2022届本科毕业设计中期检查

姓 名: 欧林妍

学 号: 2018218413

专业班级: 计算机18-1班

指导教师: 张建军 张本宏

指导教师所在系所:分布智能与物联网研

究所

汇报日期: 2022年4月16日





2022届本科毕业设计中期检查

课题名称:基于Unity的餐厅经营游戏的设计与开发

课题来源: 自拟

课题类型: 软件开发





汇报提纲

- □ 内容与要求
- □ 已完成工作和未完成工作
- □ 后续工作安排
- 口 收获
- □ 参考文献列表



内容与要求

本游戏计划基于unity来进行开发,游戏玩法主要为经营餐厅,支持多人线上游玩。在游戏中,玩家们需要根据客人进入餐厅的先后顺序来合理的为客人服务,包括分配餐桌、根据菜谱进行制作、上菜、收银等工作。在这个过程中,十分考验玩家之间的配合程度,在游戏中享受合作的快乐。整个游戏的开发包括了客户端服务端之间的网络通讯、数据存储、游戏模型以及UI美术资源的制作等。





已完成工和未完成工作

- □ 已完成的:
 - 服务端和客户端的框架
 - 餐厅3d建模
 - 开发登陆面板中 服务器访问出现问题,正在处理中



```
Client Car Assembly-CSharp Assets Scripts UIPanel C# LoginPanel.cs
                                                                                                                                   Attach
                              C# UIPanelType.cs × C# UIManager.cs × C# GameFacade.cs × C# LoginPanel.cs × C# RegisterPanel.cs × C# StartPanel.cs ×

   Unity ▼

                                                                                                                                   C# BasePa
                                      using UnityEngine.UI;
▼ < Assets</p>
                                       ▶ & References
                                       public class LoginPanel : BasePanel
  private Button closeButton;
  ► ☐ Plugins (firstpass)
  ► ≅ Resources
                                           private InputField usernameIF;
  private InputField passwordIF;
  private LoginRequest loginRequest;
     Manager
     ▶ ■ Net

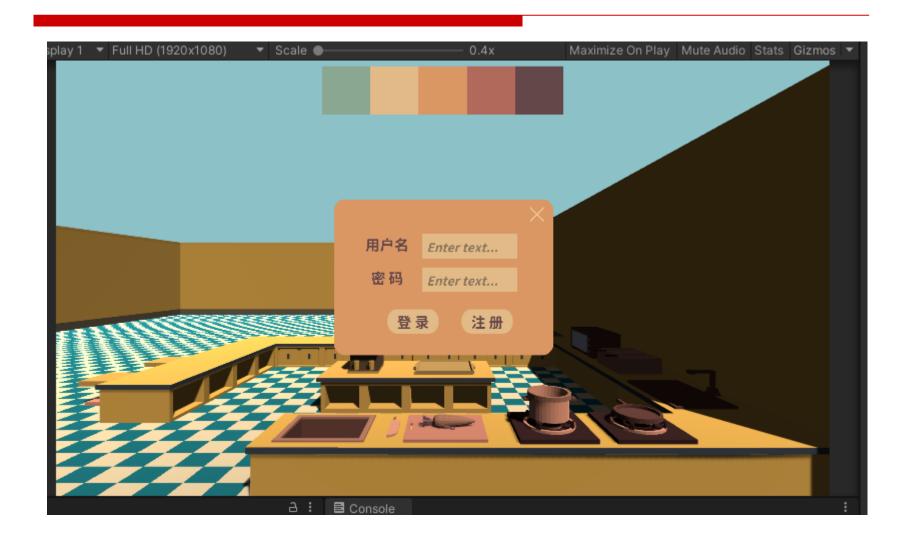
   □ 0+4 usages    ② Ouly

     客户端开发
                                           public override void OnEnter()
                              16 10
     ▼ □ UIPanel
          c# BasePanel.cs
                                               base.OnEnter();
         C# LoginPanel.cs
         c# MessagePanel.cs
                                               gameObject.SetActive(true);
         c# RegisterPanel.cs
          c# StartPanel.cs
       c# GameFacade.cs
                                               transform.localScale = Vector3.zero;
  ► ☐ UIFramework
transform.DOScale(endValue: 0.5f, duration: 0.2f);
   Scratches and Consoles
                                               transform.localPosition = new Vector3( X: 1000, Y: 0, Z: 0);
                                               transform.DOLocalMove(Vector3.one, duration: 0.2f);
                                               loginRequest = GetComponent<LoginRequest>();
                                               usernameIF = transform.Find("UsernameLable/UsernameInput").GetComponent<InputField>();
                                               passwordIF = transform.Find("PasswordLable/PasswordInput").GetComponent<InputField>();
                                               closeButton = transform.Find("CloseButton").GetComponent<Button>();
                                               closeButton.onClick.AddListener(OnCloseClick);
                                               transform.Find("LoginButton").GetComponent<Button>().onClick.AddListener(OnLoginClick);
                                               transform.Find("RegisterButton").GetComponent<Button>().onClick.AddListener(OnRegisterClick);
```

```
Server > C# Client.cs
                               🜣 — C# RequestCode.cs × C# ConnHelper.cs × C# Program.cs × C# Server.cs × C# Message.cs × C# BaseController.cs × C# Client.cs × C# ActionCode.cs
 ♠ Solution ¬
  Server · 2 projects
  ▶ 🔏 Dependencies
                                                   /// </summary>
    ► ■ Properties
       c# ActionCode.cs
       c# RequestCode.cs
       c# ReturnCode.cs
                                                       private Socket clientSocket;
   private Server server;
    ▶ & Dependencies
                                                       private Message msg = new Message();
     Properties
                                                       private MySqlConnection mysqlConn;
     ▶ ☐ Controller
     ▶ ☐ DAO
                                                       public MySqlConnection MysqlConn => mysqlConn;
    ▶ ☐ Model
     public Client(){}
         C# Client.cs
                                                                                                                服务端开发
         c# Message.cs
                                                       public Client(Socket clientSocket, Server server)
         c# Server.cs
    ► Tool
                                                           this.clientSocket = clientSocket;
       C# Program.cs
  Scratches and Consoles
                                                           this.server = server;
                                                           mysqlConn = ConnHelper.Connect();
                                                       public void Start()
                                                           clientSocket.BeginReceive(msg.Data, msg.StartIndex, msg.RemainSize, SocketFlags.None, ReceiveCallback
                                                       private void ReceiveCallback(IAsyncResult ar)
```











已完成工和未完成工作

□ 未完成的:

- 游戏的具体玩法还没有进入开发,前期的服务 端客户端框架搭建花了较多时间,在之前框架 的基础上,后面的开发速度会有所提高
- 毕业论文撰写





后续工作安排

- □ 在前面开发的基础上开发游戏的具体玩法
- □ 经过测试后撰写毕业论文



参考文献列表

[1]章国雁. 基于Unity3D的多人在线游戏案例设计与实现[J]. 安徽水利水电职业技术学院学报, 2021, 21(03):46-49.

[2] Andrew Coggan. Unity Game Audio Implementation: A Practical Guide for Beginners [M]. Taylor and Francis: 2021-07-18.

[3] Bai Ying. SQL Server Database Programming with Visual Basic.NET:Concepts, Designs and Implementations[M]. John Wiley & Sons, Inc.: 2020-06-10.

[4] Thorn Alan. Blender to Unity Workflows[M]. CRC Press: 2018-09-01.

[5] 张常清, 晏西国, 卜庆凯. 基于移动互联网的APP与服务器之间的通信设计[J]. 青岛大学学报(工程技术版), 2015, 30(02):26-30. D01:10. 13306/j. 1006-9798. 2015. 02. 005.

[6] 史宝明, 贺元香, 李岚. Unity3D随机寻路算法设计[J]. 吉林师范大学学报(自然科学版), 2022, 43(01):128-133. DOI:10. 16862/j. cnki. issn1674-3873. 2022. 01. 021.





感谢您的聆听!

欢迎指正!