CS 512 Assignment 2

The Bishops Bog Goblin (hack-and-slash RPG) based on `Unity3d`

Group Members:

`Yi Ren` (002269013) `Wentao Lu` (002276355)

Dependency

- Microsoft .NET Framework
- Unity3d
- Jetbrains Rider

Video demo

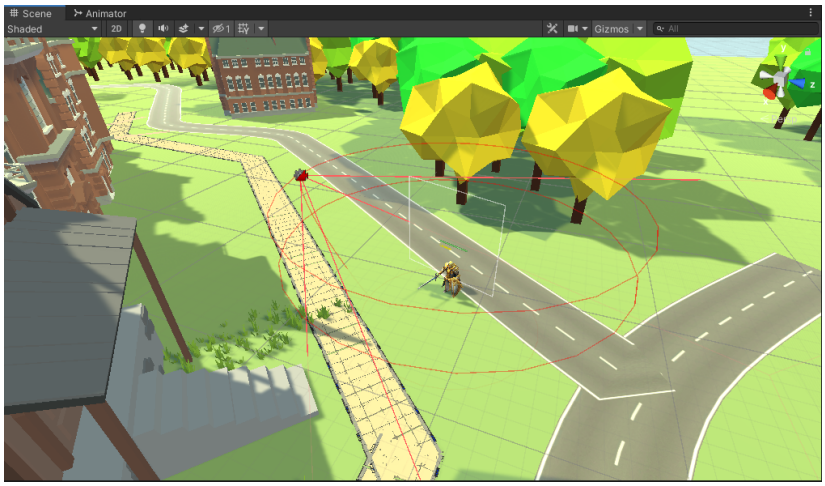
Please check Demo - Assignment2 for more information.

Basic functions

Part 1 - Basic movement and control

In this part, we built the 3rd person perspective camera with CinemaMachine, which is camera tool provieded by Unity for free. We can modify the parameters, such as axis and orbits in the Inspector to produce an appropriate 3rd camera. For the cursor part, we implemented the cursor with 3 candidate textures, which means the cursor will change it's icon when clicked on diffrent kind of objects. For example, the cursor will normally turns out to be green, and turn to yellow when clicked on terrain, turn to a sword shape icon when click on enemy.

- **a. A player controller that is viewed from 3rd person perspective camera**



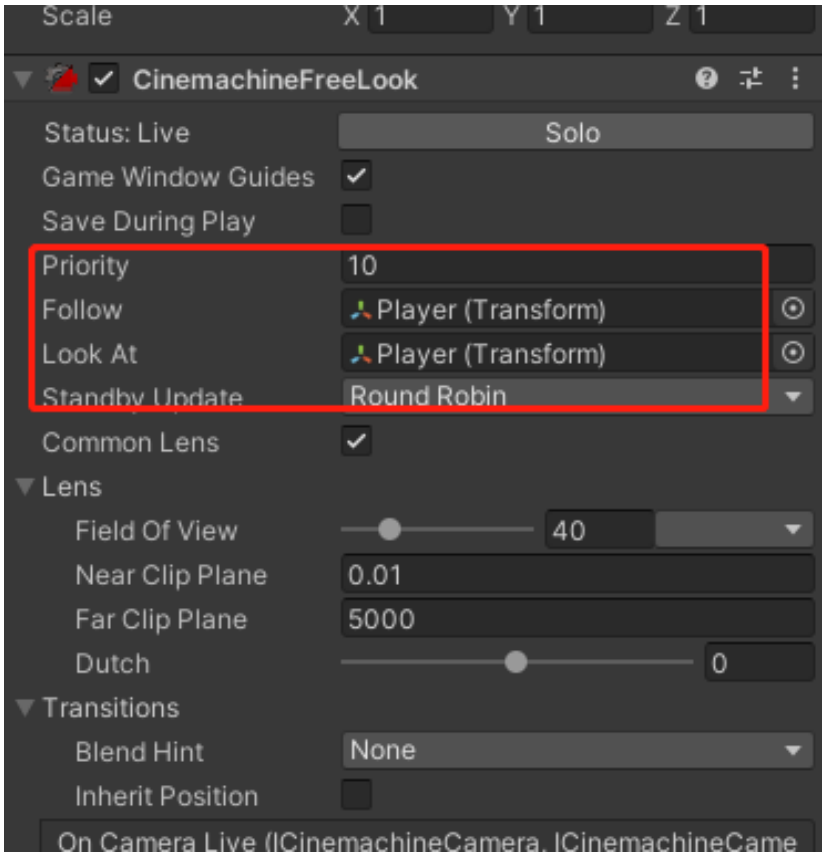
- **b. Click and hold down left mouse button to move player in that direction (youtube Diablo 3 for examples), Click on an enemy or NPC to walk towards it**

▶ Code

```
            if (Input.GetMouseButtonDown(0))
            {

                _mouse = Input.mousePosition;
                var castPoint = Camera.main.ScreenPointToRay(_mouse);
                RaycastHit hit;
                if (Physics.Raycast(castPoint, out hit, Mathf.Infinity))
                {
                    position = hit.point;
                    player.SetDestination(hit.point);
                    if (hit.collider.name.Contains("Terrain"))
                    {
                        Cursor.SetCursor(cursorTexture2, hotSpot,
CursorMode.ForceSoftware);
                        _isShowMouseInfo = true;
                    }

                    if (hit.collider.name.Contains("Goblin"))
                    {
                        Cursor.SetCursor(cursorTexture3, hotSpot,
CursorMode.ForceSoftware);
                    }

                    if (hit.collider.name.Contains("Lord"))
                    {
                        Cursor.SetCursor(cursorTexture2, hotSpot,
CursorMode.ForceSoftware);
                    }

                }
```

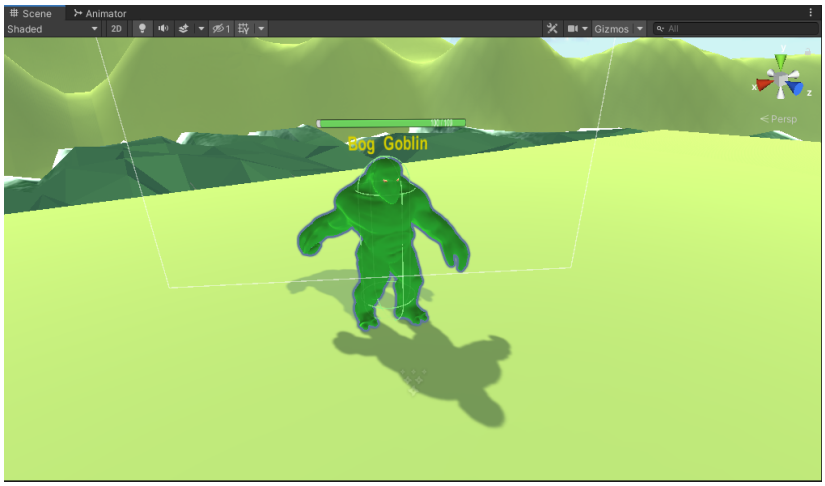
```
        private void MoveToCursor()
        {
            var playerPosition = transform.position;
            var distance = Vector3.Distance(playerPosition, _destination);
            var normalizedDirection = new Vector3((playerPosition.x -
_destination.x) * Time.deltaTime * _speed/ distance,0f,(playerPosition.z -
_destination.z)  * Time.deltaTime * _speed/ distance);
            // canvas.transform.Rotate(Quaternion.FromToRotation(Vector3.forward,
normalizedDirection).eulerAngles);
            transform.Translate(normalizedDirection);
            // ResetCanvasRotation();
        }
```

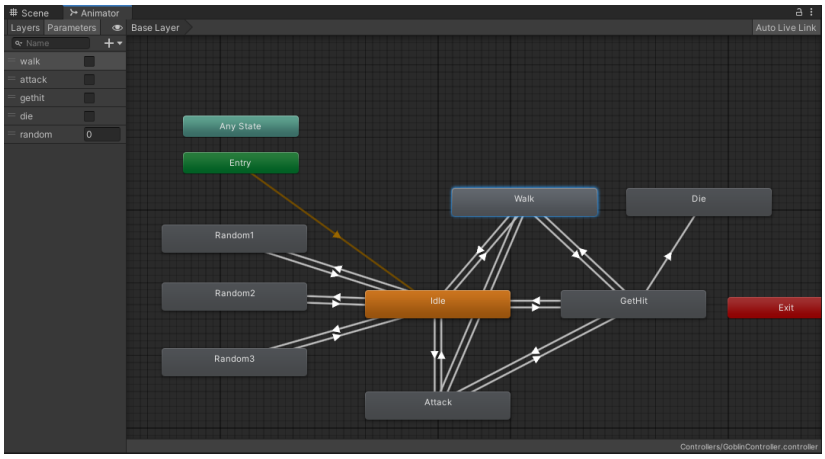
- **c. Camera follows the player when they move**



Part 2 - Combat

- **a. Enemies with an 'aggro range' – they will attack you if you move too close.**
  In this part, we selected a Monster model Asset from AssetStore, then Animator and Controller are applied to this Goblin.





▶ Code

```
        if (_range < 20 && _range > 0.1f)
        {
            _isIdle = false;
            GoblinAnimator.SetInteger(Random1, 0);
            GoblinAnimator.SetBool(Walk, true);
            MoveTowardsPlayer();
        }
        if (_range < 3.8f && _isAttack)
        {
            _isAttack = false;
            GoblinAnimator.SetBool(Walk,false);
            GoblinAnimator.SetBool(Attack,true);
            Invoke("AttackOnPlayer", 1.5f);
        }
```

```
    public void GetAttacked(float damage)
    {
        _isAttacked = true;
        _healthPoint -= Random.Range(damage, damage * 0.5f);
        slider.value = _healthPoint;
        Debug.Log(_healthPoint);
        textHealth.text = (int)_healthPoint + " / " + slider.maxValue;

        if (_healthPoint <= slider.maxValue * 0.3f)
            slider.gameObject.transform.Find("Fill
Area").Find("Fill").GetComponent<Image>().color = Color.red;
        if (_healthPoint <= 0)
        {
            // destroyAudio.Play();
            GoblinAnimator.SetBool(Attack,false);
            GoblinAnimator.SetBool("die",true);
            Lord.SetQuestStatus(true);
            TextGetter.SharedInstance.SetQuest("Misson completed!\nGo back to Lord
and get your reward!");
            Invoke("SetActiveFalse", 1.5f);
```

```
        }
    }
```

```
    private void AttackOnPlayer()
    {
        _isAttack = true;
        player.GetAttacked(5f);
    }
```

```
    private void MoveTowardsPlayer()
    {
        var playerPosition = player.transform.position;
        var position = transform.position;
        var distance = Vector3.Distance(position, playerPosition);
        var normalizedDirection = new Vector3((playerPosition.x - position.x) *
Time.fixedDeltaTime * MoveSpeed/ distance,0f,(playerPosition.z - position.z)  *
Time.fixedDeltaTime * MoveSpeed/ distance);
        // transform.Rotate(normalizedDirection);
        transform.rotation = Quaternion.Euler(normalizedDirection);
        // canvas.transform.Rotate(Quaternion.FromToRotation(Vector3.forward,
normalizedDirection).eulerAngles);
        transform.Translate(normalizedDirection);
        // ResetCanvasRotation();
    }
```

- **b. Click and hold on enemy to attack that enemy. the enemy should not overlap you when fighting.**
  In this part, we modified the collider of Goblin and Player so as to avoid overlap of models. Also, we listen to `Input.GetMouseButtonDown(0)` to monior the mouse input.

▶ Code

```
            if (Input.GetMouseButtonDown(0))
            {
                RaycastHit hit;
                var castPoint = Camera.main.ScreenPointToRay(Input.mousePosition);
                if (Physics.Raycast(castPoint, out hit, Mathf.Infinity))
                {
                    Debug.Log(hit.collider.name);
                    if (hit.collider.name.Contains("Goblin"))
                    {
                        _isAttack = true;
                    }
                }
            }
```

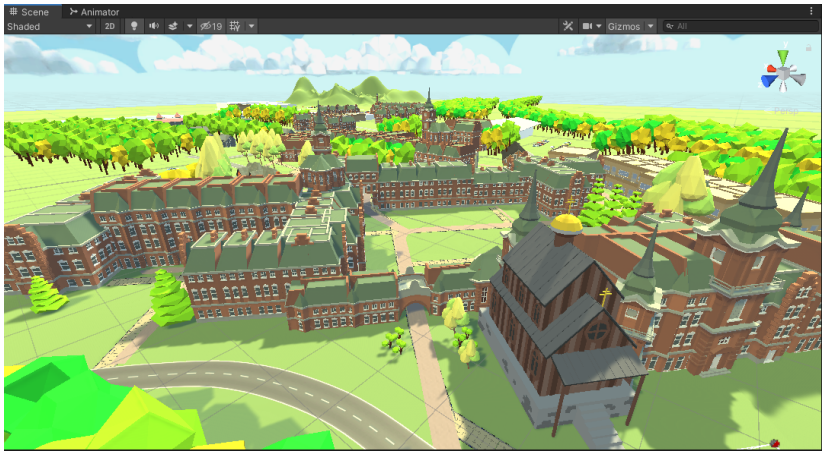
- **c. Melee attack and ranged attack that can be swapped using hotkey**
  In this part, we display melee and ranged attack with two different motions of Player. For the collision detection, we override the `OnCollisionEnter` function and detect the collision object by`collision.gameObject.name`. To specify the ranged attack, we provided Goblin with a particle system, which will start every time the Gobin is attacked by ranged attack. Finally, we listen to system input(Key R) to swap the melee and ranged attack.

▶ Code

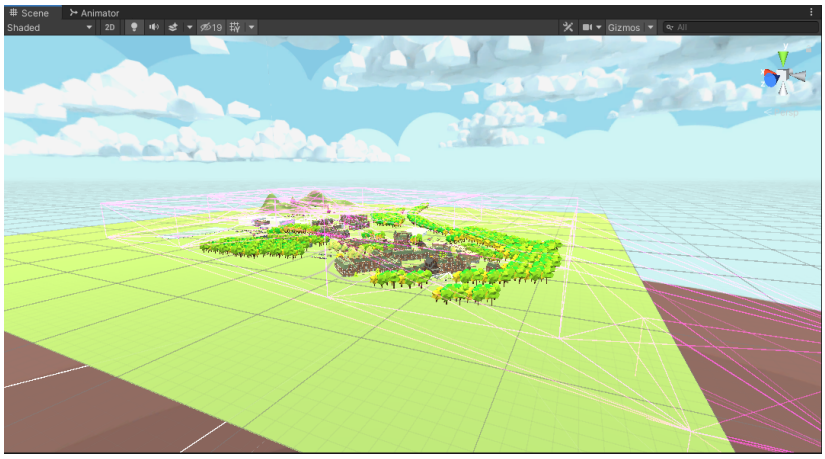
```
            if (Input.GetKeyDown(KeyCode.R))
            {
                Debug.Log("mode"+_attactMode);
                _attactMode = _attactMode == 1 ? 2 : 1;
                var mode = _attactMode == 1 ? "Melee" : "Range";
                AttackMode.text = "Attack mode: " + mode +"\n(Press R to swap
mode)";
            }
```

Part 3 - Content

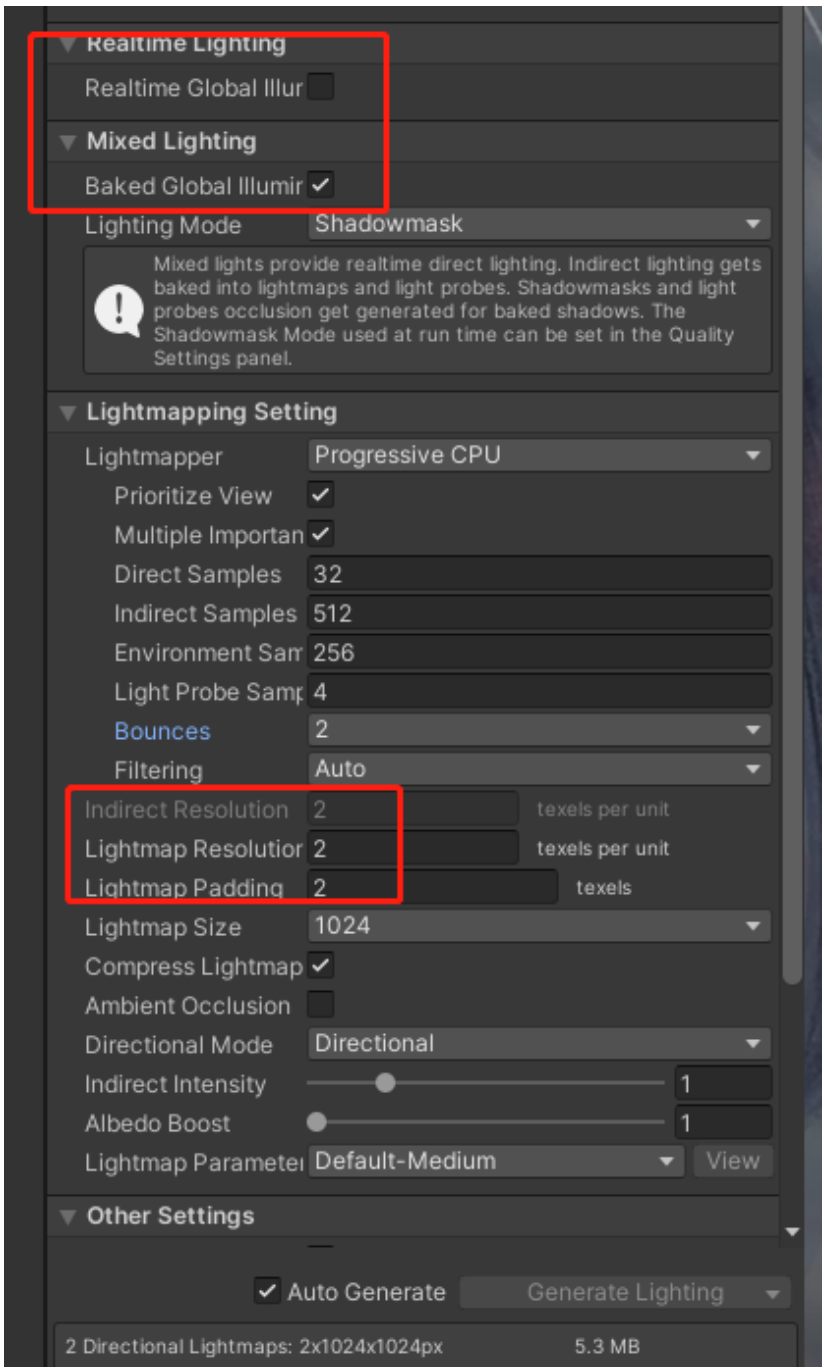
- **Level 1 - make a rough re-creation of the Bishops campus in the Google Maps photo above. Can add some scaled cubes to represent buildings, for example. Add baked lighting and light probes**
  `Campus`



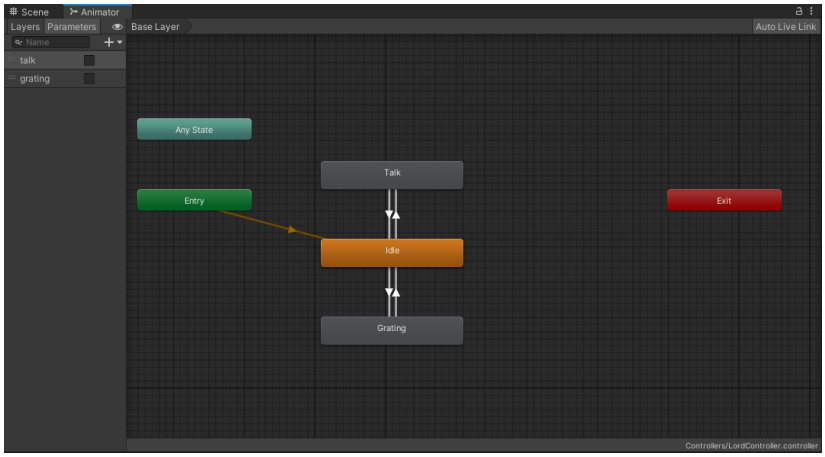
Light Probes



Baked lighting



- **b. A single NPC (Lord Goldbloom) that, if close to player, gives player instructions to kill the Bog Goblin**
  In this part, we implemented Lord Goldbloom with Asset from AssetStore, and applied some basic motions to her.



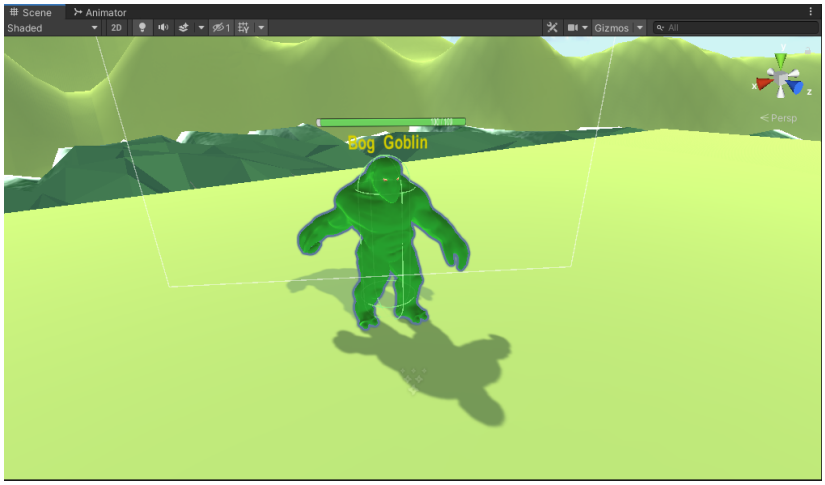
▶ Code

```
        if (_range > 20 && _setGreeting)
        {
            _setGreeting = false;
        }
```

```
        if (_range < 20)
        {
            if (!_setGreeting)
            {
                MainCanvas.SharedInstance.OnShowConversation("Lord Goldbloom:\n
Greetings human!" + "\nThe Bishops Bog Goblin, which lives behind the Athletics
center, has been terrorizing "
                    + "students and must be destroyed.");

                lordAnimator.SetBool(Talk, true);
                _setGreeting = true;
            }
            if (_isFinished && !_rewardAccept)
            {
```

```
                _rewardAccept = true;
                MainCanvas.SharedInstance.OnShowConversation("Lord Goldbloom:\n "
+ "Well done, warrior! Here's your reward!");
                TextGetter.SharedInstance.SetQuest("Your got the new title \"BU
Hero\" as reward!");
                playerTitle.text = "<BU Hero>   Berserker";
            }
        }
```

- **c. A Bog Goblin near the Marsh, that has some random behavior and attacks when you come too close.**



In this part, we implemented the random behavior with three selected motions, then randomly select one motion for every 3 seconds.

```
    public void RandomBehavior()
    {
        var r = (int) Random.Range(1, 4);
        if (_isIdle)
        {
            GoblinAnimator.SetInteger(Random1, r);
            _isIdle = false;
        }
        else
        {
            GoblinAnimator.SetInteger(Random1, 0);
            _isIdle = true;
        }
    }
```

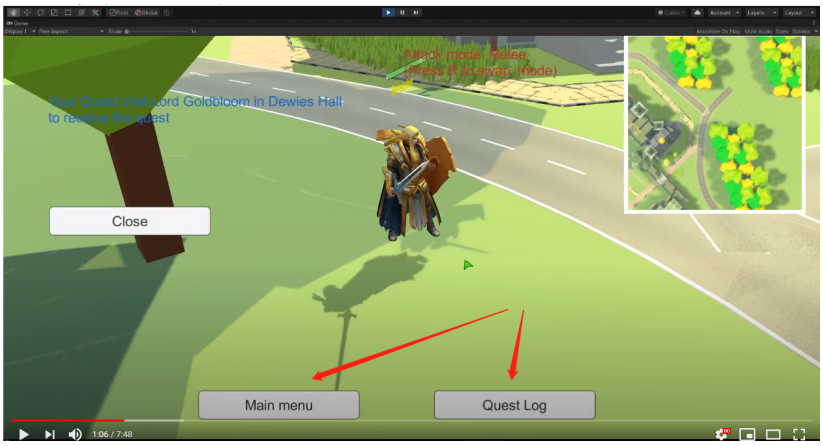
```
if (_range < 20 && _range > 0.1f)
        {
            _isIdle = false;
            GoblinAnimator.SetInteger(Random1, 0);
```

```
            GoblinAnimator.SetBool(Walk, true);
            MoveTowardsPlayer();
        }
```

- **d. Once the quest objectives are complete, return to NPC (Lord Goldbloom) to collect reward.**

Part 4 - Graphical User Interface (GUI)

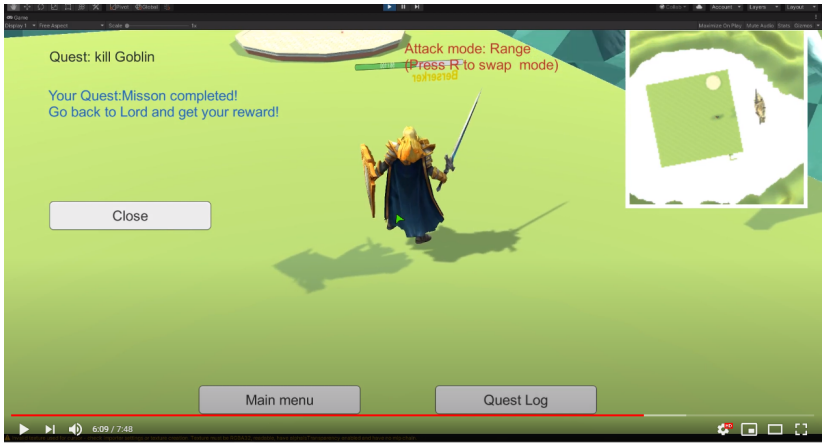
- **a. A basic GUI with two buttons at the bottom of screen**



- **b. Button 1 – opens main menu (switches to another scene, menu doesn't need to be functional)**


```
public void OnMainMenu()
{
    SceneManager.LoadScene(0);
}
```

- **c. Button 2 – opens quest log (just show some text about the status of the quest (should update when you kill the Goblin, for example)**



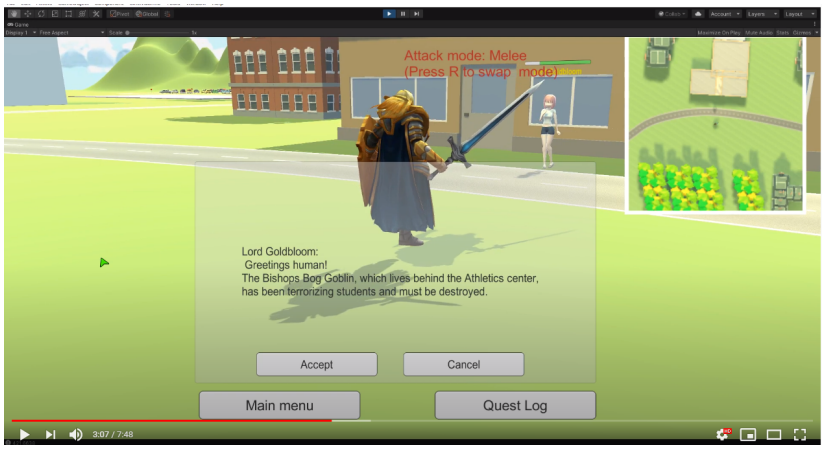

```
public void OnQuest()
{
    TextQuest.gameObject.SetActive(true);
    BtnClose.gameObject.SetActive(true);
    TextQuest.text = "Your Quest:" + TextGetter.SharedInstance.GetQuest();
}
```

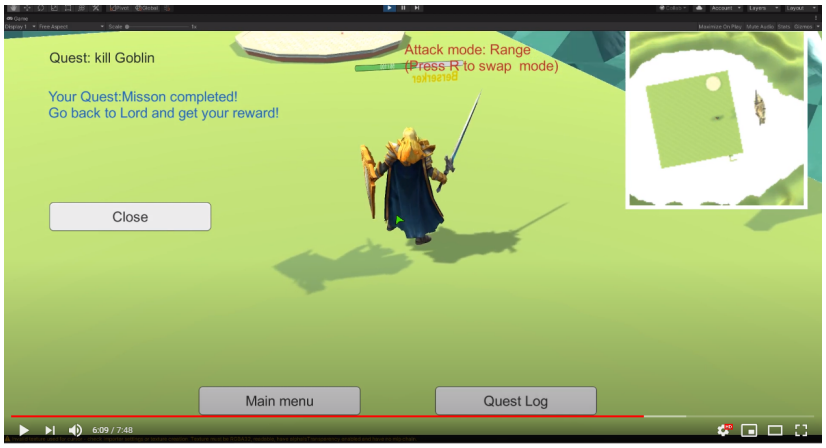
- **d. Health bars over all enemies and your own character that are updated when hit.**


```
    public void GetAttacked(float damage)
    {
        _isAttacked = true;
        _healthPoint -= Random.Range(damage, damage * 0.5f);
        slider.value = _healthPoint;
        Debug.Log(_healthPoint);
        textHealth.text = (int)_healthPoint + " / " + slider.maxValue;

        if (_healthPoint <= slider.maxValue * 0.3f)
            slider.gameObject.transform.Find("Fill
Area").Find("Fill").GetComponent<Image>().color = Color.red;
        if (_healthPoint <= 0)
        {
            // destroyAudio.Play();
            GoblinAnimator.SetBool(Attack,false);
            GoblinAnimator.SetBool("die",true);
            Lord.SetQuestStatus(true);
            TextGetter.SharedInstance.SetQuest("Misson completed!\nGo back to Lord
and get your reward!");
            Invoke("SetActiveFalse", 1.5f);
        }
    }
```

Bunos

- **a. Mini-map**
  In this part, we built the minimap with a second `camera` and a `render texture` on the top-right conner of screen.


Attack mode: Melee
(Press R to swap mode)
Lord Goldbloom:
Greetings human!
The Bishops Bog Goblin, which lives behind the Athletics center,
has been terrorizing students and must be destroyed.
Accept
Cancel
Main menu
Quest Log
3:07 / 7:48


Quest: kill Goblin
Your Quest:Misson completed!
Go back to Lord and get your reward!
Attack mode: Range
(Press R to swap mode)
Close
Main menu
Quest Log
6:09 / 7:48