |
| 12 | Lava kills player and enemies | Jump into the lava and knock enemies into the lava | The player/enemies will die over time |  |  |
| 13 | Mana bar accurately represents mana | Throw fireballs to use mana | The mana bar will decrease at a set rate per thrown fireball |  |  |
| 14 | The lava particles properly appear | Watch the lava | The particles will spray up from the lava |  |  |
| 15 | Rocks in the level can be pushed around | Walk up to rocks | The rocks will move around and not impede player movement |  |  |
| 16 | The game can be paused | Press the pause button | The game will pause |  |  |
| 17 | The game can be resumed | Press the resume button | The game will unpause and resume |  |  |
| 18 | The game can be quit to main menu | Press the quit button | The game will return to the main menu |  |  |
| 19 | The level can be restarted | Press the restart button | The level will begin again, and the game will unpause |  |  |
| 20 | The game can be started from the main menu | Press the start button | The game will start |  |  |
| 21 | The game can be quit from the main menu | Press the quit button | The game will exit |  |  |

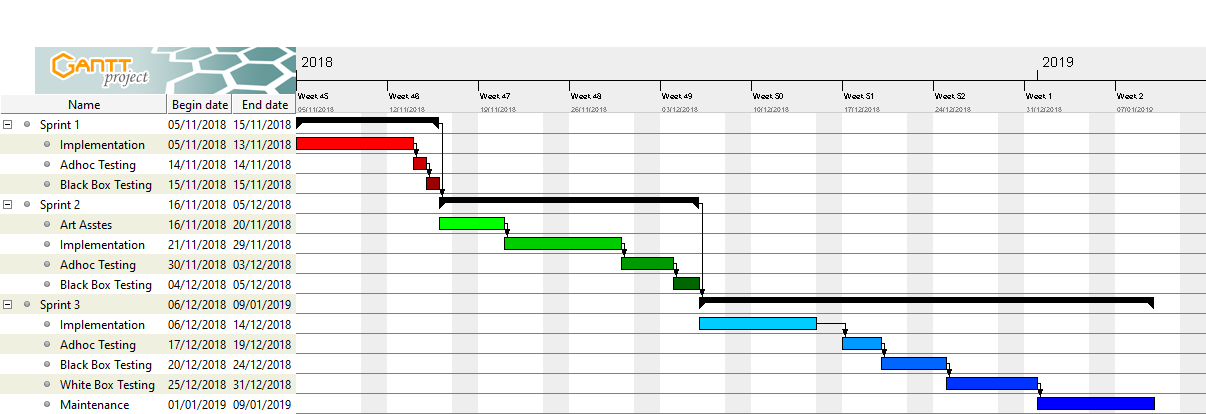
WBS – David



Task List - Alex

Gantt Chart – David





Who did what?

* User Stories – David Mincer/Alex Pearson
* Class Diagram – David Mincer
* Flowcharts/Pseudocode
  + Player Movement – David Mincer
  + Player Jumping – David Mincer
  + Player Looking – David Mincer
  + Player Throwing Fireball – David Mincer
  + Player Picking Up Mana – David Mincer
  + Player Losing Health – David Mincer
  + Player Reaching Level End – David Mincer
  + Player Respawning – David Mincer
  + Player Quitting - David Mincer
  + Enemy Move/Attack – Alex Pearson
  + Mana Bar – Alex Pearson
  + Particle System – Alex Pearson
  + Menus – Alex Pearson
* White Box Testing – Alex Pearson
* Black Box Testing – Alex Pearson
* WBS – David Mincer
* Task List – Alex Pearson
* Gantt Chart – David Mincer