论文应该按照游戏的开发流程写,从开始的立项到确定制作的内容,中期的功能实现,场景调整,最后的测试和优化完善.

论文中应该包括了项目中的核心功能是如何实现的,应该尽可能的详细说明.

以下为项目中的一些核心功能实现思路和实际开发进程:

1. 场景障碍物随机生成
2. 场景任务(两组)随机生成
3. C#事件与委托的使用
4. Unity插件OOForm的学习和使用
5. VR游戏的UI以及交互
6. 宝箱开启和几率以及动画
7. 人物行走机制
8. 预制体动态加载

系统应该分为人物行走系统,任务系统,障碍物系统,开箱子系统

目录分为:引言,相关技术介绍,需求分析,系统设计,系统实现,系统技术分析

1. 引言
   1. 项目背景

康复训练,是患者在治疗后期进行的一种特殊的恢复措施,通过这种训练,可以让患者尽可能恢复正常或接近正常状态.这里说的状态不仅仅指单单身体上的,也包括了心理上的,通过康复训练,可以让治疗者在身体上和心理上都逐渐恢复正常，会大大消除或减轻患者功能上的障碍，极大地恢复治疗者的生活与劳动能力，重返社会和家庭。

现代的康复训练已经不是只有传统意义上的谈话方式的训练,还涉及到其他多个领域上的学术,比如物理学、运动学、心理学、护理学、老年学、社会学等等.治疗手段也不再仅限于药物和手术,更加注重了理疗、体疗、工疗及心理治疗，提倡自身功能训练.

但是,在长期试验中,传统康复训练的弊端渐渐浮现出来:一是受治疗者需要长时间重复同一个动作,心理很容易产生疲倦感,这种疲倦感若不加以重视任由其逐渐积累,久而久之,患者就会对治疗产生排斥心理,最后很可能导致患者拒绝继续治疗;二是传统康复治疗往往在一段时间内只会注重一个项目(比如中风患者的行走康复训练,只会让患者重复一个动作),这段时间其它的康复训练将不再进行,那么很容易将其他的训练荒废,造成”捡了芝麻丢了西瓜”这种情况.

* 1. 项目意义

为了有效解决传统的康复训练耗时耗力，单调乏味，并且训练强度和效果得不到及时评估，很容易错失训练良机这几项问题，结合了三维虚拟与仿真技术的康复训练就很好的解决了这些问题，并且还适用于心理患者的康复训练，对完全丧失运动能力的患者也有独特效果。我们把这种类型的康复训练简称为VR康复训练.

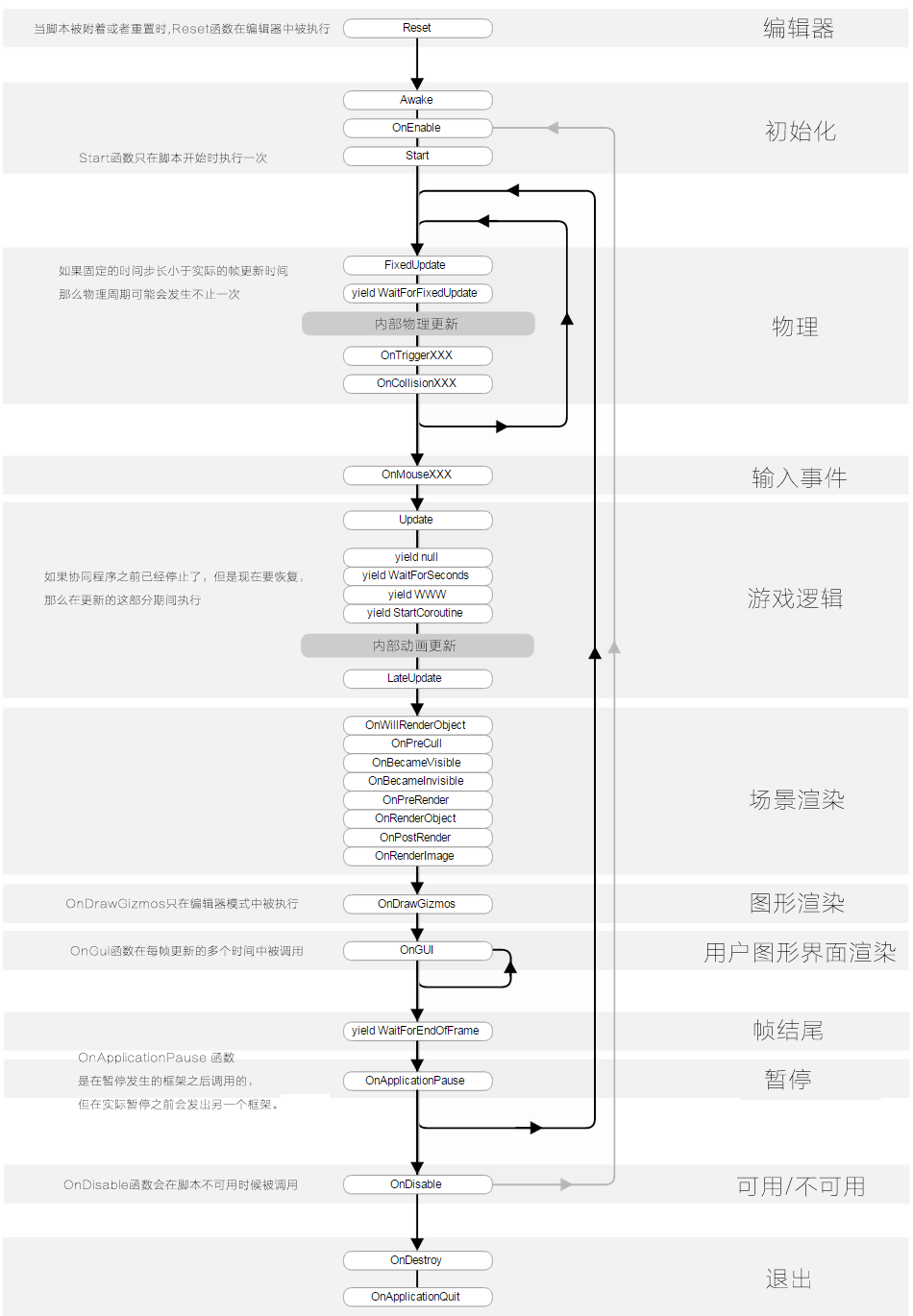
基于VR康复训练这一课题,结合当前的VR技术,此项目可以服务于行走不便的患者,让他们在助力器的帮助下,带上专业的VR设施,进行相关的康复训练,并且可以让患者在训练中不会感到过多的厌倦感,会让患者的康复周期大大缩短.不仅仅从身体上让患者渐渐恢复,更是从心理上给予患者极大地改善.

1. 相关技术简介
   1. Unity游戏引擎

Unity是目前使用最为广泛的一种游戏引擎之一,拥有极其强大的功能,几乎可以满足游戏开发中的所有需求.Unity主要运行在Windows操作系统上,有可视化图形窗口,并且操作较易,对有编程基础的开发人员来讲,上手较快.并且Unity引擎的跨平台性很强,目前支持包括手机游戏（iPhone、Android等）、PC（Windows、Mac、Linux）、网页游戏（基于各种主流浏览器）、游戏机专用游戏（Wii、Xbox、PS3、PS4）等平台的开发。

目前市场上有比较多的基于Unity引擎开发的游戏,移动端的有”炉石传说”、”神庙逃亡”、”武士2”、”神庙逃亡2”等,PC端的有”仙剑奇侠传”,”家园”等.网页游戏有

Unity脚本执行顺序见下图:



* 1. 插件OOForm以及其改进

OOForm是一个基于Unity的插件,来源为Unity Asset Store,这款插件是一款轻量级的表格管理插件,游戏开发人员可以通过这款插件进行游戏中的数据配置,属性调整.这款插件使用制表符”\t”来区分文本文件中的列,使用换行符”\n”来区分文本文件中的行,并且支持“Json”、“XML”和“CSV”格式文件,最终将文件中的数据以图形化界面展示在Unity Editorwindow中.

开发者可以使用封装的API读取一个文本文件,并根据行和列进行文件的读取或者存贮.游戏配置表中的一行数据通常作为一个数据类型,OOForm提供了接口函数public T GetObject<T>(int rowIndex),来获取一行类型为T的数据.

* 1. C#委托与事件

委托是一个类,它定义了方法的类型,允许将方法作为参数进行传递,使用委托可以执行相应的函数,免去程序中使用大量的if-else和switch语句.并且大大提升了程序的可扩展性.C#中使用delegate关键字来定义一个委托,然后使用函数调用的语法来调用委托.

delegate类能够拥有一个签名（signature），并且它"只能持有与它的签名相匹配的方法的引用"。它所实现的功能与C/C++中的函数指针十分相似。它允许你传递一个类A的方法m给另一个类B的对象，使得类B的对象能够调用这个方法m.总结C#委托如下:

1. 类似于C/C++中的函数指针；

2. 允许方法像变量一样赋值；

3. 作为回调函数来进行操作；

4. 委托以链表的方式存贮委托实例；

5. 委托实例中使用的类型并不需要与委托声明中的类型完全吻合；

6. 在C#2.0中引入了匿名委托，C#3.0引入了Lambda表达式作为委托实例。

事件是用户与应用程序交互的基础，它是回调机制的一种应用,是委托的一种应用类型,在游戏实际开发中,存在较多诸如收集,碰撞等功能需求,为了完成这一类需求,需要使用事件,在C#中使用事件主要需要如下步骤:

1. 创建一个委托
2. 将创建的委托与特定事件关联
3. 编写C#事件处理程序
4. 利用编写的C#事件处理程序生成一个委托实例
5. 将此委托实例添加到产生事件对象的事件列表中
   1. VS结合Unity动态调试

Unity开发者在对代码进行调试时,会选择UnityVS这个插件,这款插件是一款Microsoft Visual Studio C#辅助开发插件.

UnityVS通过UDP将Unity和Visual Studio连接,并且只有在存在Unity实例的时候,UnityVS才可以完成连接.在完成连接之后,游戏开发者可以在程序中设下断点并将Visual Studio附加到Unity进程上,这样在Unity实例播放的时候,当代码运行到断点时,Visual Studio会自动暂停代码运行并实时显示当前内存中的变量.方便开发者准确定位BUG并及时作出有效的修复.

1. 需求分析
   1. 概述