**3D RPG 项目计划**

1. **综合描述**

**3D Role Playing Game(RPG)是以Low Poly为主要美术风格的3D角色扮演游戏软件，其制作基于Unity3D引擎，并在Windows环境下运行。该软件中，用户可选择自己将要扮演的角色，通过键盘鼠标完成角色移动、角色打击等操作。**

1. **项目要求**
   1. **条件和限制**
      1. **开发环境**

**硬件环境：装载Windows 7及以上版本、RAM 8G及以上的计算机**

**软件环境：Unity 2020.3.x及以上版本**

* + 1. **运行环境**

**硬件环境：装载Windows 7及以上版本的计算机**

* 1. **环节定义** 
     1. **模块划分**

|  |  |  |  |
| --- | --- | --- | --- |
| **功能模块** | **功能点** | **功能说明** | **备注** |
| 角色创建与展现 | 角色选择 | 角色预设(Prefab)制作与创建 角色UI的处理 |  |
| 场景系统 | 3D场景搭建 | 搭建基本的地形，并制作完整场景  设置场景内物体的物理特性 |  |
| 3D场景气氛渲染 | 引入通用渲染管线（URP），渲染场景氛围 |  |
| 3D动画效果 | 设置不同角色（主角、NPC）的状态动画 |  |
| 控制层 | 角色控制 | 鼠标控制主角移动  键盘响应主角攻击技能 | 基本战斗系统 |
| 摄像机控制 | 键盘鼠标控制主角视角转换 |
| 游戏AI设计 | 敌人AI的开发 |
| 敌人的属性与动作设计 |
| 模型层 | 人物基本核心数值 | Player和monsters的基础数值（Max Health、Current Health、 Base Defense、Current Defense） | 技能数值平衡设计 |
| 人物战斗核心数值 | Player和monsters的攻击数值（Attack Range、Skill Range、Cool Down、Min Damage、Max Damage、Critical Multiplier、Critical Chance）  实现攻击计算 |
| 主角等级提升算法 | 实现Player LevelUP升级系统 |
| 视图层 | 登入系统 | 制作游戏主菜单 | 人物技能展示设计 |
| 数值UI系统 | 实现Player及Monster的基础数值显示 |
| 实现人物UI界面  LevelUp效果可视化 |
| 副本与战斗系统进一步开发 | 副本场景搭建 |  | 副本开发 |
| 副本场景优化 |  |
| UI预处理及场景切换 | 实现传送门 |
| 打包与发布 |  |  |  |
|  |  |  |  |

* + 1. **详细定义**

1. **场景系统**

**场景系统主要包括3D场景的搭建以及URP场景设置，其具体工作包括：**

**搭建基本的地形制作地形、在地图上设置树木、湖泊、草地、导入player和monster的模型。**

**制作风格化草地的模型、设置水体shader及模型，并在完成场景测试。**

**（以上为场景搭建）**

**设置场景内各种物理效果，主要指场景内角色及物体的碰撞检测，防止穿模**

**设置场景内AI导航，主要指地图的可行走性**

**设置场景内渲染效果，主要包括场景的环境渲染（Low Poly风格天空盒、迷雾效果等）以及角色模型的可见性（防遮挡的透视效果）**

**（以上为URP场景设置）**

1. **控制层**

**控制层主要包括人物的基本控制、镜头控制以及敌人AI系统的设计，其具体工作包括：**

1. **实现鼠标控制人物的移动，以及鼠标在不同的物体上显示不同的鼠标贴图。**
2. **实现对操作的不同动画效果的响应。角色状态可细分为行走walk、跑动run、攻击attack、死亡die、待机idle。**

**（以上为基本人物的控制）**

1. **实现摄像机跟随人物的效果以及通过键盘鼠标实现摄像头上下左右移动的效果模型层**

**（以上为镜头控制）**

1. **实现Monster对主角不同状态的动画响应。敌人的动画状态包括：行走walk、跑动run、攻击attack、死亡die、待机idle**
2. **设置敌人AI，实现范围内自动巡航、巡航范围内对Player的自动跟踪及攻击，并实现敌人脱战后的自动回复效果**
3. **设置大型敌人（Boss），实现Kill效果**

**（以上为敌人AI 导航、追踪、攻击）**

1. **模型层**

**模型层主要用于存储项目内Player以及各类Monster的关键数值，主要包括人物数值的设置以及人物升级系统的实现，其具体工作包括：**

1. **Player以及Monster的基础核心数值设置，包括：Max Health最大生命值、Current Health当前生命值、Base Defense基础防御、Current Defense等。**
2. **设置Player以及Monster的攻击核心数值，包括：Attack Range攻击伤害、Skill Range技能范围、Cool Down技能冷却时间、Min Damage最小伤害、Max Damage最大伤害、Critical Multiplier暴击倍数、Critical Chance暴击几率等。**
3. **设置算法，实现攻击数值的计算。**

**（以上为人物数值的设置）**

1. **实现Player LevelUP升级系统**

**（以上为人物升级系统）**

1. **视图层**

**视图层主要用于实现游戏内各种效果及算法的视觉实现，以各种UI设置及场景切换为主，其具体工作包括：**

1. **实现血条的显示**
2. **实现人物UI界面**
3. **制作游戏主菜单**

**（以上为UI界面）**

1. **制作传送门特效，并实现不同场景间的切换**

**（以上为场景切换）**

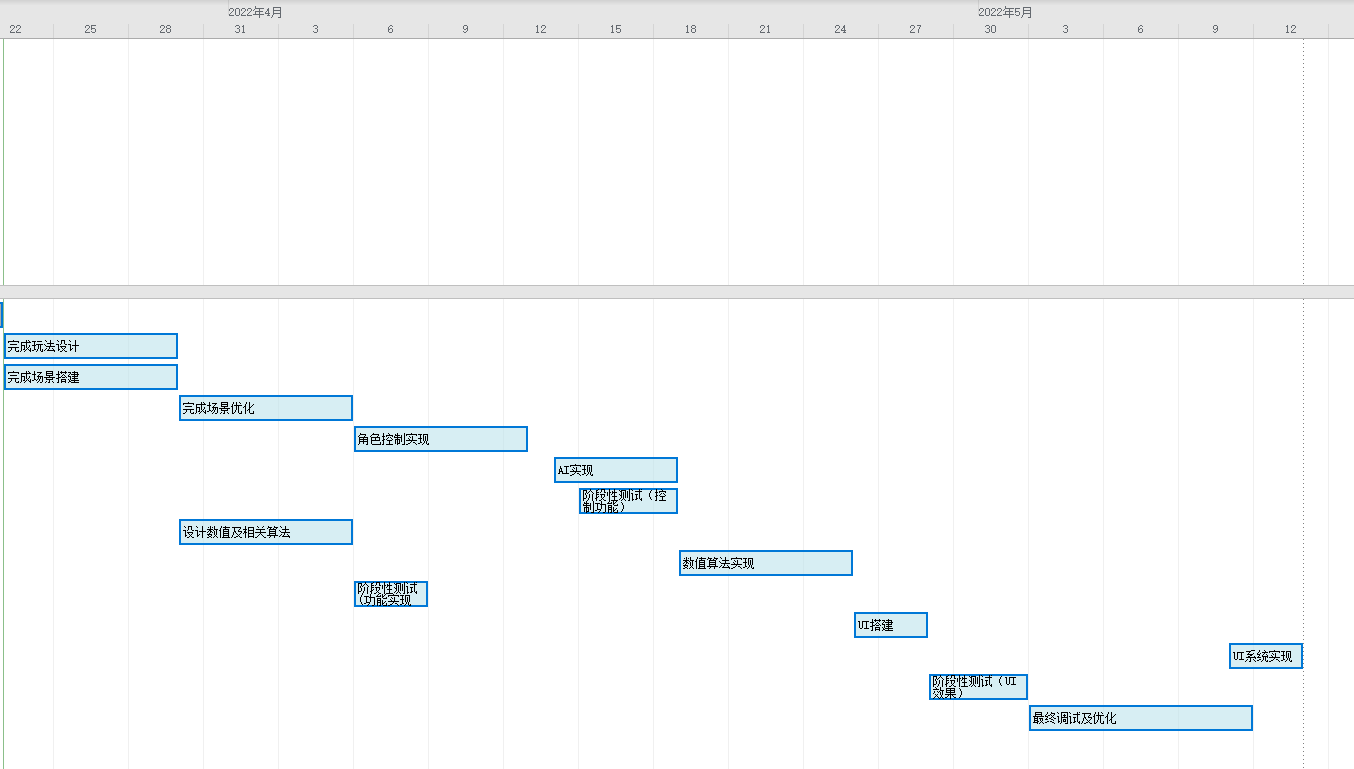
1. **数据保存**

**该部分为软件的收尾阶段。主要实现数据保存，并在测试完成后进行打包发布。**

1. **项目团队**
   1. **团队分工及职责**

|  |  |  |
| --- | --- | --- |
| **姓名** | **职位** | **具体职责** |
| **姚镕** | **场景&战斗策划** | **负责场景搭建以及场景的优化**  **设计游戏的具体内容及玩法，保证可玩性** |
| **朱佳倩** | **程序** | **负责各种数值算法及AI算法的具体实现** |
| **钟晴** | **数值策划/UI程序** | **设计数值模型，保证均衡性**  **实现数值的可视化** |

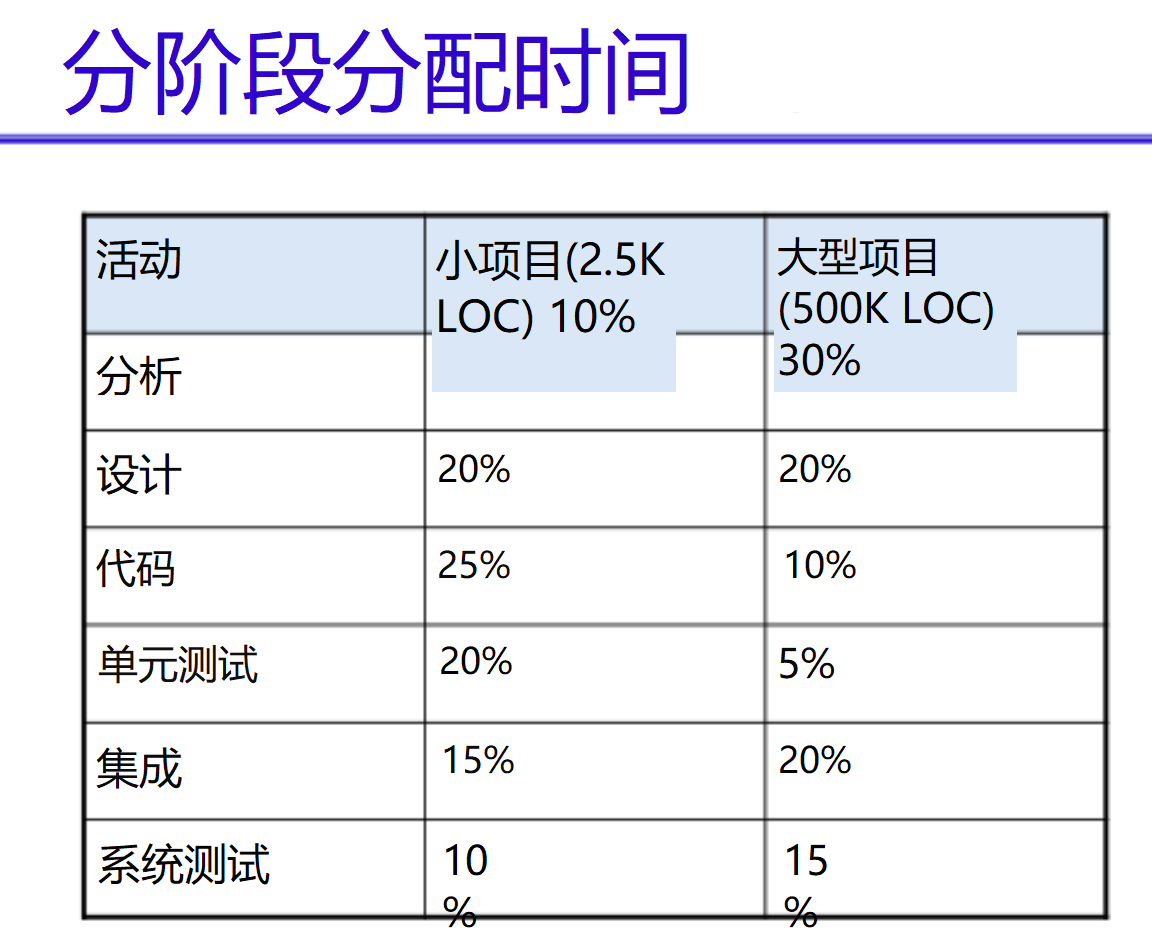
1. **日程安排**

****

**确定项目团队并定义角色和职责**

**确定项目产品和活动估计每个活动的工作量**

****

****