

PVZ base on unity

需求分析报告

所属学院：计算机与大数据科学学院

小组名称：当然是 *Relax* 最重要啦

小组成员：张一鸿、林建斌、饶清、许哲琿

杨涵轩、范彬洋、郑宜湉

俞健坤、郑子豪、王真平

指导老师：程永利

目录

一、引言	3
1.1 编写目的	3
1.2 项目介绍	3
1.3 适用人群和阅读建议	3
二、总体描述	4
2.1 产品愿景	4
2.2 产品特性	4
2.3 用户类型和特征	4
2.4 设计与实施约束	4
(1) 人员约束	4
(2) 技术约束	4
(3) 时间约束	4
2.5 假设和依赖	5
三、功能需求	5
3.1 用户用例图	5
3.2 类图	9
3.3 状态图	9
(1) 植物状态图	9
(2) 僵尸状态图	10
四、非功能需求	11
4.1 可维护性	11
4.2 可扩展性	11
4.3 可测试性	11
4.4 易用性	11

一. 引言

1.1 编写目的

本份报告目的在于明确说明“PVZ base on unity”各功能的实现方式，便于团队成员更好的了解项目情况，保证软件开发的质量和需求的完整性，同时明确所要开发的软件应具有的功能及约束，作为软件测试的总依据。

1.2 项目介绍

(1) 软件名称: PVZ base on unity;

(2) 开发团队: 软件工程-2021-C班 (福州大学) —— 当然是 Relax 最重要啦;

(3) 团队人数: 10 人;

(4) 项目简介: 基于 unity 实现的可玩性、娱乐性更强，受众更广的植物大战僵尸。

1.3 适用人群和阅读建议

- 1、 项目经理: 根据本文档了解预期产品的功能，并据此进行项目管理。
- 2、 设计员: 对需求进行分析，并据此设计游戏的基础框架。
- 3、 程序员: 配合设计人员的《设计报告》，进行游戏界面和相关代码的实现
- 4、 测试员: 根据本文档的需求对软件产品进行功能性测试和非功能性测试。
- 5、 用户: 了解产品的预期功能和相关规则。

在阅读本文档时，首先要初步了解产品的概貌，然后根据自身定位依据本文档进行进一步阅读学习。

二. 总体描述

2.1 产品愿景

基于 Unity 平台，复现经典游戏植物大战僵尸，并进一步优化游戏节奏，平衡度和创新功能。致力于实现一个可玩性、娱乐性更强，受众更广的新植物大战僵尸，让用户在休闲之余有一个操作简单、有趣、有成就感的游戏来让休息的时间变得更加美好。

2.2 产品特性

PVZ base on unity 在原版植物大战僵尸的基础上，实现了更多创新设计和优化用户体验的设计。如自动拾取阳光，铲子可攻击僵尸、调节游戏刷新率等。在游戏中开启疯狂模式后还有更多创新功能来提升游戏的可玩性与娱乐性，同时也满足不同配置、不同需求的游戏玩家提供了不同的游戏体验。并且项目可以应用于多个平台，有 Android、IOS、PC、Linux、MAC 等。

2.3 用户类型和特征

用户包括普通娱乐玩家和极限玩家类型。

普通娱乐玩家特征：较为佛性，主要目的为在游戏中获取快乐，追求简单，但又有成就感的游戏体验。主要为了游戏的通关，希望游戏关卡具有简单、花样多的特征。

极限玩家特征：追求极致的游戏体验，不仅仅为了游戏通关，还追求极限，想要以最小的代价获取最高的游戏进度。致力于挑战极限，不惧关卡的困难，甚至希望关卡更加困难。

2.4 设计与实施约束

(1) 人员约束

团队成员均为大三学生，共 10 人。植物制作 3 人，僵尸制作 3 人，UI 界面制作 2 人，文档及游戏优化 2 人。

(2) 技术约束

使用 Unity 平台进行和 C# 语言进行开发。

(3) 时间约束

开发时间约为 8 周。

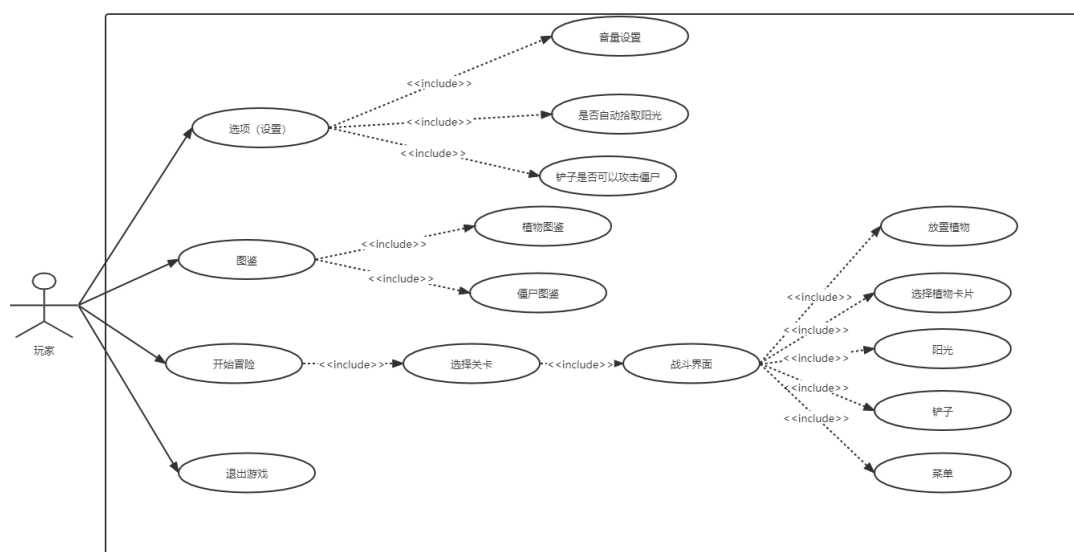
2.5 假设和依赖

本项目是否能够成功实施，主要依赖于以下的假设：

- (1) 团队成员的相互配合与积极学习，愿意为项目投入时间。
- (2) 项目的设计合理，可行。
- (3) 项目的分工合理。
- (4) 项目需求在确定后不会有较大的改动。
- (5) 项目能极大程度地还原原版游戏的逻辑。

三、功能需求

3.1 用户用例图



以下是各用例的文字描述：

<1>用例 1：修改设置

主参与者： 玩家

情景目标： 修改音量大小，设置游戏中能否自动拾取阳光，设置游戏中铲子能否攻击僵尸。

前提条件： 不是在正在游戏的状态。

触发器： 手动点击进入游戏设置面板。

场景：

1. 自定义勾选相关功能或调整音量
2. 关闭游戏设置面板

<2>用例 2：查看图鉴

主参与者： 玩家

情景目标： 可以查看任意植物或僵尸的具体已知信息。

前提条件：不是正在游戏的状态。

触发器：手动点击进入图鉴面板

场景：

1. 点击想要查看的植物或僵尸的图鉴
2. 关闭图鉴面板

<3>用例 3：开始冒险

主参与者：玩家

情景目标：进入游戏。

前提条件：当前界面在初始界面。

触发器：手动点击开始冒险。

场景：

选关后自动进入游戏，也可以不选关点直接进入游戏。

<4>用例 4：选择关卡

主参与者：玩家

情景目标：点击按钮进入对应关卡

前提条件：当前界面在选关界面

触发器：点击写有关卡数的按钮

场景：

进入对应的关卡开始游戏。

<5>用例 5：正在游戏

主参与者：玩家

情景目标：实现游戏的主体部分

前提条件：没有点击菜单

触发器：进入游戏

场景：

1. 放置植物阻止僵尸进入房子
2. 僵尸进入房子则游戏失败，僵尸全部被消灭则进入下一关

<6>用例 6：选择植物卡片

主参与者：玩家

情景目标：选择需要的植物卡片将其带入参与游戏

前提条件：当前界面在选择植物卡片界面

触发器：点击需要的卡片

场景：

选择需要的卡片，卡槽满之后除非放弃已选卡片将其空出否则不能再选择。

<7>用例 7：拾取阳光

主参与者：玩家

情景目标：拾取天空掉落或太阳花产出的阳光，并计入总阳光数

前提条件：在设置中没有勾选自动拾取阳光，正在游戏且游戏界面中出现可拾取的阳光

触发器：点击阳光

场景 a：

1. 用户点击游戏界面中掉落的阳光
2. 阳光飞向界面上边缘左侧的图标
3. 阳光总数相应增加

<8>用例 8：使用铲子消灭僵尸

主参与者：玩家

情景目标：使用铲子可消灭指定的一只僵尸

前提条件：在设置中勾选可以使用铲子攻击僵尸，正在游戏且铲子还没损坏，游戏界面中还有僵尸

触发器：点击铲子

场景：

1. 用户点击铲子
2. 再点击想要消灭的僵尸，表示使用铲子击打僵尸
3. 僵尸死亡

<9>用例 9：使用铲子铲除植物

主参与者：玩家

情景目标：使用铲子可铲去任一株植物

前提条件：铲子还没损坏

触发器：点击铲子

场景：

1. 用户点击铲子
2. 再点击想铲除的植物，表示使用铲子铲除植物
3. 植物被铲除，土地方格空出

<10>用例 10：放置植物

主参与者：玩家

情景目标：放置指定植物到指定的土地方格

前提条件：正在游戏，阳光数足够，植物卡片准备完毕，想放置的土地方格为空

触发器：点击植物卡片

场景：

1. 用户点击想使用的植物卡片
2. 用户点击想放置的土地方格
3. 植物成功种植

<11>用例 11：退出游戏

主参与者：玩家

情景目标：随时退出游戏

前提条件：无

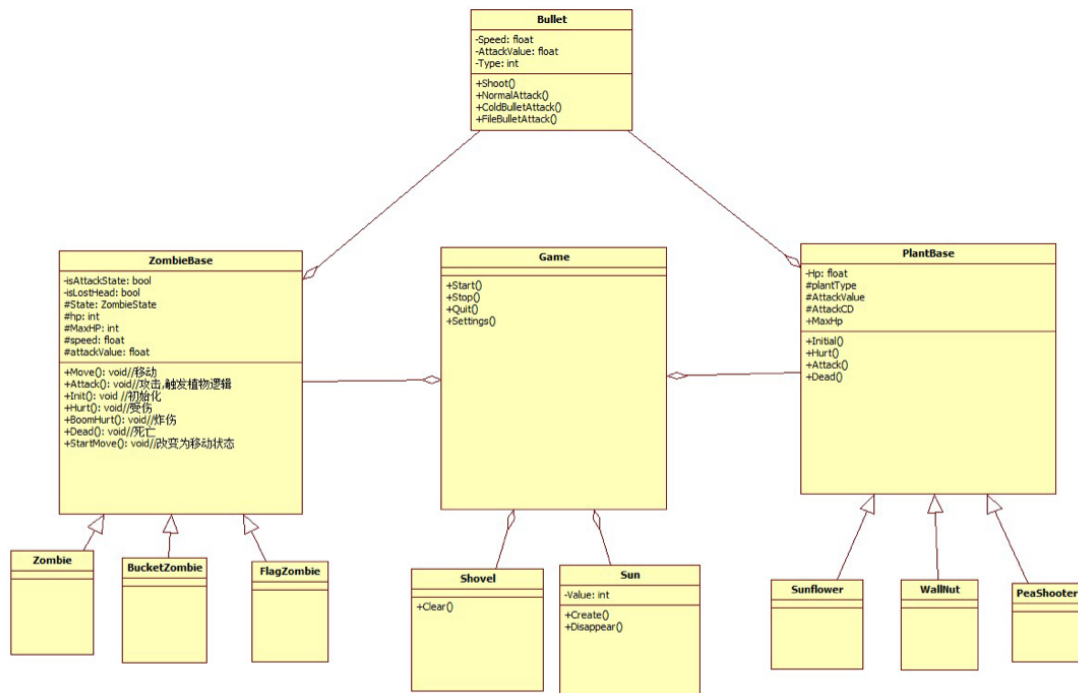
场景 a：正在游戏中

1. 点击菜单面板
2. 点击退出游戏

场景 b：当前页面是初始界面

点击退出游戏，关闭游戏程序

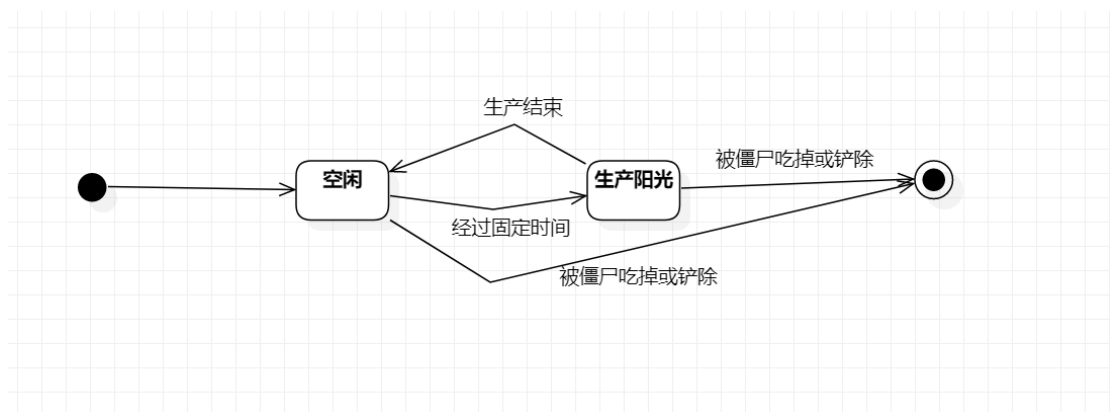
3.2 类图



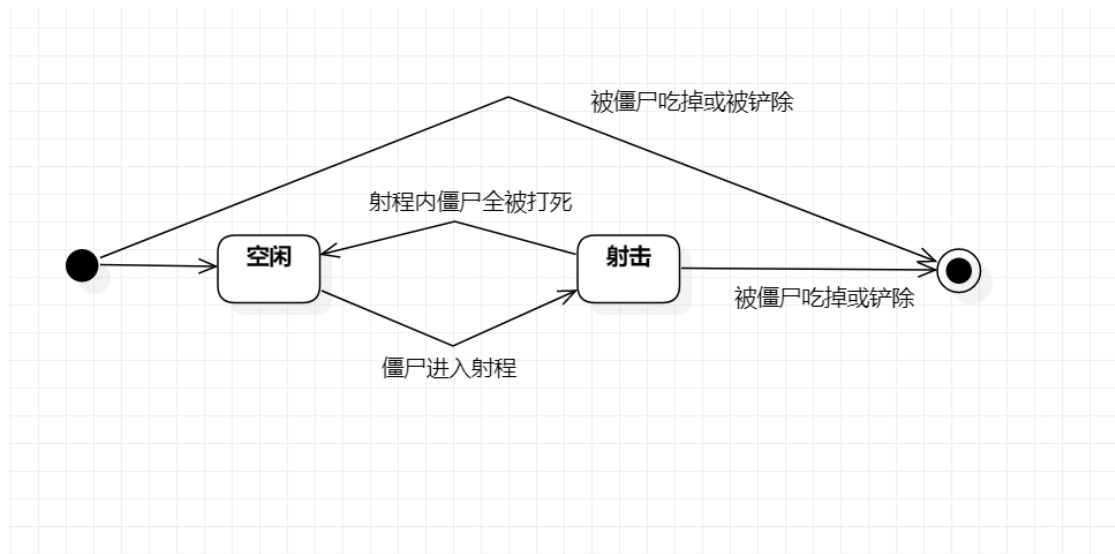
3.3 状态图

(1) 植物状态图

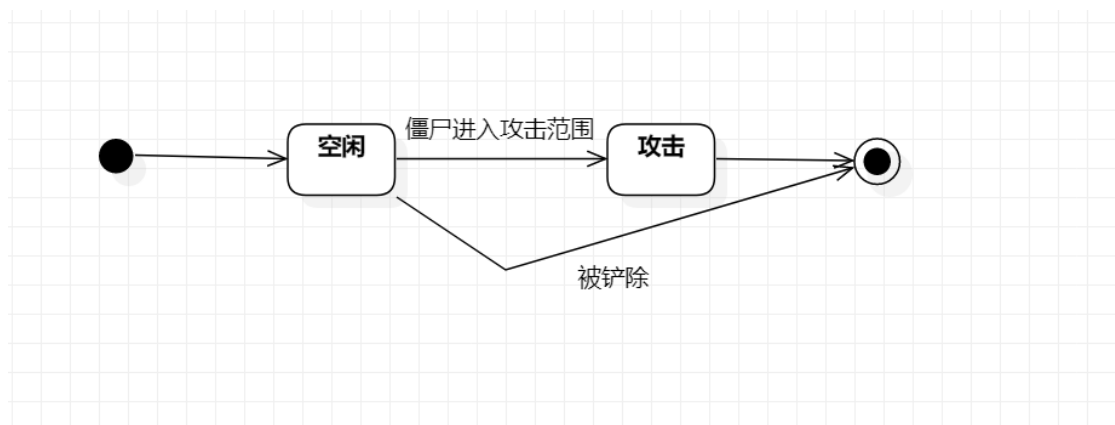
①生产类植物



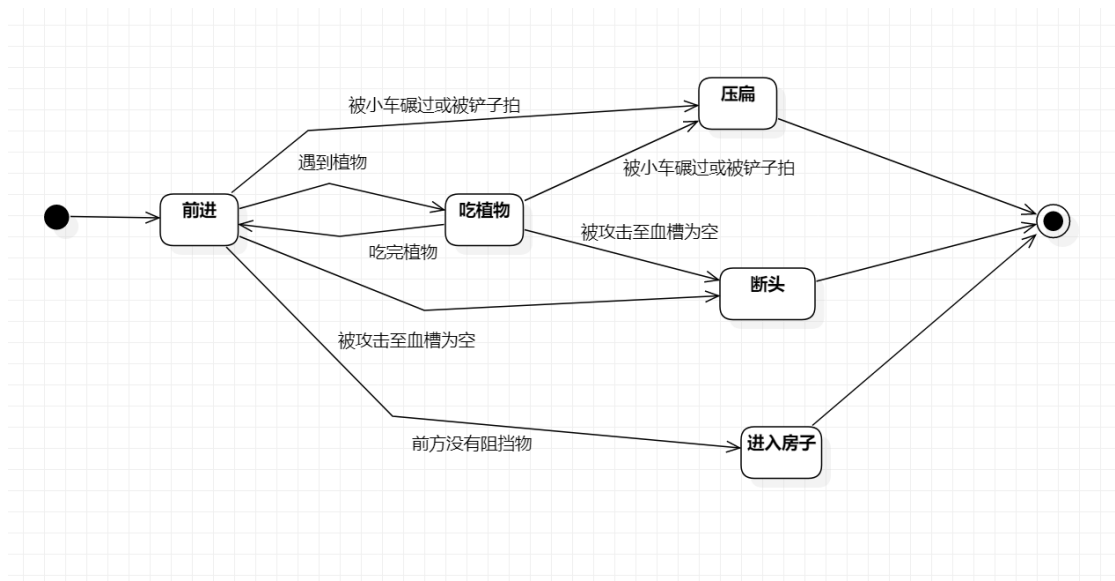
②射手类植物



③一次性攻击类植物



(2) 僵尸状态图



四、非功能需求

4.1 可维护性

本项目严格依据面向对象编程思想进行结构化设计，对于各个功能和游戏内的对象，如植物、僵尸等都要进行了类的抽象与封装。使得每一个模块单一化，更改其中一个模块时，对其他模块无影响，实现“高内聚，低耦合”的软件工程思想。

4.2 可扩展性

对植物、僵尸进行了基类的封装，同时代码文件要保证结构清晰，注释详细、清楚，以便在后期添加新植物、僵尸和新功能时，只需要添加其相关代码即可，不必进行多余的操作。

4.3 可测试性

基于 Unity 平台，能够对项目产生的问题进行快速定位与纠错，同时也能够通过可视化界面和控制台进行快速的测试和寻错。

4.4 易用性

前端 UI 界面设计要简洁、明了，利于第一次接触的玩家对于游戏的理解，同时要简化玩家除游戏内容外其他的多余操作，专注于游戏内容的品质，提高游戏的可玩性、娱乐性。