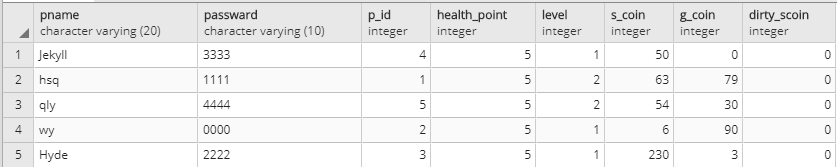
MMPROG

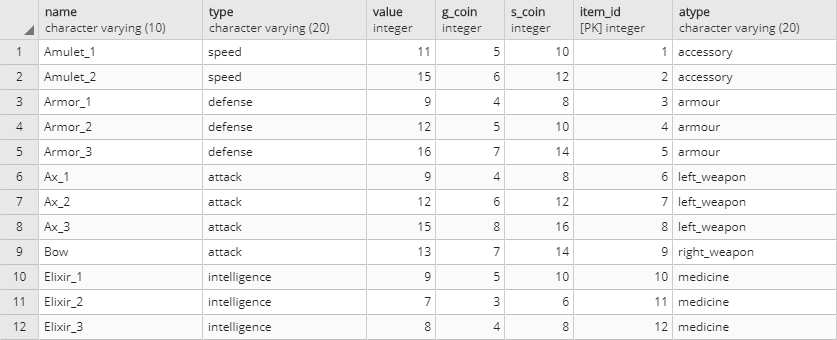
韩思绮 吴妍

1. 数据库设计

* player：存放用户基本信息
* dirty\_scoin存放用户离线中开始获得交易收益



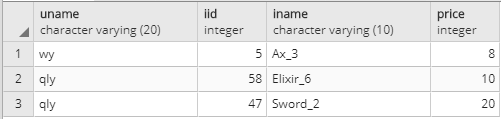
* item：存放系统商场内固定出售的物品信息



* p\_i：存放玩家物品拥有的对应关系
* 以id为Key，每次插入时去当前id最大号+1存储
* 若用户同时拥有多件物品，以单独的个体显示在背包中

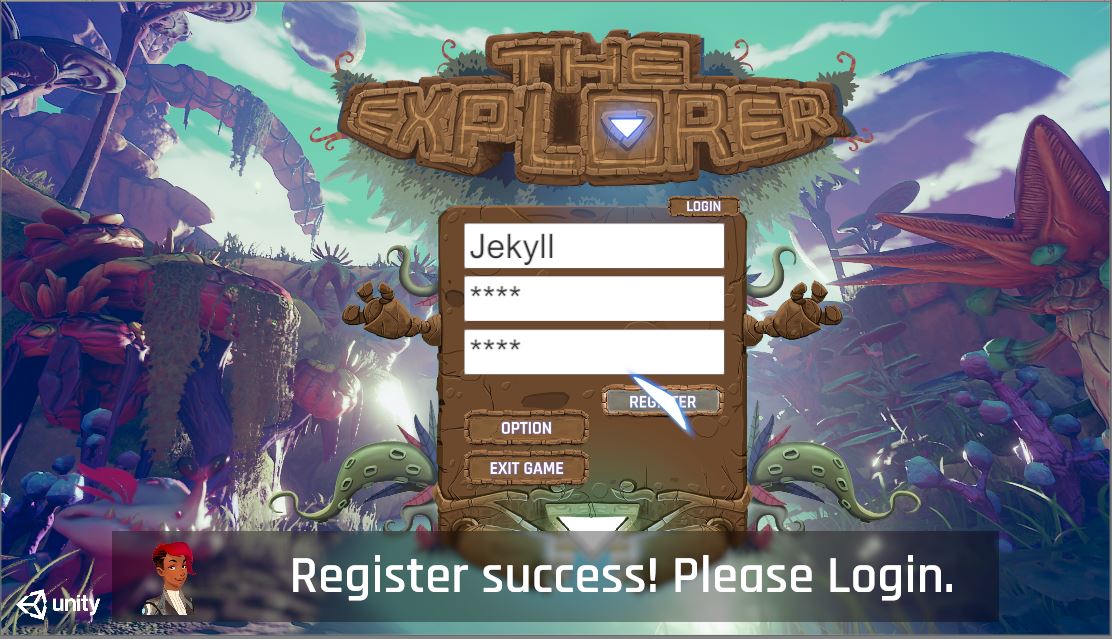


* trade：存放交易市场物品信息 （出售者，p\_i中物品iid号，物品名，设定价格）



1. 功能实现及界面展示

* 登陆注册功能
* 注册



* 后端

OnRecvRegister.cs

赋予新玩家ID和默认属性值，插入玩家表中

* 登陆



* 后端

OnRecvLogin.cs

读取数据库，将玩家各个信息存入player.cs

* 场景进入
* 通信

通过SPlayerAttribute信号传递

* 前端

OnRecvPlayerAttribute.cs

将所有信息存入前端全局类PlayerAttribute中

* 属性面板 RoleUI
* 属性值读取



* 前端

读取全局类PlayerAttribute中信息，并在RoleUI中一一绑定显示。

速度、智慧、防御、攻击属性值由佩戴宝物决定。

生命属性值由玩家当前血量决定。

* 玩家升级
* 玩家等级可由打败怪物方式升级。



* 背包功能
* 背包物品属性
* 后端

SerializableData.cs序列化创建M\_Item类用于存放物品的各项属性以及两种枚举类型。

一个物品对应一种类型及一种属性加成。

* 前端

读取全局类PlayerAttribute中玩家背包信息（以Dictionary形式），并在InventoryUI中绑定显示。

* 背包物品佩戴及卸下



* 前端

InventroyUI中绑定button脚本

* 通信（前->后）

通过CItemApply信号传递

* 后端

OnRecvItemApply.cs

将信息和player.cs中玩家宝物佩戴情况进行对比整合，判断操作是否成功，若成功修改数据库，发送成功信号。

\*可以区分一类物品、一种物品

* 通信（后->前）

通过SItemApply信号传递

* 前端

OnRecvApplySuccess.cs

操作成功则修改对应PlayerAttribute中物品相关属性，并刷新界面信息（属性值、佩戴情况、佩戴数量）

* 背包物品挂牌及撤销挂牌





* 背包已佩戴物品界面显示

EquippedUI.cs



* 背包待售物品界面显示



* 商场功能
* 前端界面设计
* 增加商品详情及拥有情况 ShelfItemUI.cs

创建DetailsView弹窗视图，从PlayerAttribute中读取显示商品详细信息

* 增加金币银币余额数显示
* 增加金币银币当前购物车总额显示
* 金币银币购买按钮脚本 CartGridUI.cs



* 购买逻辑实现
* 通信（前->后）

通过CBuyItems信号传递

* 后端

OnRecvBuyItems.cs

确认钱币足够并更新数据库及player.cs中背包情况，发送成功信号

\*更新数据库时对于ID号设置，可以唯一标识用户及该项物品

* 通信（后->前）

通过SItemBuy信号传递

* 前端

OnRecvItemBuy.cs

PlayerAttribute背包中加入新物品，并更新金币数，刷新界面信息

* 金币银币充值兑换功能
* 金币充值
* 前端

增加RechargeGoldUI.cs

放置二维码及充值数额填入框，假定用户充值完毕，向后端发送信号并刷新用户的金币数

* 通信（前->后）

通过CAddCoin信号传递

* 后端

OnRecvAddCoins.cs

更新数据库和用户金币数



* 银币兑换
* 前端

增加RechargeSilverUI.cs

假定金币兑换银币比率为1:2，放置数额填入框。确认兑换后，向后端发送信号并刷新用户的金币数和银币数。

* 通信（前->后）

通过CAddCoin信号传递

* 后端

OnRecvAddCoins.cs

更新数据库和用户金币数



* 交易商店功能
* 前端界面设计（以商场初始界面为基础）
* 增加商品详情及拥有情况 TradeShelfItemUI.cs

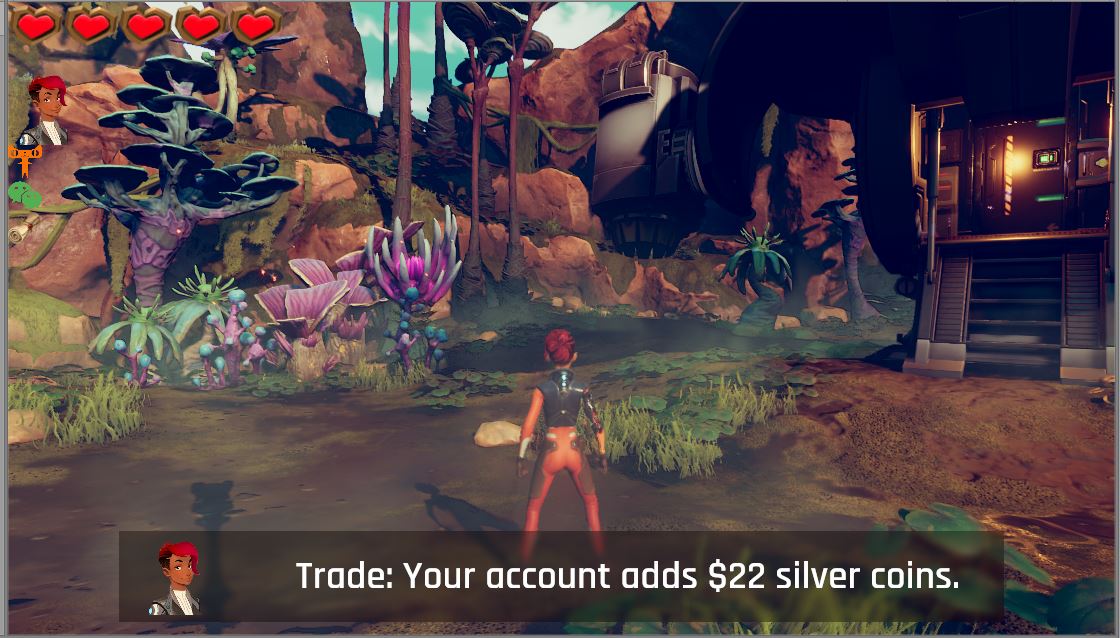
- 创建TDetailsView弹窗视图

- 从Trade\_item中读取显示商品详细信息及所有者，需新建字典类型

* 增加金币银币余额数显示
* 增加银币当前购物车总额显示
* 银币购买按钮脚本 TradeCartGridUI.cs
* 撤销商品数量增加减少功能，重新设置移出购物车的方法

· 

出售方获得交易收益提示：



* 聊天系统功能
* 前端界面(使用默认)
* 前后端通信
* 提取当前场景中所有玩家（除用户自身）作为可聊天对象。

FriendUI.cs OnRecvGetFriends.cs

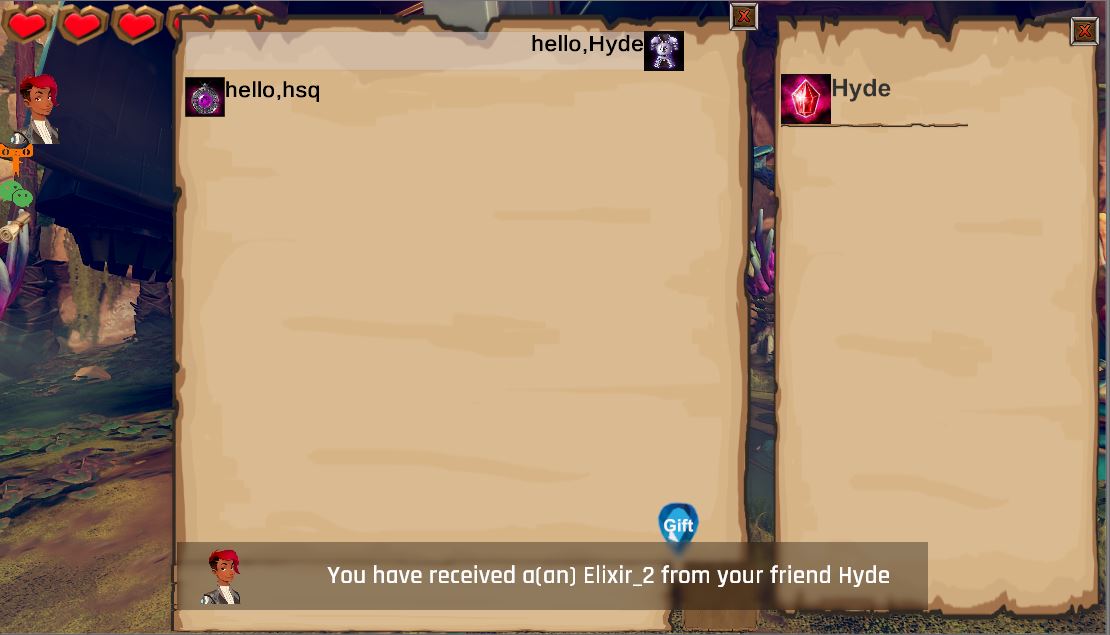
* 点击图标传递好友姓名 FriendInfoUI.cs ChatMessage.cs
* 用户打开和聊天对象的对话框可以接收到自该用户上线后接受到的和该聊天对象的所有对话内容。（由后端发送消息获得） Chat.cs
* 向聊天对象发送消息 OnRecvSendMessage.cs
* 接收来自聊天对象的消息 OnRecvRecvMessage.cs



* 好友赠礼功能
* 可以通过对话框中的按钮打开背包选择礼物赠送给朋友。GiftUI.cs



收礼方获得礼品提示：



1. 总结

历时一个多月的大作业在期末考前一周顺利验收，