Log Book

14/2/19
Progress:

Game was started in Defold using Lua as coding language.
Game currently contains a simple background image (map) and sprite (placeholder)
Sprite can move up, down, left, right and has working walking animations.

Problems:

Whenever movement keys are pressed (W, A, S, D) the sprite will move while the walking animations will pause causing sprite animations to freeze while moving.
Solution was found. Added condition where key press had to be true allowing sprite to move onscreen without paused animations.

23/2/19
Progress:

Tile set and hero sprite changed.
Sprite movement now done through physics.
Player collision with the map created through kinematic (player) and static objects (walls) and collision detection.
Basic enemy behaviour created through random movements determined through random number generator.
Music capabilities implemented.
Found out how to delete game objects e.g. (enemies)

Problems:

Don't know how to 'attack' the enemy and how to detect an attack from the player.

25/2/19
Progress:

Found out how to attack 'enemies' through the use of creating short lived objects detecting collisions when attack animation is fired.
Able to determine locations of the 'lasers' through use of vectors.

Problems:

Don't know how to attack enemies besides from the one originally spawned

5/3/19
Progress:

Able to now attack enemies separately and differentiate between different enemy objects.

Problems:

Unable to treat enemy 'health' as separate between each entity spawned. (All enemies taking damage when hit as they are treated as the same entity).
Unable to replay enemy 'idle' animation after 'damage' animation has finished

6/3/19
Progress:

Able to spawn enemies on game load
Enemy health property now treated separately
Implemented score which is updated with every enemy defeated
Implemented player health which is updated every time damage is taken
Enemy Idle animations now play after damage animation has completed
Enemies can now attack the player

12/3/19
Progress:

Fonts and colours used for health and score display updated.

Problems:

Creating a cooldown/interval on player attack.

21/3/19
Progress:

Player now has attack cooldown set between each attack.

Problems:

Preventing player movement when in attack animation

22/3/19
Progress:

Now able to prevent player movement until attack animation completes using an interval timestamp

29/3/19
Progress:

Fixed up spawning of enemies (locations now randomly generated)

14/5/19
Progress:

Started working on boss script

16/5/19
Progress:

Added mechanic where player will speed up slightly after attacking an enemy

22/5/19
Progress:

Altered mechanic where player sped up after attacking an enemy. Player now only gains an increase of speed when the enemy is defeated. (4 seconds).

25/5/19

Progress:

Added mechanic where when player loses health, text node will appear above player noted with amount of health points lost. This was done through a label node attached to the character game object.

Problems:

Unable to recreate above mechanic with enemies.

26/5/19

Progress:

Changed mechanic used to reveal damage received in both enemies and player. Now done using factories (previously done with labels) which spawn text nodes (signifying amount of health lost) when an entity receives damage. Red text used for when player takes damage and white for enemies.

29/5/19

Progress:

Player can now trigger critical attacks where damage is doubled (Currently set to a 1 in 10 chance).

30/5/19

Progress:

Added sound effects (slime enemy hit sound, player hit sound, critical hit sound).

5/6/19

Progress:

Worked on boss script, boss now takes damage, attacks and gives score when defeated.

19/6/19

Progress:

Created Boss Fight Scenario triggered when enemy slimes reach 0
Created a GUI text node noting the current objective the player is on

26/6/19

Progress:

Added bullet mechanic where player can shoot a bullet to attack enemies

27/6/19

Progress:

Slime Boss can now move and spawns a slime each time it is attacked.

12/7/19

Progress:

Added mechanic where player can fire bullets on a 2 second countdown.

13/7/19

Progress:

New enemy types/sprites added
A level 2 stage was added
Goblin enemy chosen to be used for level 2.

14/7/19

Progress:

Goblin enemy attacking and walking animations implemented
Level 2 stage completed
Boss stage for level 2 completed
Ability for game retry implemented when game over scenario is reached
All GUI text nodes updated for level 2 stage
Fixed error in bullet interval timings being affected by timestamp used for player attacks.

17/7/19

Progress:

Errors in replaying game were fixed. (Problem where game would not restart and spawn enemies correctly.)
Fixed errors in the players and goblin boss bullet firing, where the determined bullet direction was being miscalculated.
Goblin Boss fight errors fixed
Goblin Boss fight teleporting mechanic added.

23/7/19

Progress:

Level 3 stage completed
Troll enemy used for level 3 and all enemy animations and functions implemented.
Level 3 Boss completed
Laser and sound effects added
Added text nodes that display controls to player before starting the game

Data Dictionary -

Data Item	Data Type	Format	Size for display	Description	Example	Validation
health	Integer	NNN	1	Amount of player health	100	100 or 0
Score	Integer	NNNN	4	Amount of score	835	0 or above
Speed	Integer	NN	2	Player Speed	55	55 or 65
Objective Text node	String			Text node displaying current objective	“Objective: Defeat all slimes”	
Coins	Integer	NN	2	Coin amount player has	10	0 or above
Background Music	Audio file			Audio file for background music	Music.wav	
Coin sound effect	Audio file			Audio file for coin pick up sound	ding.wav	
Player hit sound effect	Audio file			Audio file for player hit	playerHit.wav	
Laser sound effect	Audio File			Audio file for laser sound effect	laser.wav	
Critical hit sound effect	Audio File			Audio file for critical hit sound effect	criticalHit.wav	
Final Score	Integer	NNNN	4	Number used to determine final player score	1820	
Hero image	Image			Image containing hero sprite	Hero.png	
healthSize.x	Integer	NNN	3	Size value determining the length of charcter health size box	80	0 or above

Slime Count	Integer	NN	2	Integer value keeping track of amount of spawned and alive ‘slime’ enemies	6	0 or above
Slime Boss Count	Integer	N	1	Integer value keeping track of amount of spawned and alive ‘slimeBoss’ enemies	1	0 or 1
Goblin Melee Count	Integer	NN	2	Integer value keeping track of amount of spawned and alive ‘goblinMelee’ enemies	9	0 or above
Goblin Melee Boss Count	Integer	N	1	Integer value keeping track of amount of spawned and alive ‘goblinMeleeBoss’ enemies	1	0 or above
Troll Count	Integer	NN	2	Integer value keeping track of amount of spawned and alive ‘troll’ enemies	9	0 or above
trollBossCount	integer	N	1	Integer value keeping track of amount of spawned and alive ‘trollBoss’ enemies	1	0 or 1

Algorithms

Character Game Object

Set GO property “attack_interval” = 0.70
Set GO property “walk_interval” = 0.25
Set GO property “is_alive” = true

```
FUNCTION Init (occurs on initiation of object)
        Acquire input focus
        Play looping background music
        scale = 1.25,1.25,1
        Set game object Scale to ‘scale’
        Self.vel = vmath.vector3()
        self.timestamp = 0
        self.fireTimestamp = 0
        heroDmgAmount = 10
        playerDirection = “down"
        health = 0
        speed = 55
        score = 0
        coins = 0
        level2 = false
        gameOver = true
ENDFUNCTION

FUNCTION update (occurs every game tick)
        Get current player position
        Player position = Current player position + Current Velocity * Every game tick
        Set position of ‘Player’ game object
        Player ‘x’ velocity = 0
        Player ‘y’ velocity = 0

        IF time since last bullet fire > 1.5 then
                canFire = true
        ELSE
                canFire = false
        ENDIF
        IF Time since last player attack < 4 then
                canTeleport = false
        ELSEIF Time since last player attack > 4 then
                canTeleport = true
                speed = 55
        ENDIF

        IF health <= 0 then
                canMove = false
```

```
            gameOver = true
            Set GO property “is_alive” = false
            Set player position to (517, -320, 0.1)
            Set position of ‘Player’ game object
        ELSEIF health >= 0 and slimeCount > 0 and gameOver = true then
            canMove = true
            slimeBossEncounter = false
            goblinMeleeBossEncounter = false
            goblinMeleeBossRoom = false
            level2 = false
            gameOver = false
            Set GO property “is_alive” = true
        ENDIF

        IF gameOver = true then
            Send Message “game_final” to Interface GUI script
        ELSEIF gameOver == false then
            Send Message “game_current” to Interface GUI script
        ENDIF
END FUNCTION

FUNCTION on_message(self, message_id, message, sender)
        IF message received is “gameFinish” then
            gameOver = true
            canMove = false
            Set player position to (-517, -320, 0.1)
            Set position of ‘Player’ game object
            Set GO property “is_alive” = false
            Send message “game_final” to interface GUI
        ENDIF

        IF slimeCount = 0 and Time since last player attack > 4 and slimeBossEncounter =
        false then
            Set player position to (110,600,0.05)
            Set position of ‘Player’ game object
            slimeBossEncounter = true
            Send message “slimeBossEncounter” to interface GUI
        ELSEIF goblinMeleeCount == 0 and level2 == true and Time since last player attack >
        4 and goblinMeleeBossRoom == false then
            Set player position to (1080,-400,0.05)
            Set position of ‘Player’ game object
            goblinMeleeBossRoom = true
            FOR goblinMeleeSpawn = 1,5 do
                Spawn goblin enemy in a random position within goblinBossRoom
            NEXT
        ELSEIF goblinMeleeBossCount == 0 and goblinMeleeBossEncounter == true and Time
        since last player attack > 4 then
```

```
            Set player position to (1080,400,0.1)
            Set position of 'Player' game object
            health = 100
            Send message "update_health" to interface GUI
            goblinMeleeBossEncounter = false
        ENDIF

        IF message received is "spawnSlime" then
            Spawn slime enemy in a random area within slimeBossRoom
        ENDIF

        IF Message received is "slimeBossDeath" then
            FOR slimeSpawn = 1,2 do
                slimeSpawn = Random area within SlimeBossRoom
                Spawn Slime at 'slimeSpawn'
            NEXT
        ENDIF

        IF Message received is "contact_point_response" then
            Set current player position to  (Current player position  + Direction of contact
            * Distance from collision)
        ENDIF

        IF Message received is "damageHero" or "damage" then
            Send Message "play_sound" to "/hero#playerHit"
            Display Text showing amount of health lost above player
            health = health - slimeDmgAmount
            Play animation "upDamage"
            Send Message "hero_damage" to interface GUI
        ENDIF

        IF Message received is "speed" then
            speed = 65
        ELSEIF Message received is "level2" and level2 = false then
            Set player position to (120, -180, 0.5)
            Set position of 'Player' game object
            level2 = true
            health = 100
            Send message "update_health" to interface GUI
            Send message "startLevel2" to map script
        ENDIF
ENDFUNCTION

FUNCTION on_input(self, action_id, action)
        IF health <= 0 and message received is "restart_game" then
            Set GO property "is_alive" = true
            health = 100
```

```
        Send message “update_health” to interface GUI
        Send message “retry” to interface GUI
        Send message “restart_game” to map script
        Set player position to (115,275,0.1)
        Set position of ‘Player’ game object
        score = 0
        coins = 0
        Send message “update_score” to interface GUI
        Send message “update_coins” to interface GUI
ENDIF

IF Input message received is "bullet" and Action key pressed down = true and canFire
= true then
        canFire = false
        self.fireTimestamp = OS Time
        IF playerDirection == "right" then
                Create bullet object to the right of the player
        ELSEIF playerDirection == "left" then
                Create bullet object to the left of the player
        ELSEIF playerDirection == "up" then
                Create bullet object above the player
        ELSEIF playerDirection == "down" then
                Create bullet object below the player
        ENDIF
ENDIF

IF OS Time > (self.timestamp + self.walk_interval) and canMove == true then
        IF Input message received is "move_up" then
                playerDirection = "up"
                Player ‘y’ velocity = speed
                IF Action key pressed down = true then
                        Play up walking animation
                ENDIF
        ELSEIF Input message received is “move_down” then
                playerDirection = "down"
                Player ‘y’ velocity = -speed
                IF Action key pressed down = true then
                        Play down walking animation
                ENDIF
        ELSEIF Input message received is “move_right” then
                playerDirection = "right"
                Player ‘x’ velocity = speed
                IF Action key pressed down = true then
                        Play right walking animation
                ENDIF
        ELSEIF Input message received is “move_left” then
                playerDirection = "left"
```

```
                    Player 'x' velocity = -speed
                    IF Action key pressed down = true then
                        Play left walking animation
                    ENDIF
                ENDIF
            ENDIF

            IF OS Time > (self.timestamp + self.attack_interval) then
                IF Input message received is "attack_right" then
                    self.timestamp = OS Time
                    Play right attack animation
                    Create attack object to the right of the player
                ELSEIF Input message received is "attack_left" then
                    self.timestamp = OS Time
                    Play left attack animation
                    Create attack object to the left of the player
                ELSEIF Input message received is "attack_up" then
                    self.timestamp = socket.gettime()
                    Play up attack animation
                    Create attack object above the player
                ELSEIF Input message received is "attack_down" then
                    self.timestamp = socket.gettime()
                    Play down attack animation
                    Create attack object below the player
                ENDIF
            ENDIF
```

Slime Game Object

```
FUNCTION init(self)
    Acquire input focus
    self.vel = vmath.vector3()
    self.slimeHealth = 20
    slimeDmgAmount = 10
    slimeCount = slimeCount + 1
    self.deathTimestamp = 0
ENDFUNCTION

FUNCTION final(self)
    score = score + random integer between 1 and 5
    Send message "update_score" to interface GUI
    slimeCount = slimeCount - 1
ENDFUNCTION
```

```
FUNCTION update (self, dt)
    IF GO property “is_alive” = false then
        Delete GO
    ENDIF

    IF self.slimeHealth > 0 then
        self.deathTimestamp = OS Time
    ELSE
        self.deathTimestamp = self.deathTimestamp
    ENDIF

    IF Time since death of GO is > 0.5 then
        Play slime death sound
        Send message “speed” to player
        Spawn coin at current GO position
        Delete GO
    ENDIF

    Get current player position
    Player position = Current player position + Current player velocity * Every game tick
    Set position of ‘Slime’ game object
    local movement = Random integer between 1 and 500
    IF movement = 1 then
        Player ‘y’ velocity = 10
    ELSEIF movement = 2 then
        Player ‘y’ velocity = -10
    ELSEIF movement = 3 then
        Player ‘x’ velocity = 10
    ELSEIF movement = 4 then
        Player ‘x’ velocity = -10
    ELSEIF movement = 20 then
        Play right attack animation
        Create attack object to the right of current position
    ELSEIF movement = 40 then
        Play left attack animation
        Create attack object to the left of current position
    ELSEIF movement == 60 then
        Play up attack animation
        Create attack object above current position
    ELSEIF movement == 80 then
        Play down attack animation
        Create attack object below current position
    ENDIF
ENDFUNCTION
```

```
FUNCTION on_message (self, message_id, message, sender)
        IF Message received is “contact_point_response” then
                Set current position to  (Current position  + Direction of contact * Distance
                from collision)
        ENDIF

        IF Message received is “damage” then
                Display text node showing health lost above current position
                Play upDamage animation
                self.slimeHealth = self.slimeHealth - heroDmgAmount
                Play slimeHit sound
        ELSEIF Message received is “animation_done” then
                Play idle animation
        ENDIF
ENDFUNCTION
```

Bullet Script

```
FUNCTION init(self)
        Scale =  0.1,0.3,1
        Set GO scale to ‘Scale’
        self.vel = vmath.vector3()
        Play laser sound
        local bulletSpeed = 250
        IF playerDirection = "down" then
                Bullet ‘y’ velocity = -bulletSpeed
        ELSEIF playerDirection = "up" then
                Bullet ‘y’ velocity = bulletSpeed
        ELSEIF playerDirection =="left" then
                Bullet ‘x’ velocity = -bulletSpeed
        ELSEIF playerDirection = "right" then
                Bullet ‘x’ velocity = bulletSpeed
        ENDIF
ENDFUNCTION

FUNCTION final(self)
        Play laser sound
ENDFUNCTION

FUNCTION update(self, dt)
        Get current GO position
        Current position = Current position + Current Velocity * Every game tick
        Set position of ‘bullet’ GO
ENDFUNCTION
```

```
FUNCTION on_message(self, message_id, message, sender)
        IF Message received is “contact_point_response” then
                enemy = Id of other object within collision
                Send message “damage” to ‘enemy’
                Delete GO
        ENDIF
ENDFUNCTION
```

Score GUI script

```
FUNCTION init(self)
        scoreText = Text node “scoreText”
        heroHealth = Text node “heroHealth”
        coinsScore = Text node “coinsScore”
        heroHealthBox = Box node “heroHealthBox”
        objective =  Text node “objective”
        gameOverText = Text node “gameOverText”
        controlTutorial = Text node “controlTutorial”
ENDFUNCTION

FUNCTION update(self, dt)
        IF slimeBossCount == 0 and slimeCount < 1 and slimeBossEncounter == true and
        canTeleport == true and level2 == false then
                Set objective text node to display “Objective: Defeat All Goblins"
                Send message “level2” to player script
        ENDIF
        IF health <= 0 or gameOver == true then
                Set objective text node to display “Press Enter to Start or Retry”
        ENDIF
ENDFUNCTION

FUNCTION on_message(self, message_id, message, sender)
        IF Message received is “game_final” then
                Set gameOverText node to display “"Final Score: " .. score”
                Enable text node controlTutorial
        ELSEIF Message received is “game_current” then
                Set text node gameOverText to display “ “
                Disable text node controlTutorial
        ENDIF

        IF Message received is “slimeBossEncounter” then
                Set objective text node to display “Objective: Defeat Slime Boss”
        ELSEIF slimeCount > 1 or Message received is “retry” then
                Set objective text node to display “Objective: Defeat All Slimes”
        ELSEIF Message received is “goblinMeleeBossEncounter” then
                Set objective text to display “Objective: Defeat Goblin Boss”
        ELSEIF goblinMeleeCount > 1 then
```

```
            Set objective text node to display “Objective: Defeat All Goblins”
        ELSEIF trollCount > 1 and goblinMeleeBossCount == 0 then
            Set objective text node to display “Objective: Defeat All Trolls”
        ENDIF

        IF Message received is “update_coins” then
            Set scoreText node to display "Score: " .. score
        ELSEIF Message received is “hero_damage” then
            Get heroHealthBox size
            Reduce heroHealthBox.x by 10
            Set heroHealthBox size
            Set heroHealth text node to display “Health: “
        ELSEIF Message received is “update_health” then
            Get heroHealthBox size
            healthSize.x = 100
            Set heroHealthBox size
            Set heroHealth text node to display “Health: “
        ELSEIF Message received is “update_coins” then
            Set coinsScore text node to display "Coins: " .. coins
        ENDIF
ENDFUNCTION
```

User Manual

How to play:

Interface –
The Health bar displays how much health the player currently and it found on the top left corner of the screen.

The left image shows the player on full health and the right image shows the player with 0 health. When the player reaches 0 health the game ends and the end game screen is shown.

When the game ends or before the game starts the final score text node displays the final score of the previous game or 0 if the game hasn’t been played yet.

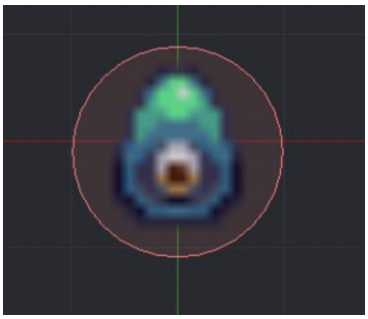
The score text node displays the players accumulative score throughout each play through of the game and is reset when the game restarts.

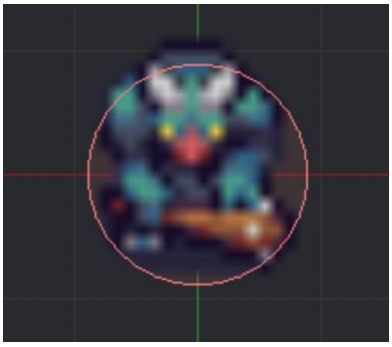
The objective text node displays the next objective the player has to meet to progress within the game. The objective text node is found at the top centre of the screen.

The Coins text node displays the players current coin count and increases as the player picks up coins in the game. The coin amount contributes to the final calculated score and is reset to 0 when the game is restarted. The coin text node is found on the top right corner of the screen.

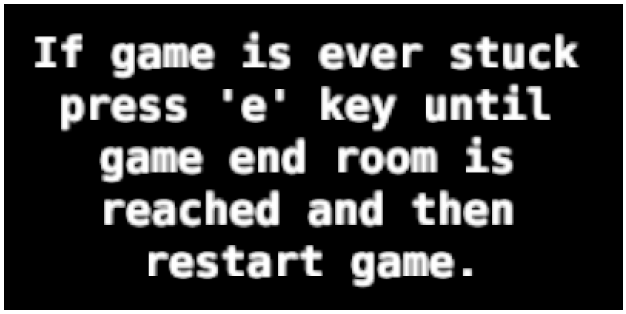
The controls text node shows the available actions the player can do and the respective keys to execute them. The control text node is enabled and shown at the start of the game and on the game over screen, it is disabled when the game is in play. The control text node when enabled is shown on the left side of the screen.

Player and Enemies –





The left image shows the character sprite and the middle and right images show examples of enemy sprites throughout the game. The aim of the game is to defeat these enemies and progress through the game until the final boss is defeated and a final score is calculated.



The above text node displayed before the game starts and during the game over screen, reveals instructions that the player can conduct if errors occur within the game and the player cannot progress further within the game. As seen within the image above if the player is ever stuck within the game the 'e' key should be pressed until the game over room is reached and then press the 'enter' key to restart the game. The text node is found on the right side of the screen when enabled.

Installation Guide

To open Mac application -
To play game once Mac application is installed/downloaded, double click application and then follow on screen instructions to play.

To install Defold –

On Mac –
The downloaded file is a DMG image containing the program.
1. Locate the file "Defold-x86_64-darwin.dmg" and double click it to open the image.
2. Drag the application "Defold" to the "Applications" folder link.
To start the editor, open your "Applications" folder and double click the file "Defold".

On Windows –
The downloaded file is a ZIP archive that needs to be extracted:

1. Locate the archive file "Defold-x86_64-win32.zip" (or "Defold-x86-win32.zip" for 32 bit Windows), press and hold (or right-click) the folder, select *Extract All*, and then follow the instructions to extract the archive to a folder named "Defold".

2. Move the folder "Defold" to "C:\Program Files (86x)"

To start the editor, open the folder "Defold" and double click the file "Defold.exe"

To open the project in Defold

1. Open Defold
2. Under the home tab found on the left side of the application, click on "Open from Disk…".
3. Locate the project folder on computer and open it.
4. Once opened locate file name "game.project" and click open it.
5. Defold will then load the project file.

Troubleshooting Guide

Problem	Fix
Enemy game object glitches out and leaves the room and the game cannot progress as player can no longer reach it and defeat it.	Press the 'e' button on the keyboard until game over room is reached and game can be reset.

Game does not progress even after objective has been achieved	Press the ‘e’ button on the keyboard until game over room is reached and game can be reset.
When in boss room and all enemies have been defeated but boss doesn’t spawn.	Press the ‘e’ button on the keyboard until game over room is reached and game can be reset.
If game freezes or does not load	Restart application
If player is stuck in the wall or has left the play zone	Press the ‘e’ button on the keyboard until game over room is reached and game can be reset.

```
if slimeCount == 0 and (socket.gettime() - self.timestamp) > 4 and slimeBossEncounter == false then
		p = vmath.vector3(110,600,0.05)
		go.set_position(p)
		slimeBossEncounter = true
		msg.post("/interface#gui", "slimeBossEncounter")
	elseif goblinMeleeCount == 0 and level2 == true and (socket.gettime() - self.timestamp) > 4 and goblinMeleeBossRoom == false then
		p = vmath.vector3(1080,-400,0.05)
		go.set_position(p)
		goblinMeleeBossRoom = true
		for goblinMeleeSpawn = 1,5 do
			goblinMeleeSpawn = vmath.vector3(math.random(1000,1200),math.random(-420,-200), 0.5)
			factory.create("/map#goblinMeleeFactory", goblinMeleeSpawn)
		end
	elseif goblinMeleeBossCount == 0 and goblinMeleeBossEncounter == true and (socket.gettime() - self.timestamp) > 4 then
		p = vmath.vector3(1080,400,0.1)
		go.set_position(p)
		health = 100
		msg.post("/interface#gui", "update_health")
		goblinMeleeBossEncounter = false
	end
```

Test Data

slimeCount == 0 , (socket.gettime() - self.timestamp) > 4 , slimeBossEncounter == false ,

Desk Check

Line	p	slimeBossEncounter	Output

1	p = vmath.vector3(110,600,0.05)		New player position is set
2			Sets player position to new location
3		slimeBossEncounter = true	
4			Sends message “slimeBossEncounter” to interface GUI script
5			

Test Data 2

goblinMeleeCount == 0 , level2 = true , (socket.gettime() - self.timestamp) > 4 , goblinMeleeBossRoom == false

Desk Check 2

Line	P	goblinMeleeBossRoom	goblinMeleeSpawn	Output
1	p = vmath.vector3(1080,-400,0.05)			New player position is set
2				Set player position to new location
3		goblinMeleeBossRoom = true		
4			1	
5			goblinMeleeSpawn = vmath.vector3(math.random(1000,1200),math.random(-420,-200), 0.5)	
6				Spawns goblin enemy at goblinMeleeSpawn

7			2	
8			goblinMeleeSpawn = vmath.vector3(math.random(1000,1200),math.random(-420,-200), 0.5)	
9				Spawns goblin enemy at goblinMeleeSpawn
10			3	
11			goblinMeleeSpawn = vmath.vector3(math.random(1000,1200),math.random(-420,-200), 0.5)	
12				Spawns goblin enemy at goblinMeleeSpawn
13			4	
14			goblinMeleeSpawn = vmath.vector3(math.random(1000,1200),math.random(-420,-200), 0.5)	
15				Spawns goblin enemy at goblinMeleeSpawn
16			5	
17			goblinMeleeSpawn = vmath.vector3(math.random(1000,1200),math.random(-420,-200), 0.5)	
18				Spawns goblin enemy at goblinMeleeSpawn

Test Data 3

goblinMeleeBossCount = 0 , goblinMeleeBossEncounter = true , (socket.gettime() - self.timestamp) > 4

Line	p	health	goblinMeleeBossEncounter	Output

1	p = vmath.vector3(1080,400,0.1)			
2				Set player position to new location
3		healt h = 100		
4				Message “update_health” is sent to interface GUI script
5			goblinMeleeBossEncounter = false	