

**成人高等教育毕业论文(设计)**

基于Unity3D的小猫快跑游戏设计与实现

|  |  |  |  |
| --- | --- | --- | --- |
| **学生姓名** | **尹圣坤** | **学 号** | **2020311645** |
| **指导教师** | **李志鹏** | **学习形式** | **函授** |
| **年 级** |  | **学习层次** | **专升本** |
| **学 院** | **合肥工业大学继续教育学院** | **站 点** | **合肥** |
| **专 业** | **计算机科学与技术** | | |
| **完成日期** | **2022 年 04 月 18 日** | | |

**合肥工业大学继续教育学院**

# 独 创 性 声 明

本人声明所呈交的毕业论文是本人在导师指导下进行的工作及取得的成果。我声明，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的论文及研究成果，也不包含为获得 合肥工业大学 或其他教育机构的学位或证书而使用过的材料。

论文作者签名： 签字日期： 年 月 日

工作单位： 电话：

通讯地址： 邮编：

# 基于Unity3D的小猫快跑游戏设计与实现

# 摘 要

随着计算机技术的发展、科技化的信息技术发展，电子游戏的形式越来越丰富了，用户对于电子游戏的娱乐性需求也越来越高，追求更加真实的感官体验成为游戏的一条发展趋势。目前3D游戏已经在电脑、手机、机顶盒等多平台得到了普及，相对于传统的2D游戏，3D游戏让用户感觉更加融入，更加享受，但所需要的计算机技术也会更复杂，计算机图形学、物理模拟等都成为3D游戏开发所需要的重点研究方向。目前我们在电子游戏领域与领先国家还有一定的差距，为了振兴电子游戏行业，对这一综合应用所涉及的技术进行整体的梳理是完全有必要的。

本设计设计的基于Unity3D的小猫快跑游戏采用的游戏场景，具有合理巧妙的安排，精心的设计了各种物理碰撞检测器，游戏场景中有鱼骨头和各种道具的设置。游戏设计采用Unity2019.2.9版本的游戏引擎，利用JetBrains Rider2019.3.1作为脚本代码编译器，开发时运用了软件工程设计过程设计和实现系统功能。

该游戏中主界面简洁，操作简单，可玩性强，游戏的各个功能模块均已经测试过，可以稳定运行。

**关键词:** 酷跑游戏；Unity 3D

# Design and Implementation of Kitten Running Game Based on Unity3D

# Abstract

With the development of computer technology and the development of technological information technology, the forms of video games are becoming more and more abundant, and the entertainment needs of users for video games are also getting higher and higher, and the pursuit of a more realistic sensory experience has become a development trend of games . At present, 3D games have been popularized on multiple platforms such as computers, mobile phones, and set-top boxes. Compared with traditional 2D games, 3D games make users feel more integrated and enjoy, but the required computer technology will also be more complex. Computer graphics, Physical simulation has become a key research direction required for 3D game development. At present, we still have a certain gap with the leading countries in the field of video games. In order to revitalize the video game industry, it is absolutely necessary to sort out the technologies involved in this comprehensive application.

The game scene used in the Unity3D-based kitten running game designed in this subject has reasonable and ingenious arrangements, and various physical collision detectors are carefully designed. There are gold coins and various props in the game scene. The game design uses the Unity2019.2.9 version of the game engine, uses JetBrains Rider2019.3.1 as the script compiler, and uses the software engineering design process to design and implement system functions during development.

The game interface is simple, the operation is simple, and the playability is strong. Each functional module of the game has been tested and can run stably.

**Keywords:** Cool Running Games;Unity 3D

# 目 录

[1 引言 1](#_Toc9123107)

[1.1 Unity 3D介绍 1](#_Toc9123108)

[1.2 Unity 3D的前景 1](#_Toc9123109)

[1.3 设计开发意义 2](#_Toc9123110)

[1.3.1 设计研究内容 2](#_Toc9123111)

[1.3.2 设计需要解决的问题 3](#_Toc9123112)

[2 平台简介 5](#_Toc9123113)

[2.1 开发平台介绍 5](#_Toc9123114)

[2.1.1脚本编辑器JetBrains Rider2019.3.1 5](#_Toc9123115)

[2.1.2 Unity3D游戏引擎 5](#_Toc9123116)

[2.2 小猫快跑架构分析 6](#_Toc9123118)

[2.2.1 游戏场景及一系列脚本代码 6](#_Toc9123119)

[2.2.2 游戏框架简介 7](#_Toc9123120)

[3 需求分析 9](#_Toc9123121)

[3.1 系统概述 9](#_Toc9123122)

[3.2 系统简介 9](#_Toc9123123)

[3.2.1 成绩 10](#_Toc9123124)

[3.2.2 设置 10](#_Toc9123125)

[3.2.3 关于 10](#_Toc9123126)

[3.2.4 开始游戏 11](#_Toc9123127)

[4 系统概要设计 12](#_Toc9123128)

[4.1 游戏设计 12](#_Toc9123129)

[4.1.1 小猫快跑游戏背景 13](#_Toc9123130)

[4.1.2 游戏功能的设计 13](#_Toc9123131)

[4.1.3路段类 13](#_Toc9123132)

[4.1.4道具类 14](#_Toc9123133)

[4.1.5障碍类 15](#_Toc9123134)

[4.1.6 游戏中对碰撞逻辑的说明 16](#_Toc9123167)

[4.1.7 小猫动画模型播放规则 16](#_Toc9123168)

[4.2 系统架构 16](#_Toc9123169)

[4.3 系统功能结构 17](#_Toc9123170)

[4.4 系统功能模块简介 17](#_Toc9123171)

[4.4.1 成绩功能 17](#_Toc9123172)

[4.4.2 关于功能 18](#_Toc9123173)

[4.4.3 开始游戏功能 18](#_Toc9123174)

[4.4.4 设置功能 18](#_Toc9123175)

[4.4.5 暂停功能 18](#_Toc9123176)

[4.4.6 退出功能 18](#_Toc9123177)

[5 游戏中主界面设计 19](#_Toc9123178)

[5.1 系统结构设计 19](#_Toc9123179)

[5.2 游戏初始界面设计 19](#_Toc9123180)

[5.2.1缓冲条的实现 20](#_Toc9123181)

[5.3 游戏主界面 20](#_Toc9123182)

[5.4 设置选项界面 21](#_Toc9123183)

[5.5 排行榜界面 22](#_Toc9123184)

[5.6 关于界面 23](#_Toc9123185)

[5.7 游戏中主界面 24](#_Toc9123186)

[5.8 游戏完成界面 26](#_Toc9123187)

[6 系统详细设计 27](#_Toc9123188)

[6.1 模型的布局分析 27](#_Toc9123189)

[6.2.1 WelcomeButton类 28](#_Toc9123190)

[6.2.2 Gradisplay类 28](#_Toc9123191)

[6.2.3 Explations类 29](#_Toc9123192)

[6.2.4 Options类 29](#_Toc9123193)

[6.3 游戏的详细设计及实体类 30](#_Toc9123194)

[6.3.1 静态类的创建 30](#_Toc9123195)

[6.3.2 小猫的跑动和动画播放 30](#_Toc9123196)

[6.3.3 森林类的实现 31](#_Toc9123197)

[6.3.4 森林路段的无限生成 32](#_Toc9123198)

[7 程序调试与测试 33](#_Toc9123199)

[7.1 程序调试 33](#_Toc9123200)

[7.1.1 小猫跳跃但不降落 33](#_Toc9123201)

[7.1.2 游戏过程中出现闪动小猫图标 33](#_Toc9123202)

[7.2 测试 33](#_Toc9123203)

[7.2.1 单元测试 34](#_Toc9123204)

[7.2.2 组合系统测试 34](#_Toc9123205)

[7.2.3 用户测试 34](#_Toc9123206)

[总 结 35](#_Toc9123207)

[参 考 文 献 36](#_Toc9123208)

[致 谢 37](#_Toc9123209)

# 1 引言

科技的迅速发展使手机硬件以惊人的速度更新，目前手机游戏已经具有了非常好的娱乐性和复杂的场景效果，以及交互性的提升和便于携带，手机游戏已经超越了电脑上的网页式游戏，手机游戏在市场的占有率会越来越大。

## 1.1 Unity 3D介绍

Unity 3D 也称 Unity，是由 Unity Technologies 公司开发的一个可以让用户轻松创建三维视频、游戏、建筑可视化、实时三维动画等类型的互动内容的多平台的综合性游戏开发引擎工具。

Unity 3D 可以运行在 Windows 和MacOS X 下，可发布游戏至 Windows、Mac、Wii、iPhone、WebGL（需要 HTML5）、Windows Phone 8 和 Android 平台。也可以利用 Unity Web Player 插件发布网页游戏，支持 Mac 和 Windows 平台的网页浏览，是一个全面整合的专业游戏引擎。

业界现有的商用游戏引擎和免费游戏引擎数不胜数，其中最具代表性的商用游戏引擎有 UnReal、CryENGINE、Havok Physics、Game Bryo、Source Engine 等，但是这些游戏引擎价格昂贵，使得游戏开发成本大大增加。而 Unity 公司提出了“大众游戏开发”（Democratizing Development）的口号，提供了任何人都可以轻松开发的优秀游戏引擎，使开发人员不再顾虑价格。

Unity 的中文意思为“团结”。Unity 的核心含义是想告诉大家，游戏开发需要在团队合作基础上相互配合完成。

时至今日，游戏市场上出现了众多种类的游戏，它们是由不同的游戏引擎开发的，Unity 3D 以其强大的跨平台特性与绚丽的 3D 渲染效果而闻名于世，现在很多商业游戏及虚拟现实产品都采用 Unity 3D 引擎来开发。

## 1.2 Unity 3D的前景

2004 年，Unity 3D 诞生于丹麦的阿姆斯特丹。

2005 年，发布了 Unity 1.0 版本，此版本只能应用于 Mac 平台，主要针对 Web 项目和 VR（虚拟现实）的开发。

2008 年，推出 Windows 版本，并开始支持 iOS 和 Wii，从众多的游戏引擎中脱颖而出。

2009 年，荣登 2009 年游戏引擎的前五，此时 Unity 的注册人数已经达到了 3.5 万。

2010 年，Unity 3D 开始支持 Android，继续扩大影响力。

2011 年，开始支持 PS3 和 XBox360，此时全平台的构建完成。

2012 年，Unity Technologies 公司正式推出 Unity 4.0 版本，新加入对于 DirectX 11 的支持和 Mecanim 动画工具，以及为用户提供 Linux 及 Adobe Flash Player 的部署预览功能。

2013 年，Unity 3D 引擎覆盖了越来越多的国家，全球用户已经超过 150 万，Unity 4.0 引擎已经能够支持在包括 MacOS X、Android、iOS、Windows 等在内的 10 个平台上发布游戏。同时，Unity Technologies 公司 CEO David Helgason 发布消息称，游戏引擎 Unity 3D 今后将不再支持 Flash 平台，且不再销售针对 Flash 开发者的软件授权。

2014 年，发布 Unity 4.6 版本，更新了屏幕自动旋转等功能。

2016 年，发布 Unity 5.4 版本，专注于新的视觉功能，为开发人员提供了最新的理想实验和原型功能模式，极大地提高了其在 VR 画面展现上的性能。

## 1.3 设计开发意义

随着科学技术的发展，手机已经成为我们生活中必不可少的一部分。并且手机游戏是其中的重要组成。现在随着智能手机的出现，手游已经具有很好的娱乐性和复杂的效果描述，交互性的提升和便捷的携带，手机游戏已经超越电脑中的网页游戏。Unity3D广泛用于开发手机平台上的应用和游戏，成为移动游戏领域较为优秀的3D游戏引擎。随着移动设备的性能不断提升，人们开始逐渐习惯于在移动设备上寻求乐趣，加之一系列物理引擎对移动设备的支持，移动端更加逼真的效果已不再遥远。

本设计是一款基于Android平台的手机休闲跑酷类游戏，本游戏使用当前最为流行的Unity 3D开发工具，结合智能移动设备的触摸技术打造的一款小型移动端游戏。用户通过滑动或者点击屏幕来实现小猫转弯、跳跃、下蹲等效果。游戏充分发挥了酷跑类型的游戏节奏，能充分发挥用户的反应能力。在游戏中用户通过滑动或者点击屏幕控制不断奔跑的小猫，通过小猫的奔跑速度和场景中的障碍来及时控制小猫的行动，以此来锻炼用户的反应能力，给用户很好的游戏体验。

### **1.3.1 设计研究内容**

目前较流行的跑酷类游戏有《小黄人快跑》《地铁酷跑》《熊出没》等等。《小猫快跑》是一款第三人称休闲跑酷类游戏。当用户点开始游戏按钮，小猫就开始向前跑并且记录成绩，路上会出现静态的障碍以及动态的障碍，用户通过上下左右滑动或者点击屏幕来躲避碰撞，收集场景中的鱼骨头得到奖励。本设计设计的游戏操作简单，界面渲染效果强，采用移动端设备为载体，操作方便，适合全部年龄段的人进行游戏，该游戏还需要用户大小脑的配合，带给用户很好的游戏操作及视觉体验。

本设计使用Unity 2019.2.9版本的游戏引擎作为开发工具和JetBriains Rider2019.3.1作为C#脚本代码的编辑工具，结合移动端设备的触摸技术。在Unity3D中新建一个Scene，导入美术做的模型资源，然后对游戏场景中的场景地图、灯光、天空盒等路段的设置来动态搭建游戏场景，再对主相机进行一系列设置。然后创建主角小猫对象，为它添加Mecanim动画组件绑定身体上的所有骨骼和一系列的动作控制脚本代码。然后是各种碰撞检测器的创建，通过碰撞检测来判断主角碰到的物体是可以拾取的还是障碍。还有鱼骨头的创建和磁铁道具、分数加倍道具、金砖道具、无敌道具的创建。本设计可以运行在Android 5.0或者更高版本的移动端设备上。

### **1.3.2 设计需要解决的问题**

（1）游戏场景路段的组合和组建方面的问题

本设计的场景搭建是一切的前提，首先需要创造一个游戏的运行路段，利用脚本代码控制路段的无限生成。设计中共有六个游戏场景路段，场景路段通过随机的方式生成下一个路段，这可以让用户看到游戏的变化。并且路段经过了特殊渲染器的处理，把模型的顶点做了一定的坐标偏移，使整个场景看起来是弯曲的，这能增强用户的3D感觉。

(2)游戏各个方面功能的问题

在小猫快跑游戏中，整体游戏的功能的实现是此设计的中心。该游戏设计有多项功能，用户可以自主地选择要体验的功能，给用户良好的游戏画面和操作体验：

1. 启动游戏，首先进入到游戏的欢迎界面，随后进入到游戏的加载数据界面，加载数据使用的是异步的加载技术，异步加载的好处是可以使用户实时的看到游戏加载时的进度。

加载完成后进入到游戏的主界面，在主界面可以通过点击不同的按钮进入到到不同的系统界面。点击”RANKING”按钮，进入到排行榜界面，排行榜界面按降序的方式排行显示每一次游戏完成的分数。点击”关闭”按钮回到主界面。点击”SETTING”按钮，进入到设置选项界面。通过滑动或者点击滚条来设置音乐的音量大小和音效音量的大小。点击”关闭”按钮回到主界面。点击设置界面中的“ABOUT”按钮，进入到关于界面。点击”关闭”按钮回到主界面。在主界面点击手机的返回按钮可以退出游戏。

1. 在主菜单选择”RUN”，进入到游戏场景界面，在这里用户开始正式玩游戏。游戏中主界面的左上角是暂停按钮，右上角是鱼骨头数量计数板。当小猫收集到道具时游戏中主界面的左下角会出现道具的剩余时间进度条。当用户在游戏中主界面点击暂停按钮后，会弹出暂停界面。点击“MAIN MENU”按钮可以返回主界面；点击“RESUME”按钮可以继续游戏。
2. 当小猫撞到障碍时弹出游戏完成界面，界面中会显示本次的分数和是前五名的排名情况。点击“MAIN MENU”按钮，返回主界面，点击“RUN”按钮，重新开始一局游戏，点击“LEADERBOARD”按钮，弹出排行榜界面，点击“STORE”按钮，弹出商城界面。点击“MISSIONS”按钮，弹出任务界面。

(3) 小猫酷跑碰撞检测逻辑的难点

在游戏开始时小猫生成在屏幕的中间道路上，游戏场景中的障碍是根据一定概率随机生成的，因此用户需要操作小猫躲避障碍。碰撞检测的代码每一帧都在检测小猫是否有碰撞行为，如果没有碰撞到障碍物，则游戏可以继续运行，如果碰撞到了障碍物就会弹出游戏完成界面。

# 2 平台简介

每一款游戏都是需要一个合适的开发平台的，手游的开发平台虽然多，但合适的平台是程序员开发时得力的助手，是游戏初始设计的灵魂。

## 2.1 开发平台介绍

在基于Unity3D的小猫快跑游戏开发中，主要运用的工具有两个：一个是Unity 3D的游戏引擎，它用于小猫快跑游戏的开源开发框架，可以为游戏的开发者构建3D游戏。通过操作Unity3D的场景界面，用户可以去完成任何他想完成的工作；一个是JetBrains Rider2019.3.1脚本代码编辑器，可以用来编辑开发C#代码。

### **2.1.1 脚本编辑器JetBrains Rider2019.3.1**

Rider 可以帮助您在 Windows、macOS 和 Linux 上开发 .NET、ASP.NET、.NET Core、Xamarin 或 Unity 应用程序。 从 C#、VB.NET 和 F# 到 ASP.NET Razor 语法、JavaScript、TypeScript、XAML、XML、HTML、CSS、SCSS、JSON 和 SQL，它为 .NET 开发中使用的语言提供了丰富的编辑支持和代码洞察。

### **2.1.2 Unity3D游戏引擎**

Unity是一种跨平台的2D和3D游戏引擎，由Unity Technologies研发，可开发跨平台的视频游戏、并延伸于基于WebGL技术的HTML5网页平台，以及tvOS、Oculus Rift、ARKit等新一代多媒体平台。除可以用于研发电子游戏之外，Unity还广泛用作建筑可视化、实时三维动画等类型互动内容的综合型创作工具。Unity最初于2005年在苹果公司的全球开发者大会上对外公布并开放使用，当时只是一款面向Mac OS X平台的游戏引擎。时至2018年，该引擎所支持的研发平台已经达到27个。

## 2.2 小猫快跑架构分析

本游戏设计开发时，在做好前期的策划和基本资源准备后，就要开始设计游戏中的场景和功能代码的开发编写，初步描绘整体游戏的架构。

### **2.2.1 游戏场景及一系列脚本代码**

基于Unity3D的小猫快跑游戏开发中，场景开发是游戏开发的主要工作。每个场景包含多个游戏对象，其中某些对象还被附加了特定功能脚本代码。本游戏包含包含三个游戏场景，接下来对几个场景进行简要的介绍。

（1）主菜单场景

“Welcome”是游戏场景的中心场景，该游戏场景中有多个游戏对象，主要包括主摄像机、小猫、森林等模型或者场景对象。在该场景中可以通过点击按钮进入到其他界面，如游戏中主界面，设置选项界面，关于界面，排行榜界面等，该场景中包含的脚本代码如图2.1所示。

小猫下滑监视器

(CrashSmallColider.cs)

小猫跳跃监视器

(CrashBigCollider.cs)

小猫(WelcomeRun.cs,WelcomeAnimation.cs)

主摄相机(WelcomeButton.cs,GradeDisplay.cs,

Options.cs,Explations.cs,SmoothFollw.cs)

主菜单

场景

**图2.1主场景的框架图**

（2）游戏场景

游戏场景”StartGame”是本游戏最重要的场景，该场景中有多个游戏对象，主要有摄像机，小猫，森林，障碍，鱼骨头，磁铁，金砖等模型或者场景对象。本游戏场景中的其他对象，如特等粒子渲染效果、鱼骨头模型等被制作成预制作。该场景中包含的脚本代码如图2.2所示。

主摄相机(DashBoard.cs,EnvGenerator.cs,

GameControler.cs,SmoothFollw.cs)

游戏结束监视器

(GameOver.cs)

小猫(DogMove.cs,

DogAnimation.cs)

小猫跳跃监视器

(CrashBigCollider.cs)

小猫下蹲监视器

(CrashSmallColider.cs)

游

戏

场

景

**图2.2游戏场景的框架图**

（3）加载场景

加载场景”Loading”用于实现游戏场景的异步加载，显示进度条。场景中包含主摄像机”Main Camera”，其功能是实现异步加载到其他的游戏场景、绘制加载进度条等，这样可以动态地显示加载进度。该场景中包含的脚本代码如图2.3所示。

主摄像机(Loader.cs)

加载场景

**图2.3加载场景的框架图**

### **2.2.2 游戏框架简介**

（1）打开本游戏，首先进入到的是主场景”Welcome”,主摄像机”Main Camera”被激活，其挂载的脚本代码”WelcomeButton.cs”和挂载在”Player”的游戏对象的脚本代码”WelcomeRun.cs”开始执行，显示出主界面，小猫原地不动。如果设置了背景音乐为可播放，则同时播放背景音乐。

（2）在主界面点击”开始游戏”按钮后，则游戏进入到场景”StartGame”.该场景的主相机激活，开始执行其上挂载的脚本代码”DashBoard.cs”,”EnvGenerator.cs”,

”GameControler.cs”,”SmoothFollw.cs”

（3）在脚本代码”Forests.cs”的作用下，在场景上生成鱼骨头，磁铁，金砖和障碍等。在脚本代码”SmoothFollow.cs”的作用下摄像机跟随小猫移动，脚本代码”DashBoard.cs”用于绘制游戏场景的暂停按钮，鱼骨头数量计数板，道具剩余时间等。

（4）“Player”游戏对象挂载的脚本代码”DogMove.cs”开始执行，小猫在”DogMove.cs”脚本代码的作用下不断地向前奔跑。“Player”游戏对象下的子对象“Dog”上挂载“Player”的”DogAnimation.cs”用于让用户控制小猫跳跃，下蹲，左转和右转。

（5）主界面点击”成绩”按钮，主摄像机上挂载的脚本代码”WelcomeButton.cs”被禁用，而脚本代码”GradeDisplay.cs”被启用，进入到排行榜界面。

（6）主界面点击”设置”按钮，主摄像机上挂载的脚本代码”WelcomeButton.cs”被禁用，而脚本代码”Options.cs”被启用，进入到设置选项界面。

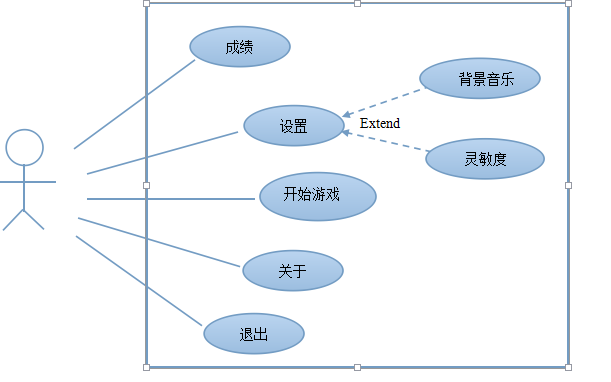
（7）主界面点击”关于”按钮，主摄像机上挂载的脚本代码”WelcomeButton.cs”被禁用，而脚本代码”Explations.cs”被启用，进入到关于界面。

# 3 需求分析

需求分析是对一个系统中疑难问题的解决方法进行详细的分析，一般在需求分析中要弄清系统的整个架构，有哪些需求要实现，通过输入一些怎样的数据，通过实现，输出怎样的结果。一个好的需求分析能够定位我们将要进行的事情和要达到的最终成果。

## 3.1 系统概述

游戏分为五个大模块：开始游戏、成绩、关于、设置、退出。游戏采用主菜单场景，用户可以自由地选择要实现的功能。成绩模块显示用户游戏信息，设置模块进行游戏设置，开始游戏模块实现运行游戏，关于模块介绍游戏开发者信息，退出模块用于退出游戏。系统的用例图如3.1所示。



**图3.1 系统用例图**

## 3.2 系统简介

在安卓版的小猫酷跑游戏系统中，总共提供五个系统功能：成绩功能、设置功能、开始游戏功能和关于功能。

### **3.2.1 成绩**

成绩里存储着游戏用户的一系列信息。成绩里有两大模块：单词游戏和总游戏时间，单词游戏中可以显示最高分、最高路程、最多鱼骨头；总游戏时间里面显示总游戏时间、总路程和鱼骨头总数。成绩功能的设计在表3.1中列出。

**表 3.1 成绩的详细设计**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Function | Precondition | Postconditions | Basic Flow |
| 成绩 | 用户可以查看自己的游戏信息 | 用户进入到了游戏主界面 | 用户点击成绩 按钮 | 用户进入到排行榜界面 |

### **3.2.2 设置**

游戏设置对小猫快跑的背景音乐和灵敏度进行了设置，用户可以根据自己的喜好进行设置，比如背景音乐的开启和关闭。游戏场景中的设置这一功能的设计用表3.2来说明。

**表 3.2 设置的详细设计**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Function | Precondition | Postconditions | Basic Flow |
| 设置 | 用户可以设置背景音乐和灵敏度 | 用户对游戏设置进行更改 | 用户对游戏属性的改变 | ①用户进入到游戏设置选项界面。  ②进行设置 |

### **3.2.3 关于**

关于部分主要介绍游戏开发者信息，让用户更进一步了解游戏。表3.3列出了游戏中关于的具体内容。

**表 3.3 关于的详细设计**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Function | Precondition | Postconditions | Basic Flow |
| 关于 | 介绍游戏的开发者信息 | 用户想要对游戏初步了解 | 点击返回退出关于界面 | 在主界面点击选项关于 |

### **3.2.4 开始游戏**

开始游戏功能让用户正式进行游戏，开始游戏功能在表3.4中列出。

**表 3.4 开始游戏的详细设计**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Function | Precondition | Postconditions | Basic Flow |
| 开始游戏 | 用户可以进入到游戏，正式开始游戏 | 进入到主游戏中主界面 | 游戏完成点击主界面按钮 | 点击开始游戏按钮 |

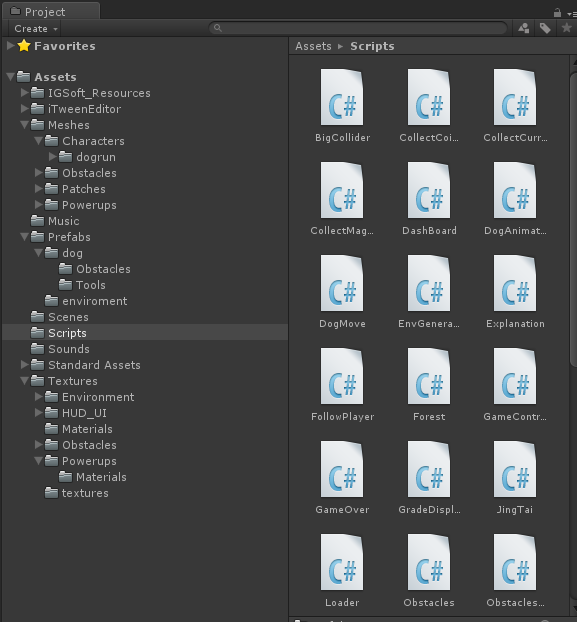
# 

# 4 系统概要设计

在系统概要设计里，包含一些系统的架构，主要功能之间的结构体系。复杂的结构可以通过简单的对象使之简化，然后再将这些分解的对象合并达到最终的系统。

## 4.1 游戏设计

游戏的设计是对整个游戏的构思和布局，在游戏实现最初要有一定的规划设计，这样在游戏实现时，就明白该怎样一步一步的进行了。小猫快跑项目文件组织设计，如图4.1所示。



**图4.1小猫快跑文件组织设计图**

### **4.1.1 小猫快跑游戏背景**

跑酷是一种非常受青少年喜爱的极限运动，但生活中由于路段和能力的限制，人们无法体验到跑酷的快感。在这种路段下跑酷类游戏孕育而生。因为跑酷类游戏能够非常发挥用户的反应能力，非常适合青少年休闲娱乐，所以现在非常流行。

### **4.1.2 游戏功能的设计**

（1）运行游戏，首先进入到的是欢迎界面，随后进入到的是本游戏的加载界面，这里使用异步加载技术，可以直观的看到游戏的加载进度。

（2）加载完成后进入到本游戏的主界面，这是游戏的中转站，从这里可以通过点击不同的功能按钮进入到不同的界面。点击”RANKING”按钮，进入到排行榜界面，点击”关闭”按钮回到主界面。点击”SETTING”按钮，进入到设置选项界面，点击”关闭”按钮回到主界面。点击设置界面中的“ABOUT”按钮，进入到关于界面。点击”关闭”按钮回到主界面。在主界面点击“退出”按钮，退出游戏。

（3）在主菜单选择”RUN”，进入到游戏中主界面，正式开始游戏。游戏中主界面的右下角为暂停按钮，右上角是鱼骨头数量计数板。当小猫收集到各种道具时游戏中主界面的左边会出现道具的剩余时间进度条。当用户在游戏中主界面点击暂停按钮后，会从屏幕上方弹出暂停界面，界面显示本次游戏的路程、分数、鱼骨头。点击”MAIN MENU”按钮，返回主界面；点击“继续游戏”按钮，继续本次游戏

（4）当小猫碰撞到障碍时，出现游戏完成界面，此时会显示出本次游戏的路程、分数、鱼骨头和是否超过最高纪录。点击”MAIN MENU”按钮，返回主界面，点击开始游戏可以继续进行下一次游戏。

### **4.1.3路段类**

路段类是游戏的一个重要的类，路段类主要定义游戏的路段路段，此游戏共有三个路段。不同的路段拥有不同的路线点集，详细设计如图表4.1所示。

**表4.1 路段的详细设计**

|  |  |  |
| --- | --- | --- |
| Name | WaypointsNumber | Characteristic |
| forest\_1 | 34 | 路线点是弯曲的，但是在同一高度上 |
| forest\_2 | 2 | 路线点是直线的，而且在同一高度上 |
| forest\_3 | 27 | 路线点是弯曲的，同时高度不一致 |

### **4.1.4道具类**

道具类定义了游戏过程的出现的各种道具，其中包括鱼骨头、磁铁和金砖。每个道具生成的概率有所不同，并且拥有各自的效果加成，道具详细设计如图表4.2所示。

**表4.2 道具的详细设计**

|  |  |  |
| --- | --- | --- |
| Name | Probability | Functions |
| ToolCoins | 98.5% | 分数加成，获取一个鱼骨头增加10分 |
| ToolCurrencys | 0.5% | 无敌状态，并且吸附鱼骨头，跑动速度加快 |
| ToolMagnets | 1% | 磁铁效果，可以吸附鱼骨头 |



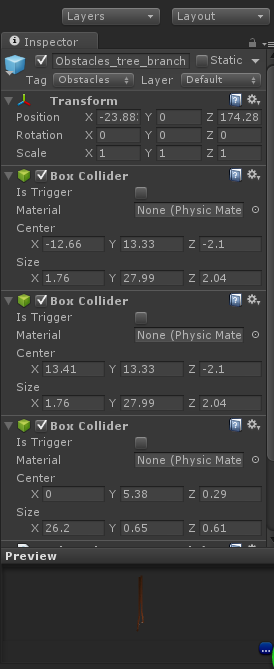
**图4.2 道具类的详细参数**

### **4.1.5障碍类**

障碍类定义游戏的多种障碍，因为障碍的长、宽、高各不相同，所以道具单行生成的个数可能不一样，详细设计如图表4.3所示。

**表4.3 障碍的详细设计**

|  |  |  |  |
| --- | --- | --- | --- |
| Name | LineMaxNumber | Collider | CenterHeight |
| Obstacles\_BarbWireMultiLane | 1 | 3 Box Collider | 6 |
| Obstacles\_BarbWireSingleLane | 3 | 3 Box Collider | 5.62 |
| Obstacles\_FallenTree | 1 | 3 Box Collider | 7 |
| Obstacles\_Rock | 3 | 1 Box Collider | 6.23 |
| Obstacles\_tree\_branch | 1 | 3 Box Collider | 0 |
| Obstacles\_TreeBark | 1 | 1 Capsule Collider | 4.37 |
| Obstacles\_TreeStump | 1 | 1 Capsule Collider | 6.25 |



**图4.3 障碍类的详细参数**

### **4.1.6 游戏中对碰撞逻辑的说明**

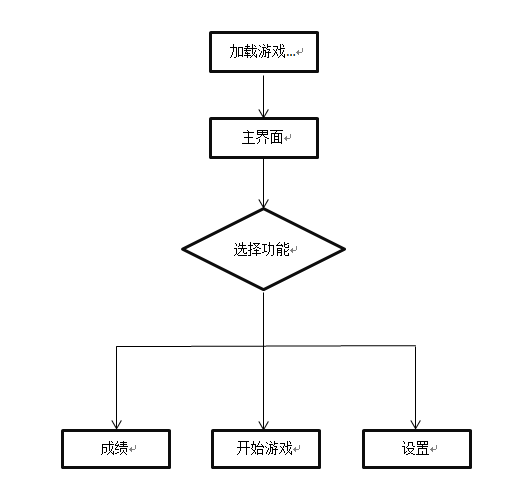
小猫一直处于屏幕中央，而游戏场景中的障碍是随机出现的，因此用户需要操作小猫躲避障碍。通过创建碰撞检测器和编写脚本代码，通过碰撞监视查看小猫是否发生碰撞，若没有发生碰撞，游戏继续运行，否则弹出游戏完成界面。

### **4.1.7 小猫动画模型播放规则**

在安卓版小猫快跑游戏中，不同的小猫动画模型播放的速度有所不同。其中奔跑动画每秒2帧，左转和右转动画每秒0.2帧，跳跃动画每秒0.5帧，下蹲动画每秒0.1帧。小猫模型不同动作的动画是根据用户操作进行切换播放的。

## 4.2 系统架构

小猫快跑的总体架构包括：游戏加载、游戏主界面、选择功能、游戏开始、设置、成绩等。游戏系统的架构如图4.4所示。



**图 4.4 系统整体架构**

加载游戏：用户在手机上点击游戏后，进入到加载游戏中主界面。

游戏主界面：用户可在主界面中看到游戏的整体包涵功能。

游戏开始：用户开始体验游戏的快乐。

成绩：用户可以查看自己历史最高成绩、获得鱼骨头数等。

设置：用户根据自己的喜好对游戏可以改变的功能进行选取。

## 4.3 系统功能结构

游戏的主要功能模块有：加载游戏、设置、关于、成绩和开始游戏。加载游戏时，会显示缓冲条和游戏图片。加载完成后进入到主界面，主界面有五个选项。游戏系统机构图结构如图4.5所示。

## 

**图4.5系统功能结构图**

## 4.4 系统功能模块简介

这一款游戏主要功能模块有：成绩、关于、开始游戏、设置、退出游戏等功能模块。

### **4.4.1 成绩功能**

排行榜界面是游戏的主要界面之一，在排行榜界面中用户可以看到一系列的游戏信息。排行榜界面显示信息主要分两个模块：单次游戏和总游戏时间。单次游戏模块汇报用户游戏场景中的最高分、最高路程和最多鱼骨头，而总游戏时间模块中统计用户游戏的总游戏数、总路程和总鱼骨头数。排行榜界面让用户可以更加了解自己的游戏信息，以激励用户挑战自我，更进一步。

### **4.4.2 关于功能**

关于功能主要介绍本游戏开发者信息和游戏操作方法。在主界面右上方，用户点击设置界面中的“ABOUT”按钮后进入到关于界面，在此界面中用户可以查看游戏的一系列信息的介绍，此功能可以让用户更快上手此游戏。

### **4.4.3 开始游戏功能**

在主界面的正中方点击开始游戏选项，小猫模型开始奔跑，用户正式进入到游戏，体验游戏的快乐。

### **4.4.4 设置功能**

设置游戏功能可以对游戏进行一系列设置，用户可以根据个人需求和爱好来调节游戏音乐的音量大小和重力感应灵敏度。进入到主菜单，然后点击”SETTING”选项，即可以进入到对背景音乐和重力感应灵敏度的设置，当开启背景音乐后，在游戏中会有相应的背景音乐出现，可以使用户玩的更酣畅。

### **4.4.5 暂停功能**

游戏运行界面中右下角有“暂停”按钮，点击按钮后游戏进入到暂停界面，用户如果想要继续游戏，点击界面中的“继续游戏”按钮，回到游戏中主界面并且有3秒的缓冲时间。当用户遇到特殊情况，可以选择暂停来保存游戏进度，这充分照顾了用户的需求。

### **4.4.6 退出功能**

在游戏主界面我们设置了退出按钮，而在游戏运行界面我们设置了暂停按钮。暂停是在用户需要暂时停止一会游戏时触发的事件，而用户在需要退出游戏时点击或点触主菜单退出游戏就可以退出小猫快跑游戏了。

# 5 游戏中主界面设计

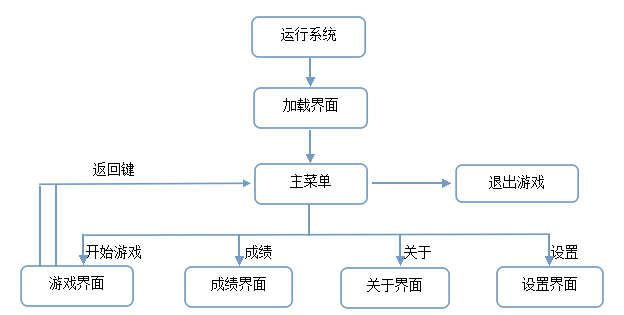
众所周知，游戏的界面是最直接与用户接触的平台，而这一平台对用户来说也是最重要的。一个比较好的游戏中主界面除了可以很快的吸引用户，还可以帮助用户快速的进入到游戏状态。

游戏中主界面的合理化设计是人和机器相互进行交互的最直接的操作方式。游戏的界面一般要求营造美观，操作简单、秩序感比较强而且要有一定的引导功能，好的游戏中主界面能为游戏的宗旨和游戏的内容所服务。

## 5.1 系统结构设计

通过分析软件的架构和体系，设计了该款Android版的小猫快跑游戏，游戏的主要界面包括：游戏加载界面、游戏主界面、游戏排行榜界面、游戏设置选项界面、游戏关于界面。

用户在手机点击游戏的图标，启动软件后屏幕会显示游戏的一个初始界面，加载界面后画面跳转到游戏主界面。在游戏的主界面，有三大块：游戏成绩，查看关于功能和游戏设置模块。不同的界面具有不同的功能，各个界面的转换关系如图5.1所示。



**图 5.1 界面转换关系图**

## 5.2 游戏初始界面设计

用户在开始游戏时候会看到一个加载界面。加载界面有三个图标：小猫形象图标、游戏名图标和加载进度条图标，加载界面的显示需要一系列脚本代码支持。运行界面如图5.2所示。



**图5.2加载界面**

### **5.2.1缓冲条的实现**

为了更好游戏的游戏体验，在各个场景之间跳转时都会显示加载界面，用于显示加载背景和加载进度。加载进度的显示主要运用以下代码：

Progress = async.progress; //获取加载进度

loadX = Screen.width \* 0.8f \* (progress - lastProgress); //设置加载条显示位置

lasProgress = progress; //将加载进度赋给上一次加载进度

## 5.3 游戏主界面

主界面有五个按钮：成绩、关于、设置、退出和开始游戏。各个按钮响应的界面和背景不同。成绩按钮响应排行榜界面，关于按钮响应关于界面，设置按钮响应设置选项界面，退出按钮响应退出游戏，开始游戏按钮响应游戏中主界面。主游戏中主界面界面友好，容易操作，鲁棒性强，可以给用户很好的游戏体验。游戏主菜单运行界面如图5.3所示。



**图5.3 主菜单运行界面**

## 5.4 设置选项界面

设置选项界面有两个选项：感应度和音量，通过滑动或者点击滚动条可以来设置它们的大小。设置的运行界面如图5.4所示。



**图5.4 设置运行界面**

## 5.5 排行榜界面

排行榜界面主要是记录用户游戏各种数据，在这个界面你可以知道最高分数、最高路程、最多鱼骨头数、总路程、总鱼骨头数、总游戏数等。用户可以查看自己的游戏信息。成绩累计的主要相应程序：

fenShu = PlayerPrefs.GetInt("fenshu"); //获得最高分数

luCheng = PlayerPrefs.GetInt("lucheng"); //跑得最高路程

jinBi = PlayerPrefs.GetInt("jinbi"); //获得最高鱼骨头数

zongLuCheng = PlayerPrefs.GetInt("zonglucheng"); //获得总路程

zongJinBi = PlayerPrefs.GetInt("zongjinbi"); //获得总鱼骨头数

zongChangShu = PlayerPrefs.GetInt("zongchangshu"); //获得总游戏数

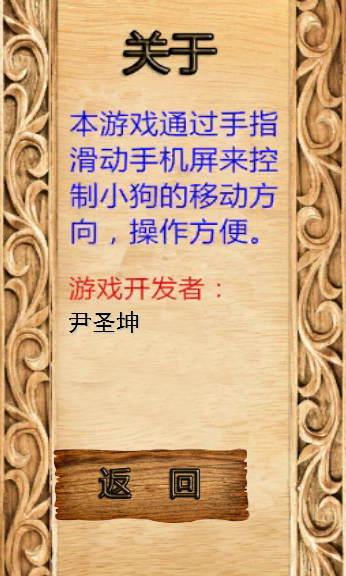
成绩的运行界面如图5.5所示。



**图5.5成绩的运行界面**

## 5.6 关于界面

关于界面主要作用就是简单地介绍游戏的玩法和游戏开发者信息，让用户充分了解游戏一系列的信息。关于界面的运行界面如图5.6所示。



**图5.6 关于的运行界面**

## 5.7 游戏中主界面

点击开始游戏后，进入到游戏中主界面，小猫对象开始跑动。仪表板开始绘制并记录路程和分数。障碍随机生成，同时鱼骨头和道具模型生成。游戏过程中用户如有需要可以选择暂停，暂停回来有三秒的缓冲时间，这充分照顾用户的需求。跑酷游戏的核心脚本代码是无限路段生成，当小猫跑动时，主摄像机的EnvGenerator.cs要检测小猫位置的z轴坐标值(即transform.postion.z)，以便生成下一个路段。其代码如下：

public class EnvironGenator : MonoBehaviour

{

public Forest forest1; //小猫对当前所处路段

public Forest forest2; //小猫下一个要进入到的路段

public int forestCount = 2; //小猫经过的路段个数

public GameObject[] forests; //路段对象数组

public void GenerateForest()

{

forestCount++;

float z = 3000 \* forestCount;

int index = Random.Range(0, 3); //0 1 2

GameObject newForest = GameObject.Instantiate(forests[index], new Vector3(0, 0, z), Quaternion.identity) as GameObject;

forest1 = forest2;

forest2 = newForest.GetComponent<Forest>(); //生成下一个路段

}

}

}

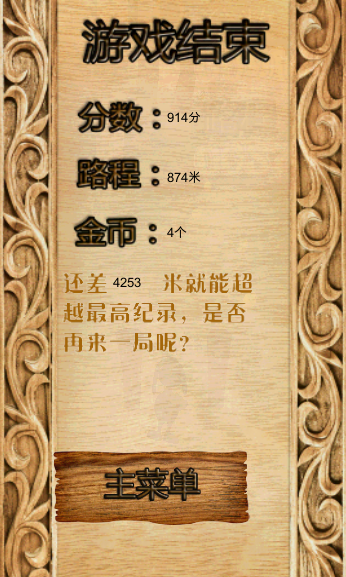
游戏进行的运行界面如图5.7所示。



**图5.7 游戏的运行界面**

## 5.8 游戏完成界面

当小猫触碰到障碍的时候，会弹出游戏完成界面。完成界面会显示用户本次游戏的小猫奔跑路程，获得的鱼骨头数以及得到的分数。而且会显示用户是否超越记录，若超越则显示新记录，反之显示与记录的差距。点击”MAIN MENU”按钮，游戏将会返回游戏主界面。游戏完成运行界面如图5.8所示。



**图5.8 游戏完成的运行界面**

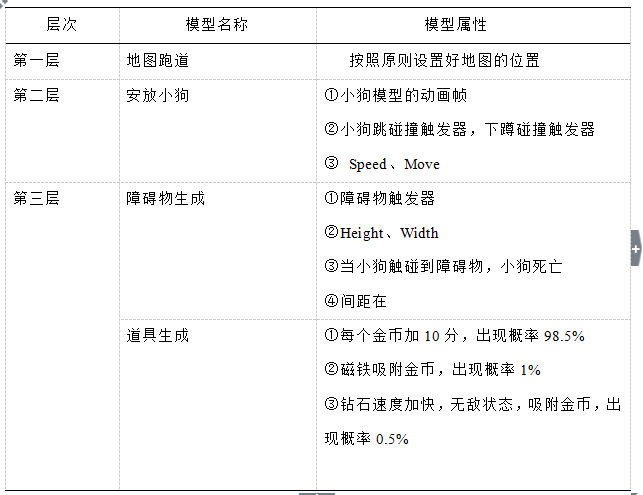
# 6 系统详细设计

小猫快跑游戏采用的是面向对象的设计方法，在最初的设计过程中是将系统中的各个事物看做不同的对象。有些对象很复杂，于是我们用一些比较简单的对象以组合的形式构建成一个复杂的对象，这样就需我们将不同的对象划分为相应的对象类，根据C#语法对这些对象类进行定义数据和方法，并且根据一系列关系来建立子类与父类，这样的结构关系使得程序员的思路更清晰明了。

## 6.1 模型的布局分析

在Unity游戏引擎中将每一个对象看做是一个模型。各个模型都会有一定的属性，相似属性的模型分配到同一层次，这样每个对象的层次关系就很明确，易于程序的调试和阅读。模型层次分布如表6.1所示。

**表6.1模型层次分布**



**6.2游戏类的实现**

在小猫快跑游戏的界面设计中，需要在游戏的静态类中定义各种不同状态的常量（布尔常量）：游戏暂停Stop、游戏排行榜界面GradeDisplay、游戏设置选项界面Options、游戏关于界面Explations。众所周知，当在系统中有很多选择时，逐个界面一次实现。绘制不同状态的游戏中主界面。

### **6.2.1 WelcomeButton类**

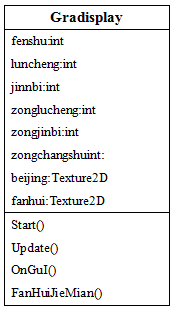
WelcomeButton类是对游戏的主界面的定义，在这个类中定义了开始游戏的按钮的位置，在Unity游戏引擎中获取背景图片和按钮图片常用代码如：GUI.Button(new Rect(图片来源），然后对其横纵坐标进行绘制。在初始化时，将所有点触按钮设置为false，并用布尔型来定义按钮的状态。当点击指定按钮位置时，一系列事件得到响应，进入到到下一个界面。WelcomeButton类图如图6.2所示。



**图6.2 WelcomeButton类图**

### **6.2.2 Gradisplay类**

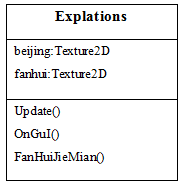
游戏的成绩类帮助游戏定义一个成绩信息的类，将用户的信息显示在一个特定图片的相应位置。用户可以查看自己的游戏状态，用户点击屏幕上的”关闭”按钮，可以将排行榜界面关闭，转换成游戏的主界面。GradeDisplay类图如图6.3所示。



**图6.3 GradeDisplay类图**

### **6.2.3 Explations类**

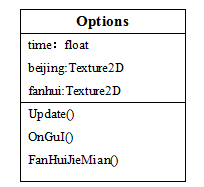
Explation类定义游戏的关于类，将有关游戏的操作方法和开发者信息附在相应的图片上。在用户最初接触游戏时打开关于，使用户更快的了解游戏，进入到角色。在查看完关于界面后，用户在屏幕上点击”关闭”按钮，将帮助窗口关闭，转换都游戏的主界面。Explation类图如图6.4所示。



**图6.4 Explation类图**

### **6.2.4 Options类**

Options类是设置游戏选项使用的类，这里用来设置游戏背景音乐和重力灵敏度的功能。用户在游戏屏幕点击设置后，然后拖动滑动或者点击条来设置的游戏功能。相应的类图如图 6.5所示。



**图6.5 Option类图**

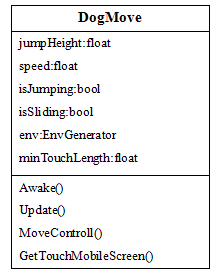
## 6.3 游戏的详细设计及实体类

### **6.3.1 静态类的创建**

游戏开发过程中往往会用到静态类，静态类无法被继承与实例化，同时会优先加载。在静态类中定义游戏场景中的某些变量可以方便管理。小猫酷跑游戏静态类中定义了小猫的奔跑状态、位置状态、跳跃下蹲标志位、道具状态和时间等变量。在“chushihua”方法中，初始化以上定义了的静态变量。

### **6.3.2 小猫的跑动和动画播放**

小猫的跑动时由DogMove类实现的在DogMove类中首先要定义用户操作枚举类型。在程序中定义小猫的跳跃高度和奔跑速度，以及各种判断标志位。在小猫跑动的过程中，Update()函数开始统计小猫奔跑的路程和分数。DogMove类的成员方法和成员变量如图6.6所示。



**图 6.6 DogMove成员方法和成员变量**

在游戏中小猫动画播放定义为DogAnimation()类，声明游戏中小猫七种状态。游戏开始时，初始化小猫模型处于Idle状态，然后根据用户的操作处于不同在游戏状态。不同的状态对应着不同的动画模型，播放速度有所不同。在LateUpdate（）函数中，实现操作的不同转化。

### **6.3.3 森林类的实现**

森林类用Forest.class来实现，在该类中定义类障碍之间的距离值和道具之间的距离值。当游戏开始时，路段地图中自动生成各种障碍和道具。每个道具生成的概率有所不同，加成效果也不同。道具生成代码如下：

void GenerateTools() //生成道具

{

float z = beginPoint; //最早出现道具的地方

while (true)

{

float length = Random.Range(mindistance, maxdistance); //两个道具相隔的距离在(minLength,maxLength)

z += length;

if (z > 3000)

{ break; }

Vector3 waypoint = GetWayPoint(z);

GenerateTools(waypoint);

}

}

void GenerateTools(Vector3 position) //生成道具

{

int index = Random.Range(0,200);

if (index < 197)

{

Tools cos = (GameObject.Instantiate(tools[1]) as Tools); //98.5%的概率生成鱼骨头

cos.InitSelf(position, this.transform);

}

else if(index < 199) //1%的概率生成磁铁

{

Tools mas = (GameObject.Instantiate(tools[0]) as Tools);

mas.InitSelf(position, this.transform);

}

else

{

Tools cys = (GameObject.Instantiate(tools[2]) as Tools); //0.5%的概率生成金砖

cys.InitSelf(position,this.transform);

}

}

### **6.3.4 森林路段的无限生成**

跑酷游戏的核心代码就是无限路段的生成，只有在此情况下用户才能挑战自我，创造更高的记录。本游戏的无限路段通过EnvGenerator类来实现，代码主要通过不断获取小猫的z坐标的大小来判定下一路段路段的生成。当小猫每跑3000米的时候，开始生成下一个路段，并且最好是小猫跑出当前的路段200米后，再消除上一路段，这样会使得路段看起来连贯，不会出现界面脱节现象。而且路段是自动随机生成，用户在游戏的过程中赏心悦目。

# 7 程序调试与测试

每个系统从开发到完成的过程中，都要多次的反复调试。一方面的原因是程序语法上的错误会使得系统不能正常运行；而另一方面就是界面的布局不合适，最后得到的界面不如人意；或者是系统中某个功能不能实现等等，这些情况都需要程序员对系统进行不断的调试，从而使系统接近完美。

## 7.1 程序调试

### **7.1.1 小猫跳跃但不降落**

在运行项目时发现小猫跳起后，就一直停在半空中没有降落。因此在代码中用Debug.log来检测。代码在执行多遍中观察小猫的y轴坐标值，发现当小猫跳跃时，y值增大后降落的程序没有执行。这是因为判断跳跃标志targetJumpHeight没有初始化，使得程序默认小猫不下降。在函数之前targetJumpHeight赋值为0，降落的程序正常运行。

### **7.1.2 游戏过程中出现闪动小猫图标**

游戏运行的时候，总是会出现小猫图标闪动，非常影响用户的游戏体验。利用写日志查看小猫的坐标值发现，当小猫无限接近路线点时，小猫有一瞬间x轴坐标增大。多次检测得出结论，原本程序中小猫是根据路线点位置来旋转调整位置，而当小猫无限接近路线点时，只剩一个y坐标值，而一个坐标值无法确定一个方向，使得小猫旋转方向不确定，出现闪动图标。而过了路线点后，小猫的旋转方向又确定了。在此情况下，设置一个精确度判断，可以避免此情况。

## 7.2 测试

在整个系统的实现过程中，为了确保软件的功能正确运行，测试是不可或缺的一部分。在开发软件时，系统出现异常是很难避免的，其中有些异常是可以简单直接发现，但有些异常则需要我们花费较大精力去调试的。一个系统没有经过仔细的测试环节，将会很难投入运行。因此，系统的测试就显的极为重要。安卓的小猫快跑游戏的测试项目有：单元测试、组合系统测试和用户测试。

### **7.2.1 单元测试**

单元测试是测试某段程序或是最基础的模块。在系统开发中，各个小的模块完成后，都要进行单元测试。比如游戏中初始界面的测试：点击游戏图标后，系统运行。进入到游戏加载界面。加载界面的测试包括：加载进度条显示。加载界面到主界面的跳转，利用了定时器的方法，用户在5秒之内没有操作即进入到主界面；在主菜单中对各个按钮的点击，测试游戏的场景转换是否顺畅，应该跳转的界面是否准确，菜单的关于功能的正常显示。在游戏中主界面测试小猫是否能跳跃下滑左转右转。还有暂停、返回功能键的测试。

### **7.2.2 组合系统测试**

组合系统测试通常有一定的顺序，分为自顶向下或自底向上两种。组合系统测试的前提是要在单元测试无误的情况下才能进行。在小猫快跑的游戏场景中通过滑动或者点击手机屏幕来操作小猫的动作，以躲避路段上的障碍，获得分数和鱼骨头奖励。这一系列的事物最终要组合成一个复杂的对象。在测试中就要相应的对整个过程中的每个环节进行测试。相对于单元测试，组合测试的测试范围更广。并且当各个模块实现后，整个系统是一个整体，Android的小猫快跑游戏的目标是在手机模拟器(ADT)上或是手机真机上进行，这就要求系统运行路段的匹配，模拟器的选择和手机硬件设备的需求都是有一定的要求。系统测试是对整个项目最后的审查，系统中涉及的所有功能实现的测试，不仅对软件进行相应的测试，硬件的配备至关重要。

### **7.2.3 用户测试**

用户测试：用户测试顾名思义就是在用户应用路段下，用户通过应用和使用软件，检测和核实软件是否实现自己预期的要求。比如操作是否流畅、功能是否达到用户使用要求、界面是否友好。一款好的游戏只有赢得越多用户的青睐才可以显现出来其质量。整个系统完成后，应该在实际应用路段下进行系统的真实测试。

# 

# 总 结

毕业设计---基于Unity3D的小猫快跑游戏的开发从最初接到任务书开始到整个系统的实现经历了长达四个月的时间，在完成毕业设计的过程中，有很多情感的掺杂：任务书下发的艰巨、游戏制作的新奇、万事开头难的困局、各种学习资料的摄取、一点一滴的积累、小有成就的兴奋、调试过程的无奈、最终成型的豁达……

毕业设计选择了安卓版小猫快跑游戏后，开始大量的搜寻素材，查找Unity一系列教学视频，学习C#语言，搭建开发平台。因为自己就很迷恋玩酷跑类游戏（神庙逃亡等），所以在做这个游戏的过程中一直都是很感兴趣的。最初设计了很多的界面，有很多创新的想法，但是在具体实现的时候屡屡碰壁，实现起来和想象的差距太大。游戏的设计和实现，让我对C#语言能够熟练的运用，虽然有很多地方还是很模糊，但是基本的语法和结构以及类、包的用法已经熟练掌握。在游戏的编程中，子类和父类用运的比较多，所以比较熟悉。游戏的设计最开始要构思整个游戏的完整结构和整体框架，这是很重要的，通过游戏设计的需求分析一步一步精化，然后再将整体框架细化，各个功能模块的实现和每个模块界面的绘制。在安卓小猫快跑游戏中，运用到Unity这个平台，此开发路段具备视觉化编辑，详细的属性编辑器和动态的游戏预览特性，让开发者更加容易操作使用。而且Unity自动同步了Visual studio项目里面的源代码，开发者可以自由调用已经封装好的很多Class，非常实用，为开发者节省很多编程时间。通过逐步设计和修改阶段增添更多的模块，功能也会更加完善。设计完成后，会对系统做全面的功能测试，使安卓的小猫快跑更完善。

# 

# 参 考 文 献

1. 加藤政树.unity游戏设计与实现 [M].人民邮电出版社.2015:327-328
2. 宣雨松.unity3d游戏开发 [M].人民邮电出版社.2016:102-103
3. 刘刚.孙文涛.unity官方案例精讲[M].中国铁道出版社.2015:79-78
4. 李果.黄帅.Autodesk Maya中文标准教程[M].上海科学普及出版社.2004:200-201
5. 刘桂元，曾志远，杨书新.基于Unity3d引擎的教育类游戏设计与实现[J]. 软件导刊，2017,（01）：50-53.
6. 李智鹏.基于Unity3D引擎的空中战机游戏设计与实现[D].吉林大学，2016.
7. 刘多多.基于OpenGL的3D游戏设计[D].吉林大学，2016.
8. 张策.基于Unity3D的三维游戏场景设计与实践[D].合肥工业大学，2016.

1. [Ryan](http://www.dangdang.com/author/Ryan_1" \t "http://product.dangdang.com/_blank) [Henson](http://www.dangdang.com/author/Henson_1) [Creighton](http://www.dangdang.com/author/Creighton_1).Unity 3D Game Development by Example Beginner's Guide [M].[Packt Publishing](http://www.dangdang.com/publish/Packt+Publishing_1" \t "http://product.dangdang.com/_blank).2016:234-234
2. Wes [McDermott](http://www.dangdang.com/author/McDermott_1). Creating 3D Game Art for the iPhone with Unity: Featuring modo and Blender pipelines [M].[Focal Press](http://www.dangdang.com/publish/Focal+Press_1).2016:125-126
3. Sue [Blackman](http://www.dangdang.com/author/Blackman_1).Beginning 3D Game Development with Unity: All-in-one, multi-platform game development[M].[Apress](http://www.dangdang.com/publish/Apress_1).2016:120-121

# 

# 致 谢

在函授的两年里，已经学习过很多课程，直到现在对老师们谆谆教诲的情景仍然历历在目，我非常感谢在这两年里所有老师给予我知识的传授和帮助。

毕业论文的完成给我的专升本函授之旅画上了一个圆满的结局，在这里我要特别感谢我的论文指导老师李志鹏老师。写论文期间，在遇到困难和疑惑的时候，老师都会悉心的给我讲解，在老师的点拨指导下，让我对自己的论文更有信心。在撰写时，老师悉心的审批我的论文，指出我论文中的错误。一遍又一遍耐心的指导，最终我顺利完成论文。

荏苒两年的时光，感谢合肥工业大学对我的培育和每个老师的精心教育，让我满载丰收。