详细设计说明书

目录

[1.界面设计： 1](#_Toc42969796)

[2．关键类设计 3](#_Toc42969797)

[2.1战斗子系统类： 3](#_Toc42969798)

[2.2人物移动控制类 4](#_Toc42969799)

[2.3装备操作类 5](#_Toc42969800)

[2.4技能列表类： 5](#_Toc42969801)

[2.5技能快捷栏类： 6](#_Toc42969802)

[2.6野怪控制类： 6](#_Toc42969803)

[2.7武器效果类： 7](#_Toc42969804)

[2.8角色生成类： 7](#_Toc42969805)

[3 关键模块核心算法设计 8](#_Toc42969806)

[3.1关于对敌人选中的关键技术 8](#_Toc42969807)

[3.2开始界面算法设计： 9](#_Toc42969808)

[3.3野怪自动控制模块设计： 10](#_Toc42969809)

## 1.界面设计：

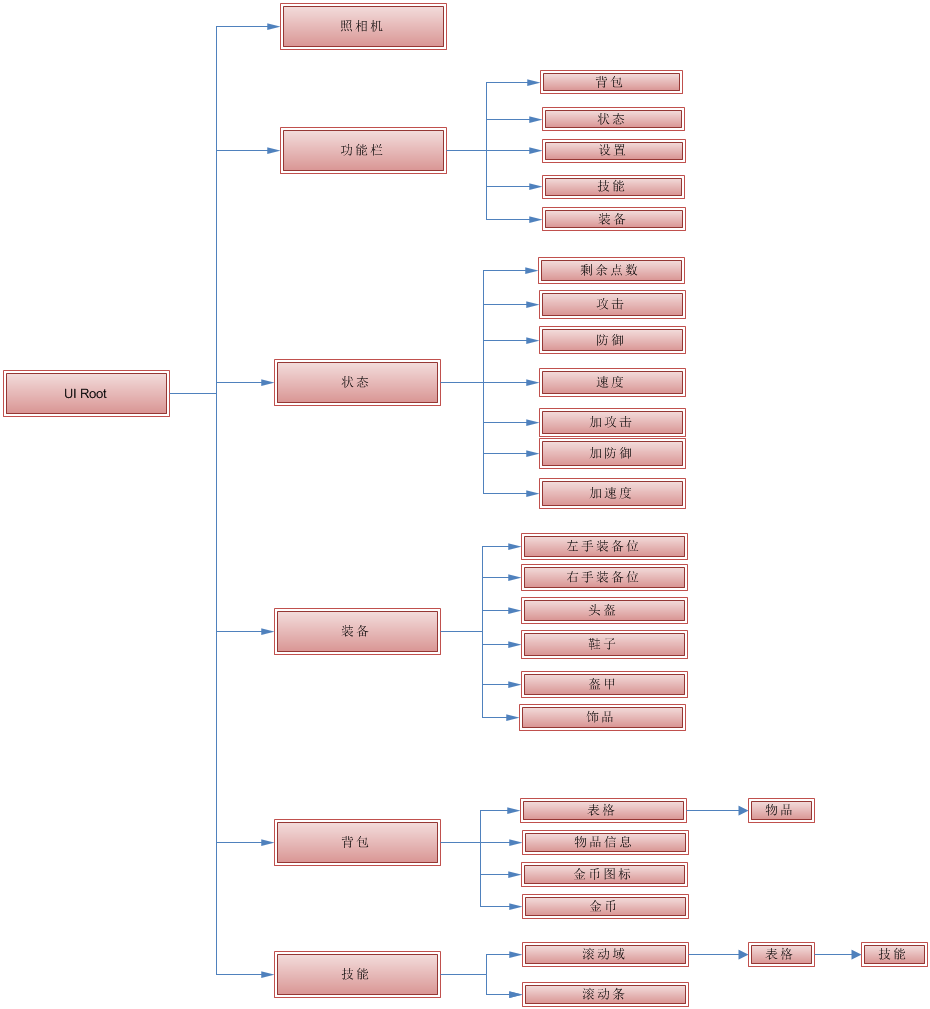


图15 游戏界面结构图

游戏界面，UI Root下有功能、状态、装备、背包、技能几大部分。

其中功能栏相当于一个快捷方式，让玩家迅速找到需要的功能。

状态显示角色信息。

角色有六个装备佩戴位置。

背包除了装物品，鼠标移动时可以查看物品具体信息外，还显示金币数量。

技能列表是主角掌握的技能，拖放到功能栏以后就可以方便的释放技能。

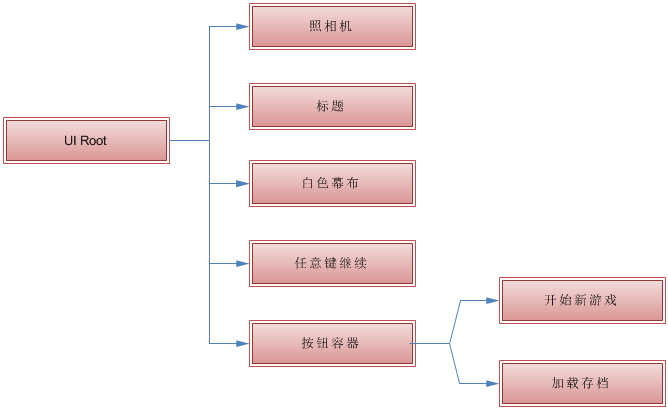


图 16 开始界面结构图

开始界面，显示标题和动画效果，主要功能是开始新游戏进入游戏场景，加载存档跳转到保存的游戏进度。

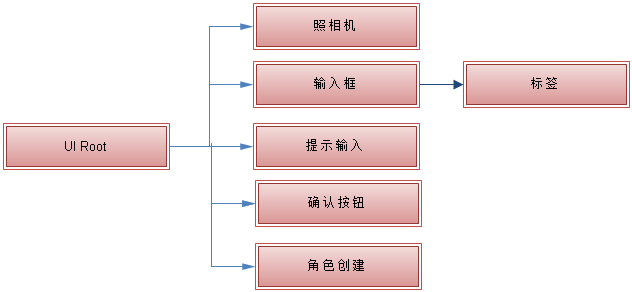


图 17 输入昵称界面结构

输入昵称界面，可以查看角色形象，输入游戏昵称。

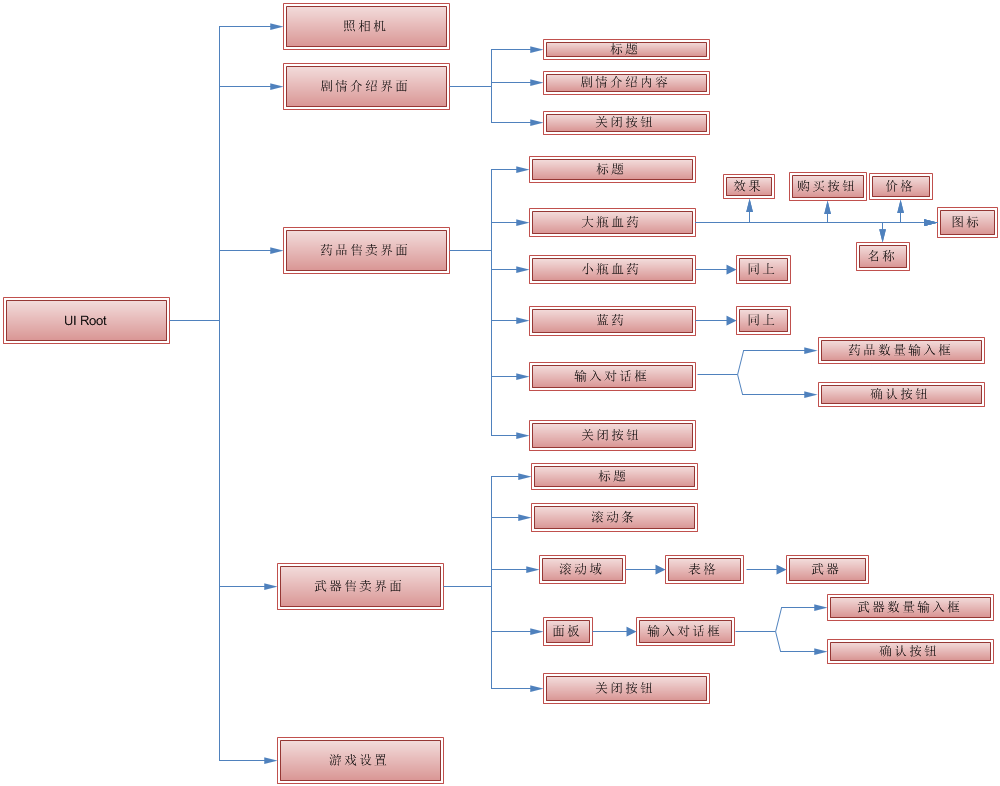


图 18 NPC商人界面

游戏场景有商店NPC，点击可以打开不同的物品售卖界面。

药品有三种，武器约二十种，游戏设置用于运行时生成武器列表。

点击购买，输入购买数量，然后扣除金币，就可以在背包中查看了。

## 2．关键类设计

### 2.1战斗子系统类：

表14 战斗子系统类详细设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | 属性 | 属性说明 | 方法 | 方法说明 |
| CharacterAttackSystem | PlayAttackState | 标识人物是否处于攻击状态 | Start() | 初始化函数 |
| aniname\_normalattack | 普通攻击动画 | Update() | 按帧刷新游戏场景 |
| aniname\_idle | 站立动画 | UseSkill（） | 使用技能 |
| me\_normalattack | 普通攻击时间 | TakeDamage（） | 造成伤害 |
| timer | 计时器 | GetAttack() | 获得角色攻击力 |
| min\_distance | 最小攻击距离 | playattack() | 播放攻击动画 |
| parameter | 按键管理器 | addAnimEve（） | 添加动画事件 |
| target\_normalattack | 目标位置 |
| animation | 动画管理器 |
| SkillInfo | 技能信息 |
| playstaus | 角色状态 |
| effect | 特效 |

### 2.2人物移动控制类

表15 人物移动控制类详细设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | 属性 | 属性说明 | 方法 | 方法说明 |
| Player\_contrller | State | 用于标识角色当前的动作 | Start() | 初始化函数 |
| normalSpeed | 行走速度 | FixedUpdate() | 刷新游戏场景 |
| gravity | 重力 | UpdateMove() | 移动时播放对应的动画 |
| cameraMoveSpeed | 摄像机移动速度 | SimpleMove（） | 向指定的位置移动 |
| mainCamera | 主摄像机位置 | AudioManagement() | 音效控制 |
| grounded | 标识角色是否在地面上 | CurrenSpeed() | 更新当前的速度 |
| parameter | 按键管理器 |

### 2.3装备操作类

表16 装备操作类详细设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | 属性 | 属性说明 | 方法 | 方法说明 |
| Equipment | Attack | 装备的攻击属性 | UpdateProperty() | 装备栏状态更新 |
| Defense | 装备的防御属性 | PlusProperty() | 控制属性增加 |
| Speed | 装备的速度属性 | DressEquip() | 装备穿上或卸下 |

### 2.4技能列表类：

表17技能列表类详细设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | 属性 | 属性说明 | 方法 | 方法说明 |
| Skill | MagicianSkillList | 魔法师角色技能编号列表 | Start() | 初始化函数 |
| SwordmanSkillList | 剑士角色技能编号列表 | UpdateShow() | 刷新技能列表显示 |
| ps | 角色 | Show() | 打开技能列表时显示 |
| grid | 所有技能的控制网格 |  |  |
| SkillItemPrefab | 单个技能的显示面板 |  |  |

### 2.5技能快捷栏类：

表18技能快捷栏类详细设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | 属性 | 属性说明 | 方法 | 方法说明 |
| Shortcut | ShortType | 技能栏是否存在技能的枚举 | SetSkill(int id) | 在技能快捷键建立技能显示 |
| Type | 用于标识技能快捷键有无添加技能 | Update() | 刷新技能快捷栏显示 |
| id | 技能编号 | OnDrugUse() | 控制角色使用技能 |
| info | 技能信息字典 |  |  |

### 2.6野怪控制类：

表19 野怪控制类详细设计

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 类名 | 属性 | 属性说明 | | 方法 | 方法说明 |
| StoreMan | state1 | | 标识野怪的状态变化 | Start() | 初始化函数 |
| aniname\_normal1attack1\_1 | | 普通攻击动画 | Update() | 按帧刷新游戏场景 |
| animall\_idle1 | | 静立动画 | Randomstate1() | 随机产生行走和静立状态 |
| aniname\_attack\_now1 | | 当前攻击动画 | AutoAttack1() | 野怪自动攻击 |
| aniname\_crazyattack\_1 | | 疯狂攻击动画 | RandomAttack() | 随机产生攻击方式 |
| minDistance1 | | 最小攻击距离 | TakeDamage() | 受到伤害 |
| attack | | 攻击造成的伤害值 | ShowBodyRed1() | 受到伤害身体变红 |
| maxDistance1 | | 最大攻击距离 |
| speed1 | | 行走的速度 |
| animation1 | | 控制动画播放 |
| hp | | 野怪的总血量 |
| normal1 | | 野怪正常体表颜色 |

### 2.7武器效果类：

表20 武器效果类详细设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | 属性 | 属性说明 | 方法 | 方法说明 |
| ShopWeaponItem | id | 武器ID | public void SetId() | 更新装备的显示 |
| ObjectInfo info | 武器信息 | public OnBuyClick() | 点击购买按钮 |
| UISprite icon\_sprite | 武器图标 |  | |
| UILabel name\_label | 武器名称 |  | |
| UIlabel effect\_label | 武器效果 |  | |
| UILabel pricesell\_label | 武器价格 |  | |

### 2.8角色生成类：

表21 角色成生成类详细设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | 属性 | 属性说明 | 方法 | 方法说明 |
| characterCreation | characterPrefabs | 角色预设 | UpdateCharacterShow() | 更新角色显示 |
| nameInput | 输入文本框 | OnNextButtonClick() | 点击下一个按钮，切换显示的角色 |
| characterGameObjects | 角色数组 | OnPrevButtonClick() | 点击上一个按钮，切换显示的角色 |
| selectedIndex | 数组下标 | OnOkButtonClick() | 存储选择的角色和昵称 |
| length | 角色个数 |  |  |

## 3 关键模块核心算法设计

### 3.1关于对敌人选中的关键技术

游戏中对于敌人的选中使用了unity提供的射线检测技术，大致流程是：当主角选中游戏场景中的某一个游戏物体时，我们需要判断玩家当前选中的物体是否为怪物。如果是怪物那么发起攻击，否则角色不产生任何行为。为了判断玩家当前选中的物体是否为怪物，我们向玩家鼠标所在的位置发出一道射线（无形的）。当射线碰撞到物体时会返回对该物体的引用，通过该引用对游戏物体的类别进行分析，检查该物体的标签是否为敌人。如果为敌人则发起攻击，否则本次选择无效，不产生任何行为。

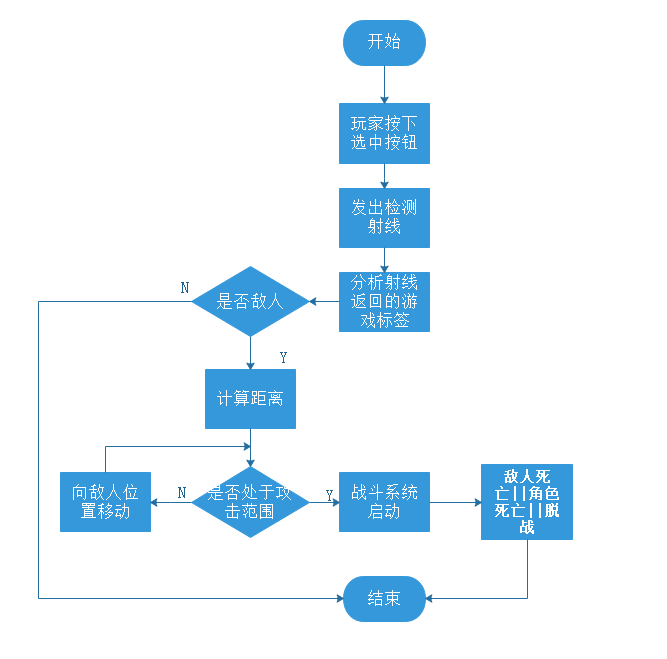


图19 敌人选中算法流程图

### 3.2开始界面算法设计：

游戏开始运行，初始场景为开始界面，摄像机移动呈现动画效果，同时显示游戏名称，任意键按钮闪烁。按任意键继续，显示两个按钮：开始新游戏按钮和存档按钮。如果点击开始新游戏按钮，跳转至输入昵称场景。如果点击加载存档按钮，跳转至存档时游戏场景，继续游戏。输入昵称场景，可以查看角色，输入玩家昵称。输入昵称后，点击确认进入游戏场景。

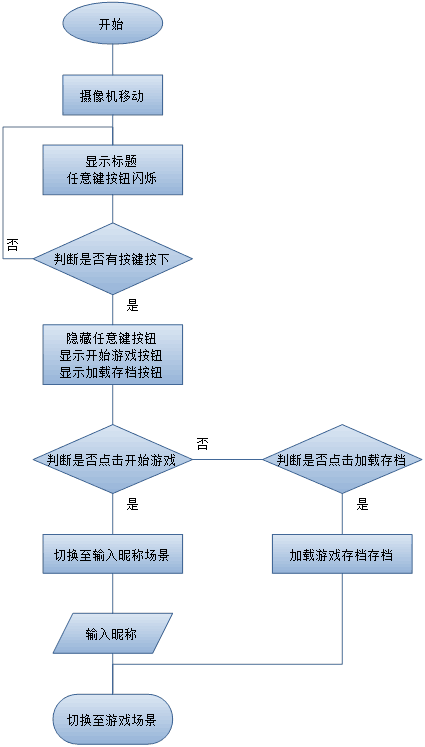


图20开始界面流程图

### 3.3野怪自动控制模块设计：

为了更加方便地描述野怪的状态，定义了一个枚举类型的变量记录野怪的5种状态，在Start（）方法里面初始化野怪的状态，初始化野怪的动画，声明GameObject对象，用于获取玩家对象，根据玩家对象可以获取到玩家的位置，定义变量存储野怪的动画，为了实现野怪攻击效果与攻击动画的同步，为每个攻击动画添加事件监听器，当播放某一个攻击动画时，保证执行一次对玩家造成伤害的动作。由于要实时检测野怪的状态，利用Update（）方法可以实现该效果，在Update（）方法中进行野怪状态更迭的判断。先判断野怪最大攻击范围内是否有玩家出现，若未检测到玩家，则野怪继续保持巡逻状态，若检测到玩家，则野怪切换到攻击状态，然后判断：如果玩家在野怪的最小攻击范围内，野怪可以直接进行攻击；如果玩家当前位置在野怪最小攻击范围到最大攻击范围之间，则野怪向玩家的位置移动，当移动到最小攻击范围后攻击玩家，同时保证当野怪与玩家进行交互时，野怪始终朝向玩家。如果野怪的状态是受到伤害状态，则播放野怪受到伤害动画；如果野怪的状态是死亡状态，则播放野怪死亡动画，然后在1.5秒后销毁该野怪对象。

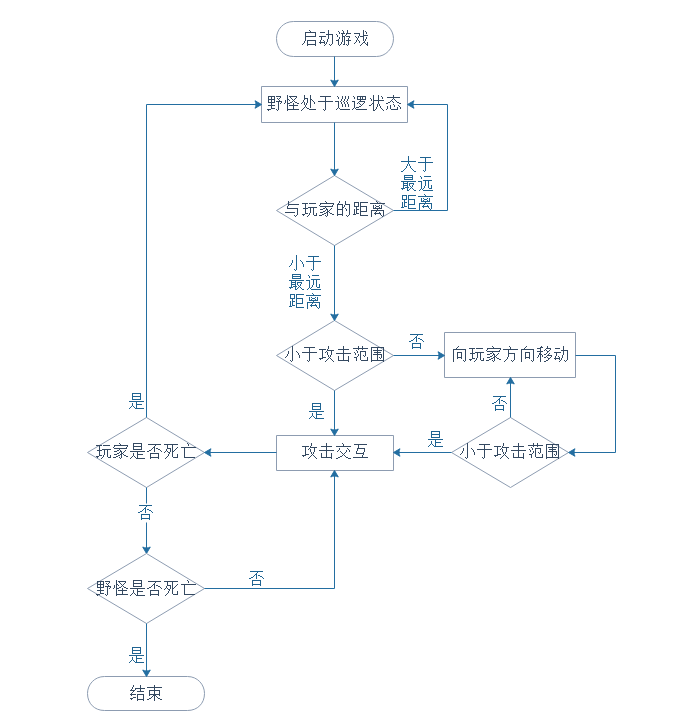


图21 野怪控制模块流程图