


Team-D

Lost In Dungeon



Jiabo Zhang 200395447
Zhimu Li 200362915



Background & Business Opportunity

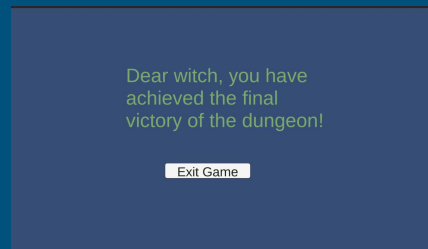
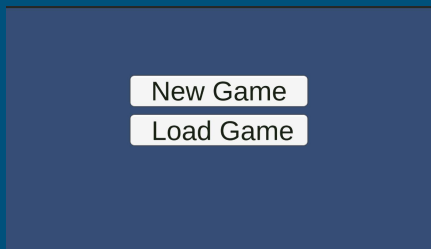
Nowadays, most people's daily lives become more and more stressful, so entertainment is an irreplaceable and necessary element in our society. And the different types games become the most common and effective solution to release pressure for many young people.

Introduction

Our project is about creating a dungeon game with 10-20 minutes play time, so player can enjoy and pass the game during a short break. And the randomness in dungeon can make the game more challenging and interesting, so player can have different experience in every different round.

Game Flow

1. Start in village and get information from NPCs
2. Enter the dungeon at end of village
3. Complete the tutorial (optional)
4. Defeat monsters and unlock items
5. Explore boss room and defeat boss
6. END of the game



Game abstract

- Unity Engine
- 2D game
- Rogue-like dungeon
- Side-scrolling battle system (KoF-like)

Main Scenes

- Village scene
- Dungeon scene
- Battle scenes



Village Scene

- Dialogs of NPCs
- Basic movement function
- Entrance of dungeon



PlayerController & Triggers

```
void Movement()
{
    float horimove;
    horimove = Input.GetAxis("Horizontal");
    float facedirection;
    facedirection = Input.GetAxisRaw("Horizontal");
    if (horimove != 0)
    {
        rb.velocity = new Vector2(horimove * speed * Time.fixedDeltaTime, rb.velocity.y);
    }
    if (facedirection != 0)
    {
        transform.localScale = new Vector3(facedirection, 1, 1);
    }
}
```

```
public class NPC1UI : MonoBehaviour
{
    public GameObject NPC1;

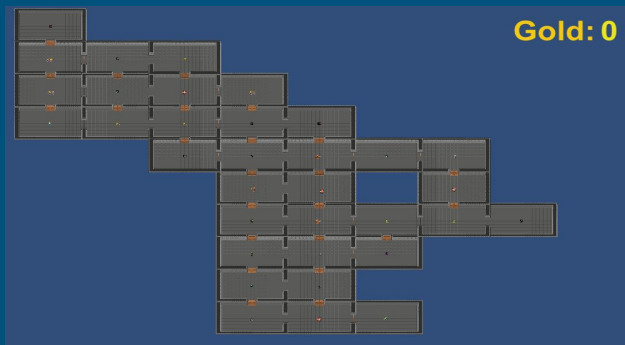
    private void OnTriggerEnter2D(Collider2D collision)
    {
        if (collision.tag == "Player")
        {
            NPC1.SetActive(true);
        }
    }

    private void OnTriggerExit2D(Collider2D collision)
    {
        if (collision.tag == "Player")
        {
            NPC1.SetActive(false);
        }
    }
}
```

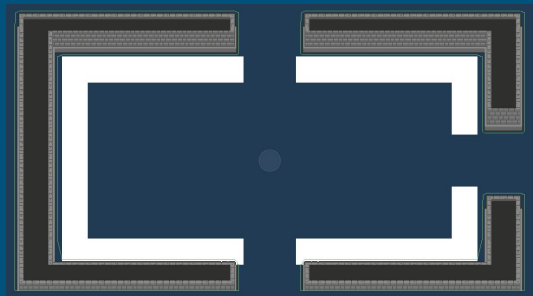
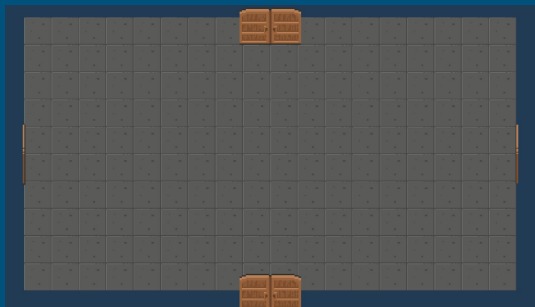
```
public class MapsChange : MonoBehaviour
{
    void Update()
    {
        if (Input.GetKeyDown(KeyCode.E))
        {
            SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex + 1);
        }
    }
}
```


Dungeon Map Scene & Events

- Over 30 rooms
- Monsters
- Items
- Healer



RoomGenerator



```
public void SetupRoom(Room newRoom, Vector3 roomPosition)
{
    newRoom.roomUp = Physics2D.OverlapCircle(roomPosition + new Vector3(0, yOffset, 0), 0.2f, roomLayer);
    newRoom.roomDown = Physics2D.OverlapCircle(roomPosition + new Vector3(0, -yOffset, 0), 0.2f, roomLayer);
    newRoom.roomLeft = Physics2D.OverlapCircle(roomPosition + new Vector3(-xOffset, 0, 0), 0.2f, roomLayer);
    newRoom.roomRight = Physics2D.OverlapCircle(roomPosition + new Vector3(xOffset, 0, 0), 0.2f, roomLayer);

    newRoom.UpdateRoom(xOffset, yOffset);

    switch (newRoom.doorNumber)
    {
        case 1:
            if (newRoom.roomUp)
                Instantiate(wallType.singleUp, roomPosition, Quaternion.identity);
            if (newRoom.roomDown)
                Instantiate(wallType.singleBottom, roomPosition, Quaternion.identity);
            if (newRoom.roomLeft)
                Instantiate(wallType.singleLeft, roomPosition, Quaternion.identity);
            if (newRoom.roomRight)
                Instantiate(wallType.singleRight, roomPosition, Quaternion.identity);
            break;
        case 2:
            if (newRoom.roomLeft && newRoom.roomUp)
                Instantiate(wallType.doubleLL, roomPosition, Quaternion.identity);
            if (newRoom.roomLeft && newRoom.roomRight)
                Instantiate(wallType.doubleLR, roomPosition, Quaternion.identity);
            if (newRoom.roomLeft && newRoom.roomDown)
                Instantiate(wallType.doubleLB, roomPosition, Quaternion.identity);
            if (newRoom.roomUp && newRoom.roomRight)
                Instantiate(wallType.doubleUR, roomPosition, Quaternion.identity);
            if (newRoom.roomUp && newRoom.roomDown)
                Instantiate(wallType.doubleUL, roomPosition, Quaternion.identity);
            if (newRoom.roomRight && newRoom.roomDown)
                Instantiate(wallType.doubleRB, roomPosition, Quaternion.identity);
            break;
        case 3:
            if (newRoom.roomLeft && newRoom.roomUp && newRoom.roomRight)
                Instantiate(wallType.tripleLUR, roomPosition, Quaternion.identity);
            if (newRoom.roomLeft && newRoom.roomRight && newRoom.roomDown)
                Instantiate(wallType.tripleLRB, roomPosition, Quaternion.identity);
            if (newRoom.roomUp && newRoom.roomDown && newRoom.roomRight)
                Instantiate(wallType.tripleURB, roomPosition, Quaternion.identity);
            if (newRoom.roomLeft && newRoom.roomUp && newRoom.roomDown)
                Instantiate(wallType.tripleLUB, roomPosition, Quaternion.identity);
            break;
        case 4:
            if (newRoom.roomLeft && newRoom.roomUp && newRoom.roomRight && newRoom.roomDown)
                Instantiate(wallType.fourDoors, roomPosition, Quaternion.identity);
            break;
    }
}
```



Wall_B



Wall_Four



Wall_L



Wall_LB



Wall_LR



Wall_LRB



Wall_LU



Wall_LUB



Wall_LUR



Wall_R



Wall_RB



Wall_U



Wall_UB



Wall_UR



Wall_URB

Battle Scenes & System

- 3 difficulty levels
- 3 different battle AI





Monster damage = Monster attack - Player DEF

BattleControl & Enemy AI

Asset packages

- Sunny Land Forest Assets
- Sunnyland,
- Gothicvania-Town
- Orc,Custom Character_Vol3
- Aekashics Librarium - Megapack III
- Free_packs



Game Demo

Q & A

Thank You For Watching!
