《World of one：一个人的世界》项目文档

组名：小众工作室

程序：刘琦、潘天瑶

美工：尹瑞

1. **概述**

本项目基于unity引擎开发，是一款PC端2D休闲闯关小游戏。

玩家将通过简单轻松的操作控制主角嘻嘻嘻来避开关卡中的陷阱，利用关卡机关，努力到达终点。为了贴合生生不息的主题并增加游戏趣味性，本游戏在关卡中会不断生成随机移动的玩家克隆体，因此玩家需要在人潮中努力辨认主角，带领TA突出重围，取得胜利！

本文将从设计思路、玩法及操作控制、UI及音乐、进程及关卡、程序设计五点对本游戏进行简要阐述。

1. **游戏设计思路**
2. **游戏类型**

2D休闲闯关类游戏。

1. **游戏背景**

在一个风和日丽的早晨，嘻嘻嘻开始了TA的冒险之旅。但是，一个人的旅程总是枯燥无味的，因此异次元的小伙伴们也纷纷赶来，加入了这段奇妙的旅程……

1. **核心玩法**

玩家需努力在人潮中辨认主角，并通过简单的操作控制主角避开陷阱、利用机关到达关卡终点，达成胜利。

1. **平台和人群**

PC端Windows平台，1024x768分辨率，全年龄段适合。

1. **美术风格**

简约休闲像素画风，色彩简明靓丽，风格轻松愉快。

1. **游戏玩法及操作控制**
2. **游戏机制**

**玩家能力：**左右移动、跳跃。

**胜负判定：**掉出边界或碰到陷阱人物死亡，到达关卡终点的小屋游戏胜利。

1. **游戏挑战**

准确判断主角位置，把握操作时机，躲避陷阱。

1. **操作键位**

**AD：**左右移动

**Escape：**暂停界面

1. **游戏UI及音乐设计**
2. **UI设计**

**开始界面：**



**暂停界面：**



**通关界面：**





1. **音乐设计**

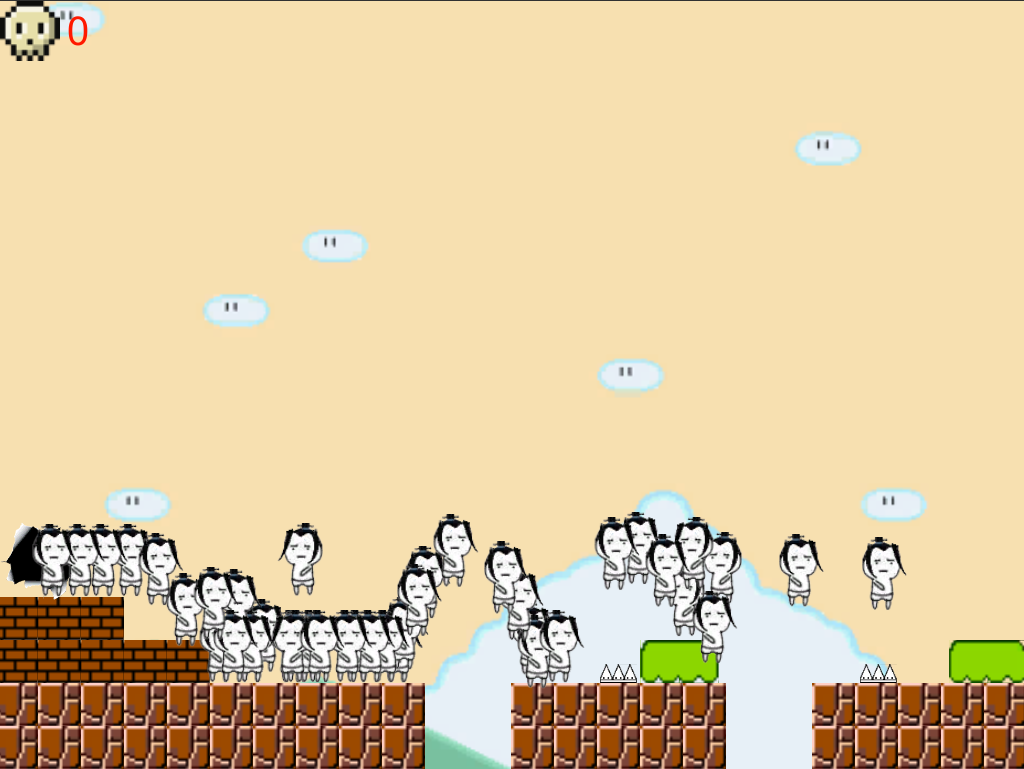
为配合游戏轻松愉快的气氛，背景音乐采用节奏轻快的电子合成音乐，音效也采用简洁有趣的卡通音效。为了配合游戏的不同难度，游戏两个章节采用不同的背景音乐。

1. **游戏进程和关卡设计**

**第一章开场：**



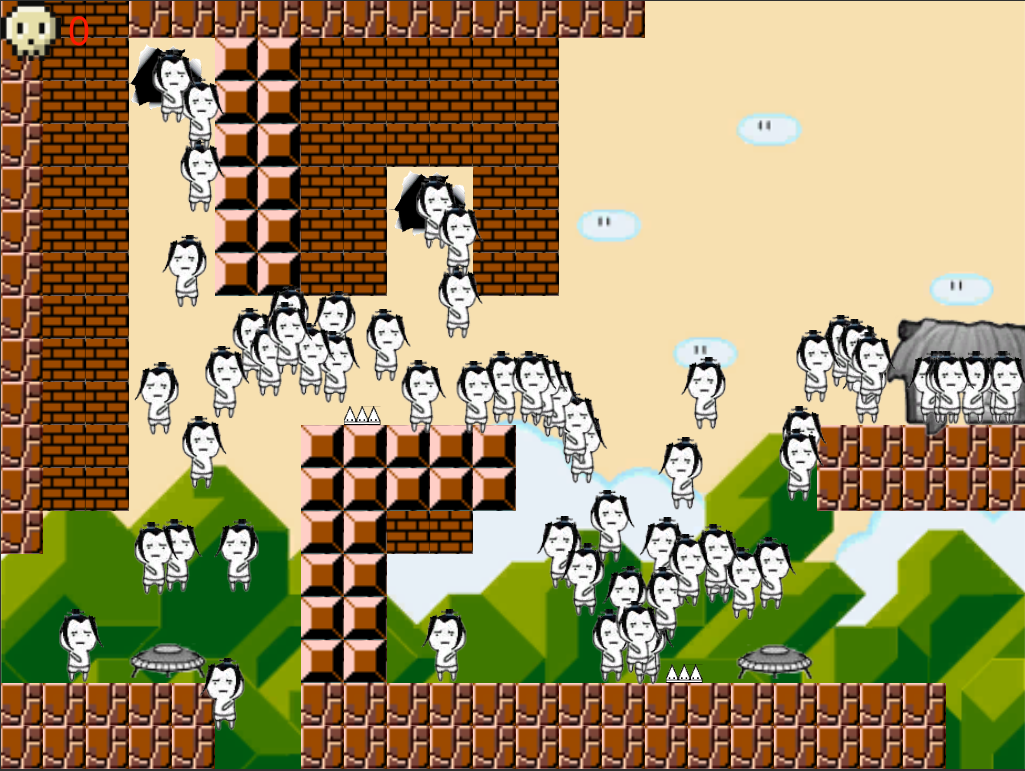
**Level1：**



**Level2：**



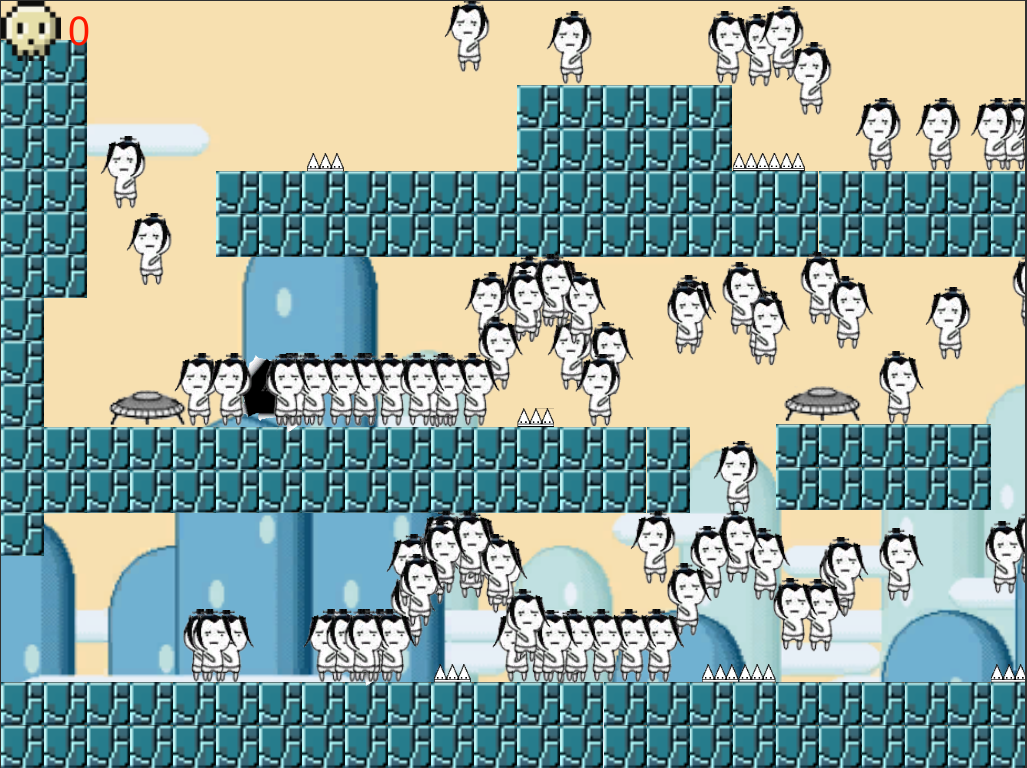
**Level3：**



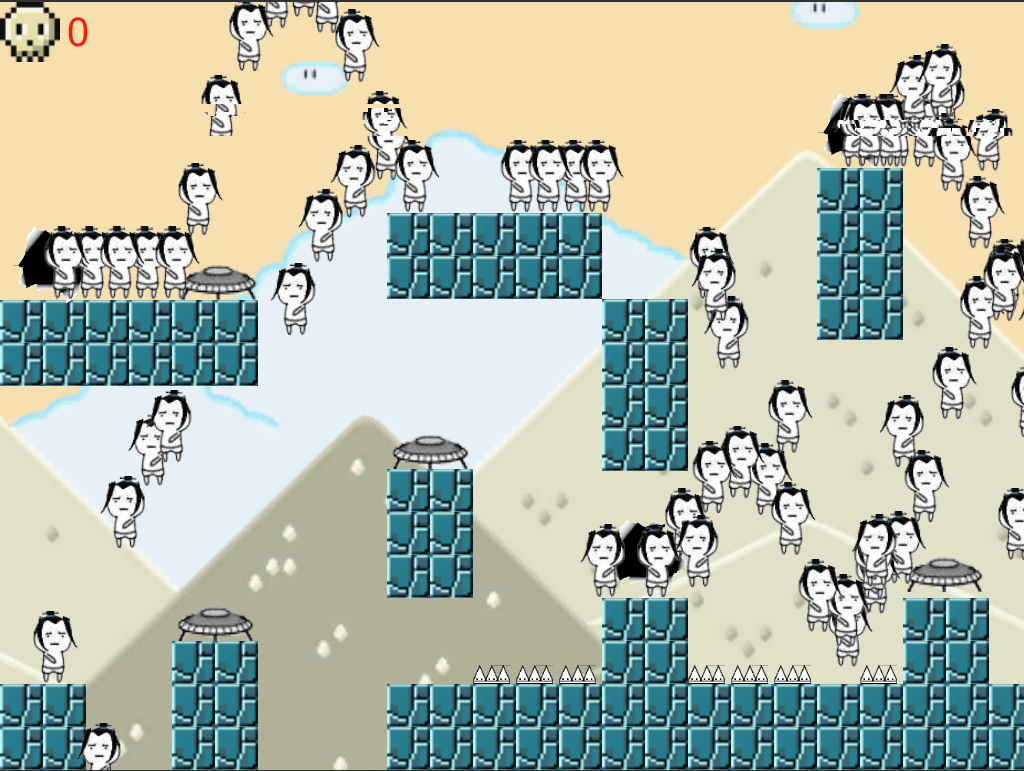
**第二章开场：**



**Level4：**



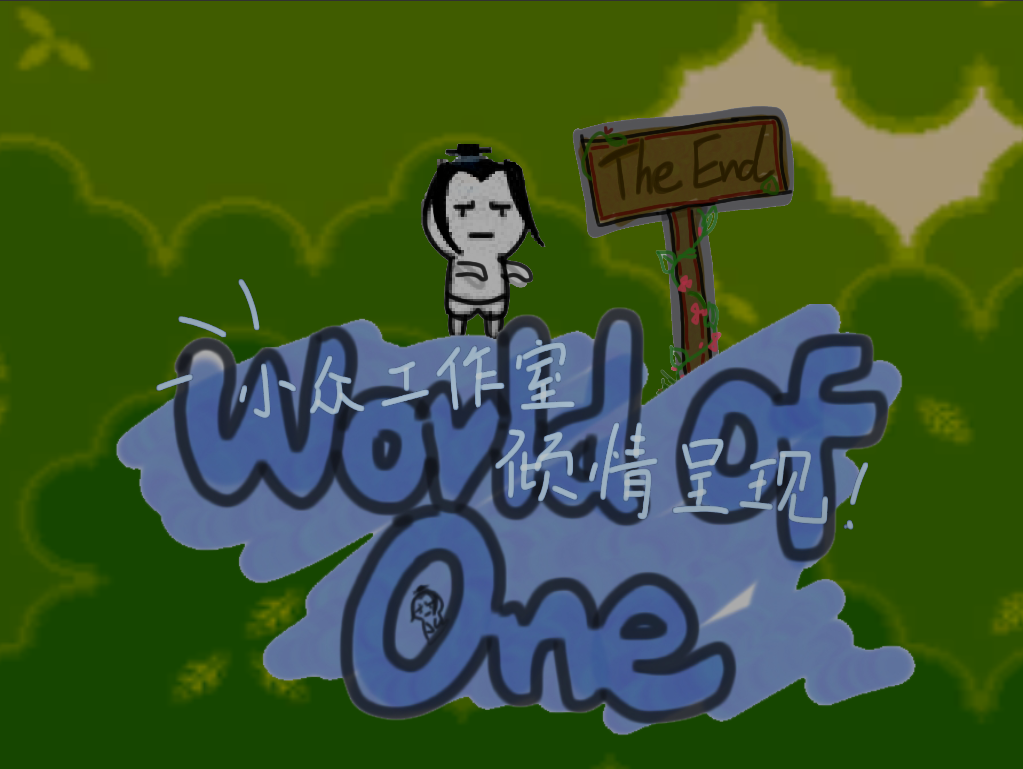
**Level5：**



**通关场景：**







1. **程序设计**

**克隆体随机移动脚本（Type1、Type2）：**

**Type1：**

1. **using** System.Collections;
2. **using** System.Collections.Generic;
3. **using** UnityEngine;
5. **public** **class** Type1Move : MonoBehaviour {
6. **private** Rigidbody2D rig;
7. **private** **float** lastTime = 0;
8. **private** **float** timeSpan = 1.5f;
9. **private** **float** walkSpeed = PlayerMovement.walkSpeed;
10. **private** **float** jumpSpeed = PlayerMovement.jumpSpeed;
11. **private** **int** direction = 0;
12. **private** **bool** touchEdge = **false**;
13. // Use this for initialization
14. **void** Start () {
15. rig = GetComponent<Rigidbody2D>();
16. lastTime = Time.time;
17. }
19. // Update is called once per frame
20. **void** Update () {
21. **if** (touchEdge)
22. {
23. Turn();
24. }
25. **else**
26. {
27. Move();
28. }
29. Death();
30. }
31. **private** **void** Move()
32. {
33. **if** (Time.time - lastTime > timeSpan)
34. {
35. **if** (Random.value > 0.5)
36. {
37. direction = 0;
38. walkSpeed = PlayerMovement.walkSpeed;
39. }
40. **else**
41. {
42. direction = 180;
43. walkSpeed = -PlayerMovement.walkSpeed;
44. }
45. lastTime = Time.time;
46. }
47. transform.eulerAngles = **new** Vector3(0, direction, 0);
48. rig.velocity = **new** Vector2(walkSpeed, rig.velocity.y);
49. }
50. **private** **void** Turn()
51. {
52. direction = (direction + 180) % 360;
53. walkSpeed = -walkSpeed;
54. touchEdge = **false**;
55. }
56. **private** **void** OnTriggerEnter2D(Collider2D collision)
57. {
59. **if** (collision.tag == "JumpTrigger")
60. {
61. //Debug.Log("Jumptype1");
62. rig.velocity = **new** Vector2(rig.velocity.x, jumpSpeed);
63. }
64. **if** (collision.tag == "TurnTrigger")
65. {
66. touchEdge = **true**;
67. }
68. }
69. **private** **void** Death()
70. {
71. **if** (transform.position.y < -6)
72. {
73. Debug.Log("death");
74. Destroy(gameObject);
75. }
76. }
77. }

**Type2：**

1. **using** System.Collections;
2. **using** System.Collections.Generic;
3. **using** UnityEngine;
5. **public** **class** Type2Move : MonoBehaviour {
6. **private** Rigidbody2D rig;
7. **private** **float** lastTime = 0;
8. **private** **float** timeSpan = 1.5f;
9. **private** **float** walkSpeed = PlayerMovement.walkSpeed;
10. **private** **float** jumpSpeed = PlayerMovement.jumpSpeed;
11. **private** **int** direction = 0;
12. **private** **bool** canJump = **false**;
13. **private** **bool** touchEdge = **false**;
14. // Use this for initialization
15. **void** Start () {
16. rig = GetComponent<Rigidbody2D>();
17. lastTime = Time.time;
18. }
20. // Update is called once per frame
21. **void** Update () {
22. **if** (touchEdge)
23. {
24. Turn();
25. }
26. **else**
27. {
28. Move();
29. }
30. Jump();
31. Death();
32. }
33. **private** **void** Move()
34. {
35. **if** (Time.time - lastTime > timeSpan)
36. {
37. **if** (Random.value > 0.5)
38. {
39. direction = 180;
40. walkSpeed = -PlayerMovement.walkSpeed;
41. }
42. **else**
43. {
44. direction = 0;
45. walkSpeed = PlayerMovement.walkSpeed;
46. }
47. lastTime = Time.time;
48. }
49. transform.eulerAngles = **new** Vector3(0, direction, 0);
50. rig.velocity = **new** Vector2(walkSpeed, rig.velocity.y);
51. }
52. **private** **void** Jump()
53. {
54. **if** (Input.GetKeyDown(KeyCode.Space )&& Random.value > 0.5 && canJump)
55. {
56. rig.velocity = **new** Vector2(rig.velocity.x, jumpSpeed);
57. canJump = **false**;
58. }
59. }
60. **private** **void** Turn()
61. {
62. direction = (direction + 180) % 360;
63. walkSpeed = -walkSpeed;
64. touchEdge = **false**;
65. }
66. **private** **void** OnTriggerEnter2D(Collider2D collision)
67. {
68. **if** (collision.tag == "Ground")
69. {
70. canJump = **true**;
71. }
72. **if** (collision.tag == "TurnTrigger")
73. {
74. touchEdge = **true**;
75. }
76. **if** (collision.tag == "JumpTrigger")
77. {
78. rig.velocity = **new** Vector2(rig.velocity.x, jumpSpeed);
79. }
80. }
81. **private** **void** Death()
82. {
83. **if** (transform.position.y < -6)
84. {
85. Debug.Log("death");
86. Destroy(gameObject);
87. }
88. }
89. }