

基于 Unity3D 引擎的手机端游戏 的设计与开发

作者姓名 黄辑林 学校导师姓名、职称 褚华副教授

领 域 软件工程 企业导师姓名、职称 曹德邦高工

申请学位类别 工程硕士 提交学位论文日期 2015 年 3 月

学校代码 10701

学 号 1210122622

分 类 号 TP31

密 级 公开

西安电子科技大学

硕士学位论文

基于 Unity3D 引擎的手机端游戏 的设计与开发

作者姓名：黄辑林

领 域：软件工程

学位类别：工程硕士

学校导师姓名、职称：褚华 副教授

企业导师姓名、职称：曹德邦 高工

提交日期：2015 年 3 月

**Design and Development of a Mobile Game
using Unity3D Engine**

A thesis submitted to
XIDIAN UNIVERSITY
in partial fulfillment of the requirements
for the degree of Master of Engineering

By
Jilin HUANG
(Software Engineering)
Supervisor: Hua CHU Debang CAO
March 2015

西安电子科技大学

学位论文独创性（或创新性）声明

秉承学校严谨的学风和优良的科学道德，本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果；也不包含为获得西安电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中做了明确的说明并表示了谢意。

申请学位论文与资料若有不实之处，本人承担一切的法律责任。

本人签名：_____ 日 期：_____

西安电子科技大学

关于论文使用授权的说明

本人完全了解西安电子科技大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属西安电子科技大学。学校有权保留送交论文的复印件，允许查阅和借阅论文；学校可以公布论文的全部或部分内容，可以允许采用影印、缩印或其它复制手段保存论文。同时本人保证，毕业后结合学位论文研究课题再撰写的文章一律署名单位为西安电子科技大学。

本人签名：_____ 导师签名：_____

日 期：_____ 日 期：_____

摘要

近年来,随着智能手机、平板电脑等移动设备的迅速普及,移动游戏数量急速增加,推动了中国游戏市场规模高速增长。越来越多的大中型游戏厂商、小型游戏工作室加入到移动游戏的开发阵营中,游戏市场竞争相当激烈。

Unity3D 游戏引擎以其易用性、稳定性、扩展性和跨平台等特性,获得了目前游戏引擎市场的很大的市场占有率,在国内外游戏开发团队、甚至在科学研究领域中都有广泛的使用。

本文主要叙述了手机游戏《悠游厦门》的开发工作。开发工作中,首先对游戏项目进行了需求分析,在明确项目背景和开发任务后,使用 UML 用例图对游戏开发需要实现的主要功能进行了需求建模和用例分析,使用实体-联系图对游戏中的数据进行了数据建模,并使用数据流图对游戏中主要的业务流程进行了过程建模;其次对游戏进行了详细的设计,在确定了游戏的客户端-服务器总体架构之后,将游戏的功能分解为游戏主体、游戏场景、游戏模式、游戏关卡、道具系统、成就系统等多个模块进行设计,并使用物理数据模型图进行了游戏的数据结构设计;然后对游戏进行具体实现,分析游戏中的各个模块中的各个对象的属性以及相互之间的关系,并将其抽象成为 UML 类图,实现了游戏的程序结构的建模,再使用 Unity3D 游戏引擎对游戏的角色、场景、动画、特效、图形用户界面等内容进行实际开发;最后对游戏进行测试与发布,针对游戏的功能需求设计了多个测试用例,定义了每个测试的目的、操作步骤、预期结果,并根据测试用例的内容对游戏进行单元测试和实机测试,记录下每个测试用例的实际测试结果和测试状态,完成测试之后,使用应用平台提供的接口,实现了游戏在多个应用平台上的接入,并成功发布于多个应用平台上。游戏开发工作中使用的关键技术有 Unity3D 游戏引擎和统一建模语言。

游戏项目的创新点在于:突破性地将传统旅游业与当前非常流行的手机游戏相结合,同时使用了三维的游戏画面,使玩家能够在进行游戏的同时身临其境般地欣赏到旅游景点的美景和感受到当地的风俗文化,从而带动了游客旅游的积极性。

《悠游厦门》手机游戏开发工作取得了良好的成果:负责游戏开发项目的厦门时代华亿动漫有限公司的游戏开发任务顺利完成,游戏顺利完成在 iOS、Android 等主流平台、设备上的测试运行,并成功于中国移动动漫基地平台、谷歌应用商店、苹果应用商店等平台上发布,被两万余名玩家下载安装。

关键词: 移动游戏, Unity3D, UML

论文类型：应用软件技术

Abstract

In recent years, along with the fast popularization of mobile phones, tablets and other mobile devices, the number of mobile games has been growing at a high speed, which drives the rapid growth in the scale of China mobile game market. More and more large and medium game companies and small game studio join the group of developing mobile games, making the mobile game market competing fiercely.

Unity3D game engine wins a large share of game engine market with its usability, stability, scalability, cross-platform and other features, and it is now widely used by many game developing teams and even in academic research field all around the world.

This paper mainly described the development of mobile game Amoy Monopoly. During the development, we first analyzed the requirement of the game project. After figuring out the background of the project and the developing task, we modeled the requirements and did use case analysis for the main functions of the game by UML use case diagram, and we modeled the game data by entity-relationship diagram and modeled the game processes by data flow diagram. Secondly, we designed the game in detail. After defining the Client-Server framework of the game, we designed the game functions by dividing them into game body, game scenes, game stages, item system, achievement system and several other modules. Also, we designed the data structure of the game by physical data model diagram. Thirdly, we implemented the game. We analyzed the objects and their relationships in each module of the game and made an abstraction of them into UML class diagrams, which implemented the modeling of the game program structure. And we used Unity3D game engine to develop the game characters, scenes, animations, special effects, graphical user interface and so on. Finally, we tested and published the game. Focusing on the functional requirements, we designed many test cases by defining the test purpose, operation steps and expected result of each test case and ran unit tests and machine tests according to the test cases with the actual test results and test statuses recorded. After the tests, we implemented the connection to several application platforms using the interfaces provided by the platforms and published the game on the platforms successfully. The key technologies used in the game project are Unity3D game engine and unified modeling language.

The innovation point of the game project is that the game combined the traditional tourism and the currently popular mobile game, which is a breakthrough. What's more, the game uses 3D graphic, which makes the players be able to appreciate the beautiful scene in the travel spots

and feel the local customs and culture realistically. Thus, the tourists are encouraged to travel. The development of mobile game Amoy Monopoly achieved good results: the developing task of Xiamen Time Waii Co. Ltd. was successfully completed. The testing of the game on iOS, Android these mainstream platforms was smoothly done and the game was published on China Mobile ACG platform, Google Play and Apple Store and was downloaded by more than 20,000 players.

Key words: mobile game, Unity3D, UML

Type of Dissertation: Applied Software Technology

插图索引

图 3.1	游戏基本流程活动图	5
图 3.2	游戏主要功能用例图	12
图 3.3	游戏数据实体-联系图	13
图 3.4	新闻公告数据实体-联系图	14
图 3.5	成就数据实体-联系图	15
图 3.6	卡牌数据实体-联系图	16
图 3.7	地图事件数据实体-联系图	17
图 3.8	关卡数据实体-联系图	17
图 3.9	游戏上下文数据流图	18
图 3.10	游戏业务活动数据流图	19
图 4.1	总体软件体系结构	21
图 4.2	客户端界面迁移状态图	22
图 4.3	游戏客户端模块划分图	22
图 4.4	Game 类的 UML 类图	23
图 4.5	道路对象继承关系图	24
图 4.6	道路和地产地块 UML 类图	26
图 4.7	游戏事件 UML 类图	27
图 4.8	游戏事件 UML 活动图	28
图 4.9	角色行动流程 UML 活动图	28
图 4.10	角色类 UML 类图	29
图 4.11	卡牌和道具 UML 类图	30
图 4.12	NPC 控制器 UML 类图	31
图 4.13	NPC 控制器使用卡牌策略	31
图 4.14	激情赛艇小游戏 UML 类图	32
图 4.15	宝箱迷踪 UML 类图	32
图 4.16	胡里炮台小游戏 UML 类图	33
图 4.17	天赐富贵小游戏 UML 类图	34

图 4.18 勇闯迷宫小游戏 UML 类图.....	34
图 4.19 成就系统 UML 类图.....	36
图 4.20 玩家物理数据模型图.....	37
图 4.21 新闻公告物理数据模型图.....	38
图 5.1 资源导入流程.....	39
图 5.2 模型动画配置.....	39
图 5.3 图片导入设置.....	40
图 5.4 Game 类中配置 XML 文件	42
图 5.5 XML 文件使用流程.....	42
图 5.6 角色动画控制器设定.....	43
图 5.7 测试场景构建流程.....	44
图 5.8 图形用户界面实现流程.....	44
图 5.9 主菜单界面结构.....	46
图 5.10 主菜单界面结构处理响应数据.....	47
图 6.1 调用支付接口核心代码.....	55
图 6.2 加载 OBB 文件	56

表格索引

表 4.1	User 实体属性详细设计	37
表 4.2	Item 实体属性详细设计	37
表 4.3	Achievement 实体属性详细设计	37
表 4.4	News 实体属性详细设计	38
表 6.1	软硬件环境表	49
表 6.2	角色行动流程测试用例	50
表 6.3	五缘湾场景测试用例	51
表 6.4	博饼模块测试用例	52
表 6.5	游戏下载测试用例	52
表 6.6	积分兑换奖励测试用例	53
表 6.7	实机测试结果	53

符号对照表

符号

符号名称

缩略语对照表

缩略语	英文全称	中文对照
UML	Unified Modeling Language	统一建模语言
GUI	Graphical User Interface	图形用户界面
XML	Extensible Markup Language	可扩展标记语言
GPU	Graphics Processing Unit	图形处理单元
HLSL	High-Level Shading Language	高级着色语言
NGUI	Next Generation User Interface	新一代用户界面
HTTP	HTTP-Hypertext transfer protocol	超文本传送协议
JSON	JavaScript Object Notation	JavaScript 对象表示法
SDK	Software Development Kit	软件开发工具包
APK	Android Package	安卓安装包
OBB	Opaque Binary Large Object	不透明二进制大对象
NPC	Non-Player Character	非玩家角色
Cg	C for Graphics	图形编程 C 语言
GLSL	OpenGL Shading Language	OpenGL 着色语言

目录

摘要	I
Abstract	III
插图索引	V
表格索引	VII
符号对照表	VIII
缩略语对照表	IX
目录	XI
第一章 绪论	1
1.1 选题背景及意义	1
1.1.1 中国游戏产业发现现状	1
1.1.2 研究的目的及意义	1
1.2 论文工作内容	1
1.3 论文组织结构	2
第二章 相关技术概述	3
2.1 游戏引擎	3
2.2 Unity3D 游戏引擎	3
2.3 本章小结	4
第三章 需求分析	5
3.1 业务陈述	5
3.1.1 项目背景	5
3.1.2 开发任务概述	5
3.2 整体需求	6
3.3 功能需求	6
3.3.1 游戏主体功能分析	6
3.3.2 游戏场景功能分析	8
3.3.3 游戏模式功能分析	8
3.3.4 小游戏功能分析	9

3.3.5 游戏关卡功能分析.....	10
3.3.6 道具功能分析.....	10
3.3.7 成就功能分析.....	10
3.3.8 游戏商城功能分析.....	10
3.3.9 新闻公告功能分析.....	11
3.3.10 其他客户端功能分析.....	11
3.4 用例分析	11
3.5 数据建模	12
3.5.1 玩家数据建模.....	13
3.5.2 新闻公告数据建模.....	14
3.5.3 成就数据建模.....	15
3.5.4 卡牌数据建模.....	16
3.5.5 地图事件数据建模.....	16
3.5.6 关卡数据建模.....	17
3.6 过程建模	18
3.7 本章小结	19
第四章 游戏设计.....	21
4.1 总体软件体系结构	21
4.2 客户端功能模块设计	22
4.2.1 游戏主体设计.....	23
4.2.2 小游戏设计.....	32
4.2.3 成就系统设计.....	35
4.3 游戏数据库设计	36
4.3.1 玩家数据.....	36
4.3.2 新闻公告数据.....	38
4.4 本章小结	38
第五章 游戏实现.....	39
5.1 资源导入与管理	39
5.2 使用 XML	41

5.3 设定角色动画	42
5.4 构建测试场景	43
5.5 实现图形用户界面 (GUI)	44
5.6 实现客户端与服务器之间的通信	47
5.7 本章小结	47
第六章 游戏测试与发布	49
6.1 游戏测试	49
6.1.1 游戏测试环境	49
6.1.2 测试用例及过程	49
6.1.3 测试结果分析	53
6.2 游戏发布	54
6.2.1 在中国移动动漫基地平台发布	54
6.2.2 在谷歌应用商店 Google Play 发布	56
6.2.3 在其他平台发布	57
6.3 本章小结	57
第七章 结束语	59
7.1 论文工作总结	59
7.2 后续工作展望	59
参考文献	61
致 谢	63
作者简介	65

第一章 绪论

1.1 选题背景及意义

1.1.1 中国游戏产业发现现状

2014 年上半年,中国游戏市场规模受网页游戏和移动游戏的影响,整体市场规模高速增长,增幅达到近五年新高。中国游戏(包括网络游戏市场、移动游戏市场、单机游戏市场等)用户数量 4 亿人,同比增长 9.5%,实际销售收入达到 496.2 亿元,同比增长 46.4%,其中,客户端游戏市场实际销售收入 255.7 亿元,网页游戏市场实际销售收入 91.8 亿元,移动游戏市场实际销售收入 125.2 亿元,社交游戏市场实际销售收入 23.4 亿元,单机游戏市场实际销售收入 0.1 亿元。

由此可见,中国的游戏产业整体上是在快速发展的。

1.1.2 研究的目的及意义

在中国游戏产业快速发展、移动游戏市场规模快速增长的背景下,厦门时代华亿动漫有限公司与厦门市旅游局决定合作开发一款旅游主题的移动游戏《悠游厦门》,通过移动游戏平台来达到宣传厦门旅游景点、吸引游客的目的。

为了使游戏画面中的旅游景点、城市建筑更逼真,游戏采用三维画面。Unity3D 是一个技术领先的三维游戏引擎,以其易用性、稳定性、扩展性和跨平台等特性,获得了目前游戏引擎市场的很大的市场占有率,在国内外游戏开发团队、甚至在科学研究领域中都有广泛的使用。因此游戏采用 Unity3D 游戏引擎进行开发。

《悠游厦门》游戏突破性地传统旅游业与当前非常流行的手机游戏相结合,并使用了三维的游戏画面,使玩家能够在进行游戏的同时身临其境般地欣赏到旅游景点的美景和感受到当地的风俗文化,从而带动了游客旅游的积极性。同时,这又是一次非常有意义的游戏行业与传统旅游行业的合作尝试,是游戏行业发展形式的一次有意义的探索。

1.2 论文工作内容

本人在《悠游厦门》手机游戏开发项目中担任程序员,主要负责游戏客户端的开发工作。工作的大致内容如下。

- (1) 根据项目需求和策划文案进行需求分析,并对游戏进行建模。
- (2) 游戏核心系统的设计与实现。游戏核心系统主要包括游戏主体、道具系统、成就系统、人工智能系统等。
- (3) 实现游戏客户端与服务器之间的通信,使客户端能够与服务器通信,完成购买道具、充值等事务。

(4) 将游戏发布到中国移动动漫基地平台、谷歌应用商店等平台上。

1.3 论文组织结构

本论文主要研究和实现了《悠游厦门》手机游戏的开发，叙述了软件工程思想在游戏开发中的应用。本文的组织结构如下：

第一章为绪论部分。通过对中国游戏产业的现状的介绍，分析游戏开发研究的必要性，叙述本论文的研究目的和意义，并叙述了本人在《悠游厦门》手机游戏开发项目中完成的主要工作，最后列出了本论文的组织结构。

第二章介绍游戏开发项目中使用到的相关技术。首先介绍游戏引擎技术，并对 Unity3D 游戏引擎以及游戏开发项目中使用到的引擎技术进行基本介绍，最后简单介绍统一建模语言。

第三章为游戏项目的需求分析。首先陈述游戏的基本业务，包括游戏项目背景和开发任务，然后对游戏项目分别进行需求建模、数据建模和过程建模，详细分析游戏项目的需求。

第四章叙述根据需求分析进行的游戏设计。首先分析游戏系统的整体软件体系结构，并将游戏客户端的按功能进行模块划分，然后对每个模块进行详细设计与建模，最后对游戏的数据结构进行设计。

第五章叙述游戏的实现过程。首先对游戏进行 UML 建模，对游戏的客户端的各个模块及其结构进行分析并使用统一建模语言将其清晰地描述、记录下来；然后叙述游戏的具体实现过程，描述具体的开发过程中的重要步骤和技巧，依次叙述了游戏资源的导入与管理、XML 技术的使用、角色动画的设定、测试场景的构建、游戏中使用的图形渲染与特效技术、图形用户界面的实现和客户端与服务器之间的通信的实现。

第六章叙述游戏开发完成后的测试和发布过程。首先叙述游戏的单元测试和实机测试过程，分析并总结测试结果；然后分别叙述在中国移动动漫基地平台、谷歌应用商店、苹果应用商店以及微软应用商店上发布游戏的过程。

第七章对整个游戏开发项目以及论文工作进行总结，并对未来提出展望。

第二章 相关技术概述

论文中使用到的关键技术主要有游戏引擎和统一建模语言两个技术，其中游戏引擎使用的是 Unity3D 游戏引擎。

2.1 游戏引擎

游戏引擎指已制作好的可制作交互式实时图像应用程序或普通的电子游戏系统的核心组件。此核心组件为游戏开发者提供各种制作游戏时所需要的工具，通过使用游戏引擎，游戏开发者花费较少时间就可以轻松地编写出游戏程序，而不必自己从头开始一步一步搭建游戏程序所需的各种基础模块。游戏引擎一般由以下模块组成：物理引擎、碰撞检测系统、图形引擎、音效引擎、脚本引擎、动画系统、网络系统以及场景管理系统。^[3]

经过多年的发展，现在的游戏引擎的功能已经变得非常复杂非常强大了。下面对各个模块进行简单介绍。

首先是物理引擎。物理引擎通过模拟现实中的物理规律，可使物体遵循固定的规律运动，更具有真实感。例如，球类游戏中的球的滚动、反弹，战争游戏中炮弹爆炸时飞起的石块。

碰撞检测是物理引擎的重要的一部分，它用于探测物体是否发生碰撞。当游戏中的两个物体靠近并碰撞的时候，碰撞检测系统能够检测到这个碰撞，并通知管理物体运动的模块处理物体的接触、碰撞。例如，当游戏中角色投出篮球时，碰撞检测可以使篮球撞击到篮板时不会穿过篮板，因为当篮球和篮板发生接触的时候，碰撞检测系统探测到了这个接触，并触发碰撞和反弹事件，让篮球反向运动。

其次是光影效果。光影效果即光线和阴影效果，它是图形系统的一个部分，是游戏场景中的光源对人和物体的影响方式。游戏引擎根据折射、反射等基本光学原理来计算游戏场景中的光线和阴影，及动态光源、彩色光源等高级效果都是通过引擎的图形技术来实现的。

动画系统中，模型动画是在模型的基础上直接进行变形，骨骼动画则利用模型中内置的骨骼带动物体产生运动。为方便动画制作人员为人物角色或动物设计丰富的动作造型，游戏引擎把这两种动画系统预先植入了游戏当中。

渲染是指计算机从模型生成图像的过程。

2.2 Unity3D 游戏引擎

Unity3D 游戏引擎是由 Unity Technologies 开发的一个支持跨平台的综合型游戏开

发工具，它能够让开发者轻松制作二维或三维电子游戏、实时动画、建筑可视化等类型的互动内容，是一个全面整合的专业游戏引擎。

2014 年，Unity3D 游戏引擎占据了功能游戏引擎市场 45% 的份额。Unity 全球用户已经超过 330 万人，每月活跃用户数高达 60 万。全世界有 6 亿的玩家在玩使用 Unity 引擎制作的游戏，用 Unity3D 制作的应用和游戏目前的累计下载量达到了 87 亿次。

Unity3D 引擎的特点有：

- (1) 开发环境方面，Unity3D 采用层级式的用户界面，能够可视化地对游戏进行编辑，对游戏中的对象和对象的属性的管理也非常方便。
- (2) 跨平台方面，Unity3D 支持非常多平台的游戏应用开发，可开发 Windows、Mac OS X、Linux、Web 平台、Play Station、Xbox、Wii、Android、iOS、Windows Phone 等平台上的游戏应用。
- (3) 资源管理方面，Unity3D 会自动将资源导入游戏中，使开发者可以直接使用图片、声音、模型、贴图等游戏资源，并根据资源的改动自动更新。
- (4) 图形编程方面，Unity3D 使用 ShaderLab 语言来编写着色器，同时支持 Cg、GLSL 语言编写的着色器。
- (5) 物理引擎方面，Unity3D 内置对 Nvidia 公司的 PhysX 物理引擎的支持。
- (6) 游戏脚本方面，Unity3D 使用基于 Mono 的脚本系统，一个基于 .NET Framework 的开源语言，开发者可以使用 C# 语言、JavaScript 语言或 Boo 语言来编写游戏脚本。

2.3 本章小结

本章主要叙述了论文工作中使用的主要技术——游戏引擎技术，并详细介绍了项目中使用的 Unity3D 游戏引擎的特点。

第三章 需求分析

与一般软件开发相同，游戏开发的过程中，需求分析是相当重要的一个阶段。需求分析的质量对游戏开发的影响是深远的、全局性的，高质量的需求分析往往对游戏开发起到事半功倍的效果。需求分析决定了游戏的内容、风格、定位的走向，为游戏的设计和开发奠定了基础。本章对游戏的需求进行分析。

3.1 业务陈述

3.1.1 项目背景

《悠游厦门》手机游戏是 2013 年中国厦门中秋旅游嘉年华活动的一部分，由厦门时代华亿动漫有限公司负责手机游戏的开发。

中秋旅游嘉年华活动是由厦门市旅游局主办的旅游活动。活动主题为“中秋博饼，好运厦门”，活动目的为，深入挖掘厦门特有的中秋民俗文化内涵，凸显厦门旅游的文化特质，丰富厦门市旅游产品，提升厦门市在境内外旅游市场的影响力，持续打造面向境内外的、主题鲜明的、具有地方特色的、国际知名的旅游节庆品牌，加快建设国际知名、国内一流的旅游目的地城市。

《悠游厦门》手机游戏活动的内容为，向全球游客开放游戏，采取累积分和奖励分的方式进行活动。具体奖励方式如下：

(1) 下载游戏并最终通关者，直接赢取闽南旅游卡中秋套票及全球游客博厦门总决赛资格；

(2) 下载游戏并累积到一定分数，可相应获得闽南旅游卡中秋套票或全球游客博厦门网络博饼资格一次；

(3) 下载游戏并获得某些特定分数时，可奖励全球游客博厦门网络博饼资格一次。

玩家进行游戏的基本流程如图 3.1 所示：

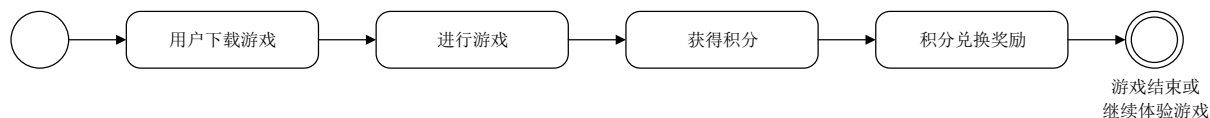


图 3.1 游戏基本流程活动图

3.1.2 开发任务概述

《悠游厦门》手机游戏开发项目的总体任务为：开发一个能够运行在移动平台上的游戏，游戏基于大富翁规则，借用鼓浪屿等厦门知名景点设置游戏场景，借用博饼等厦门特色文化元素设置游戏道具。

3.2 整体需求

根据项目合作方提出的要求，从整体上看，游戏项目有以下需求。

（1）大富翁的游戏规则

大富翁（英语：Monopoly）是一种桌面游戏。游戏参与者通过投掷骰子进行购买地产、建设楼房和收取租金等投资活动，最后获得游戏胜利。^[9]

游戏基本规则为，游戏开始时每名玩家拥有一定的资产，轮流投掷骰子，根据投掷出的点数控制角色在地图上移动，移动结束时可对停留的地点进行投资，或需向地点的拥有者缴纳地租。最后资产最多的玩家获胜。

（2）用鼓浪屿等厦门知名景点设置游戏场景

游戏的一个重要的功能需求为，在游戏场景中设置鼓浪屿等厦门知名景点，使玩家在游戏过程中能够根据游戏的场景、画面，认识到厦门的各个旅游景点的名称和大致的景观，成为旅游者认识厦门、接近厦门的第一步。

知名景点区域主要有鼓浪屿、五缘湾、白鹭洲、环岛路和岛外区域。

（3）用博饼等厦门特色文化元素设置游戏道具

博饼，起源于泉州，是闽南地区几百年来独有的中秋传统活动，是一种独特的月饼文化，也是对历史的一种传承。具体的玩法大致为，10个人围在一张桌子周围，每个人每次要把6个骰子一起投进大碗里，众人依次投骰子。根据投出的6个骰子的点数的不同特殊组合，参与者可获得不同等级的奖品。

（4）玩家可从互联网上下载游戏

玩家参与游戏的途径为，通过互联网将游戏下载到自己的移动设备中并安装后，在移动设备上玩游戏。因此，在互联网上发布游戏，使玩家能够下载游戏也是重要的一个功能需求。

（5）积分兑换奖励

玩家在游戏中完成关卡或达成某些特殊条件时可获得积分，使用积分可兑换旅游卡等厦门旅游相关的实体奖励。游戏的奖励是吸引玩家参与到游戏当中的一个重要因素，能够大大提高玩家的游戏积极性。

3.3 功能需求

根据对整体需求的详细分析，游戏必须提供多个功能来满足整体需求。下面叙述游戏的功能需求。

3.3.1 游戏主体功能分析

游戏主体，即游戏的核心玩法，或基本玩法，指建立在基本规则之上的进行游戏的

流程。它是一个游戏的根本，是一个游戏的灵魂，游戏中的其他的所有内容都是为游戏主体服务的。玩家在游戏中花费时间最长、得到的享受最多的部分就是游戏主体部分。因此，游戏主体的功能非常重要。

《悠游厦门》游戏的主体即建立在大富翁规则之上的游戏玩法：投掷骰子、角色移动、投资地产和缴纳租金。具体功能如下。

（1） 投掷骰子

根据需求，投掷骰子采用博饼的形式。根据博饼游戏的规则，每次必须同时投掷 6 个骰子，因此选择 6 个骰子中的其中一个骰子的点数作为投掷之后角色行走的步数。同时，根据博饼的规则，若 6 个骰子的点数出现了特殊的组合，角色即可获得一些特殊的奖励。

（2） 道路

道路是游戏场景中逻辑上首尾相接的线性路径，游戏中玩家所控制的角色在道路上行走。道路被分为若干个“格子”，方便起见将其称为“地块”。角色每移动几步，就在道路上走过几个地块。角色停留在不同的地块上时会触发不同的事件。相邻的若干个地块组成一个路段。

（3） 地产

地产是游戏场景中的一个重要元素，每个地产都与一个地块相连接，当角色停留在对应的地块时，就会触发地产的事件。地产分为两类，一类为普通地产，另一类为商业地产。普通地产有 4 个等级：空地（0 级）、1 级、2 级、3 级。当角色停留在空地上时，可以选择购买该地产，此时角色会取得该地产的所有权，同时地产会升级到 1 级，此后再次停留在该地产上时，可以选择继续升级至 2 级、3 级。当角色停留在所有权为其他角色的地产上时，需要支付租金。租金与地产的等级以及同一路段上地产所有者拥有的地产数量有关。商业地产也有 4 个等级：空地（0 级）、1 级、2 级、3 级。当角色停留在商业地产的空地上时，可以选择购买该地产，并选择要建设的建筑类型。建筑类型有三种：酒店、古玩市场、写字楼。酒店的功能为，在地产的非所有者停留时，该角色须进入酒店住宿若干回合，并支付住宿费用；古玩市场的功能为，在地产的非所有者停留时，该角色须支付费用；写字楼的功能为，每隔若干回合使写字楼的所有者获取一定的租金。

（4） 事件地块

与地产连接的地块之外，还有很多种其他的地块，当角色停留在不同的事件地块上，将会触发不同的游戏事件。

（5） 卡牌

每个角色都可以持有若干张卡牌，可以持有的最大数量由角色的卡牌背包的大小决定。每种卡牌都有特定的效果，主要分为主动效果和被动效果。拥有主动效果的卡牌须持有者在自己的回合时使用，拥有被动效果的卡牌在满足使用条件时会自动触发效果。

（6） 神仙

神仙随机出现在地块上，当角色停留在神仙所在的地块上时，神仙会附身于角色上，接下来的 4 回合内对角色产生影响。4 回合结束后神仙会自动离开角色。如果角色已有神仙附身并再次停留在有神仙的地块上时，原有的神仙会离开角色，新的神仙再进行附身。

（7） 胜利条件

对于每一局游戏，有一个获得游戏胜利的条件，当角色率先满足胜利条件时，该名角色即获得游戏的胜利。

3.3.2 游戏场景功能分析

游戏场景，或称为游戏地图，即为游戏主体进行的背景、所处的空间。好的游戏场景设定能够充实游戏主体的内容，使游戏更富有真实感，使玩家更有身临其境的感觉。更重要的是，游戏的一个重要需求是“用鼓浪屿等厦门知名景点设置游戏场景”，因此游戏场景的功能尤为重要。

游戏有五个主要场景，分别对应厦门的五个知名旅游观光区域：鼓浪屿、环岛路、白鹭洲、五缘湾、岛外。同时，每个场景中还必须设置该旅游观光区域中的知名景点或标志性建筑物，在游戏中将以 3D 模型的方式展现。

3.3.3 游戏模式功能分析

游戏主体确定之后，还需要围绕游戏主体继续丰富游戏的内容，首先是游戏模式的功能分析。《悠游厦门》主要分为两种游戏模式：剧情模式和自由模式。

（1） 剧情模式

剧情模式即为闯关模式，每一个关卡都有一些角色对话和剧情，关卡之间的剧情都是连贯的，共同组成一个完整的故事。故事的背景即为厦门中秋嘉年华活动和国际虚拟投资旅游大赛，这不仅丰富了游戏的内容，还紧紧贴合了游戏的功能需求和定位——厦门中秋嘉年华活动的一部分。

游戏的剧情模式中有五个大关卡，每个大关卡中有 10 个小关卡，每个小关卡对应一次游戏主体的进行。起初玩家只能够进入第一个大关卡中的第一个小关卡，其余关卡暂时锁定。当玩家获得关卡的胜利后，下一个小关卡即被解锁，解锁后玩家即可进入下一个关卡中继续闯关。每个关卡中的出场角色由剧情的发展而定，玩家所扮演的角色在

不同的关卡中也不尽相同，玩家扮演的角色之外的角色由 NPC 控制器来控制。游戏关卡的详细功能分析在下文中会详细叙述。

（2） 自由模式

自由模式也是建立在游戏主体上的，玩家可以自由设定游戏场景、游戏人数、起始资金、游戏时间限制和胜利条件，控制一个角色与其他由 NPC 控制器控制的角色一同进行游戏。自由模式中可选择的场景与剧情模式相关，只能够选择剧情模式中已经被解锁了的场景。

（3） 寻宝模式

寻宝模式即为博饼模式，博饼模式与游戏主体无关，只是模拟现实中的博饼游戏。每次博饼游戏中有若干数量的奖品，奖品均为游戏道具。4 名角色轮流投掷骰子，当投掷出特定的点数组合时，可以获得相应的奖品。博饼游戏可以一直进行到所有奖品发放完毕为止，也可中途退出。进行了一次博饼游戏后，需要等待一段时间后才能进行下一次博饼游戏。

3.3.4 小游戏功能分析

小游戏是游戏中的游戏，规则独立，与游戏主体的其他内容无关。“小”体现在规则简单，时间短，进行一次小游戏只需要不到一分钟时间。角色停留在小游戏地块时，将触发小游戏事件，进行一次小游戏后，根据小游戏获得的积分，角色能够获得相应数量的点券。

游戏中有五个小游戏，其详细功能分析如下。

（1） 激情赛艇。一辆赛艇在水面上向前开，玩家可以控制赛艇左右移动。水面上有金币和汽油，当赛艇撞到金币时，可以获得积分，当赛艇撞到汽油时，可以加速一段时间。游戏时间为 30 秒。

（2） 宝箱迷踪。游戏开始时，有 5 点、10 点、50 点、200 点、500 点五个点券，分别将其装入 5 个箱子中，并快速随机移动箱子的位置，移动停止后，玩家选择其中的一个箱子，获得其中的点券。

（3） 胡里炮台。有黄色、红色、紫色三个炮台，玩家可以控制其开炮攻击前方驶过的船只，船只同样有黄色、红色、紫色三个种类，当炮台发出的炮弹击中相同颜色的船只时，可以获得积分，如果击中不同颜色的船只，则积分会减少，如果没有击中船只，则积分不变。游戏时间为 30 秒。

（4） 天赐富贵。天空中有一辆飞艇，随机扔下气泡，玩家操作地上的手左右移动，用针刺破落下的气泡。游戏时间为 60 秒。刺破不同的气泡会触发不同的效果。

（5） 勇闯迷宫。迷宫有 4 个入口和 4 个出口，4 个出口上分别有 10 点、50 点、

200 点、500 点的点券。玩家须在游戏开始后的 6 秒内选择 4 个入口中的其中一个入口进入迷宫，角色会自动在迷宫里走，最后走到出口，并获得该出口处的点券。

3.3.5 游戏关卡功能分析

剧情模式总共有五个大关卡，每个大关卡中有 10 个小关卡，其中有 2 个小游戏关卡，7 个小场景关卡，1 个大场景关卡。

小场景关卡中，道路的地块数、地产的数量较少，角色可以在短时间内走完道路的一个回环，使对自己的地产升级的机会以及给别的角色付租金的机会增加，从而能够使游戏更快速的进行，使玩家在短时间内能够完整地体验到完整的一局游戏。

每个大场景关卡都是对应的旅游观光区域的完整场景，道路的回环较大，地产数量较多，进行一局游戏需要的时间也较长。但是由于场景较大，能够容纳的内容也较多，场景的地形能够更细致地表现旅游观光区域的地貌特征，场景中出现的旅游景点数量也较多，玩家能够在享受游戏的同时，逐渐对厦门的旅游景点进行了解。

3.3.6 道具功能分析

游戏道具，也是玩家在游戏主体的进行过程中可以使用的物品，每个道具都有特殊的效果，效果的类型与游戏卡牌相似：主动效果和被动效果。与游戏卡牌不同的是，游戏卡牌只是和每一局游戏相关，卡牌不会被带到新的一局游戏当中，当新的一局游戏开始时，角色身上的卡牌将会被重置。而游戏道具是和游戏玩家相关的，只要玩家拥有道具，在任何一局游戏中都可以使用。

道具有两种类型：限时道具和永久道具。限时道具主要通过寻宝模式获得，有使用时间的限制，在游戏主体的进行过程中，道具的使用时间会减少，当使用时间减少为零时，道具即消失。永久道具主要在游戏商城通过元宝购买。没有使用时间限制，可以永久使用。

3.3.7 成就功能分析

成就，即玩家在游戏中完成的一些事件。每个成就包含一个或多个达成条件，当玩家在游戏中达成这些条件时，即可获得成就，同时获得积分。获得的积分将被记录在游戏服务器中，作为发放中秋旅游嘉年华活动奖品的依据。

根据玩家的成就积分，积分排名前若干名的玩家将会出现在排行榜中。

3.3.8 游戏商城功能分析

在游戏商城中，玩家可以使用元宝购买游戏道具。元宝的获得途径主要为充值、即花费现实中的货币来购买。对于不同的平台，充值策略也不同，即每次需要花费多少现实货币来购买多少元宝。默认的充值策略如下表。

3.3.9 新闻公告功能分析

新闻公告模块的功能是将游戏的相关信息传达给玩家。在游戏启动时会出现新闻公告的界面，上面显示具体的新闻公告信息。

新闻公告的信息包含三项内容。

- (1) 游戏更新公告。主要是游戏版本的更新信息、BUG 的修复信息。
- (2) 游戏新闻。游戏相关的新闻。
- (3) 中奖玩家名单。公布游戏每月抽奖的中奖玩家名单。

3.3.10 其他客户端功能分析

为了使游戏更完整、功能和内容更充实，需要加入一些小功能。

- (1) 显示 LOGO（徽标）

在游戏程序启动后，依次显示厦门时代华亿动漫公司的 LOGO、厦门中秋旅游嘉年华的 LOGO、“悠游游戏”的 LOGO，对游戏的制作商、相关活动做一个提示性的介绍。

- (2) 存档、取档

由于基于大富翁的游戏规则，一次游戏主体的完整进行通常需要 30 分钟时间，大场景关卡甚至有可能超过一个小时，因此对于移动设备用户来说，连续长时间进行游戏有时是比较困难的。玩家有可能在游戏进行途中突然需要去处理别的更重要的事情，此时需要对游戏进行存档，将游戏进度记录下来，便可暂时关闭游戏，之后再次启动游戏时可以取档，恢复之前的游戏进度，将游戏继续进行下去。

- (3) 调整游戏速度

由于基于大富翁的游戏规则，游戏进行的节奏比较慢，玩家通常只需要点击投掷骰子，然后就一直等待游戏中的角色轮流走路，等待的过程会非常枯燥。因此需要提供调整游戏速度的功能，使等待的过程能够以较快的速度进行，从而缩短游戏时间，使玩家能够更好地享受游戏。

3.4 用例分析

如图 3.2 所示，在游戏中，参与者只有玩家一个，主要有以下用例：

- (1) 下载游戏。玩家通过互联网从游戏平台上下载游戏，并在自己的设备上安装游戏。
- (2) 选择游戏模式。玩家运行游戏后，通过游戏界面中的操作，选择需要开始的游戏模式。
- (3) 进入游戏关卡。玩家通过游戏界面中的操作，进入游戏关卡，开始游戏的主要内

容。

- (4) 投掷骰子。投掷骰子为游戏的最主要的操作，玩家通过点击游戏画面中央的“GO”按钮来投掷骰子，使角色行走。
- (5) 购买地产。当玩家的角色停留在地产上时，玩家选择消费游戏中的货币，投资、购买地产，之后在游戏中可以通过地产获得收益。
- (6) 进行小游戏。玩家通过触发小游戏事件，进行游戏内部的小游戏。
- (7) 充值。玩家使用手机话费、支付宝等支付途径，购买游戏内的货币——元宝。
- (8) 购买游戏道具。玩家通过消费元宝来换取游戏道具。
- (9) 使用游戏道具。玩家在游戏过程中，通过打开游戏的道具界面，使用游戏道具，触发特殊效果。
- (10) 获得游戏成就。玩家在游戏中，通过达成一些特殊条件，获得游戏成就并获得积分。
- (11) 阅读新闻公告。玩家在游戏中阅读游戏的新闻公告信息，获取游戏更新、奖励发放等与游戏相关的最新动态、消息。
- (12) 使用积分兑换奖励。玩家通过消费在游戏中获得的积分，兑换旅游门票等实体奖励。

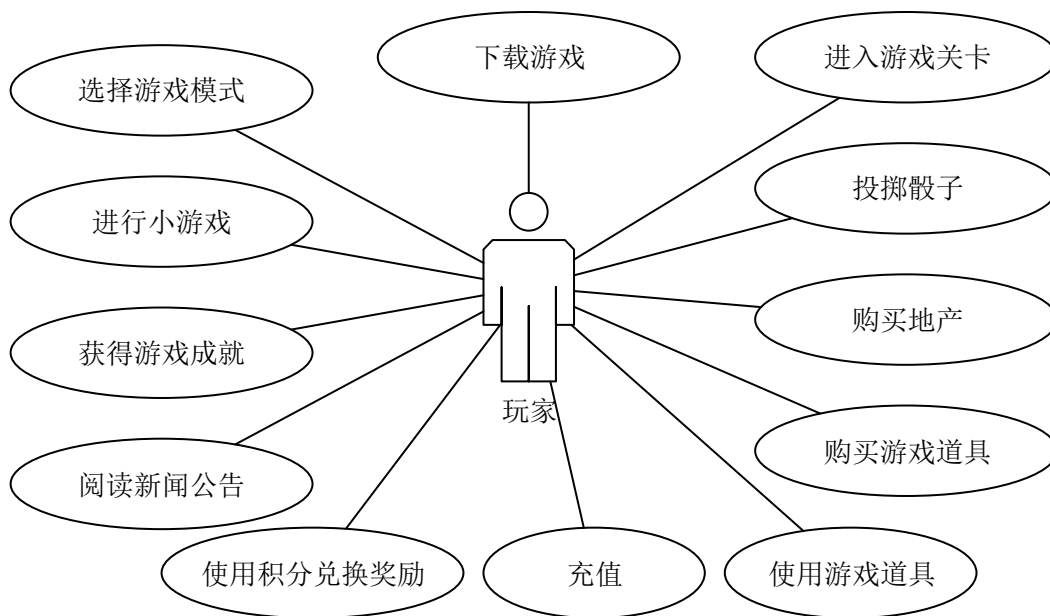


图 3.2 游戏主要功能用例图

3.5 数据建模

游戏中包含两类数据，一类是存储在游戏服务器中的数据，包括玩家数据和新闻公告数据，另一类是存储在游戏客户端中的数据，包括成就数据、卡牌数据、地图事件数

据、关卡数据等。

3.5.1 玩家数据建模

每一名玩家在进行游戏的过程中，需要记录许多数据，如：玩家使用的角色名、已通关的关卡数、持有的道具、已获得的成就和已获得的积分。这些数据通过实体-联系图表示如图 3.3 所示：

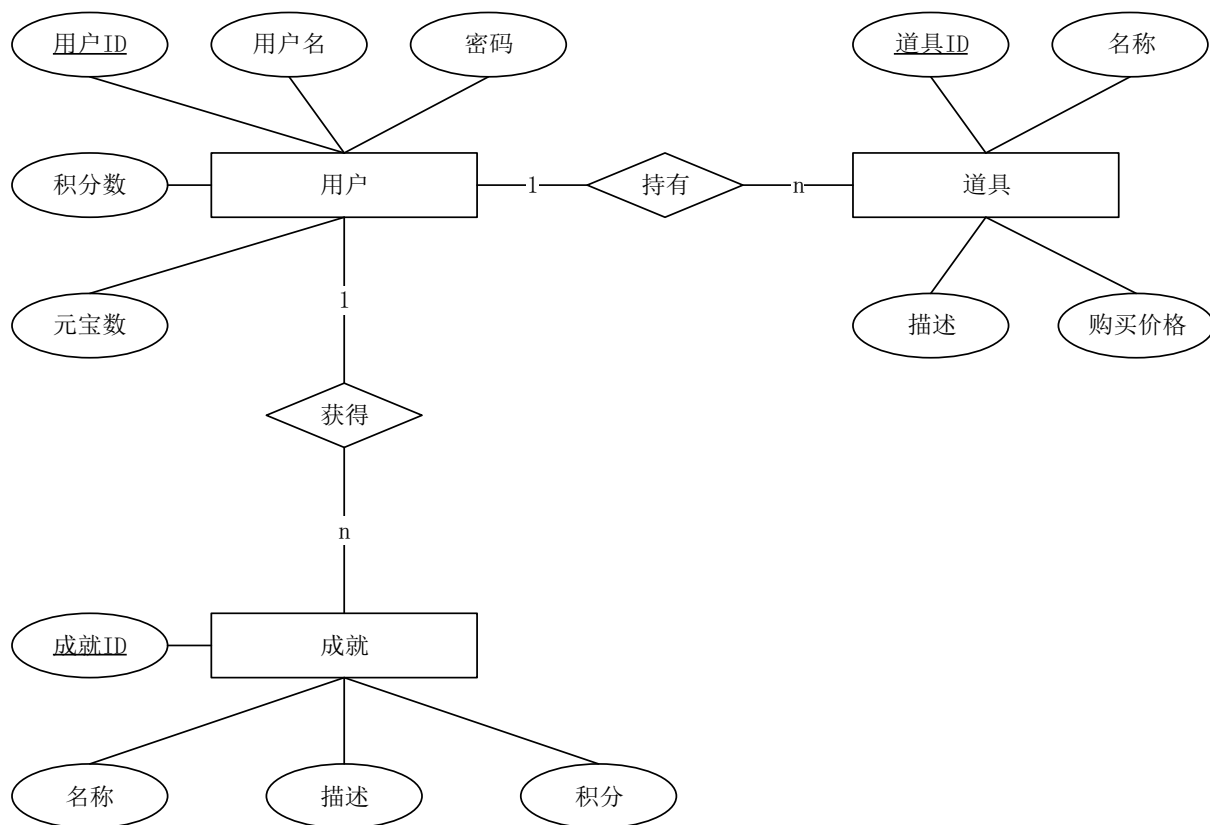


图 3.3 游戏数据实体-联系图

“用户”实体主要记录玩家的各种游戏数据，主要包括以下属性：

- (1) 用户 ID。玩家在注册游戏时，游戏服务器会给用户分配一个唯一的数字 ID，用于唯一地标识每个用户。
- (2) 用户名。玩家在注册或登录游戏时使用的用户名。
- (3) 密码。玩家在注册或登录游戏时使用的密码。
- (4) 积分。玩家在游戏中获得的累计积分。
- (5) 元宝数。玩家通过充值获得的元宝的数量，元宝主要用于购买游戏道具以及开通一些特殊游戏功能。

“道具”实体主要记录游戏中玩家可使用的道具信息，主要包括以下属性：

- (1) 道具 ID。用于唯一地标识道具。

- (2) 名称。道具的名称。
 - (3) 描述。道具的使用效果的文字描述，使玩家通过阅读道具描述可以了解道具的使用效果。
 - (4) 购买价格。表示玩家需要消耗指定数量的元宝来获得该道具的使用权。
- “成就”实体主要记录玩家在游戏中获得的成就的信息，主要包括以下属性：

- (1) 成就 ID。用于唯一地标识成就。
- (2) 名称。成就的名称。
- (3) 描述。成就的达成条件的文字描述，使玩家通过阅读成就描述可以了解到如何达成条件并获得成就。
- (4) 积分。获得成就时能够累积的积分。

每个用户均可以持有若干个道具，并获得若干个成就，因此“用户”实体与“道具”实体的关系为 1 对 n 的关系，“用户”实体与“成就”实体的关系也为 1 对 n 的关系。

3.5.2 新闻公告数据建模

游戏的更新内容、奖励的发放公告等消息需要通过服务器发送给玩家。这些数据通过实体-联系图表示如图 3.4 所示。

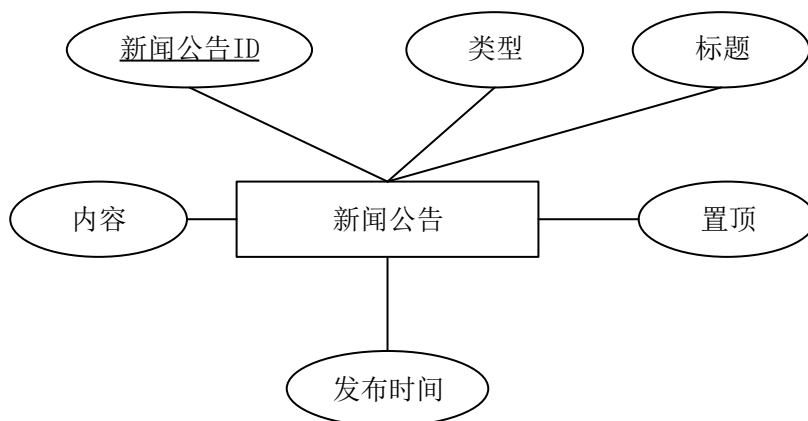


图 3.4 新闻公告数据实体-联系图

“新闻公告”实体主要记录新闻公告的详细信息，主要包括以下属性：

- (1) 新闻公告 ID。用于唯一地标识新闻公告。
- (2) 新闻公告类型。有三种类型的新闻公告：游戏更新公告、游戏新闻和中奖玩家名单。
- (3) 新闻公告标题。每条新闻公告都有一个标题，表示消息的主题。
- (4) 新闻公告内容。叙述消息的详细内容。

- (5) 置顶。表示是否将消息置于新闻公告列表的顶部，使玩家能够第一时间看到重要的消息通知。
- (6) 发表时间。表示新闻公告的发布时间。

3.5.3 成就数据建模

游戏客户端中记录着玩家可以获得的成就数据，客户端会自动根据玩家进行游戏的状态来判断玩家是否达成了获得成就的条件。成就数据通过实体-联系图表示如图 3.5 所示：

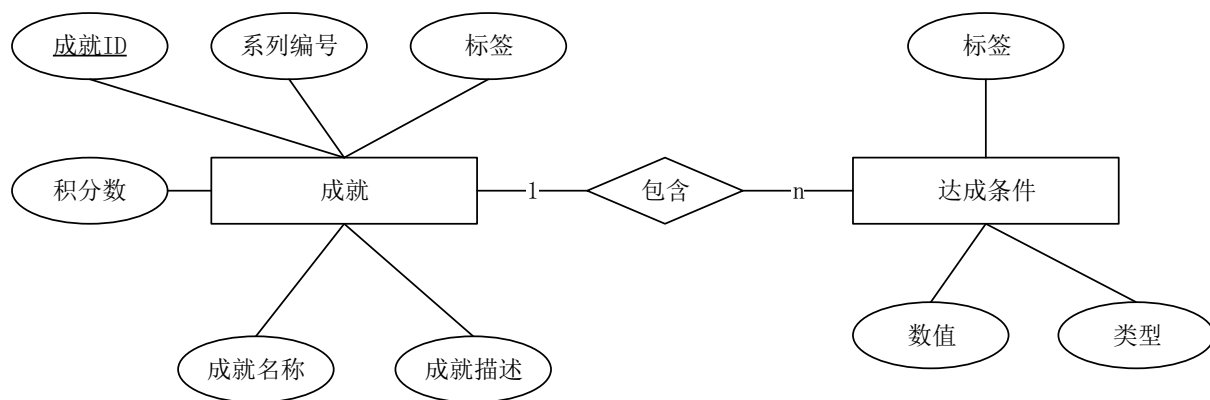


图 3.5 成就数据实体-联系图

“成就”实体主要记录成就的基本信息，主要包括以下属性：

- (1) 成就 ID。用于唯一标识成就信息。
- (2) 成就名称。
- (3) 成就描述。用于描述完成成就所需要达成的条件。
- (4) 积分。完成成就时获得的积分。
- (5) 系列编号。达成条件相似的一组成就为系列成就，通常会依次先后地完成系列中的成就，这一组成就有相同的系列编号。
- (6) 标签。根据成就的类型进行标记，便于分类。

“达成条件”实体主要记录达成条件的信息，主要包括以下属性：

- (1) 类型。达成条件的类型。
- (2) 数值。与类型属性一起组成条件的判断依据，当玩家的状态满足当前条件类型的当前数值时，条件才算达成。
- (3) 标签。根据达成条件的类型进行标记，便于分类。

每个成就包含若干个达成条件，当这些达成条件同时满足时，玩家才可以完成成就。

3.5.4 卡牌数据建模

卡牌和道具是大富翁游戏规则的重要部分。卡牌数据通过实体-联系图表示如图 3.6 所示：

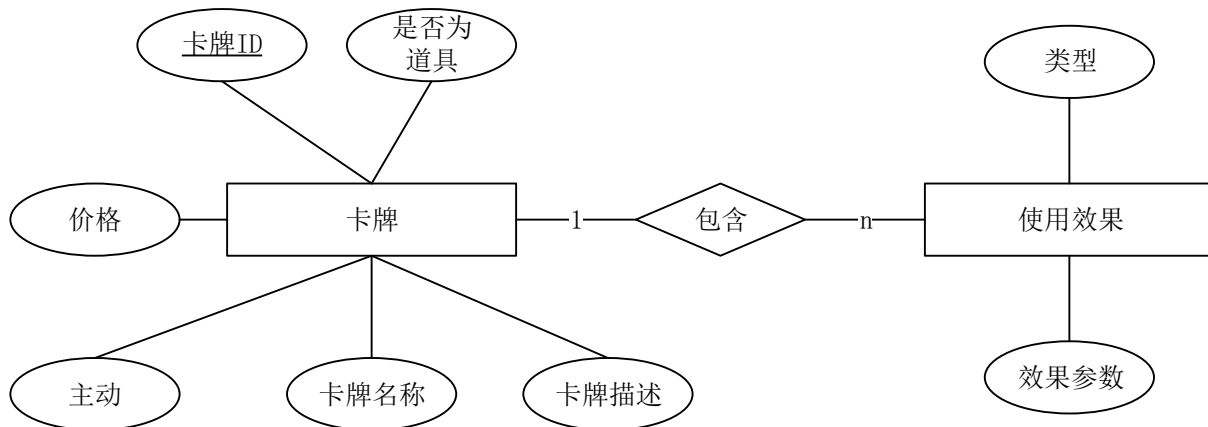


图 3.6 卡牌数据实体-联系图

“卡牌”实体主要记录卡牌的基本信息，主要包括以下属性：

- (1) 卡牌 ID。用于唯一标识卡牌信息。
- (2) 卡牌名称。
- (3) 卡牌描述。用于描述完卡牌的功能、使用效果、使用条件等基本信息。
- (4) 是否为道具。道具是一种特殊的卡牌。
- (5) 价格。购买卡牌所需要的点券数或元宝数。
- (6) 主动。表示卡牌是否可以被主动使用，不能主动使用的卡牌效果均为自动触发类。

“卡牌效果”实体主要记录卡牌使用效果的信息，主要包括以下属性：

- (1) 类型。使用效果的类型，不同的效果类型会触发不同的游戏事件。
- (2) 效果参数。使用卡牌时产生的效果多少、大小的依据。

每张卡牌包含若干个卡牌效果，当卡牌被使用的时候，这些效果会依次被触发。

3.5.5 地图事件数据建模

地图事件是游戏主体中会发生的一些随机事件，其数据通过实体-联系图表示如图 3.7 所示：

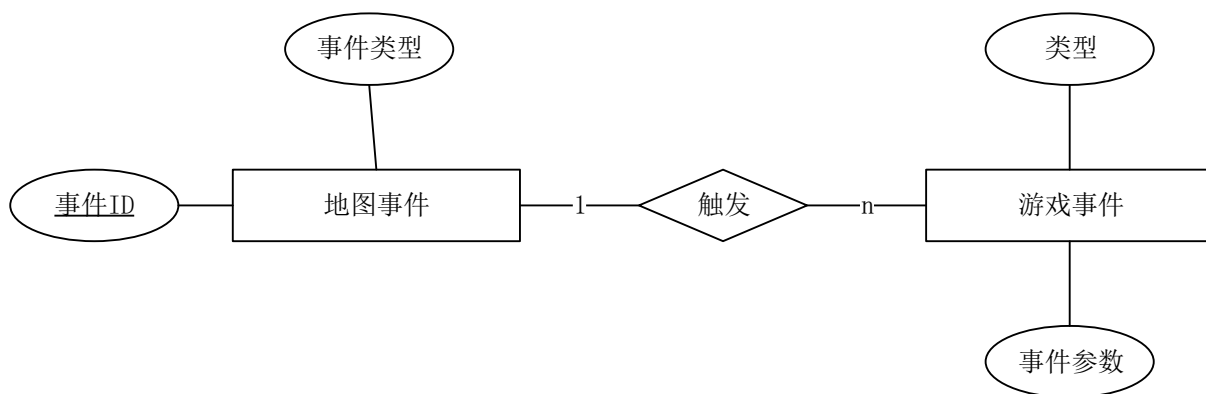


图 3.7 地图事件数据实体-联系图

“地图事件”实体主要记录地图事件的基本信息，主要包括以下属性：

- (1) 事件 ID。用于唯一标识地图事件信息。
- (2) 事件类型。不同的事件类型出现的情况不同。

“游戏事件”实体主要当地图事件出现时会触发的游戏事件信息，主要包括以下属性：

- (1) 类型。游戏事件的类型，不同类型的游戏事件会在游戏中表现出不同的效果。
- (2) 事件参数。某些事件类型的参数，如“获得现金 500”中，500 即为些事件的参数。

3.5.6 关卡数据建模

关卡数据为游戏中设置的关卡的基本信息，其数据通过实体-联系图表示如图 3.8 所示。

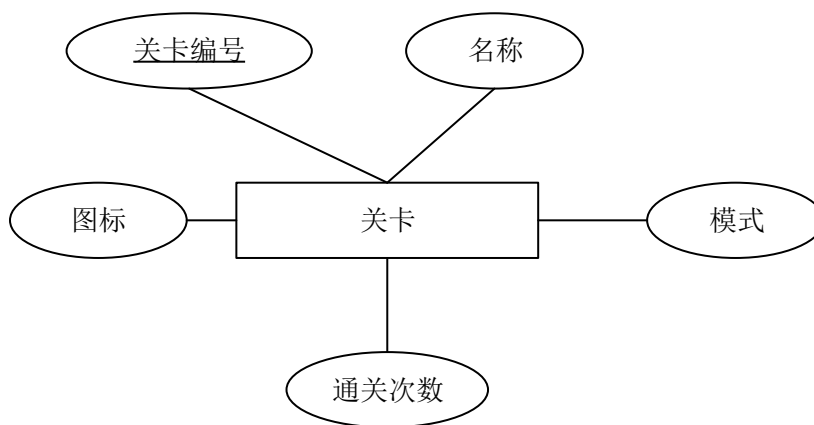


图 3.8 关卡数据实体-联系图

“关卡”实体用于记录关卡的基本信息，主要包括以下属性：

- (1) 关卡编号。用于唯一地标识关卡，其格式为“大关卡号-小关卡号”，如“3-1”。

- (2) 名称。关卡的名称。
- (3) 图标。每个关卡都有一个能抽象表示该关卡的小图标。
- (4) 模式。表示该关卡的游戏模式。
- (5) 通关次数。表示玩家在该关卡的通关次数。

3.6 过程建模

参与游戏的用户主要有游客、会员两种类型，游客通过注册功能可以成为会员，从而能够累积积分，并通过兑换积分获取奖励。游戏总体的上下文数据流图如图 3.9 所示。

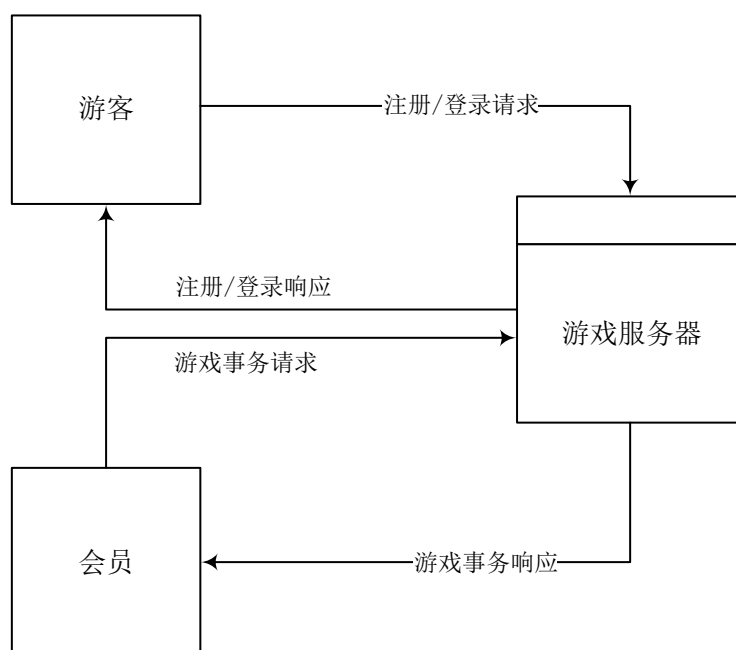


图 3.9 游戏上下文数据流图

其中，游客主要有“注册”和“登录”两个业务活动。“注册”活动中，游客将用户名与密码发送给游戏服务器，游戏服务器在数据库中检查是否有重复的用户名，若没有重复则新增用户数据记录，并将注册结果作为响应发送给游客。“登录”活动中，游客将用户名与密码发送给游戏服务器，游戏服务器从数据库中取出该用户的密码，检查密码与游客发送的是否一致，并将登录结果作为响应发送给游客。

游客登录成功后，则可以会员身份发起游戏事务。会员主要有“获得积分、成就”和“购买道具”两个业务活动。“获得积分、成就”活动中，会员将所获得的成就信息发送给游戏服务器，游戏服务器在数据库中记录该会员所获得的成就以及相应的积分变化情况。“购买道具”活动中，会员将想要购买的道具信息发送给游戏服务器，游戏服务器将用户的元宝数量变化情况记录到数据库中，并新增道具数据，最后将购买道具的结果作为响应发送给会员。

由以上分析，游戏主要业务活动的数据流图如图 3.10 所示。

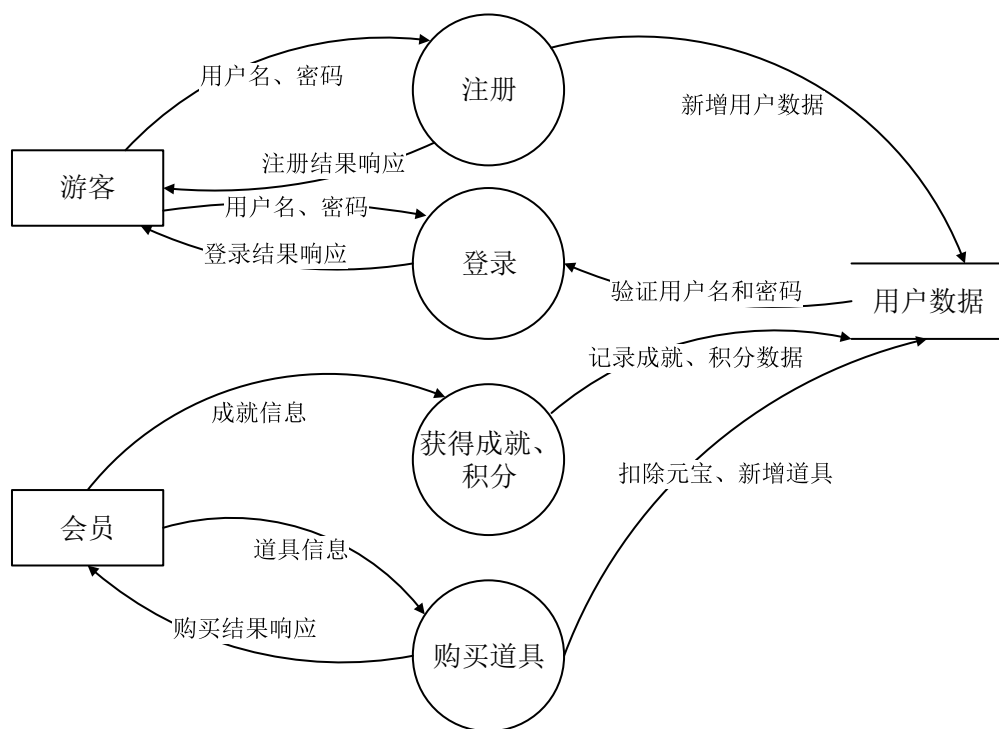


图 3.10 游戏业务活动数据流图

3.7 本章小结

本章首先陈述了《悠游厦门》游戏开发项目的基本业务，叙述了项目背景和开发任务，然后对功能需求进行了建模，并通过实体-联系图分析了游戏中需要使用到的数据结构，最后通过数据流图分析了游戏的基本流程和活动，从而清晰完整地分析了游戏开发项目的整体需求，为开发工作指明了方向，做好了准备。

第四章 游戏设计

在正式开始游戏开发工作之前，首先要对游戏的内容和游戏系统进行详细的设计。游戏的规则是游戏的基本，所有的游戏内容都是围绕着游戏的基本规则展开的。然而，游戏规则仅仅是游戏的极小的一部分内容，围绕着游戏规则展开的大量内容才是构成游戏的主要成分，依靠这些内容，游戏才具有实体，才具有可玩性。同时，从程序开发的角度上看，支撑起游戏内容的则是游戏系统以及各个模块。

本章对游戏系统及游戏内容进行详细设计。

4.1 总体软件体系结构

根据游戏的需求分析，游戏不需要提供多人联机共同进行游戏的功能，因此游戏为单机游戏。但是，玩家的积分等数据需要进行统计，以根据积分给玩家发放奖励。同时，若将玩家的游戏数据保存在玩家的设备中，容易被玩家擅自修改，导致数据的不真实，因此，需要将玩家的数据保存在游戏的服务器中。

根据以上分析，游戏采用客户端/服务器（C/S）的体系结构。系统分为两大部分，一部分是游戏的客户端部分，即在玩家的设备中运行游戏程序。客户端是在玩家设备上运行的游戏程序，是游戏的最主要的部分，游戏的大部分内容都在客户端上。由于内容比较多，结构比较复杂，需要将客户端划分为多个系统、模块来进行设计。另一部分是游戏的服务器端部分，主要用于储存玩家的积分等游戏数据。

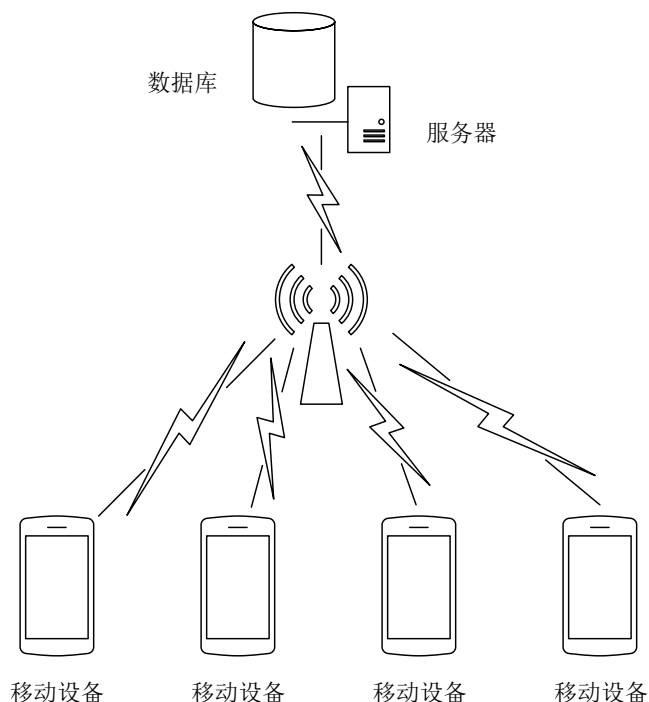


图 4.1 总体软件体系结构

4.2 客户端功能模块设计

游戏的客户端是游戏的最重要的部分，也是内容最多、结构最复杂的部分，因此对游戏客户端的设计是尤其重要的。对客户端的设计，需要将客户端分为多个层次、多个模块，分别进行设计。

《悠游厦门》游戏选择从大到小对客户端进行设计，首先对客户端的总体结构进行设计。

根据游戏的需要分析，客户端的功能主要分为：三个游戏模式、充值元宝并购买道具、查看个人成就和积分排行榜。通过客户端的界面状态变化关系，可以将客户端的主要功能串联在一起，如图 4.2 所示。图中每个状态（即界面或场景）都可以通过点击“关闭”或“退出”返回到前一个状态或直接退出游戏。由于图中的状态转移数量较多，为使状态图的重要部分更清晰，“关闭”操作的状态转移在图中省略了。

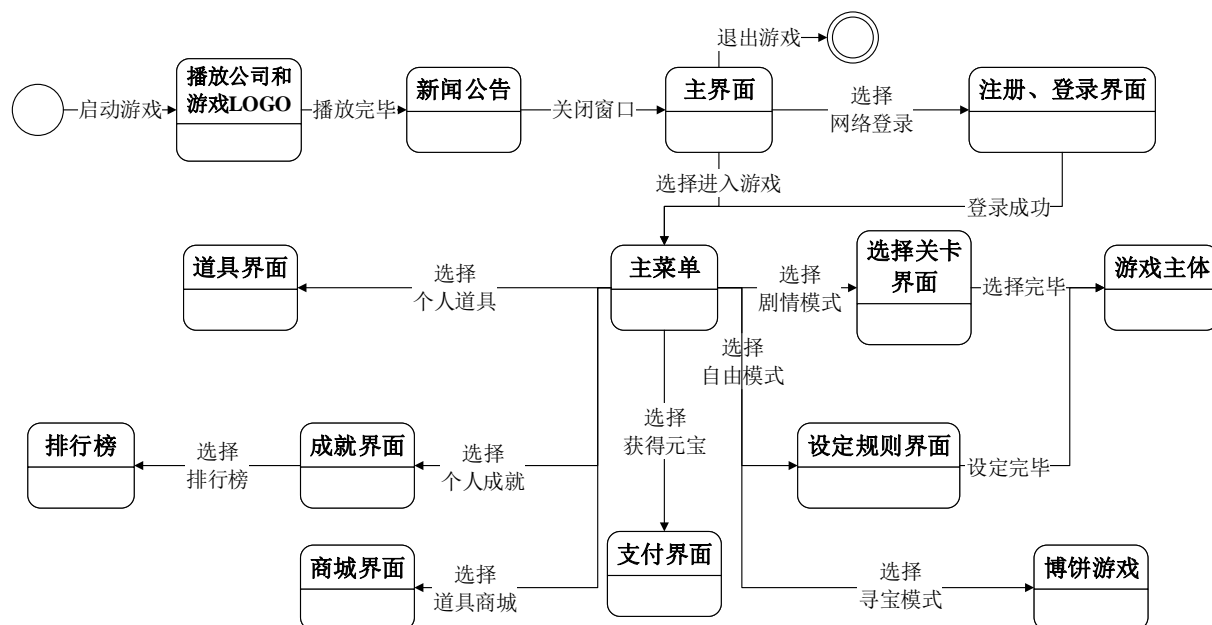


图 4.2 客户端界面迁移状态图

根据客户端的功能，将客户端的模块划分为：游戏主体模块、小游戏模块、成就系统模块。模块划分如图 4.3 所示。



图 4.3 游戏客户端模块划分图

4.2.1 游戏主体设计

游戏主体是游戏的最核心的部分，游戏的所有功能都是围绕着游戏主体展开的，都是为游戏主体服务的，因此首先进行游戏主体的设计。

游戏主体中的对象种类较多，关系错综复杂，因此同样需要将其分为多个层次、多个模块进行设计。

(1) 游戏主体的核心类——Game 类

游戏主体的核心类——Game 类是游戏主体即游戏核心玩法进行的中枢、引擎，是整个游戏主体的管理中心。Game 类的 UML 类图如图 4.4 所示。

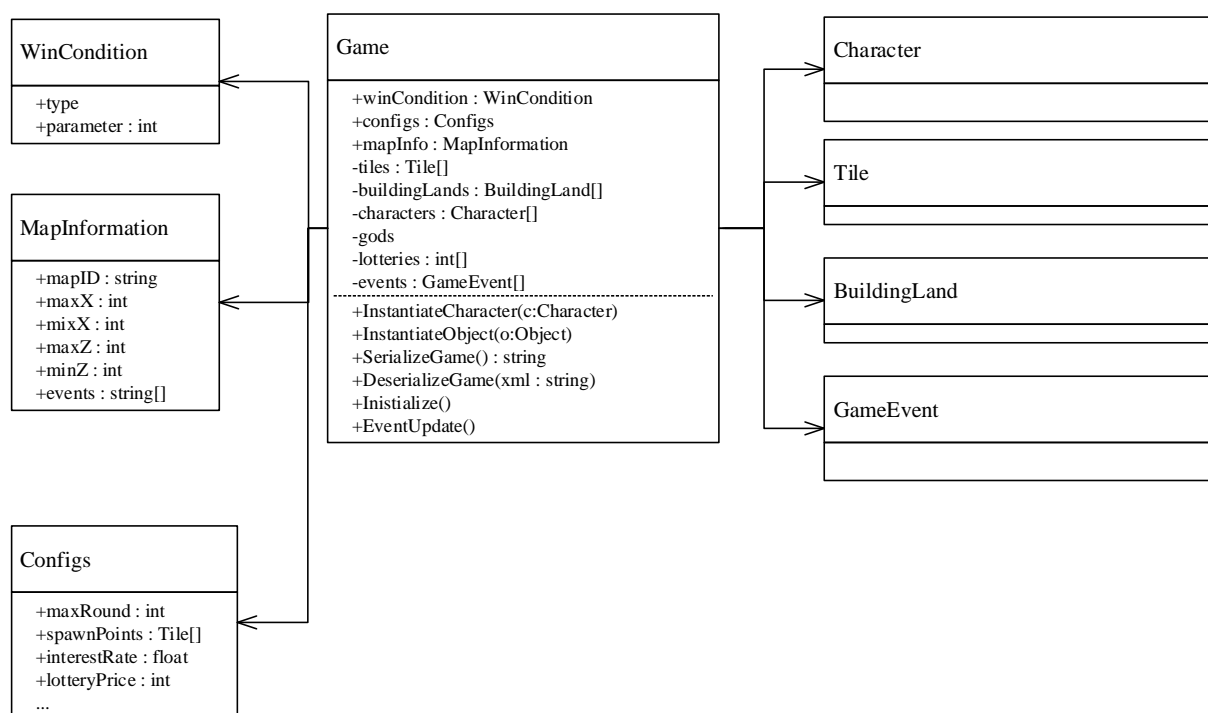


图 4.4 Game 类的 UML 类图

Game 类主要包含以下功能。

- 游戏参数配置。游戏的胜利条件（WinCondition 类）、最大回合数、角色的出生点、银行利率、彩票奖金等各项参数配置（Configs 类）。
- 管理场景信息。场景信息由 MapInformation 类保存，具体信息包括：场景 ID、场景的镜头限制范围、新闻事件中“发生意外”的具体意外事件内容。
- 动态加载场景中的对象。对象主要为游戏中的角色（Character 类）、道路上出现的神灵。
- 管理所有道路（Tile 类）和地产（BuildingLand 类）地块的信息。
- 管理所有角色以及角色的行动顺序。

- 管理在场景中的所有神灵。当神灵因为附身或其他原因消失,经过若干回合后,会再次随机在场景中刷新。
- 管理彩票站奖金的累计和发放。
- 存档和取档。存档即为将当前的场景基本信息和场景中的各角色、地产、道路等所有对象的当前状态保存成游戏存档记录。取档即存档的逆过程,将游戏存档记录中的数据全部恢复到游戏场景当中。
- 初始化游戏。在新的游戏开始或者取档完成时,初始化游戏场景中的所有对象。
- 管理和驱动游戏事件(**GameEvent** 类)。使游戏中发生的各种事件轮流依次进行。

(2) 道路对象

道路对象,即能够在游戏场景中的道路上活动的游戏对象。其 UML 类图如图 4.5 所示。

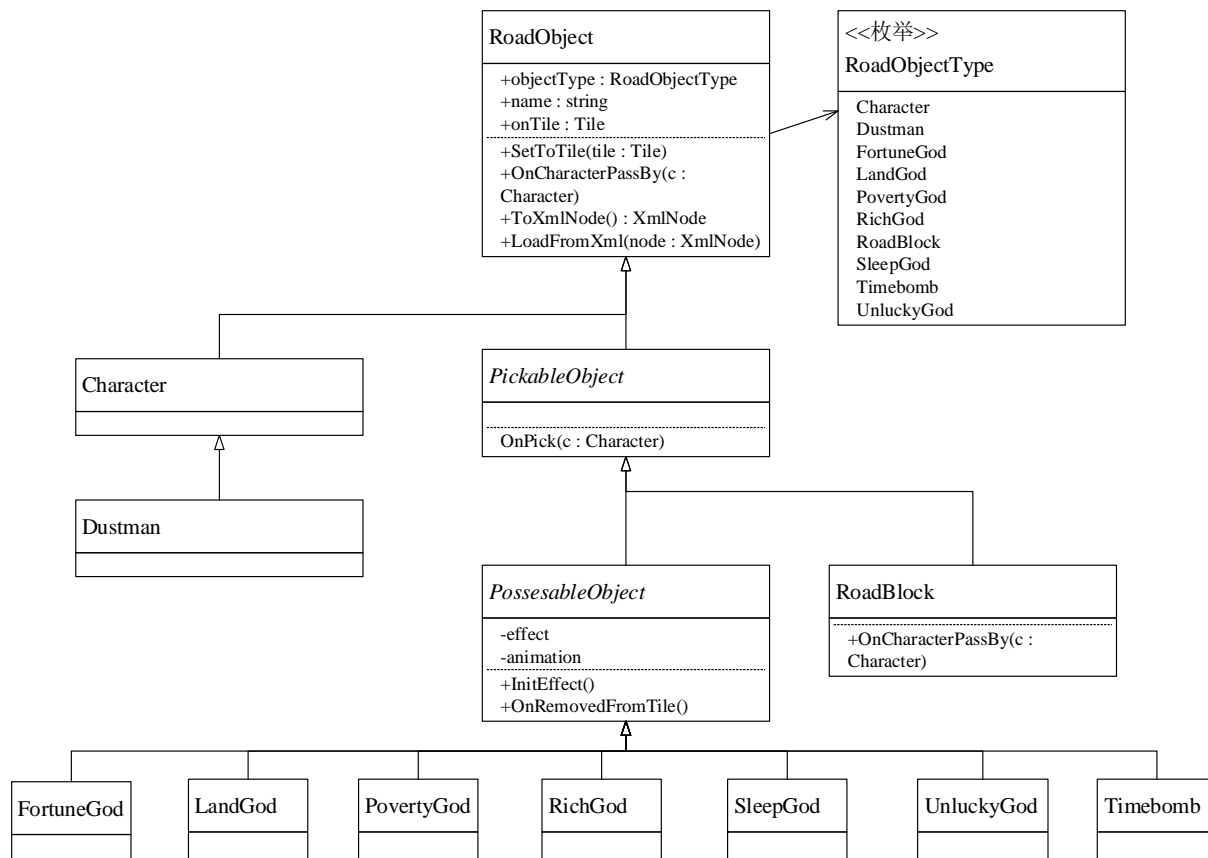


图 4.5 道路对象继承关系图

所有的道路对象可以抽象为一个虚基类 **RoadObject**, 它具体三个属性: 道路对象类型、对象名称、所在地块, 同时有两个关键的方法: 一个是 **SetToTile**, 作用为将对象移动到指定的地块去, 另一个是 **OnCharacterPassBy**, 当角色行走经过该对象时, 此方法会被调用。

根据游戏设计，道路对象有：游戏角色、卡牌中的清洁工（Dustman）、路障（RoadBlock）、定时炸弹（Timebomb）、福神（FortuneGod）、土神（LandGod）、穷神（PovertyGod）、财神（RichGod）、睡神（SleepGod）、衰神（UnluckyGod）。根据道路对象各自的特点，可以将其分为以下三类。

- 角色类对象。可以在场景中的道路上行走，并和道路上的其他对象发生交互。由于清洁工也是在道路上行走的，因此清洁工也属于角色类对象。角色类的详细设计在下文中叙述。
- 可拾取对象（PickableObject）。当角色停留在道路上的某个地块时，如果地块上有可拾取对象，则会触发该拾取该对象的事件。可拾取对象类为虚基类，成员中有一个虚方法 OnPick，当角色停留在对象所在的地块时，OnPick 方法会被调用。路障类以及可附身对象类均从可拾取对象类中派生。
- 可附身对象（PossessableObject）。当角色停留在道路上的某个地块时，如果地块上有可附身对象，则会触发该对象的附身事件。当发生附身之后，可附身对象会转变为角色身上的一个状态效果。可附身对象类也虚基类，拥有状态效果这一成员。定时炸弹类和所有神灵类都从可附身对象类中派生。

（3）道路和地产地块

道路和地产地块都是在游戏场景的地面上的对象，游戏中只使用一个平面作为道路和地产放置的空间，因此所有的道路和地产地块都在此平面上，互不重叠。因此，为方便道路和地产地块对象的管理，将场景地面的平面像围棋棋盘一样，横竖垂直地划分成若干个方块，每个方块都有自己的 X 坐标和 Y 坐标。所有的道路和地产地块都被放进地图上的方块之中，因此每个道路和地产地块也拥有 X 坐标和 Y 坐标。但因为道路地块类（Tile 类）和地产地块类（BuildingLand 类）必须继承 Unity3D 的脚本基类 MonoBehaviour 类，所以此处只能将 X 坐标和 Y 坐标的属性抽象为 IPoint2 接口，接口有两个方法，功能分别为获取 X 坐标和 Y 坐标。

道路地块类除了拥有 X 坐标和 Y 坐标两个属性外，还有三个重要的属性：（1）与相邻的其他道路地块的链接：角色类对象在道路上行走时，需要通过道路地块之间的链接寻找行走的路径。（2）当前停留在地块上的道路对象：数量可能不只一个，例如多名角色同时停留在同一个道路地块的情况是允许发生的。（3）与道路地块想链接的地产地块：每个地产地块都有相应的道路地块与其链接，当角色停留在该道路地块时，将触发地产的事件。每个道路地块最多有一个地产地块与其链接，但每个地产地块可能会有同时有两个道路地块与其链接（商业地块）。

同时道路地块类还有三个重要的方法，分别当道路对象进入道路地块时、当道路对象离开道路地块时、当角色在道路地块上停留时被调用。这三个方法是实现各种特殊道

路地块的功能的基础，所有特殊道路地块类都由 `Tile` 类派生，然后通过覆盖（`override`）这三个方法，实现特殊地块的特殊功能。

道路和地产地块的 UML 图如图 4.6 所示。

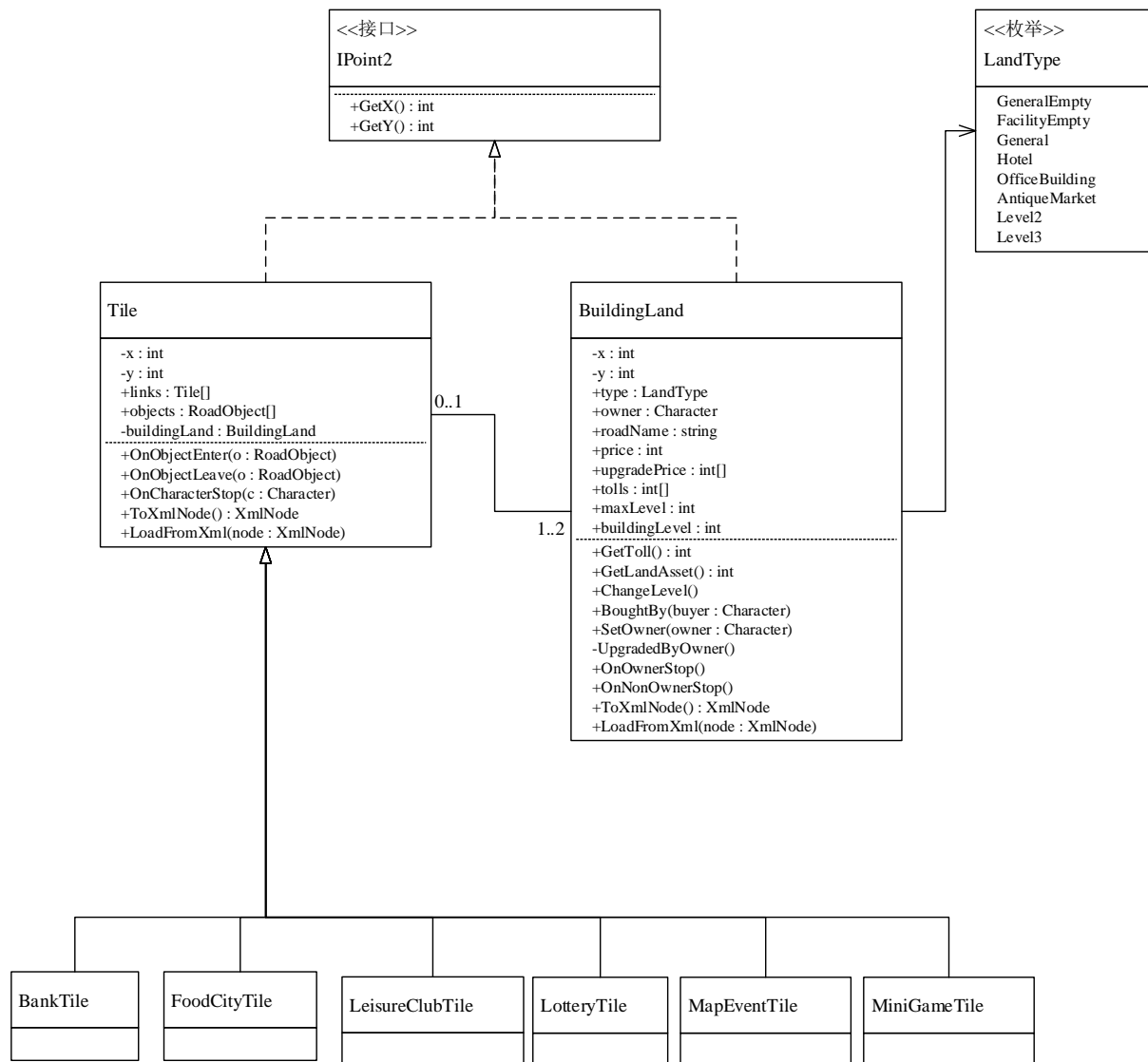


图 4.6 道路和地产地块 UML 类图

地产地块类的属性和方法较多，主要的属性如下。

- X 坐标和 Y 坐标。
- 地产类型。为配合地产模型的显示，地产类型中包含了两类信息：建筑的类型和建筑的等级。通过位运算将这两类信息保存在枚举类型中。
- 拥有者。拥有该地产所有权的角色。
- 空地购买价格和每个等级对应的升级价格、收取的租金。
- 最高等级。能够升级到的最高等级，达到最高等级的建筑无法继续升级。

- 当前等级。地产的当前等级，即空地（0 级）、1 级、2 级或 3 级。
- 地产地块类的主要方法如下。
- 获取当前等级对应的租金数。
- 获取地产的总价值数。地产的总价值为到达当前等级所需要的购买费用与升级费用之和。
- 改变地产的等级。升级或降级。
- 地产被角色购买。
- 更换或移除地产的拥有者。
- 地产拥有者停留在对应的道路地块上，触发升级事件。
- 非地产拥有者停留在对应的道路地块上，触发付费事件。

（4） 游戏事件驱动

游戏主体的进行，是通过一个个游戏事件的依次执行完成的。**Game** 类中保存着一个游戏事件的队列，每当一个事件执行完毕时，**Game** 类会从队列的头部取出下一个游戏事件并开始执行。

游戏事件的执行包含三个主要过程：事件开始、执行循环、事件结束。执行循环是通过 **Game** 类的主循环驱动的。这三个主要过程分别对应游戏事件类（**GameEvent** 类）的三个方法：**Start()**、**Update()**、**OnEnd()**。**GameEvent** 类是一个抽象类，具体的游戏事件类——如投掷骰子事件（**RollDiceEvent** 类）、角色行走事件（**MoveEvent** 类）、地产等级改变事件（**ChangeBuildingLevelEvent** 类）、播放动画事件（**PlayAnimationEvent** 类）等等均继承自 **GameEvent** 类，并覆盖上述的三个方法，实现每个事件的特殊逻辑。

游戏事件的类图和事件执行的活动图分别如图 4.7 和图 4.8 所示。

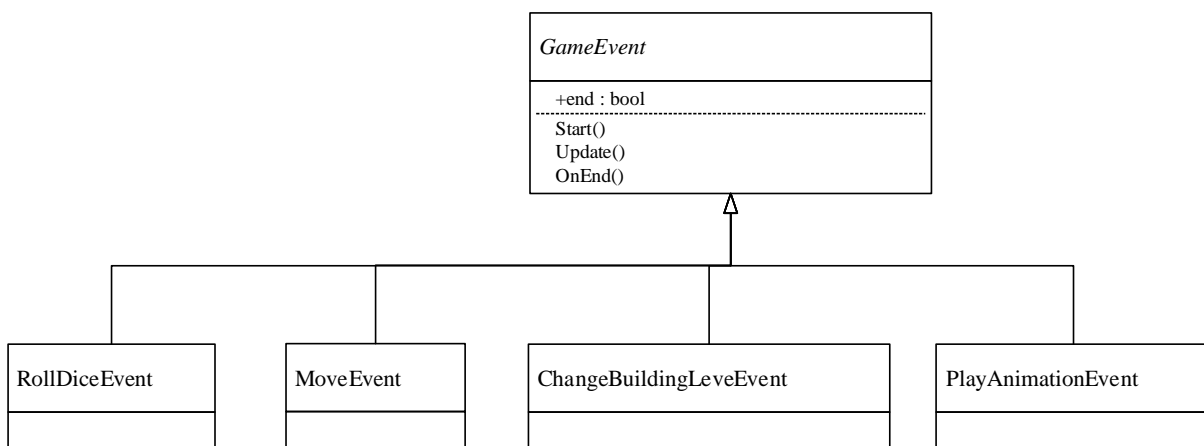


图 4.7 游戏事件 UML 类图

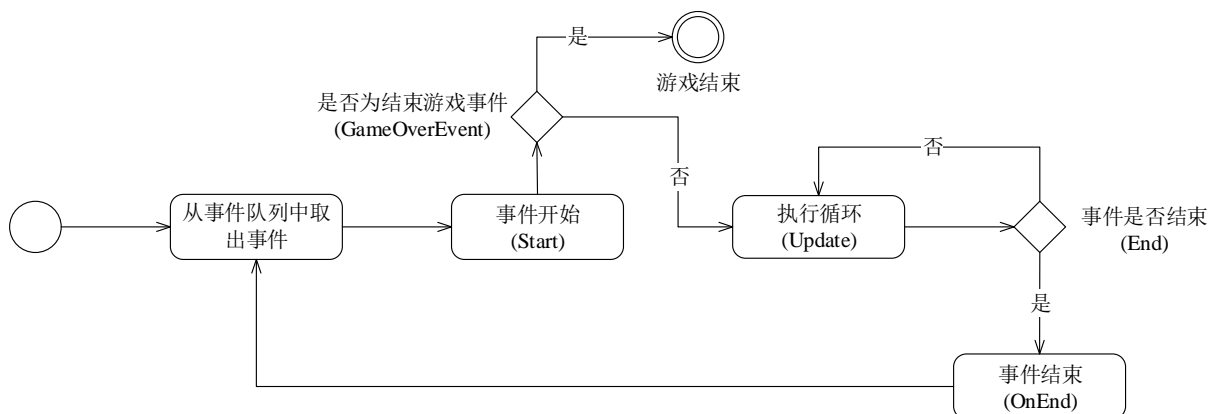


图 4.8 游戏事件 UML 活动图

(5) 角色行动流程

游戏的每一回合中各个角色是按顺序轮流行动的。每个角色的行动流程大致如下：

- 回合开始 (OnTurnStart)。若角色正在酒店住宿或者正处于其他休息状态中，则回合直接结束。
- 使用卡牌。角色可以选择使用卡牌包裹中的卡牌。
- 投掷骰子。投掷骰子，确定要行走的步数。
- 行走。在场景中的道路上行走。
- 停留。行走完由骰子确定的步数后，角色会停留在地块上，触发该地块的停留事件。
- 回合结束 (OnTurnEnd)。回合结束，开始下一个角色的回合。

根据以上分析，可以作出角色行动流程的 UML 活动图，如图 4.9 所示。

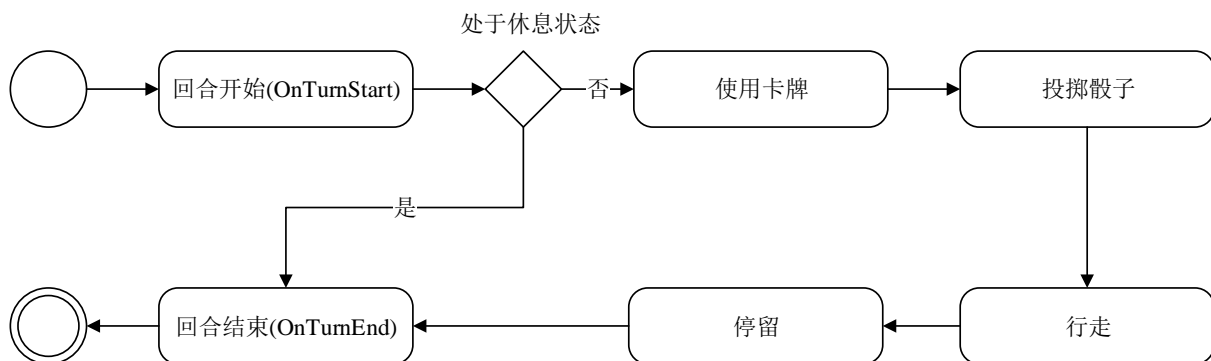


图 4.9 角色行动流程 UML 活动图

(6) 角色类 (Character 类)

游戏中的角色除了上文中叙述的主要属性以及行动流程相关的方法之外，还有一个重要的属性——状态效果。状态效果 (status effect) 是一个游戏术语，它表示对角色状态或能力值的改变，分为增益效果 (buff) 和减益效果 (debuff) 两类，分别表示对角色有益的状态效果和对角色有害的状态效果。根据游戏设计，游戏中的状态效果主要有：神灵的附身效果，游戏卡牌中蜗牛卡、泥鳅卡、包租婆卡、定时炸弹的效果，游戏道具中的漆线雕、珠绣、土笋冻的效果。

状态效果类的 UML 类图如图 4.10 所示。

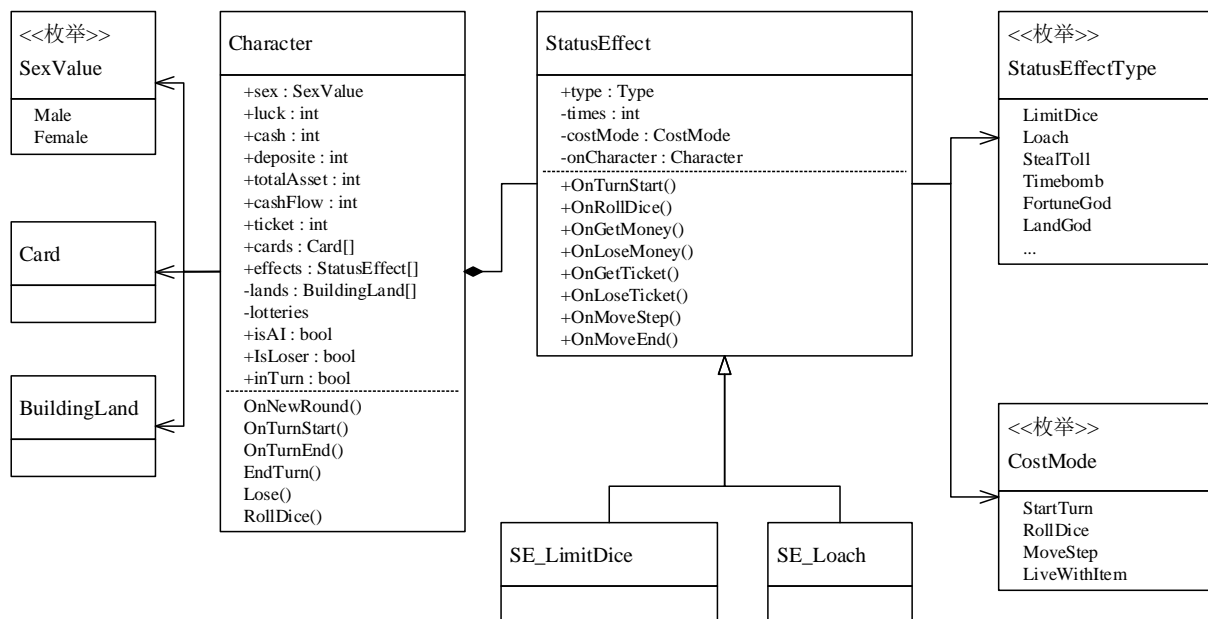


图 4.10 角色类 UML 类图

状态效果类 (**StatusEffect** 类) 有以下成员：

- 效果类型 (**Type**)。
- 可使用次数 (**Times**)。当满足某种条件时，可使用次数就减少一次。当可使用次数减少为零时，状态效果从角色身上移除。
- 使用次数减少方式 (**CostMode**)。表示状态效果的可使用次数减少的条件。主要有四种减少方式：角色回合开始、角色投掷骰子、角色行走、角色持有某道具。
- 生效角色 (**OnCharacter**)。表示状态效果作用的对象角色。
- 触发事件。当生效角色发生某些事件时，状态效果会被触发。触发事件主要有：角色回合开始 (**OnTurnStart**)、投掷骰子 (**OnRollDice**)、获得/失去金钱、获得/失去点券、行走、停留等。每个事件对应一个 **StatusEffect** 类的成员方法，具体的状态效果类均继承自 **StatusEffect** 类并覆盖事件方法，实现特殊的状态效果触

发逻辑。

(7) 游戏卡牌和道具

游戏卡牌和道具虽然在内容和持有方式上不同，但本质上属于同一类，因此将它们都归为 Card 类。Card 类的 UML 类图如图 4.11 所示。

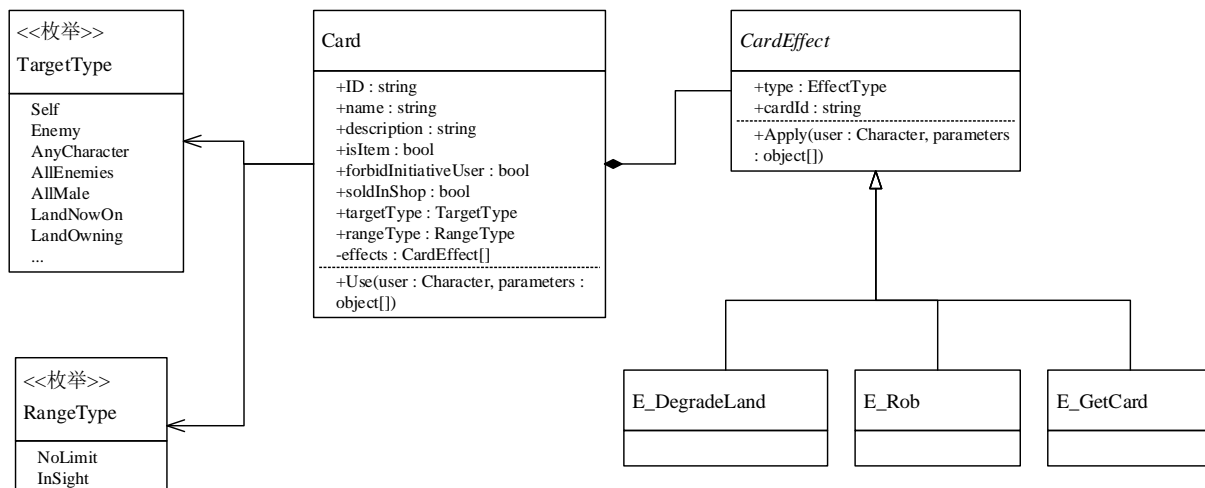


图 4.11 卡牌和道具 UML 类图

Card 类主要有以下成员：

- 卡牌 ID。卡牌的标记符。
- 卡牌名称 (Name)。
- 描述 (Description)。对卡牌的作用、效果的描述。
- 是否为道具 (IsItem)。区分卡牌和道具。
- 禁止主动使用 (ForbidInitiativeUse)。某些只有被动效果的卡牌不允许主动使用。
- 在卡牌商店中出售 (SoldInShop)。大部分卡牌都在可以游戏中的卡牌商店中使用点券购买，但少数高级卡牌只能通过游戏中的特殊事件获得或通过元宝购买。
- 目标类型 (TargetType)。表示卡牌的使用对象的类型，主要有：对自己使用、对一个其他角色使用、对任意一个角色使用、对所有其他角色使用、对所有男性角色使用、对当前所在位置的地产使用、对自己拥有的地产使用、对其他角色的地产使用、对任意建筑使用、对已被购买的地产使用、对任意地产使用、对空闲道路地块使用。可以为多个目标类型的复合，因此目标类型是一个使用位运算表示的枚举类型。
- 范围类型 (RangeType)。表示使用卡牌时选择目标的范围，有“在视野内”和“全场景”两种类型。
- 卡牌效果 (Effects)。一些卡牌可能有多个效果，当使用卡牌时，这些效果会被

轮流触发。卡牌效果类（CardEffect 类）为卡牌效果的基类，它有一个 Apply 方法，当卡牌效果被触发时，Apply 方法会被调用。具体的卡牌效果类都继承自 CardEffect 类，并覆盖 Apply 方法，实现特殊的卡牌效果逻辑。

（8）NPC 控制器

由于游戏是一个单机游戏，游戏中的对手均为电脑控制的角色，因此需要 NPC（Non-Player Character，非玩家角色）控制器来控制角色。NPC 控制器主要对发生在角色身上的各种事件做出响应，如停留在地产时需要响应购买和升级的事件，停留在彩票站时需要响应购买彩票的事件，经过银行时需要响应存取款的事件。同时在轮到 NPC 控制器所控制的角色回合时，NPC 控制器需要选择合适的卡牌来使用。选择并使用卡牌的策略、流程大致为：（1）从卡牌列表中选出可以主动使用的卡牌，如果没有可以主动使用的卡牌则结束选择卡牌流程；（2）随机选出一张卡牌；（3）根据卡牌的效果、目标类型、范围类型，寻找出卡牌的使用目标，若选不出合适的目标，则结束选择卡牌流程；（4）对目标使用卡牌。

根据以上分析，可以作出 NPC 控制器的 UML 类图以及 NPC 控制器使用卡牌逻辑的 UML 活动图，分别如图 4.12 和 4.13 所示。

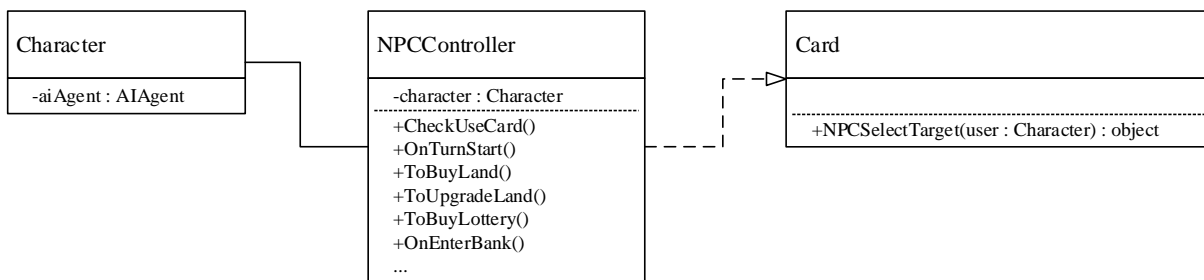


图 4.12 NPC 控制器 UML 类图

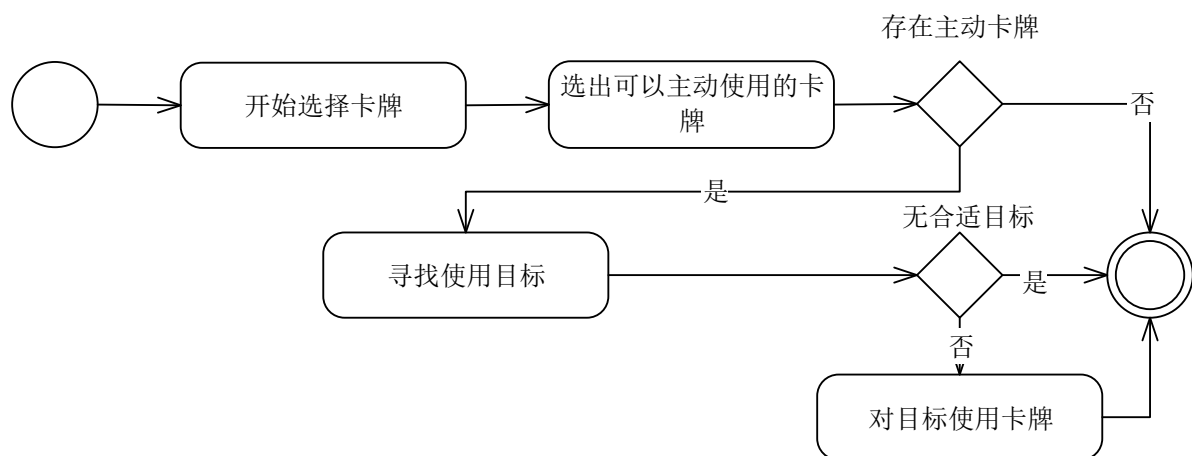


图 4.13 NPC 控制器使用卡牌策略

4.2.2 小游戏设计

(1) 激情赛艇

根据需求分析，赛艇小游戏中有以下对象：

- 赛艇游戏核心（BoatGame 类）。负责加载和生成赛艇游戏中的赛艇、金币、汽油对象，记录游戏时间，记录所获得的积分，显示游戏图形用户界面（GUI）。
- 赛艇控制器（BoatController 类）。负责配置赛艇的行动区域、最大速度、拿取汽油时的加速度，接收玩家的操作输入，控制赛艇移动。
- 金币（BoatPointObject 类）。负责配置拿取金币时获得的积分，与赛艇接触时通知 BoatGame 类累计积分。
- 汽油（BoatSpeedUp 类）。与赛艇接触时让赛艇加速。

根据以上分析，可作出赛艇小游戏的 UML 类图，如图 4.14 所示。

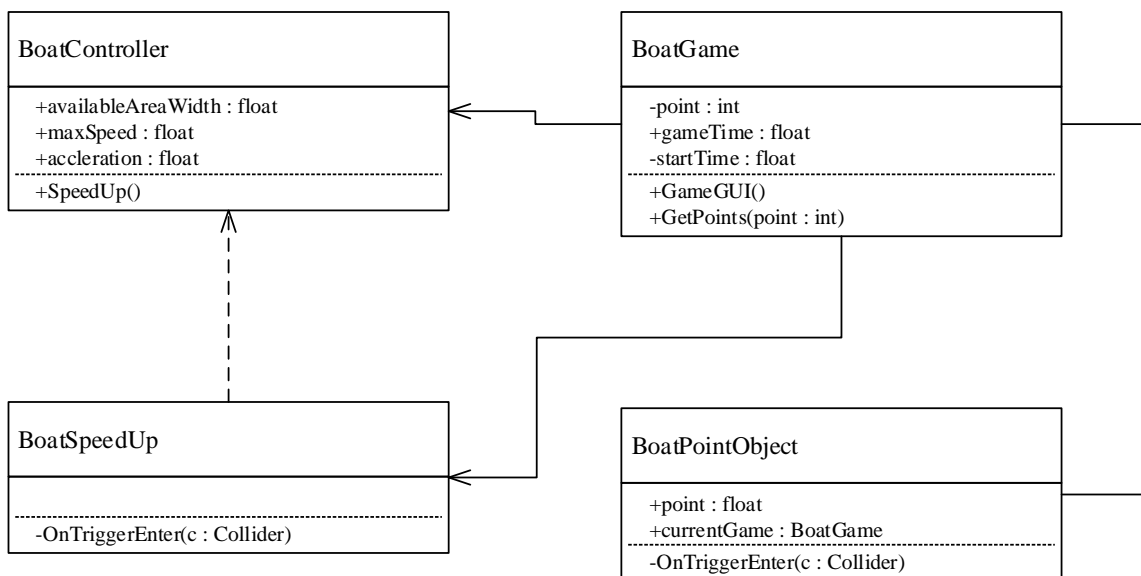


图 4.14 激情赛艇小游戏 UML 类图

(2) 宝箱迷踪

根据需求分析，宝箱迷踪小游戏的 UML 类图如图 4.15 所示。

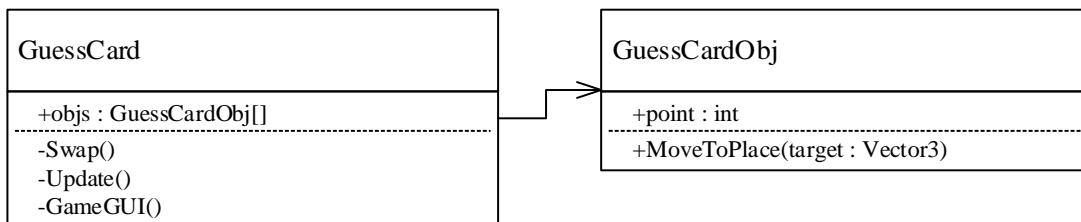


图 4.15 宝箱迷踪 UML 类图

宝箱迷踪小游戏中有以下对象：

- 宝箱游戏核心（**GuessCard** 类）。负责移动箱子、接收玩家的选择箱子操作、显示 GUI。
- 箱子中的点券（**GuessCardObj** 类）。配置猜中时获得的积分，执行箱子移动逻辑。

（3）胡里炮台

根据需求分析，胡里炮台小游戏中有以下对象：

- 炮台游戏核心（**CannonGame** 类）。负责配置游戏时间长度、船只出现的速率，动态生成船只、计时、计分、显示 GUI。
- 炮台（**Cannon** 类）。负责配置各个炮台的颜色、开炮间隔时间，接收玩家点击输入，播放开炮动画，计算开炮结果并通知炮台游戏核心计分。
- 船只（**CannonBoat** 类）。负责配置船只的颜色、移动速度和击中时获得的积分。

胡里炮台小游戏的 UML 类图如图 4.16 所示。

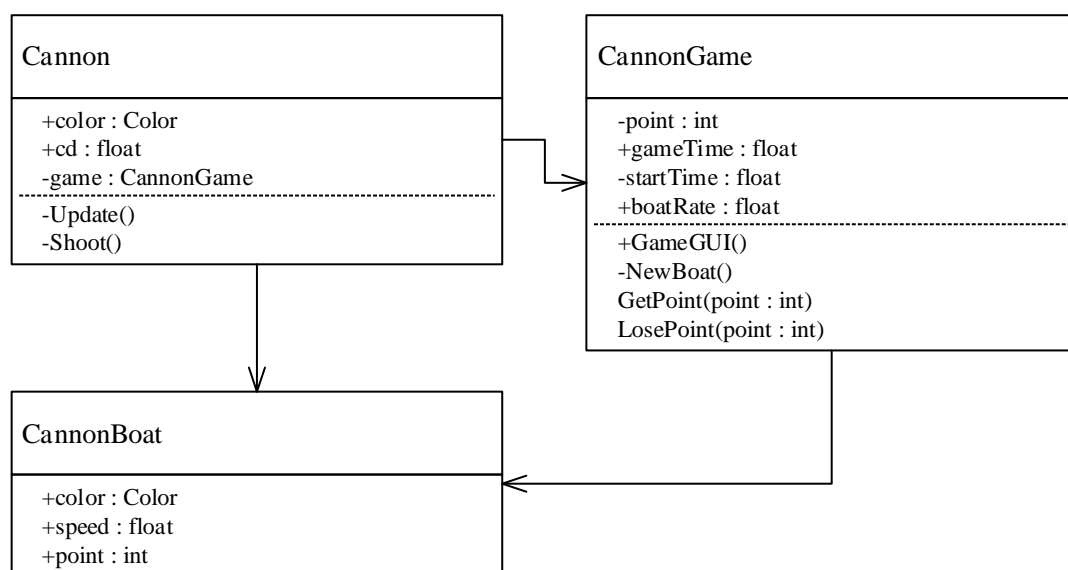


图 4.16 胡里炮台小游戏 UML 类图

（4）天赐富贵

根据需求分析，天赐富贵小游戏中有以下对象：

- 天赐富贵小游戏核心（**CatchTreasure** 类）。负责配置游戏各项参数，动态生成气泡，计时，计分，显示 GUI。
- 气泡（**FallingTreasure** 类）。可配置气泡的类型、降落速度，被针刺破时根据类型将结果通知 **CatchTreasure** 类。气泡的类型有：获得积分、积分按比例变化、结束游戏、加速、减速。

- 接收器（TreasureCatcher 类）。即上文中叙述的手和针，用于接收落下的气泡。当接触到气泡的时候，触发气泡的效果。

天赐富贵小游戏的 UML 类图如图 4.17 所示。

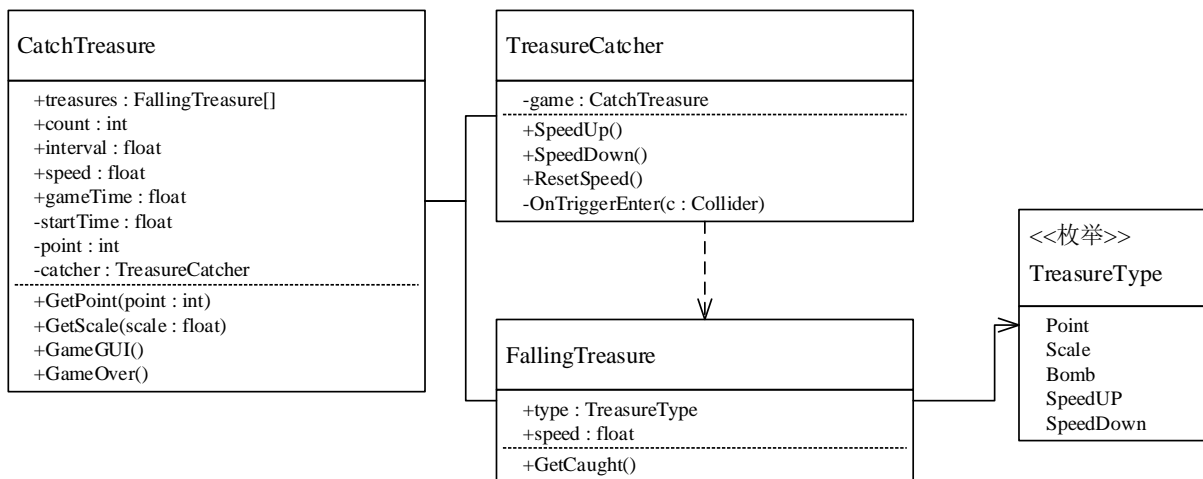


图 4.17 天赐富贵小游戏 UML 类图

(5) 勇闯迷宫

根据需求分析，勇闯迷宫小游戏中有以下对象：

- 勇闯迷宫小游戏核心（GuessWay 类）。负责初始化迷宫路径和路径终点的奖励，显示 GUI。
- 竖方向路径（Way 类）。一条竖方向路径连接着若干个横方向路径。
- 横方向路径（HorizontalWay 类）。每条横方向路径的两端都分别连接着竖方向路径。

勇闯迷宫小游戏的 UML 类图如图 4.18 所示。

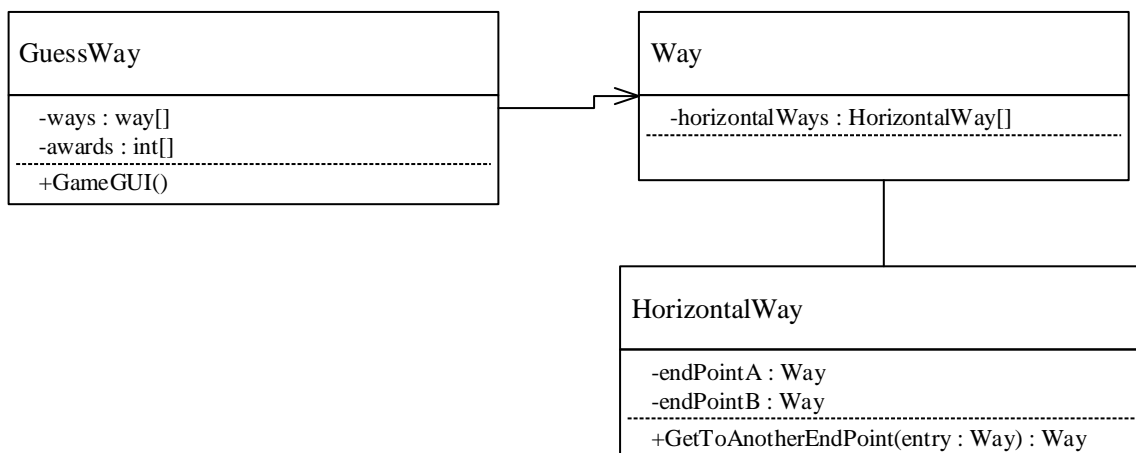


图 4.18 勇闯迷宫小游戏 UML 类图

迷宫的生成算法如下：

- (1) 生成 4 条竖方向路径 (**HorizontalWay**)，并在每条路径的两端分别设置起点和终点奖励。
- (2) 对于相邻的两条竖方向路径，在其之间随机生成 2 至 5 条横方向路径，使每条横方向路径的两端分别与两条竖方向路径相连接。

4.2.3 成就系统设计

由成就系统的设计部分可知，成就达成的条件均为游戏内的数值的比较，因此成就系统是建立在游戏的数据统计上的。

数据统计类 (**DataStatistics** 类) 负责统计游戏内的各项数据，如获得点券的数量、使用卡牌的次数、收取租金的数量、购买土地的数量、关卡通关的次数等等。数据的统计分为两类，一类为某一局游戏中的数据统计，另一类为对于某个玩家的整体数据统计，即成就描述中的“生涯中”。这两类数据分别记录在两个字典 (**Dictionary** 类) 中，字典的键为数据类型与参数，字典的值即为数据统计的值，如数据统计项“关卡 1-1 通关的次数”的数据类型为“关卡通关的次数”，参数为“1-1”，具体统计到的关卡 1-1 的通关次数即为数据统计的值。

成就类 (**Achievement** 类) 包含以下成员：

- (1) 成就 ID。成就的标识符。
- (2) 成就系列 ID (**SeriesId**)。成就系列，表示成就达成的条件类型相同，但是数值不同的一系列成就。如“关卡 1-1 通关 1 次”和“关卡 1-1 通关 2 次”为同一系列的成就。
- (3) 成就名称 (**Name**)。
- (4) 成就描述 (**Description**)。成就达成条件的描述。
- (5) 获得积分 (**Score**)。成就在达成后能够获取的积分数。
- (6) 成就达成条件 (**Requirements**)。需要满足所有达成条件后，成就才能够达成。成就条件类 (**AchievementRequirement** 类) 的内容为某个数据统计项与设定值的大小比较。
- (7) 成就标签 (**Tag**)。只要有发生了特定的游戏事件时，才会检查某一类成就是否满足达成条件，因此成就按标签来进行分类。

成就系统类 (**AchievementSystem** 类) 负责成就的管理，主要功能为将所有成就按标签分类排序、按系列分类排序、按标签检查成就是否满足达成条件、处理成就达成后玩家数据的记录。

成就监视器（AchievementInspector 类）负责在达成成就时，将成就的内容以及获得的积分显示在 GUI 当中，它包含一个成就缓存队列，当同时完成多个成就时，成就会被缓存在队列之中，之后依次在 GUI 中显示成就的内容和获得的积分。

根据以上分析，可以作出成就系统的 UML 类图，如图 4.19 所示。

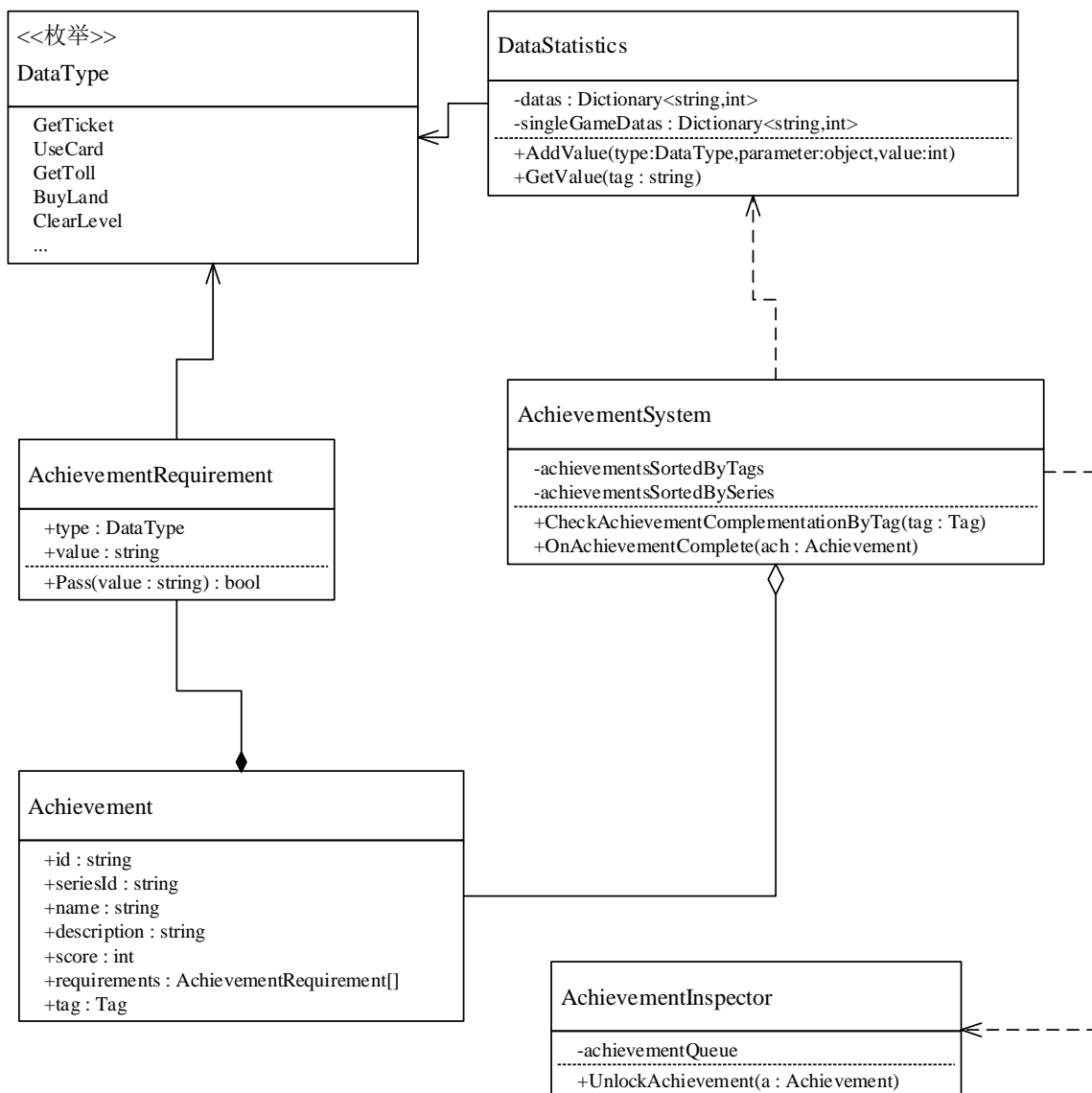


图 4.19 成就系统 UML 类图

4.3 游戏数据库设计

4.3.1 玩家数据

根据需求分析中的数据建模，游戏中需要记录玩家的一些数据，其数据结构通过物

理数据模型图表示如图 4.20 所示：

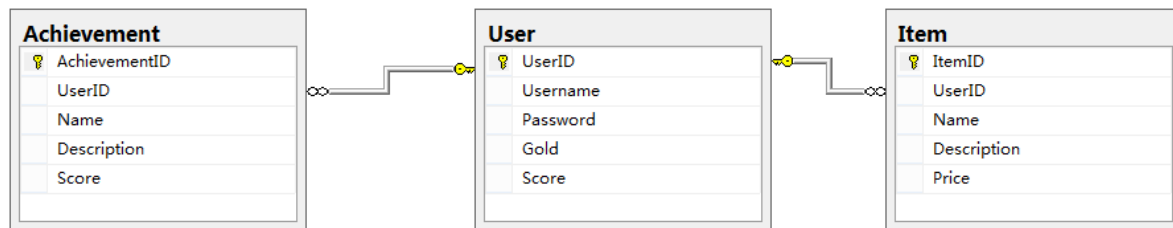


图 4.20 玩家物理数据模型图

User、Item、Achievement 三个表分别对应“用户”、“道具”、“成就”三个实体，其属性的详细设计分别如表 4.1、表 4.2 和表 4.3 所示：

表 4.1 User 实体属性详细设计

User			
列名	类型	约束	备注
UserID	int	NOT NULL, PRIMARY KEY	用户 ID（标识列）
Username	nchar(20)	NOT NULL	用户名
Password	int	NOT NULL	密码
Gold	int	NOT NULL	元宝数
Score	int	NOT NULL	积分

表 4.2 Item 实体属性详细设计

Item			
列名	类型	约束	备注
ItemID	int	NOT NULL, PRIMARY KEY	道具 ID（标识列）
UserID	int	NOT NULL, FOREIGN KEY	用户 ID
Name	nchar(10)	NOT NULL	道具名称
Description	nvarchar(100)	NOT NULL	道具描述
Price	int	NOT NULL	道具价格

表 4.3 Achievement 实体属性详细设计

Achievement			
列名	类型	约束	备注
AchievementID	int	NOT NULL, PRIMARY KEY	成就 ID（标识列）

UserID	int	NOT NULL, FOREIGN KEY	用户 ID
Name	nchar(10)	NOT NULL	成就名称
Description	nvarchar(100)	NOT NULL	成就描述
Score	int	NOT NULL	成就积分

4.3.2 新闻公告数据

新闻公告数据结构通过物理数据模型图表示如图 4.21 所示：

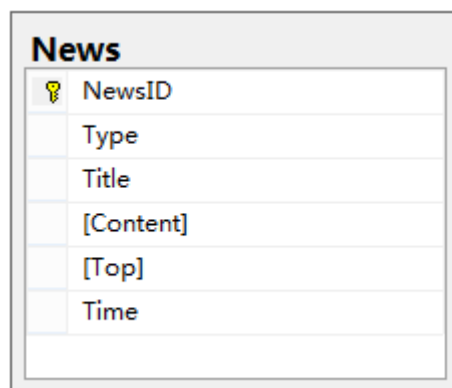


图 4.21 新闻公告物理数据模型图

新闻公告实体的详细设计如下：

表 4.4 News 实体属性详细设计

News			
列名	类型	约束	备注
NewsID	int	NOT NULL, PRIMARY KEY	新闻公告 ID（标识列）
Type	smallint	NOT NULL	新闻公告类型
Title	nvarchar(50)	NOT NULL	标题
Content	nvarchar(MAX)	NOT NULL	内容
Top	bit	NOT NULL	置顶
Time	Datetime	NOT NULL	发布时间

4.4 本章小结

本章根据游戏的需求分析，先分析了游戏的总体软件体系结构，然后将游戏客户端的功能按模块划分并对每个模块进行了详细的设计和建模，分析了每个模块的结构和功能，使得程序编码工作能够清晰、有序地进行，同时编写出的程序拥有较高的健壮性。最后设计了游戏的数据结构，确定了游戏的详细内容，为实际的开发做好了充分的准备。

第五章 游戏实现

需求分析和游戏设计已经为游戏的具体实现做好了充分的准备。本章描述游戏的具体实现过程，包含了使用 Unity3D 引擎进行实际游戏开发时的具体步骤。

游戏开发是一个复杂的过程，要把开发过程中的每一个详细的步骤、细节都记录下来是不现实的，因此本节主要叙述使用 Unity3D 进行游戏开发的过程中的比较常见、重要、关键的步骤和技巧。

5.1 资源导入与管理

游戏资源指游戏中使用到的模型、图片、文字等组成游戏的元素。Unity3D 中资源的管理是非常简便的，只要将资源文件放置在游戏项目的 Assets 目录下，资源文件即被 Unity3D 自动识别并管理。资源的导入流程如图 5.1 所示。

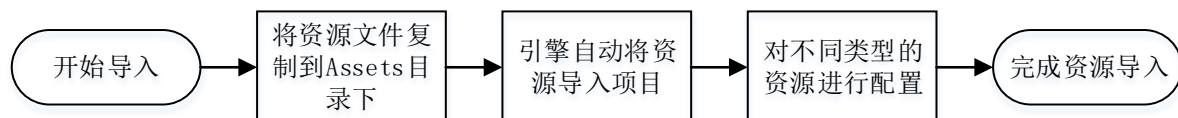


图 5.1 资源导入流程

资源的配置过程如下。

(1) 建筑、角色等模型、贴图及动画



图 5.2 模型动画配置

作为一个 3D 游戏，组成游戏场景画面的最重要的元素无疑是模型了。3D 模型由项目组中的美术工程师使用 3D 建模工具制作，并将制作好的模型保存成模型文件和贴图（或称为纹理，Texture）文件，本项目中使用的模型文件的格式为 FBX，贴图文件的格式为 JPG。需要注意的是，因为 Unity3D 对中文等编码的支持不太好，所以文件名最好不要使用中文等特殊字符，而是使用英文、数字和下划线的组合。另外，贴图尺寸的长宽分别都应为 2 的整数次幂，这样有利于贴图的压缩。

将模型文件和对应的贴图文件放置到 Assets 目录下后，Unity3D 会自动识别模型文件中的模型信息和动画信息。保存在模型文件中的动画信息是一个多个动作连续的动画，因此在导入 Unity3D 后，需要在 Unity3D 的 Project 面板中选中模型文件，在 Inspector 面板中选中 Animations 标签，将动画分割成多个动画片断（Clip）。如图 5.2 所示，角色鹭鹭模型的动画被分割成了空闲（Idle）、行走（Walk）、欢呼（Cheer）、失败（Fail）、使用技能（Skill）五个动画片断。

（2） 图片资源

除了用于模型贴图的图片之外，还有用于图形用户界面（GUI）等 2D 对象的图片，这些图片需要在导入设定（Import Settings）中设置其纹理类型（Texture Type），纹理类型有：Texture、GUI、Sprite、Advanced 等。Texture 用于模型的贴图，GUI 用于 Unity3D 的旧图形用户界面系统，Sprite 则用于 Unity3D 的新 2D 系统，Advanced 则可以自定义图片的各项参数。用于 GUI 的图片一般不进行压缩，因为压缩之后在游戏中会变得模糊，影响显示效果，因此 Format 参数选择真彩色（Truecolor），如图 5.3 所示。

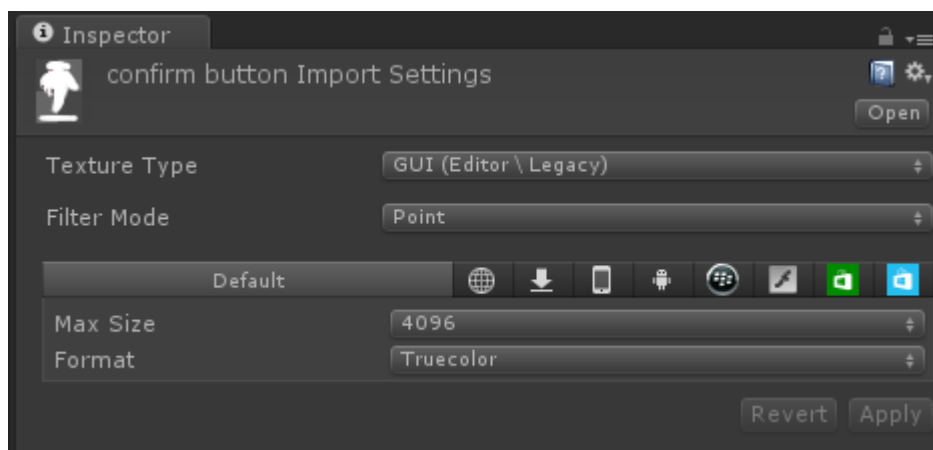


图 5.3 图片导入设置

（3） 文字资源

游戏中，界面、角色对话等位置中都会出现文字，这些文字的内容通常不写在代码文件中，也不在 Inspector 面板中设置，原因主要是中文等文字在某些平台中因为编码问题显示成乱码，同时不便于管理和修改，特别是需要开发多语言版本的时候。因此，文

字内容主要保存在文本文件当中，在游戏运行时再加载文本文件中的文字内容。比较常用的文件格式为 XML。

5.2 使用 XML

XML 是 Extensible Markup Language 的缩写，即可扩展标记语言。它是一种用来存储数据的语言，可以存储结构化的数据，优点是极其简单，可以很轻松地使用它来存储各种数据。

游戏项目中，XML 主要被使用在以下模块：

- 存储剧情对话等文字资源。
- 成就信息。存储成就的达成条件、获得积分等信息。
- 卡牌、道具信息。存储卡牌和道具的各种参数、效果等信息。
- 寺庙求签中签的参数、效果等信息。
- 大转盘中各区域的参数、效果等信息。
- 玩家数据。包括持有的元宝数、道具、已获得的成就等信息。虽然在游戏服务器中也保存着玩家数据，但只有当玩家重新安装游戏或更换设备登录时才会再从游戏服务器中获取玩家数据，一般情况下玩家数据是以 XML 形式保存在玩家的设备中的。

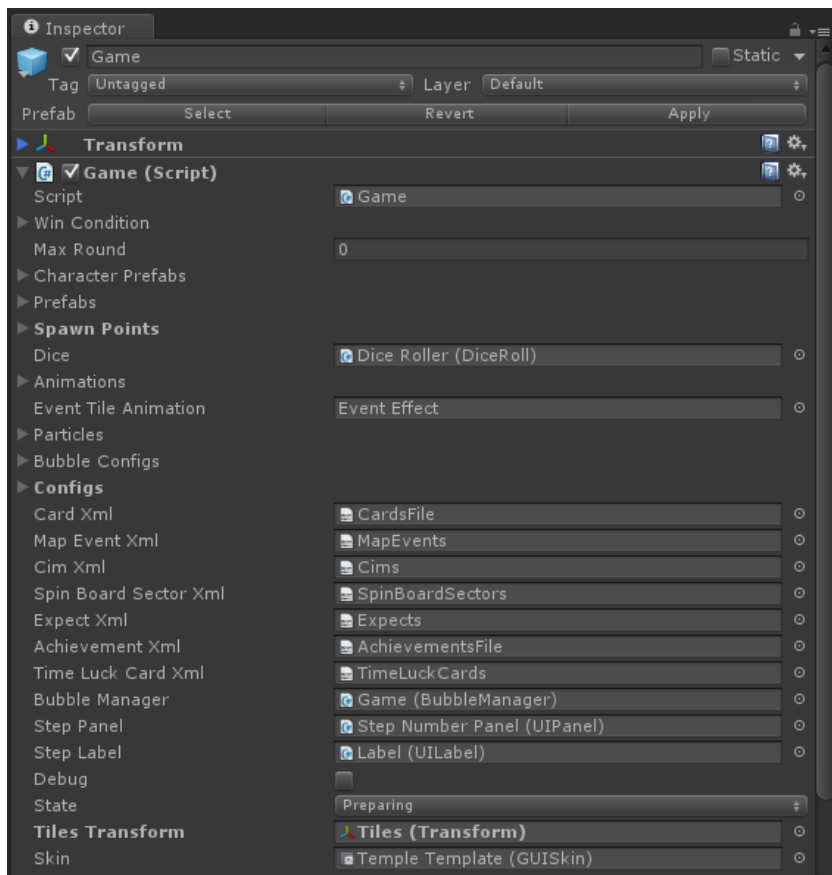


图 5.4 Game 类中配置 XML 文件

XML 在游戏项目中的应用主要有以下两个方面：

(1) 使用.NET 的 XML 序列化功能

XML 序列化功能是.NET 框架提供的一个很便利的功能，它可以将程序中的对象转换为 XML 字符串（即序列化），同时也可以再将 XML 字符串转换为程序中的对象（即反序列化）。通过这个功能，可以将游戏中保存在对象中的数据序列化成 XML 字符串，并写入文本文件中，在需要的时候，再将从文件中读取 XML 字符串，通过反序列化再转换为数据对象。

提供 XML 序列化功能的类为 `System.Xml.Serialization` 命名空间中的 `XmlSerializer` 类，`Serialize` 方法的功能为序列化，`Deserialize` 方法的功能为反序列化。

(2) 在 Unity3D 中读取 XML 文件

在 Unity3D 程序中读取 XML 文件的方法为，首先在脚本的类中声明一个访问级别为 `public` 的 `TextAsset` 类成员，然后在 `Inspector` 面板中将该 `TextAsset` 成员的值设为需要读取的 XML 文件，在游戏启动后即可通过访问该 `TextAsset` 类成员的 `text` 属性获取 XML 文件的内容。图 5.4 所示的面板为 `Game` 类的 `Inspector` 面板，`Game` 类中通过读取 XML 文件加载卡牌信息、新闻事件等游戏信息。

使用 XML 文件存储数据的总体流程如图 5.5 所示。

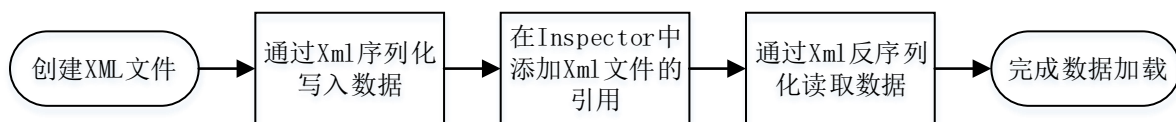


图 5.5 XML 文件使用流程

5.3 设定角色动画

在导入模型的时候已经将动画剪切成片断，要使动画根据游戏逻辑进行播放，还需要进一步设定。

Unity3D 中的动画系统为一个基于有限状态机的状态转换系统，每一个动画状态播放一个动画片段，当满足设定的转换条件时，会由一个动画状态转换到另一个动画状态，并开始播放该动画状态对应的动画片段。在动画系统中还可设定若干个参数，状态转换的条件即为对参数当前的值的判断。在游戏运行过程中，可以通过脚本对参数的值进行改变，从而满足不同的状态转换条件，触发动画状态转换。

本游戏项目中的每个角色主要使用三个动画片段：空闲（Idle）、行走（Walk）和欢呼（Cheer），因此需要创建三个动画状态，分别播放这三个动画片段，同时经过分析可

以知道有三个状态转换：空闲转换至行走、行走转换至空闲、空闲转换至欢呼。设定的参数有两个：**Move** 和 **Victory**，均为布尔型，分别表示角色是否在行走、角色是否已经获得胜利。空闲转换至行走的条件为 **Move** 的值为真，行走转换至空闲的条件为 **Move** 的值为假，空闲转换至欢呼的条件为 **Victory** 的值为真。由此分析，角色的动画控制器（Animator）设定如图 5.6 所示。图中空闲状态为默认状态。

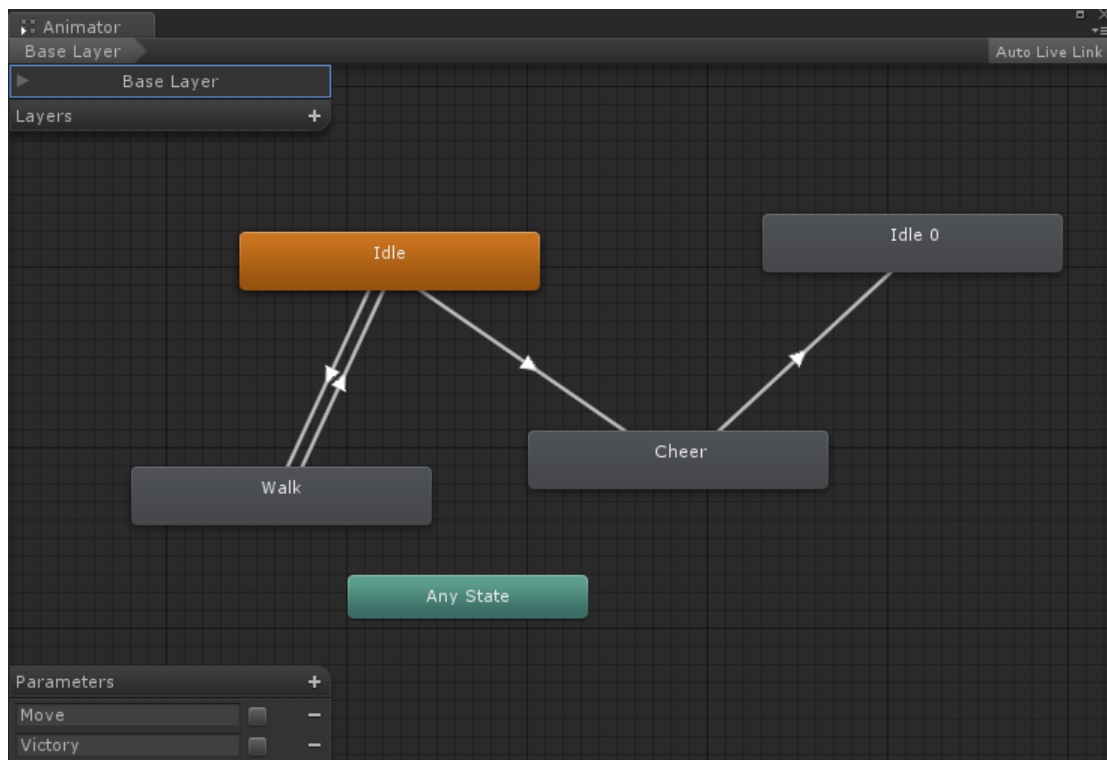


图 5.6 角色动画控制器设定

5.4 构建测试场景

测试场景是用于构建游戏对象、测试游戏主体各项功能的场景，它不会出现在正式发布的游戏中。测试场景的优点是地图很小，可以快速地进行游戏主体各功能的单元测试。下面叙述构建测试场景的过程。

（1）构建游戏对象

游戏主体中的对象主要有角色、道路地块、地产地块和游戏核心。首先在 Unity3D 的 Project 面板中分别将角色、道路地块、地产对应的模型拖入场景中，创建基础对象。角色的基础对象需要配置动画控制器，使其能够进行上文中叙述的动画状态转换，同时需要添加 **Character** 类的脚本，道路地块和地产地块的基础对象分别需要添加 **Tile** 类的脚本和 **BuildingLand** 类的脚本。由于在使用卡牌或道具时，角色、道路地块和地产地块都有可能被选为目标，因此需要给对象分别加上碰撞体（Collider），以接收玩家在点击操作。新建一个空对象并添加 **Game** 类的脚本即可创建游戏核心对象。

最后将创建好的游戏对象保存为 Prefab，使其在其他场景中也能够使用。

(2) 构建简易地图并配置游戏参数

通过复制操作，创建多个道路地块和地产地块，并将其排列整齐，使道路构成封闭的环，然后在 Inspector 中配置 Tile 类的 Links 和 BuildingLand 属性，将相邻的地块互相连接起来。对其中几个道路地块，将 Tile 类替换成由 Tile 派生的各个事件地块的派生类，并在相应的地块上添加适当的标记，方便辨认事件地块的具体类型。

Game 类中需要配置多项参数，如需要配置角色对象的 Prefab，使游戏核心能够初始化角色，还需要配置当前场景中的地块，使游戏核心可以对各地块进行访问。

测试场景的总体构建流程如图 5.7 所示。

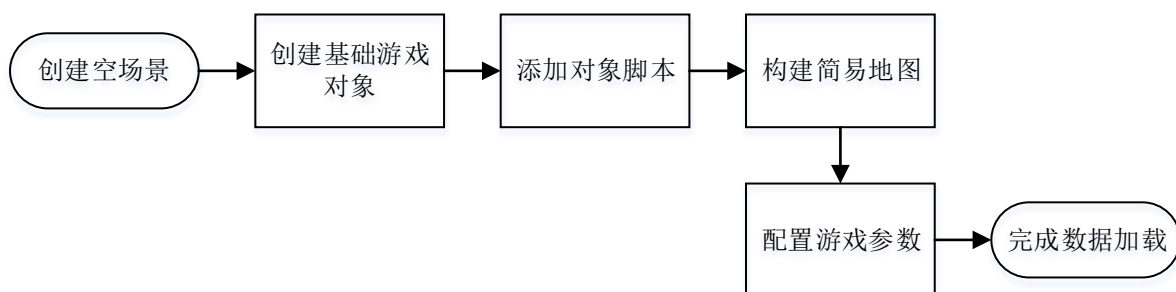


图 5.7 测试场景构建流程

在测试场景中对游戏主体各项功能测试完毕后，即可将对象移至游戏的正式场景中，正式场景的构建主要由美术工程师完成。

5.5 实现图形用户界面（GUI）

游戏中的图形用户界面的实现过程如图 5.8 所示。

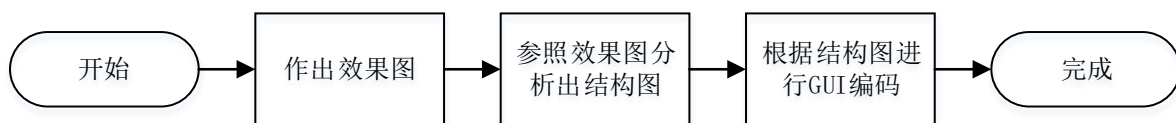


图 5.8 图形用户界面实现流程

首先由团队中的美术工程师用绘图软件作出界面的理想效果图，程序员再根据效果图分析界面中的元素、结构并作出结构图，然后根据结构图来在游戏引擎中进行 GUI 的编码、构建工作。

Unity3D 中主要使用两种 GUI 系统：Unity3D 自带的 GUI 系统(Unity GUI)和 NGUI 插件。Unity GUI 是 Unity3D 官方的 GUI 系统，它有非常多的缺点：使用代码编写 GUI、无可视化界面、运行效率较低，因此出现了第三方开发的 Unity3D GUI 插件——NGUI。NGUI 使 GUI 的编辑可视化，运行效率也比 Unity GUI 高，是 GUI 系统的一个较好的选

择。

《悠游厦门》游戏项目开发初期使用的是 Unity GUI，后发现其运行效率较低，使游戏主体部分的画面不够流畅，因此项目开发后期将一部分的 GUI 开发改为使用 NGUI 系统。本节分别对两种 GUI 系统的使用方法进行简单叙述。

（1） Unity GUI

Unity GUI 是使用脚本与 Inspector 配合实现 GUI 编辑的系统，脚本中使用代码定义 GUI 元素，Inspector 面板中配置 GUI 元素的样式。GUI 元素主要有以下几种：

- 标签（Label）。用于显示文字。
- 按钮（Button）。用于显示按钮，接收用户的点击操作输入。
- 文本输入区域（TextArea）。用于接收用户键盘输入的文字。
- 选择（Toggle）。用于设置布尔值。
- 窗口（Window）。在 GUI 中显示逻辑上的窗口。
- 区域布局（Area）。定义一个逻辑上的 GUI 区域，使 GUI 元素可以在区域中排列。
- 横向布局（Horizontal）、纵向布局（Vertical）。定义一个逻辑上的 GUI 区域，使 GUI 元素在区域内横向排列或纵向排列。

每个 GUI 元素都可以定义 GUI 样式，如背景图、与相邻的 GUI 元素的间距、对齐方式、字体字号等。

下面以游戏的主菜单为例叙述 Unity GUI 在游戏项目中的使用过程。

图 5.9 为主菜单的界面结构图，它是 GUI 编码、构建工作的依据。游戏的主菜单为一个窗口，位于游戏画面中央，窗口的内部分为玩家数据区域、左侧按钮区域和右侧按钮区域。代码中首先通过 GUI.Window 方法定义窗口，窗口的坐标根据游戏画面的高和宽（Screen.height 与 Screen.width）计算出中央的位置，因此在任何大小的屏幕中都可以保持居中。在窗口中，使用 GUILayout.BeginArea 方法和 GUILayout.EndArea 方法分别定义三个 GUI 区域。在玩家数据区域中，使用 GUILayout.BeginHorizontal 方法和 GUILayout.EndHorizontal 方法定义横向区域，并在其中使用 GUILayout.Label 方法显示玩家的账号名、积分和元宝数；在左侧按钮区域中，使用 GUILayout.Button 方法定义四个按钮，同时使用 GUILayout.FlexibleSpace 方法在四个按钮之间插入可以自动调节的间隔；在右侧按钮区域中，需要定义两个横向区域，第一个横向区域中定义剧情模式和寻宝模式按钮，第二个横向区域中定义自由模式和继续游戏按钮。最后再在窗口右上角定义关闭按钮。

GUI 元素定义完毕后，在主菜单的 Inspector 面板配置各元素的样式。需要配置的样

式主要有：窗口以及各按钮的背景图，玩家数据部分的文字字体、字号和对齐方式。

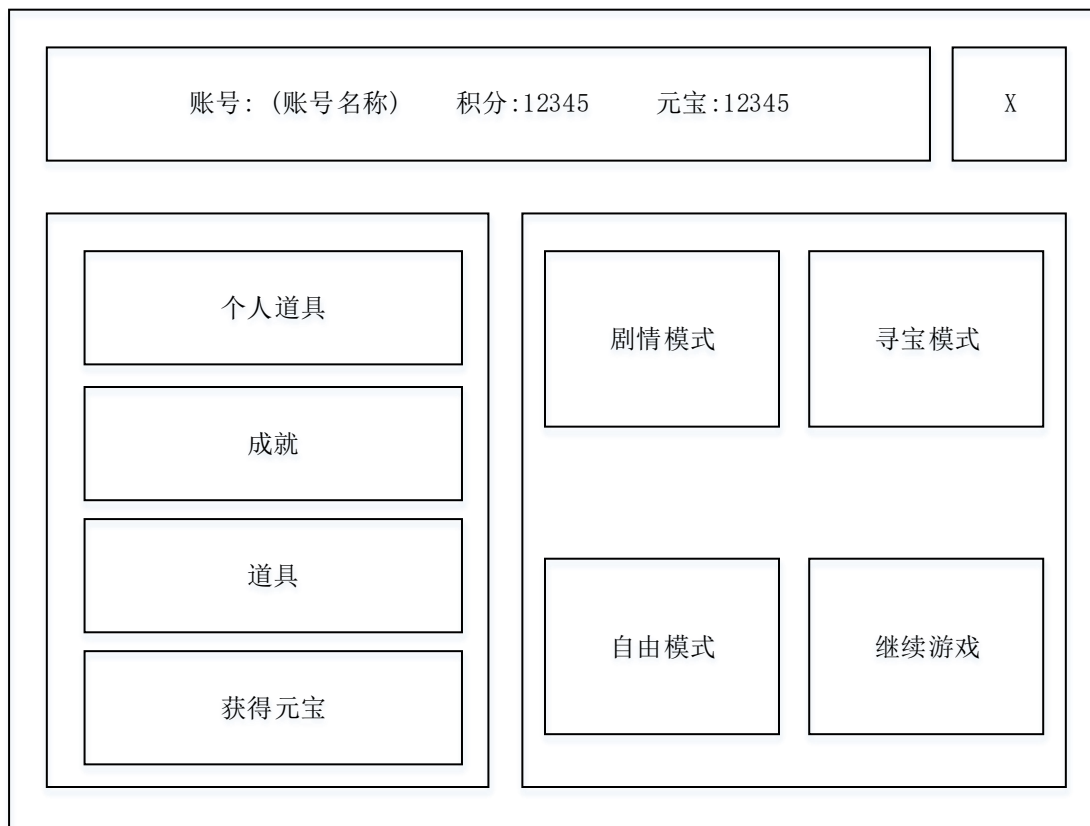


图 5.9 主菜单界面结构

(2) NGUI

NGUI 是 Next Generation User Interface 的缩写，即新一代用户界面的意思，表示比 Unity GUI 更新的 GUI 系统。NGUI 的实现原理是将图形、文字等界面元素绘制在只有一个平面的 3D 模型上，从而可以实现在 Unity3D 的编辑器中可视化地编辑。NGUI 提供了多种现成的 GUI 元素，如标签（Label）、按钮（Button）、可拖动窗口（Draggable Windows）、滚动视图（Scroll View）等。通过这些 GUI 元素可以轻松构建出功能强大的而美观的游戏界面。

NGUI 使用一种与 Unity GUI 不同的布局方式——Anchor 方式。游戏界面中有 9 个 Anchor，分别位于左上角、上边缘中央、右上角、右边缘中央、右下角、下边缘中央、左下角、左边缘中央、画面正中央。所有的 GUI 元素的位置均为与某个 Anchor 的相对位置，如画面中左上角的角色头像、持有的现金和存款数与左上角的 Anchor 的相对位置不变，左下角的商城等按钮与左下角的 Anchor 的相对位置不变，画面中央的 Go 按钮与画面正中央的 Anchor 相对位置不变。这样的好处在于，不管游戏画面的大小如何，各个 GUI 元素的位置都可以得到自动调整。

5.6 实现客户端与服务器之间的通信

游戏的服务器提供基于 SSH 框架的 Web 服务，客户端通过 POST 方法向服务器发送 HTTP 请求，服务器处理请求后再给客户端发送响应。

(1) 发送 HTTP 请求并接收响应

Unity3D 提供了一个非常方便的 HTTP 接口——WWW 类，通过 WWW 类可以很轻松地构造并发送 HTTP 请求。游戏项目中发送 HTTP 请求并接收响应过程大致如下：创建 WWWForm 类实例，将请求参数添加到 WWWForm 类的实例中，使用 WWWForm 类的实例和预先设定的服务器 URL 创建 WWW 类实例，创建好 WWW 类实例的同时，HTTP 请求会自动发出，之后每个游戏循环中检查 WWW 类实例的 isDone 属性是否为真，若为真表示收到了服务器的响应，检查是否有出错信息，若无出错信息，则取出响应数据并返回请求成功。流程图如图 5.10 所示。

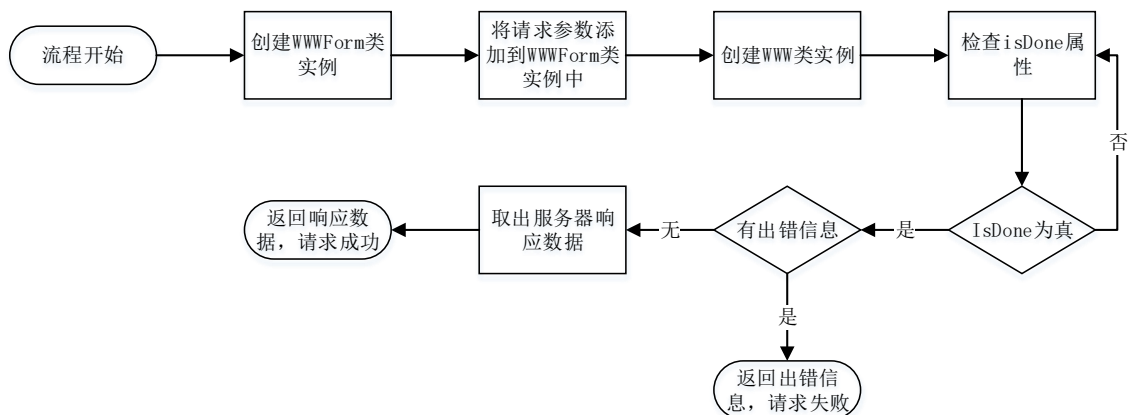


图 5.10 主菜单界面结构处理响应数据

服务器返回的响应数据为 JSON 格式的。JSON 比 XML 更轻便，存储同样的数据，使用 JSON 比使用 XML 花费的空间更小，因此游戏项目中使用 JSON 作为服务器与客户端之间的数据传输格式。

在 Unity3D 中可以使用很多第三方的程序库来处理 JSON，如 LitJSON、Newtonsoft.Json 等。游戏项目中选择使用 SimpleJSON，因为它非常小巧，源代码仅仅一千行，同时又能够满足游戏项目中 JSON 的使用需求。

5.7 本章小结

游戏实现是整个游戏项目中最关键的、耗时最长的阶段，这个阶段的工作使游戏从无变为有、从计划变为实际、从设计变为实现。

本章对使用 Unity3D 引擎开发游戏的主要步骤和技巧进行了详细的描述，使游戏开发得到完整的实现。

第六章 游戏测试与发布

游戏实现完成之后，还需要进行游戏测试与游戏发布。游戏测试的目的是找出并修正游戏中的程序方面的错误或者图形方面的错误，验证游戏的每个功能都能够正常运行，给游戏玩家最好的游戏体验。最后还需要发布游戏，只有将游戏发布到了正确的渠道上，玩家才可以下载、安装并进行游戏。

6.1 游戏测试

游戏测试的内容主要为检查游戏的各项功能、画面表现是否正常，主要包括单元测试和实机测试。单元测试为针对游戏中的每项功能，单独测试其是否正常，主要在 Unity3D 引擎的 Editor 环境中进行；实机测试为在实际的移动设备上运行游戏，完整地进行游戏的每个关卡，对实际运行效果进行测试评估。

6.1.1 游戏测试环境

游戏的单元测试主要在开发用的个人电脑上使用 Unity3D 引擎直接进行，实机测试则在多个不同平台的移动设备上运行，游戏测试的详细软硬件环境如表 6.1 所示：

表 6.1 软硬件环境表

设备名称	硬件环境	软件环境
个人电脑	CPU: Core i7-2600K 显示核心: NVIDIA GeForce GTX 550 Ti RAM: 8GB	Windows 8 Unity3D 4.4
HTC Desire HD	CPU: 单核 1024MHz RAM: 768MB	Android 2.3.5
三星 G3508	CPU: 双核 1.2GHz RAM: 768MB	Android 4.1
iPad Mini	CPU: 双核 1GHz RAM: 512MB	iOS 6.0
HTC 8S	CPU: 双核 1GHz RAM: 512MB	Windows Phone 8

6.1.2 测试用例及过程

根据游戏的功能需求建模，游戏的测试分为以下几个部分进行。

（1） 游戏主体测试

游戏主体测试即测试功能需求中的“大富翁的游戏规则”，先用单元测试对各个功能、模块进行测试，再在实机测试中对游戏中的每个关卡的运行情况进行检查。游戏主体的主要部分——角色行动流程的测试用例如表 6.2 所示。

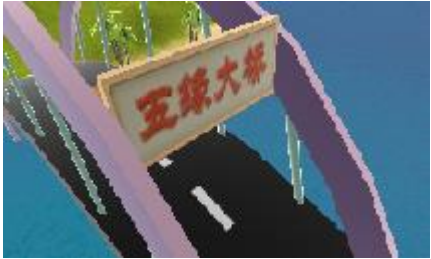


表 6.2 角色行动流程测试用例

用例编号	M1			
用例目的	测试角色行动流程			
子用例编号	操作步骤	预期结果	实际结果	状态
M101	在角色回合开始时，点击游戏画面中的“GO”按钮	出现博饼动画，角色根据骰子的数字向前行走	出现博饼动画，骰子在碗中随机散开，约 2 秒后停止，绿色骰子的数字为 5，角色向前行走 5 步并停留	通过
M102	利用调试工具使角色行走至空地并停留	出现是否购买的提示	出现提示“筭筭路，价格 700，是否买下此地”	通过
M103	利用调试工具使角色行走至其他角色拥有的土地	出现是否缴纳过路费的提示，并扣除现金	出现提示“鹭鹭支付了 1100 过路费”，现金由 58800 变为 57700	通过
M104	角色停留后，完成投资土地或缴纳过路费的操作	角色回合结束，切换到下一角色的回合	画面由角色鹭鹭切换到角色王老五，并开始角色王老五的回合	通过

（2） 游戏场景测试

游戏场景测试主要测试游戏场景中的地形、建筑等对象的模型、贴图、特效是否正常渲染。五缘湾场景的测试用例如表 6.3 所示。

表 6.3 五缘湾场景测试用例

用例编号	S1			
用例目的	测试五缘湾场景			
子用例编号	操作步骤	预期结果	实际结果	状态
S101	移动游戏画面到“五缘大桥”，通过手势操作旋转游戏画面，从各个角色检查“五缘大桥”的模型、贴图。	模型、贴图正常。	桥的模型、贴图正常，可看到“五缘大桥”四个字。 	通过
S102	移动游戏画面到“湿地公园”，通过手势操作旋转游戏画面，从各个角色检查“湿地公园”的模型、贴图、粒子。	模型、贴图、粒子正常。	公园的地面、水、雕像的模型、贴图正常，喷水的粒子效果正常。 	通过
S103	移动游戏画面到“高崎机场”，通过手势操作旋转游戏画面，从各个角色检查“高崎机场”的模型、贴图。	模型、贴图正常。	机场的航站楼、飞机模型和贴图正常。 	通过

(3) 博饼模块测试

博饼是厦门文化的重要体现，博饼功能也是厦门市旅游局对《悠游厦门》游戏提出的一个重要需求。博饼功能的测试主要是对博饼中投掷骰子后出现特定组合时获得的奖励进行测试，测试用例如表 6.4 所示。

表 6.4 博饼模块测试用例

用例编号	B1			
用例目的	测试博饼模块			
子用例编号	操作步骤	预期结果	实际结果	状态
B101	通过调试工具使投掷出的 6 个骰子中只有一个骰子的点数是 4	获得“一秀”奖励	游戏画面中出现“一秀”提示,并获得 10 点券的奖励。	通过
B102	通过调试工具使投掷出的 6 个骰子中正好有两个骰子的点数是 4	获得“二举”奖励	游戏画面中出现“二举”提示,并获得 500 现金的奖励。	通过
B106	通过调试工具使投掷出的 6 个骰子中正好有 4 个骰子的点数是 4	获得“状元”奖励	游戏画面中出现“状元”提示,播放烟花动画,并同时获得 10 幸运值、100 现金和 100 点券的奖励。	通过

(4) 游戏下载测试

游戏主要在应用平台上发布,同时也在游戏自己的网站上提供 Android 平台的安装包下载,因此需要测试此安装包下载的功能是否正常。游戏下载的测试用例如表 6.5 所示。

表 6.5 游戏下载测试用例

用例编号	D1			
用例目的	测试游戏 Android 安装包的下载功能			
子用例编号	操作步骤	预期结果	实际结果	状态
D101	在桌面电脑上打开浏览器,登录 http://yoyoxm.cn , 点击页面中的“免费下载”按钮	获得游戏安装包的下载	出现“yoyoxm.apk”的下载提示	通过
D102	在 HTC 手机上打开浏览器,登录 http://yoyoxm.cn , 点击页面中的“免费下载”按钮	获得游戏安装包的下载	出现“yoyoxm.apk”的下载提示	通过

(5) 积分兑换奖励测试

积分兑换奖励的功能是调动玩家参与游戏的积极性的一个重要功能，其测试用例如表 6.6 所示。

表 6.6 积分兑换奖励测试用例

用例编号	E1			
用例目的	测试使用积分兑换奖励的功能			
子用例编号	操作步骤	预期结果	实际结果	状态
E101	在游戏主界面中进入“积分排名”界面，点击“豪华大礼”标签，点击“率意揽胜”奖品	成功兑换“率意揽胜”奖品，并扣除积分	获得“率意揽胜”奖品的兑换码“684BA42E”，积分由 1000 变为 900。	通过
E102	在游戏主界面中进入“积分排名”界面，点击“豪华大礼”标签，点击“惬意午茶”奖品	成功兑换“惬意午茶”奖品，并扣除积分	获得“惬意午茶”奖品的兑换码“F23CE697”，积分由 9000 变为 700。	通过

6.1.3 测试结果分析

从对游戏的各项功能的单元测试和实机测试的结果中可以看到，各项功能的测试均已通过。在实机测试中，不同设备的具体表现如表 6.7 所示。

表 6.7 实机测试结果

测试设备	网络功能	图形功能	关卡测试
HTC Desire HD	网络通信正常，能够正确完成游戏事务	图形表现正常，游戏画面清晰，流畅度一般	所有关卡可以正常通关，关卡载入速度较慢
三星 G3508	网络通信正常，能够正确完成游戏事务	图形表现正常，游戏画面清晰，流畅度较高	所有关卡可以正常通关，关卡载入速度一般
iPad Mini	网络通信正常，能够正确完成游戏事务	图形表现正常，游戏画面清晰，流畅度非常高	所有关卡可以正常通关，关卡载入速度较快
HTC 8S	网络通信正常，能够正确完成游戏事务	图形表现正常，游戏画面清晰，流畅度较低	所有关卡可以正常通关，关卡载入速度一般

由此可见，游戏的各项功能都可以正常运行，并在各平台的设备上基本稳定运行，已满足开发需求，具备了发布的条件。

6.2 游戏发布

游戏制作完成后, 需要将游戏发布, 玩家才可以下载并安装。国内外有很多移动应用平台供游戏发布, 同时各平台还提供充值付费接口, 使用户可以使用平台提供的付费渠道, 购买游戏内的元宝。谷歌、微软、苹果分别为 Android、Windows Phone 和 iOS 系统开放了应用商店, 同时还有许多非官方的应用平台, 如国内的 91 平台、UC 九游平台、360 移动平台等。

下面结合《悠游厦门》游戏的发布情况, 叙述游戏的发布过程。

6.2.1 在中国移动动漫基地平台发布

中国移动动漫基地的平台主要为安卓应用的平台, 平台中的应用多为漫画在线阅读等动漫相关的应用。平台提供了一个非常便利的付费方式——使用手机话费付费, 只要手机中的 SIM 卡为中国移动的 SIM 卡, 在应用中只要点击一个按钮, 即可使用手机话费付费购买应用中的内容, 无需输入密码等繁琐的操作。在中国移动动漫基地平台发布主要包括以下两个步骤:

(1) 接入中国移动动漫基地能力平台 SDK

要使游戏应用能够使用中国移动动漫基地平台的付费功能, 首先需要接入动漫基地能力平台 SDK。SDK 为 Software Development Kit 的缩写, 即软件开发工具的意思, SDK 中提供了一些接口, 通过调用这些接口来实现付费功能的调用。

SDK 是一个面向安卓应用的 Java 程序库, 因此需要使用 Eclipse 来编写 Java 程序进行 SDK 接口的调用。Unity3D 中的脚本是使用 C#语言编写的, 因此需要使用 Java Native Interface(JNI)技术, 使 C#程序可以调用 Java 程序中的接口。Unity3D 提供了两个类来实现 JNI 的功能: AndroidJavaClass 类和 AndroidJavaObject 类, 这两个类分别对应 Java 语言中的类和对象。接入 SDK 的流程大致如下:

- a) 在 Eclipse 中创建一个新的 Android 项目, 在 Build Path 配置中添加对 SDK 的 jar 的库引用。
- b) 从 Unity3D 的安装目录中找到 classes.jar, 并在 Android 项目中对其添加引用。
- c) 创建一个继承自 UnityPlayerActivity 的类, 在 onCreate 方法中进行 SDK 的初始化, 编写 order 方法, 在方法中调用 SDK 的支付接口。由于 SDK 的支付功能本身带有 GUI, 因此需要使 SDK 的支付界面在 Unity 程序中的 UI 线程中运行, 否则程序会出错。使 SDK 的支付界面在 UI 线程中运行的方法为, 调用 UnityPlayer.currentActivity.runOnUiThread 方法, 将调用 SDK 支付接口的程序代码写在 Runnable 的接口实现中作为参数传进 runOnUiThread 方法中。《悠游厦门》游戏项目中创建的继承自 UnityPlayerActivity 的类为

waii.huanying.youyouxiamen.YoyoActivity, 其中 order 方法没有返回值, 并且有一个 int 类型的参数, 表示购买的商品的编号。

- d) 当 SDK 支付完成, 返回结果时, 需要将支付结果发送给 Unity 程序, 以在游戏中处理支付完成后的逻辑。将支付结果发送给 Unity 程序的方法为调用 UnityPlayer.UnitySendMessage 方法, 这个方法的作用为将一个字符串发送到游戏当前场景中的某个游戏对象 (GameObject) 的某个参数为 string 的方法中。UnitySendMessage 方法有三个参数, 分别表示要发送到的游戏对象名称、发送的目标方法名、要发送的字符串。当然在游戏的支付模块中要准备好这一接收消息的方法, 并在方法中写处理好支付结果的代码。
- e) 在 Unity3D 中调用支付接口。如上文所述, 需要使用 AndroidJavaClass 类和 AndroidJavaObject 类来实现调用, 具体的代码如图 6.1 所示。

```
public void CallJavaOrder(int goodIndex)
{
    var unityPlayer = new AndroidJavaClass("com.unity3d.player.UnityPlayer");
    var currentActivity = unityPlayer.GetStatic<AndroidJavaObject>("currentActivity");
    currentActivity.Call("order", goodIndex);
}
```

图 6.1 调用支付接口核心代码

图中的 currentActivity 对象即为 (3) 中创建的 YoyoActivity 类实例, 因此可以调用其成员方法 order, 并将商品编号作为参数传进去。

- f) 配置 AndroidManifest.xml。AndroidManifest.xml 为安卓应用的配置文件, Unity3D 会在编译打包 APK 的时候自动生成该配置文件, 但在接入 SDK 时, 需要添加 SDK 的一些配置, 如 Activity 声明、用户权限等, 因此需要手动制作该配置文件。制作该配置文件的一般方法为: 使用 Unity3D 将游戏编译成 APK, 此时在游戏项目目录下的 Temp\StagingArea 目录中会生成一个 AndroidManifest.xml, 将其复制到游戏项目目录下的 Assets\Plugins\Android 目录中, 并对其编辑, 在其基础上添加 SDK 的配置信息, 同时将默认启动的 Activity 名改为 waii.huanying.youyouxiamen.YoyoActivity。
- g) 将 YoyoActivity.java 导出为 jar 文件, 与 SDK 的 jar 文件以及 libs 目录、res 目录一同复制到游戏项目目录下的 Assets\Plugins\Android 目录中。

(2) 编译并打包 APK

由于动漫基地的 SDK 是为安卓应用设计的, 与 Unity3D 有细微的不兼容的地方, 即如果直接使用 Unity3D 编译游戏并打包成 APK, 游戏在启动时无法正确初始化 SDK, 以至于不能使用支付功能。经过长时间的研究发现, 在 Unity3D 的 Build Settings 中选择将游戏导出为 Google Android Project, 再由 Eclipse 打开导出的项目并进行编译、打包成

APK，即可解决这个问题。

6.2.2 在谷歌应用商店 Google Play 发布

谷歌公司已经退出中国市场，在谷歌应用商店中的应用是不能在中国大陆地区进行收费的，但中国大陆地区的用户仍然可以从谷歌应用商店中下载免费应用。因此，为了进一步推广游戏，决定将游戏以免费应用的方式发布到谷歌应用商店中。在谷歌应用商店 Google Play 发布主要包括以下四个步骤：

(1) 注册谷歌商店开发者账号

在谷歌商店开发者控制台（Google Play Developer Console）网站上进行注册，需要注意的是，开发者所在地址无法选择中国大陆，因此只能填写香港、澳门或国外的地址。

(2) 设置签名

在 Unity3D 的 Player Settings 的 Publishing Settings 中，创建 Keystore 和 Key，并设置密码。

(3) OBB 文件

谷歌应用商店对单个 APK 文件的大小有 50M 字节的限制，《悠游厦门》所编译出的 APK 文件约为 80M 字节，因此需要将 APK 文件拆分为 OBB 文件。

在 Unity3D 的 Player Settings 的 Publishing Settings 中，对 Split Application Binary 选项打钩，之后在编译的时候，会自动生成一个 APK 文件和一个 OBB 文件。此时的 APK 文件中只有游戏的第一个场景的内容，而 Unity3D 是不会自动加载 OBB 文件的，因此需要编写脚本来进行 OBB 文件的加载。需要下载 Unity3D 官方提供的一个插件——Google Play OBB Downloader，其使用方法如图 6.2 所示：

```
void Start()
{
    string expansionFilePath = GooglePlayDownloader.GetExpansionFilePath();
    string mainObbPath = GooglePlayDownloader.GetMainOBBPath(expansionFilePath);
    StartCoroutine(Load(mainObbPath));
}
IEnumerator Load(string path)
{
    WWW www = WWW.LoadFromCacheOrDownload("file://" + path, 0);
    yield return www;
}
```

图 6.2 加载 OBB 文件

(4) 编译并打包 APK 并上传、发布

一切准备就绪后使用 Unity3D 编译游戏并打包成 APK 和 OBB 文件，然后在谷歌应用商店中上传，填写应用介绍等信息，添加游戏截图，最后就可以在谷歌应用商店中发布游戏了。

6.2.3 在其他平台发布

除上述平台之外,《悠游厦门》游戏还在苹果应用商店 App Store 和微软应用商店 Windows Phone Store 进行了发布。其发布过程与谷歌应用商店的过程大同小异,主要的区别在于:

(1) 苹果和微软应用商店是允许中国大陆的用户付费的,因此需要分别编写苹果和微软应用商店支付接口的调用程序。苹果应用商店的支付接口需要使用 Objective-C 语言进行编写,并在 Unity3D 的脚本中使用 DllImport 进行调用。

(2) Windows Phone 虽然也是使用 .NET 框架,但 Windows Phone 上使用的 .NET 框架是经过精简的,有一部分功能是无法使用的。如 .NET 2.0 的 XmlDocument 类无法使用,而必须使用 LINQ 版的 XML 库。因此可以使用条件编译代码来对 Windows Phone 平台中出现的不兼容代码进行特殊处理。

6.3 本章小结

游戏测试与发布是游戏项目的收官阶段,同样也是很重要的阶段。完整的游戏测试为良好、流畅的游戏体验提供了保障,游戏的成功发布则提供了游戏玩家接触到游戏的途径,意味着游戏开发工作的顺利完成。

本章前半部分叙述了游戏测试的环境和测试用例、过程,通过分析测试结果,确认了游戏可以在各移动平台上正常运行,为游戏的发布做好了充分准备。后半部分详细叙述了将游戏在中国移动动漫基地平台和谷歌商店发布的过程、简略叙述在苹果商店和微软商店发布时的注意事项,概括了使用 Unity3D 发布移动游戏应用的一般过程。

第七章 结束语

7.1 论文工作总结

2013 年是移动游戏迅速发展的一年,在这移动游戏发展的潮流中,本人非常荣幸地以一个实习生的身份加入厦门时代华亿动漫有限公司的游戏开发团队,参与《悠游厦门》旅游主题游戏的开发,负责程序开发的工作。经过团队数月的共同努力,游戏项目主要取得了以下成果:

- (1) 《悠游厦门》游戏顺利完成开发,厦门时代华亿动漫有限公司与厦门市旅游局签订的合同中的游戏开发任务顺利完成;
- (2) 游戏于中国移动动漫基地平台、谷歌应用商店 (Google Play)、微软应用商店 (Windows Phone Store)、苹果应用商店 (App Store) 发布;
- (3) 至少有两万名玩家下载并安装游戏;
- (4) 作为“厦门中秋旅游嘉年华”活动的一部分,对厦门的文化起到了宣传作用,吸引更多游客,带动旅游和消费,创造了旅游收入。

游戏项目获得的成果也许不能算是非常显著的,但开发团队中的每个人都从中获得了丰富的经验和巨大的收获,同时游戏本身作为一个旅游主题的游戏,是游戏行业与传统旅游行业有机结合、互惠互利的一次非常有益的尝试。

本文主要研究并实现了手机游戏《悠游厦门》的开发过程,叙述了软件工程思想在游戏开发中的应用。游戏开发是一个非常复杂的过程,但是仍然有规律、套路可循,应用软件工程思想就是改善游戏开发过程的一个有效的方法。同时,软件工程思想的应用并不是生搬硬套,而是根据游戏软件的特征和游戏项目的实际需求制定开发计划和方法,使软件工程与游戏开发有机地结合。本文根据软件工程思想,首先对游戏项目的需求进行分析,再根据需求进行游戏内容的设计、建模,最后对游戏进行实际开发、测试并发布,较为完整地总结了软件工程思想在游戏开发项目中的应用。

7.2 后续工作展望

游戏项目虽然圆满完成,但仍然存在着许多不足的地方,如:

- (1) 客户端与服务器的异步通信模块的代码条理不清,需要改动事务时容易出现错误。
- (2) 图形用户界面对于不同的屏幕分辨率适应能力较差,在某些特殊的屏幕分辨率下会出现锯齿。
- (3) 游戏数值设定不佳,导致游戏性较差。

针对这些不足，今后需要做以下几点改进：

- （1） 充分利用 C# 语言和 Unity3D 脚本的特性，对于缺乏条理的模块进行分析并设计出更合理的程序结构，提升代码的可读性，提高程序的条理性和可维护性。
- （2） 进一步学习 NGUI 系统的使用方法，充分了解并分析图形用户界面在不同屏幕分辨率下表现出的区别，针对区别与项目组中的美术人员进行更细致的沟通，从而能够进一步优化图形用户界面的表现。
- （3） 收集玩家反馈意见，不断优化游戏的数值设定，使游戏更具有趣味性。

Unity3D 作为一个使用简便、功能强大、扩展性强、具有跨平台特性的游戏引擎，在游戏引擎市场中获得了巨大的成功。然而其使用领域不仅仅局限于制作游戏，在科学研究领域也同样有着应用，例如利用其物理系统可以实现各种物理模拟。相信在现在以及不久的将来，Unity3D 或游戏引擎技术在不同领域的不同应用会陆陆续续、源源不断地出现。

游戏引擎技术的高速发展，使进入游戏行业的技术门槛大大降低，各种应用平台的出现也使游戏的发布和推广变得非常轻松，游戏开发者们可以更专注于游戏内容的创造。一个游戏的最重要的部分不是使用了什么游戏引擎，也不是在什么平台上运行，而是游戏内容本身，游戏的最重要的作用是给人们带来快乐、享受。在游戏技术发展越来越快的今天，相信越来越多的人会加入到游戏行业当中，好游戏也会络绎不绝地出现在人们的眼前。

参考文献

- [1] 中国音数协游戏工委 (GPC), CNG 中新游戏研究 (伽马数据), 国际数据公司 (IDC). 2014 年 1-6 月中国游戏产业报告. 2014.
- [2] 陈俊锋. 基于 Unity3D 的跨平台手机网络游戏的研究与实现[D]. 中山大学, 2013.
- [3] Game engine - Wikipedia. http://en.wikipedia.org/wiki/Game_engine
- [4] 游戏引擎_百度百科. <http://baike.baidu.com/view/33343.htm>
- [5] 太平洋电脑网. 游戏不单只靠烧硬件 主流游戏引擎大盘点. 2012.
http://diy.pconline.com.cn/graphics/study_gra/1202/2672414_all.html
- [6] 51CTO. 主流手机游戏引擎介绍. 2013. http://mobile.51cto.com/aengine-387752_all.htm
- [7] Unity (game engine) – Wikipedia. [http://en.wikipedia.org/wiki/Unity_\(game_engine\)](http://en.wikipedia.org/wiki/Unity_(game_engine))
- [8] SSH 框架_百度百科. <http://baike.baidu.com/view/2090973.htm>
- [9] Monopoly (game) – Wikipedia. [http://en.wikipedia.org/wiki/Monopoly_\(game\)](http://en.wikipedia.org/wiki/Monopoly_(game))
- [10] 博饼_百度百科. <http://baike.baidu.com/view/328580.htm>
- [11] 林深华, 范忠尚, 将建兵, 等. 基于 Android 平台 Unity3D 游戏设计与实现[J]. 企业科技与发展: 下半月, 2013 (5): 40-42.
- [12] 林剑, 张帅, 伍传敏. 基于 Unity3D 的手机触屏类游戏关键技术研究实现[J]. 三明学院学报, 2014, 2: 005.
- [13] 龙艳军, 俞佳曦. Unity3D 游戏开发中的版本控制研究[J]. 软件导刊, 2012, 11(11): 12-14.
- [14] 姚剑. 基于 UML 的网络游戏模型的实现和研究 [D]. 武汉: 武汉大学, 2004.
- [15] 段丽, 薛永毅, 赵亦松. 基于 UML 的手机网络对战游戏系统设计分析[J]. 北京联合大学学报: 自然科学版, 2008, 22(1): 38-40.
- [16] 孙晓鹏, 冯俊杰, 邵英安, 等. UML 在游戏系统分析与设计中的应用[J]. 计算机工程与应用, 2010, 46(13): 70-72.
- [17] 刘晓瑞. 基于 UML 的网络游戏建模研究[J]. 洛阳工业高等专科学校学报, 2006, 16(1): 42-44.
- [18] 胡必波. 基于 Android 系统的手机游戏设计与实现[D]. 华南理工大学, 2012.
- [19] 石彦, 桂志海. 软件工程需求分析与质量保障[J]. 无线互联科技, 2014 (2): 45-45.
- [20] 卢梅, 李明树. 软件需求工程——方法及工具评述[J]. 计算机研究与发展, 1999, 36(11): 1289-1300.
- [21] 李师贤, 张珞玲. 需求分析的常见问题及其对策分析[J]. 计算机工程, 2002, 28(1): 6-8.

- [22] 黄怡强, 黄怡胜. 浅谈软件开发需要分析阶段的主要任务[J]. 中山大学学报论丛, 2002, 22(1): 262-265.
- [23] 王庆育. 软件工程[M]. 清华大学出版社, 2004.
- [24] Heineman G T, Councill W T. Component-based software engineering[J]. Putting the Pieces Together, Addison-Westley, 2001.
- [25] Flynt J, Salem O. Software Engineering for Game Developers (Software Engineering Series)[J]. Course Technology PTR, 2004.
- [26] Eberly D H. 3D game engine design[M]. San Francisco: Morgan Kaufmann Publishers, Inc, 2001.
- [27] Peeper C, Mitchell J L. Introduction to the DirectX® 9 high level shading language[J]. ShaderX2: Introduction and Tutorials with DirectX, 2003, 9.
- [28] 麒麟子(Alex). 可编程渲染管线与着色器语言.
<http://www.cnblogs.com/geniusalex/archive/2012/07/20/2601847.html>
- [29] Wang S, Mao Z, Zeng C, et al. A new method of virtual reality based on Unity3D[C]//Geoinformatics, 2010 18th International Conference on. IEEE, 2010: 1-5.
- [30] Lee W S, Kim S D, Chin S. Subsurface Scattering-Based Object Rendering Techniques for Real-Time Smartphone Games[J]. Mathematical Problems in Engineering, 2014, 2014.
- [31] Unity Scripting Reference. <http://docs.unity3d.com/ScriptReference/>
- [32] MSDN Library. <http://msdn.microsoft.com/en-us/library/ms123401.aspx>
- [33] NGUI: Next-Gen UI kit. http://www.tasharen.com/?page_id=140
- [34] SimpleJSON. <http://wiki.unity3d.com/index.php/SimpleJSON>
- [35] Support for Split Application Binary (.OBB). <http://docs.unity3d.com/Manual/android-OBBsupport.htm>

致 谢

首先非常感谢褚华老师对本论文开题和撰写的指导，也非常感谢厦门时代华亿动漫有限公司给本人的实习机会，特别要感谢游戏项目组中的曹德邦经理和各位同事，同时也要感谢带领本人走进游戏开发领域、指导我各种游戏开发技术的黄佳维同学。

作者简介

1. 基本情况

男，广西南宁人，1990 年 7 月出生，西安电子科技大学软件学院软件工程专业 2012 级硕士研究生。

2. 教育背景

2008.8～2012.6 就读于西安电子科技大学经济管理学院信息管理与信息系统专业，获管理学学士学位

2012.8～ 西安电子科技大学软件学院软件工程专业硕士研究生