

基于 iOS 的学生个人计划管理系统的设计与实现

作者姓名： 孙嘉玮
指导教师： 孙永佼 副教授
单位名称： 信息科学与工程学院
专业名称： 计算机科学与技术

东 北 大 学
2015 年 6 月

Design and Implementation of iOS-Based Students Planning Management System

by SUN Jiawei

Supervisor: Associate Professor SUN Yongjiao

Northeastern University

June 2015

毕业设计（论文）任务书

毕业设计（论文）题目：

基于 iOS 的学生个人计划管理系统的设计与实现

设计(论文)的基本内容：

设计并实现一个基于 iOS 的学生个人计划管理系统。要求基于 iOS 环境，使用 object-c、C++语言，在 Cocos2-dx 框架下实现学生的学习和健身计划的定制和完成情况的记录功能，并且在完成计划的过程中，加入好友之间的社交关系，能让好友了解自己的计划完成进度，达到相互监督的目的。

系统应包含的主要功能：

学习模块，包括自习的计划和记录，上课的计划和记录；运动模块，包括在体育场和体育馆运动的时间和项目的计划；社团计划，记录参与社团活动的项目和时间的计划。

毕业设计（论文）专题部分：

题目：_____

设计或论文专题的基本内容：

学生接受毕业设计（论文）题目日期

第 1 周

指导教师签字：

2015 年 3 月 5 日

基于 iOS 的学生个人计划管理系统的设计与实现

摘要

近年来，随着信息技术的不断进步与发展，智能手机越来越成为学生生活中不可缺少的一部分，但同时也给学生的家长带来了不少的烦恼，很多学生在智能手机中的游戏上花费了大量的时间，一些自制力不强的学生甚至经常在课堂上玩弄手机。本系统致力于给东北大学的大学生们提供一款基于 iOS 平台和 Cocos2d-X 引擎的个人计划软件，让学生们对个人的学习生活有着更明确的计划，减少学生花费在智能手机游戏上的时间。

本系统的主要功能是给东北大学大学生提供鸟瞰视角的东北大学地图，学生可以在浏览东北大学地图的同时，设定自己的学习运动计划并记录自己的完成情况。本系统采用 C/S 的架构，客户端使用 Cocos2d-X 的引擎，显示东北大学的地图，并且实现计划制定、完成计划、好友交互的功能；服务器负责处理数据库操作，将用户的数据存储在服务器中。本系统融合了游戏中可视化的特点和生活计划软件的元素，极大地提高了学生对使用计划软件的兴趣。

本文首先进行了系统的需求获取和业务分析。然后采用分层的方法对系统进行设计，叙述了系统界面、数据库和核心功能以及服务器端的设计和实现。并通过对系统的测试，对其功能和性能进行评估。最后，本文进行了总结，介绍了系统不足和有待改进的地方，指出了下一步的工作方向。

关键词：学生个人计划系统；可视化；Cocos2d-X 引擎；iOS

Design and Implementation of iOS-Based Students Planning Management System

Abstract

Recently, with the rapid development of information technology, smart phones have become an indispensable part of students' daily life. However, it also brings a lot of troubles to students' parents at the same time. Many students spend their time on smart phone games. Some of them with weak self-control even play phone games in class. This system focuses on providing an iOS and Cocos2d-x based personal planning software for Northeastern University students, in order to make students clear their study life plans and reduce waste time on the games.

The main function of this system is to provide a bird's eye view perspective map of Northeastern University for students. While operating the map, students can set their own learning exercise plan and record their completion. The system uses C/S architecture. The client part adopt Cocos2d-x engine to load the map, and implement the function of making and completing the plans as well as friends communication. The server part is responsible for handling database operations and then storing the users' data into the database. The system merges the game visual characteristics with the life planning software elements, which greatly improve the students' interests in using planning software.

In this paper, we first elicit the requirement elicitation and analyze the business. Then, we adopt a hierarchical approach to design the system. This paper describes the design and implement of the system interface, database, core function and web servers. Next, we evaluate the performance by test result. Finally, we summarize the whole paper and point out the deficiencies of the system for future work.

Keywords: Student personal planning system; visualization; Cocos2d-X; iOS

目录

毕业设计（论文）任务书	I
摘要.....	II
Abstract.....	III
第一章引言	1
1.1 课题研究的背景和来源	1
1.2 课题研究成果应用的实际意义	2
1.3 本文的内容和主要工作	2
第二章相关技术简介	3
2.1 Cocos2d-X 引擎	3
2.2 CocoStudio 介绍	5
2.3 SQLite 和 MySQL 介绍	6
2.4 Tomcat+Java Servlet 介绍	6
2.5 本章小结	7
第三章需求分析	9
3.1 系统实现目标	9
3.2 系统的运行环境	10
3.3 系统的开发环境	10
3.4 功能性需求描述	10
3.4.1 用户注册	11
3.4.2 用户登录	11
3.4.3 制定学习生活计划	12
3.4.4 学习生活活动计时	12
3.4.5 邀请好友共同学习/运动	12
3.4.6 添加好友	12
3.4.7 查看好友的邀请信息	13
3.4.8 查看个人计划完成信息	13
3.5 非功能性需求描述	13

3.6 本章小结	14
第四章系统设计	15
4.1 系统总体设计	15
4.2 数据库设计	15
4.3 系统 UI 设计	17
4.4 系统功能设计	17
4.6 服务器设计	19
4.7 本章小结	20
第五章系统实现	21
5.1 地图的制作和显示	21
5.2 角色的移动和碰撞检测	22
5.3 客户端界面的设计与实现	25
5.3.1 系统开发环境的语言、搭建和配置	25
5.3.2 UI 控件的实现	25
5.3.3 UI 界面跳转的实现	26
5.4 用户登录注册功能模块的实现	28
5.5 制定计划和完成功能的实现	30
5.6 好友功能模块实现	32
5.6.1 添加好友功能实现	32
5.6.2 发送好友邀请信息和查看信息功能	33
5.7 本章小结	34
第六章系统测试	35
6.1 单元测试	35
6.2 集成测试	36
6.3 系统测试	36
6.4 本章小结	39
第七章结论	41
7.1 本文工作总结	41
7.2 进一步展望	42
参考文献	43

致谢.....	45
---------	----

第一章引言

1.1 课题研究的背景和来源

智能手机是目前大家用的最多的移动终端设备，我们的生活也与之息息相关，随着 3G，4G 网络的发展，智能手机市场也随之变化。现在流行智能手机的操作系统主要分为 iOS，Android，Windows Phone 三个阵营，本文将简要的介绍一下这三个平台：

iOS 是苹果公司开发的操作系统，它的系统架构分为核心系统层，核心服务层，媒体层，可触碰层^[1]。iOS 系统拥有大量的应用支持，同时因为不开源的原因，所以安全性相对比较高^[2]。缺点是对第三方应用的限制比较多，高昂的设备价格等。

Android 是 Google 公司推出的基于 Linux 的开源手机操作系统，它的系统架构由四层组组层，分别为操作系统，中间件，用户界面和应用软件。Android 系统收益与代码开源，所以优点是开放性高，拥有最广大的开发者群体，硬件要求比较宽松，应用开发也不受到开发商的限制。对应的缺点是存在很多的垃圾软件。

Windows Phone 是微软公司开发的手机操作系统，具有桌面定制、图标拖拽、滑动控制等一系列前卫的操作体验。缺点是不支持存储卡，无法在手机上管理文件等。

智能手机确实给人们的生活带来了越来越多的便利，但同时也带来了不少的烦恼，对于当代的大学生来说，几乎已经离不开智能手机而生活了：有些学生自制力不强的甚至会在课堂上玩手机而不听老师讲课。很多学生花费了大量的时间在手机和一些电子产品的娱乐功能之上，对于自身在大学生活里面的学习生活、职业规划和人生目标的设定反而越来越模糊，导致荒废了很多大学时光的情况的出现。

列宁指出过：“任何计划都是尺度、准则、灯塔、路标。”可从此看出。制定一个详细可行的计划对于一个学生的学习和生活来说是非常重要的。所以本课题希望能够借助智能手机的普遍性，让当代大学生树立更明确的学习生活计划和目标。

此外，一般的计划性软件只有一些文本框让用户输入时间来制定计划，用户只能看到一些数字，比较抽象，用户体验也比较差。本课题改变了这种设定计划的模式，融入了可视化的元素，用户可以在界面看到学校的地图，并且操纵一个虚拟的任务在校园中行走，在学校中行走的同时完成计划的制定。

综上所述，需要一个可视化的手机学习生活计划软件，帮助学生更好的制定学习和生活计划。

1.2 课题研究成果应用的实际意义

本课题主要目的是实现一个可视化的基于 cocos2d-x 引擎的学习生活计划软件，东北大学的学生可以通过这个软件，在大致了解东北大学的地形的同时，给自己的学习、运动和生活做出计划。

由于一般的计划性软件是只有一些文本框完成计划功能的，用户只能看到一些数字和表格，用户的体验不是很好。再加上很多东北大学的新生对于学校的环境设施的位置都不是很了解：看到课程表上的建筑名有时也不知道具体在哪。所以本课题提供了一个鸟瞰视角的东北大学视角，新生在用软件熟悉学校环境的同时也可以对自己的学习和生活有一个明确的计划。再加上一般的计划性软件都是不联网的软件，所以只能自己看到自己的进度。本课题加入了联网的功能和社交的元素，可以添加好友，邀请好友一起上课、自习或是运动。从而有了好友之间相互监督的效果，督促学生更好的完成自己的计划。

1.3 本文的内容和主要工作

本文介绍了 Cocos2d-x 引擎、其中的核心基本类、配合 Cocos2d-x 进行开发的 CocoStudio, TiledMap 等工具以及 java servlet 服务器的相关内容，利用这些工具和知识的整合，实现了可视化学习生活计划软件的基本功能。

其余各章的内容组织如下：

第二章介绍了 Cocos2d-x 框架、核心基本类，CocoStudio 和 TiledMap Editor 等辅助制作软件，以及 Java Servlet, MySQL 数据，和 Tomcat 服务器等知识。

第三章分析了系统的开发环境，利用用例建模分析系统的功能需求，并且根据实际情况补充了部分需求，提出了非功能性需求，并描述了系统术语表。

第四章对系统总体进行设计，并详细分析了数据库的设计，系统 UI 逻辑的设计，系统功能的设计，以及服务器端的分层设计。

第五章针对系统的各个功能的实现进行了描述，重点阐述了地图设计，角色移动，碰撞检测，制定和完成计划以及好友功能的实现。

第六章介绍了对系统进行测试的情况，并且对出现的错误进行了分析解决。

第七章结论部分是对整个论文的总结以及下一步工作的展望，指出我们工作中有待改进的地方，同时指明了下一步的研究方向。

第二章相关技术简介

2.1 Cocos2d-X 引擎

2.1.1 Coco2d-X 的介绍

Cocos2D 是一个开源框架，用于构建二维游戏、演示程序和其他图形界面交互应用等^[3]。它于 2008 年 2 月诞生于 Cordoba 市附近的 Los Cocos，最早是由 Ricardo 和他的朋友们用 Python 开发的。最早引擎取名自开发地 Los Cocos。后来引擎的 0.1 版本改名为 Cocos2D^[4]。

随着智能手机平台的发展，在其他平台上的游戏开发需求也开始多了起来，此时，Cocos2D 的各种平台、各种编程语言移植版也开始涌现。ShinyCocos、Cocos2D-Andriod、CocosNet、Cocos2D-JavaScript 相继诞生。

Cocos2D-X 是 Cocos2D 家族的一个重要分支，因为它的跨平台特性，开发者编写一套代码就可以横跨 Android 和 iOS 两大平台。从而提高开发者的开发效率^[5]。Cocos2D-X 不但具有跨平台的特性，它还保留了 Cocos2D 的全部特性，这样 Cocos2D 开发者可以很轻松地掌握 Cocos2D-X^[6]。

随着智能设备平台的普及，用户数量的增多，智能平台的应用，尤其是游戏异常火爆，从植物大战僵尸和愤怒的小鸟，移动平台游戏的开发进入了新的阶段。但另一方面平台的多样性也给开发者带来诸多不便，怎样将一款游戏发布到多个平台也是开发者们一直在探索的。特别是随着 Android 和 iOS 平台的兴起，开发者有了游戏跨平台的需求，随即 Cocos2D-X 分支诞生，其中“X”包含两个意思：一方面是 C++ 的文件扩展 CXX，“X”标志着该项目是由 C++ 编写的；另一方面意味着交叉，这个开源项目目标是让用户创建跨平台的代码^[7]。Coco2D-X 是 Cocos2D 框架用 C++ 重写的，因为 Andriod 和 iOS 均支持 C++。Coco2D-X 具有跨平台的特点，再加上 Cocos2D 全球社区的支持以及 Cocos2D 框架简单易学，开发者十分容易快速上手^[8]。

Cocos2D-X 的主要功能有流程控制（Flow control）：非常容易地管理不同场景（scenes）之间的流程控制；精灵（Sprites）：快速而方便的精灵；动作（Actions）：告诉精灵们该做什么。可组合的动作如移动（move）、旋转（rotate）和缩放（scale）等更多；特效（Effects）：特效包括波浪（waves）、旋转（twirl）和透镜（lens）等更多；平面地图（Tiled Maps）：支持包括矩形和六边形平面地图；转换（Transitions）：从一个场景移动到另外一个不同风格的场景；菜单（Menus）：

创建内部菜单；文本渲染（Text Rendering）：支持标签和 HTML 标签动作；文档（Documents）：编程指南 + API 参考 + 视频教学 + 很多教用户如何使用的简单测试例子；基于 Pyglet：没有外部的依赖；基于 OpenGL：支持硬件加速；3D 对象：MD2 模型支持；脚本语言：支持 Lua，JavaScript 语言^[9]。

2.1.2 Cocos2d-x 的核心类介绍

（1）坐标系简介

Cocos2D-X 以 OpenGL 和 OpenGL ES 为基础，所以自然支持 OpenGL 坐标系，坐标系原点在屏幕左下角，x 轴正方向向右，y 轴正方向向上^[10]。

由于屏幕坐标系使用的是不同的坐标系统，原点在屏幕左上角，x 轴正方向向右，y 轴正方向向下。在 3.0 以前的版本中，屏幕触摸事件 Touch 传入的位置信息使用的是屏幕坐标系。因此在 Cocos2D-X 中对触摸事件做出响应之前，需要首先把触摸点转化到 OpenGL 坐标系。而在 3.0 版本之中，Touch 把转化的一步封装在自己的函数里，可以直接调用函数来获得坐标，这样就不需要再进行转化了，简化了调用函数和接口^[11]。

（2）节点类简介

节点类 Node 是 Cocos2D-X 中的主要类。任何需要画在屏幕上的对象都是节点类，最常用的节点类包括场景类（Scene）、布景层（Layer）、精灵层（Sprite）、菜单类（Menu）^[12]。

Node 类的主要功能包括：每个节点都可以含有子节点；节点含有周期性回调的方法；可以含有动作；重写初始化的方法和周期性回调方法；在时间线上控制回调；重写渲染的绘制方法。

（3）导演类简介

导演类 Director 是 Cocos2D-X 游戏引擎的核心，它创建并且控制着主屏幕的显示，同时控制着场景的显示时间和显示方式，在整个程序里一般只有一个导演^[13]。程序的开始、结束、暂停都会调用 Director 类的方法，Director 类负责的功能有：

1. 初始化 OpenGL 会话。
2. 设置一些 OpenGL 的一些参数和方式。
3. 访问和改变场景，并访问 Cocos2D-X 的配置细节。
4. 访问视图。
5. 设置投影和朝向。

在 Cocos2D-X 中，在游戏的任何时间，只有一个场景对象实例处于运行状态。

而导演类就是负责流程的总指挥。负责程序全过程的场景切换^[14]。

（4）布景层类

布景层类 Layer 是 Node 的子类，每个游戏场景中可以有多个层，每一层负责各自的任务；每一层上可以放置不同的元素，包括文本、精灵图片和菜单等。通过层与层之间的组合关系，就可以构成游戏现实的界面 UI^[15]。

2.2 CocoStudio 介绍

CocoStudio 是由触控科技引擎团队研发的一款基于 Cocos2D-X 的免费游戏开发工具集。CocoStudio 支持多种类型游戏的编辑，满足不同开发者和公司的开发需求，并且上手容易，简单易懂，同时提供了丰富的技术支持和开发资源^[16]。CocoStudio 研发团队在经历长期的游戏开发和技术研发后，总结出游戏中必须的四套工具，功能分别如下：

动画编辑器：用于编辑游戏中使用的角色动画、特效动画、场景动画等动态的游戏资源。主要使用人员是美术设计师，美术设计师可以将绘制好的图片资源导入到动画编辑器中，然后按照策划设计来编辑各种动画效果，并且在编辑器中演示编辑后的内容。动画编辑器支持关键帧动画，序列帧动画，骨骼动画等多种动画编辑方式。并支持碎图导出，整图导出等多种资源优化方式。同时动画编辑器支持对 Flash，PSD，Plist 资源的解析^[17]。

UI 编辑器：用于编辑游戏中的所有的图形界面，主要使用人员是美术设计师。美术设计师可以将绘制好的界面图片资源导入到 UI 编辑器中，按照相应的设计方案来摆放、设计、编辑、拼接、修改相应的游戏图形界面。并将编辑好的 UI 资源直接整体导出，可以在游戏代码中直接呈现出来。UI 编辑器支持 Cocos2d-X 的全部的控件，并进行了扩充^[18]。满足了游戏开发中对各种 UI 控件的需求。UI 编辑器支持碎图导出、整图导出、UI 模板、UI 动画等实用功能。

数据编辑器：用于将数值策划编辑的数值表导入数据编辑器中，将复杂表进行分解，导出成属性表，数值表等几种常用的数据文件。数据编辑器支持索引表，导出可以选择多种数据排列方式^[19]。

场景编辑器：用于编辑游戏中的场景元素、游戏关卡。主要使用人员是策划或者关卡设计师。策划可以将多种游戏资源进行整合。包括动画资源、UI 资源、地图资源、粒子资源、图片资源、音乐音效等游戏资源在编辑器中按照设计文档进行编辑^[20]。同时场景编辑器可以编辑游戏中碰撞、触发器、控制器等逻辑资源。场景编辑器支持模拟运行，可以在模拟器中或者无线连接到手机上直接运行当前游戏场景。

2.3 SQLite 和 MySQL 介绍

SQLite，是一款轻型的数据库，是遵守 ACID 的关系型数据库管理系统，它包含在一个相对小的 C 库中。它是 D.RichardHipp 建立的公有领域项目。它的设计目标是嵌入式的，而且目前已经在很多嵌入式产品中使用了它，它占用资源非常的低，在嵌入式设备中，可能只需要几百 K 的内存就够了。它能够支持 Windows/Linux/Unix 等等主流的操作系统，同时能够跟很多程序语言相结合，比如 Tcl、C#、PHP、Java 等，还有 ODBC 接口，同样比起 Mysql、PostgreSQL 这两款开源的世界著名数据库管理系统来讲，它的处理速度比他们都快^[21]。SQLite 第一个 Alpha 版本诞生于 2000 年 5 月。至 2015 年已经有 15 个年头，SQLite 也迎来了一个版本 SQLite 3 已经发布。

不像常见的客户-服务器范例，SQLite 引擎不是个程序与之通信的独立进程，而是连接到程序中成为它的一个主要部分。所以主要的通信协议是在编程语言内的直接 API 调用。这在消耗总量、延迟时间和整体简单性上有积极的作用。整个数据库(定义、表、索引和数据本身)都在宿主主机上存储在一个单一的文件中。它的简单的设计是通过在开始一个事务的时候锁定整个数据文件而完成的。

MySQL 是一个关系型数据库管理系统，由瑞典 MySQLAB 公司开发，目前属于 Oracle 公司。MySQL 最流行的关系型数据库管理系统，在 WEB 应用方面 MySQL 是最好的 RDBMS (Relational Database Management System，关系数据库管理系统) 应用软件之一^[22]。MySQL 是一种关联数据库管理系统，关联数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。MySQL 所使用的 SQL 语言是用于访问数据库的最常用标准化语言。MySQL 软件采用了双授权政策，分为社区版和商业版，由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，一般中小型网站的开发都选择 MySQL 作为网站数据库。由于其社区版的性能卓越，搭配 PHP 和 Apache 可组成良好的开发环境。

2.4 Tomcat+Java Servlet 介绍

Tomcat 是 Apache 软件基金会 (Apache Software Foundation) 的 Jakarta 项目中的一个核心项目，由 Apache、Sun 和其他一些公司及个人共同开发而成。由于有了 Sun 的参与和支持，最新的 Servlet 和 JSP 规范总是能在 Tomcat 中得到体现，Tomcat 5 支持最新的 Servlet 2.4 和 JSP 2.0 规范^[23]。因为 Tomcat 技术先进、性能稳定，而且免费，因而深受 Java 爱好者的喜爱并得到了部分软件开发商的认可，成为目前比较流行的 Web 应用服务器。目前最新版本是 8.0。

Tomcat 服务器是一个免费的开放源代码的 Web 应用服务器，属于轻量级应用服务器，在中小型系统和并发访问用户不是很多的场合下被普遍使用，是开发和调试 JSP 程序的首选^[24]。对于一个初学者来说，可以这样认为，当在一台机器上配置好 Apache 服务器，可利用它响应 HTML（标准通用标记语言下的一个应用）页面的访问请求。实际上 Tomcat 部分是 Apache 服务器的扩展，但它是独立运行的，所以当运行 tomcat 时，它实际上作为一个与 Apache 独立的进程单独运行的。

Servlet 是 Java2.0 中新增的一个全新功能，运行在请求/响应模式的服务器的模块，具有平台无关性的，没有图形用户界面的 Java 程序。它可以在许多方面扩展 Web 服务的功能^[25]。因为 Servlet 是用 Java 编写的，所以它能够访问 Java 全部的高级特性：数据库连接，网络有关的特性，面向对象特性和多线程处理的支持等。这些特性可以用来分发完全功能的应用程序到 Web 客户而不需要任何特殊的客户端配置。因为使用 HTTP 作为其接口，所以在任意的浏览器环境下运行都是一样的。

Servlet 运行于由 Servlet 引擎所管理的 JVM（Java Virtual Machine，Java 虚拟机）中。可以使用 Java Servlet 技术编写 HTTP Servlet 类，用来扩展服务器的功能，相应客户的请求，它比较类似于传统的 CGI（Common Gateway Interface，公共网关接口）程序，但比 CGI 技术和很多其他类似于 CGI 的技术更先进。首先 Servlet 能够有效的利用服务器端的资源^[27]。而 Servlet 在每次被调用的时候由 Web 服务器进程创建一个新的 Servlet 线程，相同的请求由同一个线程处理，从根本上解决了 CGI 浪费系统资源的问题。其次，传统的 CGI 程序不具备跨平台的特性，系统环境发生变化，CGI 程序就必须修改。而 Servlet 具备 Java “一次编写，到处可用”的特性，在系统的开发过程当中保持了系统的夸张性和可重用性。第三，在传统 Web 应用中，CGI 程序与数据库之间建立连接的速度成为系统的瓶颈，死机、数据库死锁的现象频繁发生，数据库一致性得不到保证。而 Servlet 有连接池的概念，它可以利用多线程的优点，在系统缓存中事先建立好若干个与数据库的连接，当需要与数据库打交道的时候，可以随时向系统申请一个连接，这样连接建立速度比 CGI 快得多。第四，Servlet 编译过后以字节码运行，而 CGI、ASP、PHP 都在脚本级运行，所以 Servlet 的运行效率比它们高^[28]。

2.5 本章小结

本章首先对 Cocos2D-X 进行了介绍，包括了 Cocos2D-X 的来源，特点，优缺点等；接着对 Cocos2D-X 引擎的基本核心类进行了介绍，包括坐标系的设定、

导演类、节点类、布景层类的介绍。之后介绍了辅助开发基于 Cocos2D-X 程序的工具：CocoStudio，包括了动画编辑器、UI 编辑器、数据编辑器和场景编译器的功能。说明 Cocos2D-X 引擎已经逐步成为大众认可的游戏开发引擎，开发者能够更方便的使用该引擎开发自己的程序；接着介绍了本系统采用的两种数据库：Sqlite 和 Mysql。根据他们不同的特点，分别用于存储本地数据和服务器数据。最后介绍了 Tomcat+Java Servlet 的 MVC 模式的服务器的基本概念和特点。

由本章的分析可知，Cocos2D-X 的开发已经越来越受到开发者们的重视，并且提供了大量的辅助软件帮助设计程序。并且随着新版本的更新，Cocos2D-X 的语句也更加规范和易于使用。同时，经典的 Tomcat 和 Java Servlet 搭建的服务器仍然受到很多开发者的钟爱，可以满足该系统所要求的服务器端的功能需求。

第三章需求分析

需求分析是软件开发过程中重要的一个环节，需求分析的基本任务是根据用户的需求，准确定义要完成的系统的目标。它的主要用途是明确系统需求，是用户和开发人员之间进行讨论的基础，并作为系统设计、实现以及测试的依据。

只有准确全面的需求获取与分析才能保证系统设计、系统实现的顺利进行，为后期测试提供必要的支持。在参考了一般性的计划软件并且询问了一些在校学生的建议之后，制定了本系统的最终的具体需求。

3.1 系统实现目标

在可视化学习生活计划系统中，制定学习生活计划和记录完成的进度是最核心的功能。用户应能够针对不同的分类，如上课时间，自习时间，运动时间等分别制定计划，之后系统应能在对应的不同事件中自动记录用户完成的时间和进度。同时在好友之间要实现交互的功能从而达到相互监督的目标。为了满足可视化系统的要求，系统应提供友好的用户界面和学校的鸟瞰地图供用户进行操作。

为保证可视化学习生活计划系统能有效的为学生所使用，从功能上必须达到以下目标：

（1）实现学习生活计划系统的各项功能：用户分别制定每周学习、生活、运动的总时间；用户在不同建筑进行不同活动时，系统能记录活动的时间并完成记录：如学生在教学馆进行自习时，系统对用户自习时间进行统计，并且记录在本周完成的时间中，供用户查看自己的完成进度；用户在邀请好友一起自习或者运动时，系统能给用户的好友发送邀请信息，从而达到好友交互的目的。保证系统实现功能的有效性。

（2）保证地图数据、精度和信息传递的准确。本系统面向的对象中还有对学校的地理环境不是很了解的新生团体，所以学校内部的各个建筑的相对位置信息必须保证相对准确。在邀请好友功能时，保证传递信息的时间地点信息准确。保证系统信息的准确性。

（3）保证系统功能操作的简单性。系统所提供的功能必须能让用户方便使用：用户能在较短时间内完成制定计划、完成进度和邀请好友的操作。同时用户在查看自己的信息和添加好友时也必须保证操作的简单性，保证系统功能的易操作性。

3.2 系统的运行环境

本系统的用户群体是使用 iOS 操作系统的东北大学学生。根据系统需求，设定了本系统的运行环境。

硬件环境：

手机设备：iphone4s 版本以上的系列；

服务器主机配置：CPU 使用 Core i5 处理器，内存 4GB。

软件环境：

客户端：操作系统版本 iOS8 以上；

服务器：Tomcat 5.5 服务器，MySQL5.6 数据库。

3.3 系统的开发环境

本系统采用开源的引擎、框架和软件进行开发，使用相对简单且以达到减少成本的目的。

服务器环境：Web 服务器采用 Tomcat5.5。在 MyEclipse 里集成使用 Tomcat，可以比较方便的完成部署和配置工作。

开发工具与支持环境：系统开发使用 Xcode6。这是一款苹果公司提供的编译软件，能够编写包括 C++ 在内的多种语言程序。框架则使用 Cocos2D-X 引擎，它提供了多种封装好的接口供开发者使用。地图编辑则使用 TiledMap Editor。使用它比较方便的绘制出鸟瞰视角的地图。UI 设计采用 CocoStudio。使用它可以辅助开发者更方便的设计 UI 界面。

3.4 功能性需求描述

在面向对象的系统分析中，获取功能需求的最常用的技术就是用例建模技术。它能够直观地向客户展示系统的功能，方便技术人员准确地获取需求。在项目开发的前期，它帮助开发人员理解系统的需求而不考虑具体的实现方法。用例，表示系统的一种行为或者某个功能。它是在特定条件下，为了达到某种目的，系统和用户之间进行交互的一个序列。用例模型由参与者和用例组成。参与者是与系统进行交互的外部实体，可能是某个人或者是某个事物。图 3.1 就是本系统的用例图。系统的参与者就是用户本身，有七个基本的用例：用户注册、用户登录、制定学习生活计划、学习生活活动计时、添加好友、查看好友的邀请信息以及查看个人计划完成信息。对于用例分析来说，只有用例图和用例描述相结合才能表示系统的需求，相关的用例描述如图 3-1。

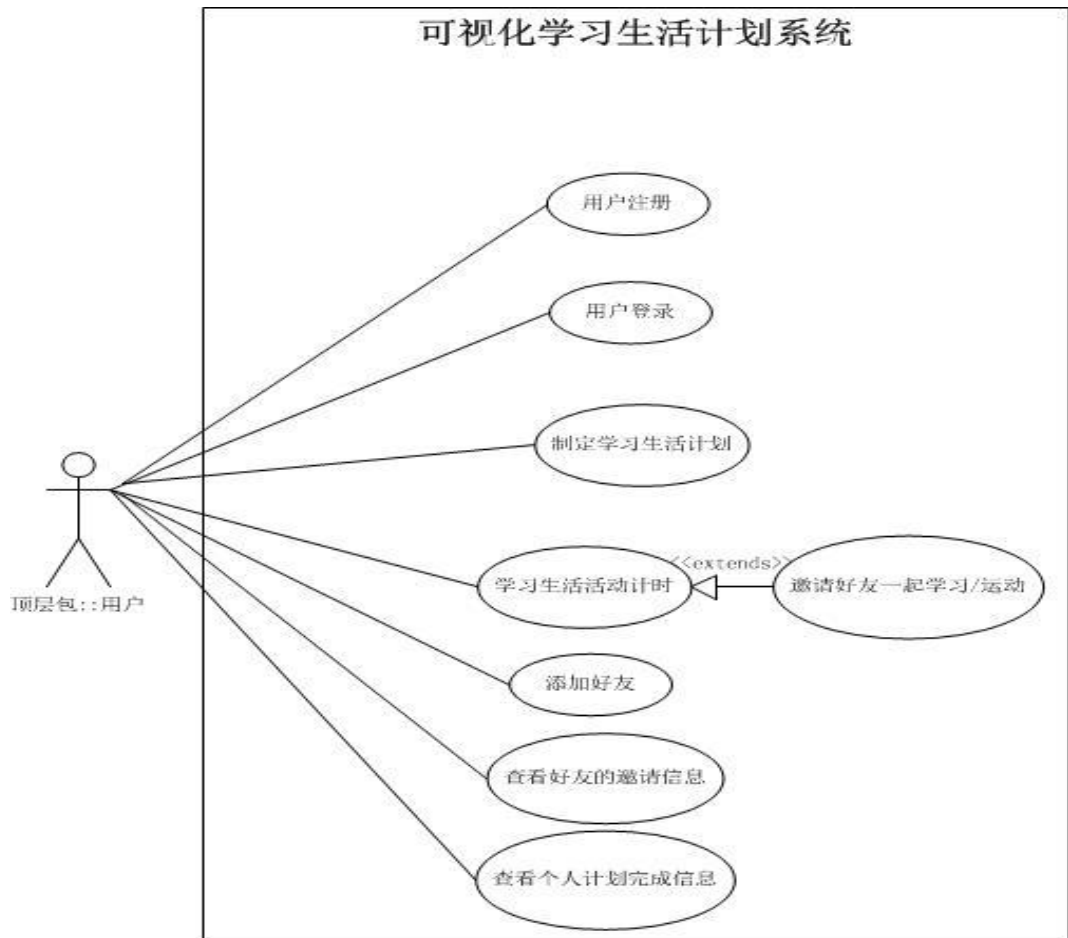


图 3.1 可视化学习生活计划系统用例图

3.4.1 用户注册

参与者：未注册用户

简要说明：用户进入系统输入账号密码进行注册。

基本事件流：

- (1) 未注册用户在输入框内输入账号和密码。
- (2) 点击注册按钮，进行注册确认。
- (3) 经过系统检测，账号不重复则可以注册，系统提示注册成功，否则提示注册失败。
- (4) 提示注册成功之后，用例结束。

3.4.2 用户登录

参与者：已注册用户

简要说明：用户输入正确的账号密码进行登录操作。

基本事件流：

- (1) 已注册用户在输入框内输入账号和密码。

(2) 用户点击登录按钮。

(3) 系统验证用户输入的账号和密码是否正确。若输入正确则跳转地图界面，反之则提示输入错误，继续停留在此界面，用例结束。

3.4.3 制定学习生活计划

参与者：已登录用户

简要说明：制定自己的学习/生活计划。

基本事件流：

- (1) 已登录用户点击 **Study/Sports** 按钮。
- (2) 根据系统提示输入计划的学习/运动时间。
- (3) 系统记录个人的计划时间。用例结束。

3.4.4 学习生活活动计时

参与者：已登录用户

简要说明：记录用户自己学习/运动的时间。

基本事件流：

- (1) 已登录用户移动视角到特定建筑，并点击触发按钮。
- (2) 系统提示选择记录自习时间/上课时间/运动时间。
- (3) 用户选择完之后系统开始计时，直到用户点击结束按钮。
- (4) 系统将统计的时间记录到个人信息档案中，用例结束。

3.4.5 邀请好友共同学习/运动

参与者：已登录用户

简要说明：用户邀请好友与自己共同学习/生活。

基本事件流：

- (1) 用户点击建筑物触发事件中的邀请好友按钮。
- (2) 系统判定用户是否存在好友，若存在则向好友发出邀请信息，反之则提示邀请失败。
- (3) 系统邀请成功之后将信息存至服务器，用例结束。

3.4.6 添加好友

参与者：已登录用户

简要说明：根据用户名添加好友。

基本事件流：

- (1) 用户点击添加好友按钮。
- (2) 用户根据提示填入用户姓名并点击继续按钮。
- (3) 系统根据用户名向服务器请求是否找到姓名。
- (4) 若找到好友则系统提示添加好友成功，并向服务器请求更新数据库信息，反之则提示添加失败。用例结束。

3.4.7 查看好友的邀请信息

参与者：已登录用户

简要说明：查看好友邀请自己的信息。

基本事件流：

- (1) 用户点击查看信息按钮。
- (2) 系统向服务器请求查看信息接受者为当前用户名的信息。
- (3) 若获得信息则在系统界面中显示邀请信息，反之则显示为空。用例结束。

3.4.8 查看个人计划完成信息

参与者：已登录用户

简要说明：查看自己已经完成和计划完成的学习运动时间。

基本事件流：

- (1) 用户点击我的信息按钮。
- (2) 系统查询当前用户名的学习运动统计时间。
- (3) 系统将查询到的统计时间显示在界面上，用例结束。

3.5 非功能性需求描述

一个软件产品的需求可以分为功能性需求和非功能性需求，其中非功能性需求是常常被轻视，甚至被忽视的一个重要方面。但是，软件产品非功能性定义不仅决定产品的质量，还在很大程度上影响产品的功能需求定义。下面列出本系统的非功能性需求：

- (1) 系统应具有一定的可靠性。

可靠性是指在规定的一段时间和条件下软件维持其性能水平的能力有关的一组属性。考虑到本系统的实际情况，本系统需保持 24*7 不间断地稳定运行。并且在网络发生意外错误时具备一定的恢复数据的能力。

- (2) 系统应具有一定的易用性。

应保证用户在使用该系统时能在尽量短的时间内使用本系统的全部功能。

（3）系统应具有一定的效率。

系统与服务器的信息交互过程应控制在一定时间内完成，查询功能在 2 秒之内完成。

3.6 本章小结

本章首先从总体上分析系统的实现目标，提出了总体的功能要求，之后介绍了系统的运行环境和开发环境。然后通过用例分析的方式描述了系统的功能性需求分析，对系统的功能性需求有了详细的了解。最后对系统的需求分析做了补充，提出了非功能性需求描述。从而完整的掌握系统的整体需求。

从本章的需求分析中可以看出，功能性需求分析对业务的逻辑和流程有着更多的要求，而非功能性需求则对整体系统的框架和硬件设施有着更严格的要求。设计系统时应充分考虑这两个方面的因素，而不是偏重于某一个方面。

第四章系统设计

4.1 系统总体设计

本系统分为客户端和 Web 服务器两个部分，其中 Web 服务器的部分负责连接服务器端的数据库和进行数据处理操作，根据 MVC 的思想，把 Web 服务器层分成 Entity（实体）层，DAO（数据访问）层，Servlet（服务）层。Entity 层负责数据库类的封装，DAO 层负责数据库的连接和各个基本数据操作的函数的实现，Servlet 层负责接受和处理不同的 HTTP 请求；而客户端部分负责实现系统的各个功能，UI 界面的显示，地图和移动功能的完成和本地数据的处理。图 4-1 是本系统的模块功能图。

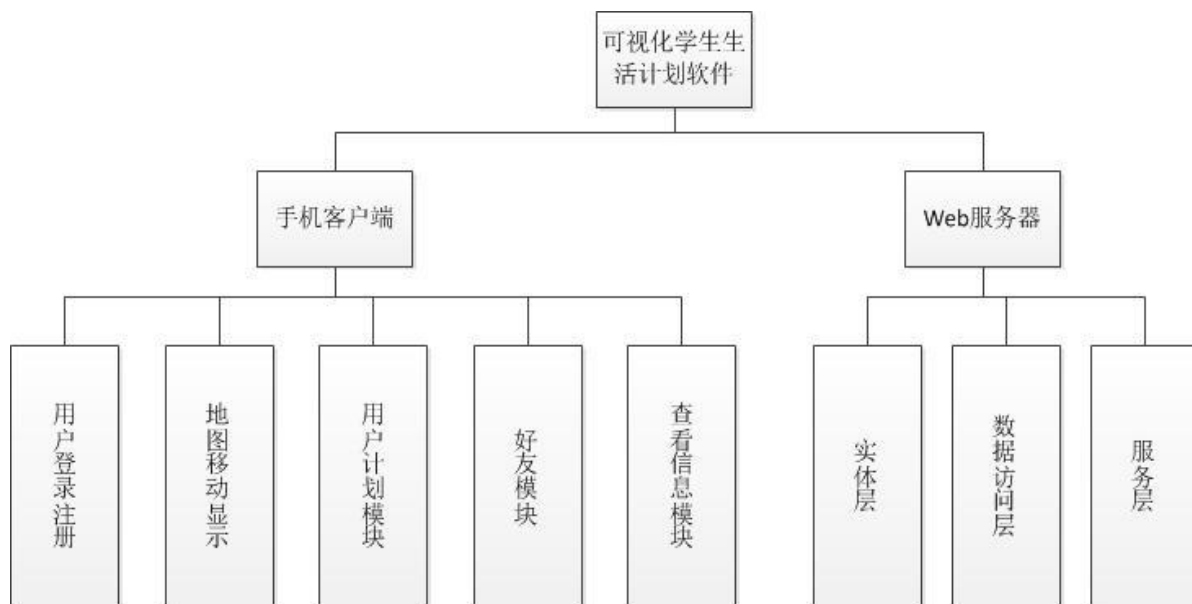


图 4-1 系统功能模块图

4.2 数据库设计

4.2.1 用户登录表设计

用户登录表主要负责验证用户的注册和登录。因为只负责验证登录和注册，所以用户登录表的设计会很简单，由两个字段构成：用户名（username）和密码（password）。其中 username 设置为主键，这样可以利用数据库的主键不能重复的特点减少用户注册时验证用户名是否存在的步骤。另外因为登录和注册功能是比较常用的功能，且不需要别的用户的信息，所以这样设计数据库可以避免把所有用户信息放在一个表内产生多余的内存开销。

表 4-1 用户登录表

字段	类型	可空	字段描述
username	varchar(40)	false	用户名，主键
password	varchar(40)	false	用户密码

4.2.2 用户信息表设计

用户信息表主要负责记录用户计划的总学习/运动时间，以及用户已经完成的学习/运动时间。根据本系统的需求，该表将记录计划上课时间（totalClassTime），计划自习时间（totalStudyTime），计划运动时间（totalSportsTime），已完成的上
课时间（classTime），已完成的自习时间（stduyTime），已完成的运动时间（sportsTime）。以上这些时间的单位均为分钟。当然，该表的主键是用户名（username），是引用的用户登录表的外键。这样的设计保证只有用户在注册成功之后才能有相应的数据产生。详细的设计见表 4-2.

表 4-2 用户信息表

字段	类型	可空	字段描述
username	varchar(40)	false	主键，指向用户登录表 username 字段的外键
totalClassTime	integer	true	计划上课时间，默认值空
totalStudyTime	integer	true	计划自习时间，默认值空
totalSportsTime	integer	true	计划运动时间，默认值空
classTime	integer	true	已完成上课时间，默认值空
studyTime	integer	true	已完成自习时间，默认值空
sportsTime	integer	true	已完成运动时间，默认值空

4.2.3 用户好友表

用户好友表主要是为了存储用户和好友的关系，并且实现给好友发送信息的功能。该表的主键是 fid，并且添加自增的属性，用来表示不用的好友关系。之后的两个字段分别是 邀请者（me），被邀请者（friend）。因为考虑到好友的关系并不一定是相互的，存在单方面加为好友的这种情况的出现，所以邀请者字段和被邀请者的字段是允许重复的。在该表后面添加一个消息（message）字段，便可实现给好友发出邀请信息的功能。把消息作为好友关系的一个附属，这样可以实现保存最近一次好友发出的信息，这样做既实现了功能，也减少了多开设一个数据表的开支。详细的设计见表 4-3。

表 4-3 用户好友表

字段	属性	可空	描述
fid	integer	false	自增，主键
me	varchar(40)	false	自身的用户名
friend	varchar(40)	false	好友的用户名
message	varchar(200)	true	存储好友之间的信息，默认空

4.3 系统 UI 设计

软件的 UI 系统是非常重要的一个部分，应本着简单易用的原则，使用户能够在短时间内学会操作系统。

该系统的 UI 设计采用了简洁的风格，在加载东北大学地图的界面，集成了所有的主要功能：首先，为了实现地图的显示和视角的移动，该系统在屏幕的上下左右四个位置分别加载了指向上下左右四个方向的箭头，用户可以在使用这四个箭头直接控制软件中主角的移动，从而完成对事件的触发操控和地图视角的移动；该系统在系统的显眼位置设置了两个按钮，分别是“学习”和“运动”。用户可以点击这两个按钮触发设定计划学习和运动时间的事件；系统的在屏幕的右上角绘制了一个“我的信息”的按钮，以方便用户随时点击按钮查看自己计划的学习运动时间和已经完成的学习运动时间；系统在屏幕的左边安排了一系列的按钮，分别对应添加好友和查看邀请信息的按钮。系统通过设置以上这些按钮菜单，方便用户在短时间内选择要执行的功能。

系统 UI 界面只有一层也是不够的。针对不同的功能，如设定计划学习时间功能或学习自习计时功能，系统采用新增一个 UI 层的方法，使用户还能看见主角在地图中的位置，同时在新的一个 UI 层上进行操作，用户只需要点击每个新弹出层右上角的关闭按钮，便可关闭此弹出层并回到之前操作的界面。

综上所述，本着易用的原则，该系统的 UI 设计方便用户进行自己所需要的操作。

4.4 系统功能设计

4.4.1 用户登录注册模块

用户进入系统时调用的第一个模块。在第一次使用系统时需要向系统注册一个用户，输入用户名和密码进行注册，当系统验证通过时，将返回登录界面，输入账号和密码，之后系统将获取到的用户填的账号密码信息请求服务器进行验证，如果服务器返回值为真，则用户将进入系统主界面，反之则继续停留在登录界面，并且提示登录失败。

4.4.2 学习生活计划功能模块设计

该系统的核心是提供给用户制定学习生活计划，并记录用户完成情况的功能。所以用户在进入系统主界面之后就可以在显眼的地方点击学习/运动的按钮，系统弹出一个新的界面，提示用户输入计划的学习/运动时间，点确定按钮之后，系统将获取的数据存储到本地的数据库中。存储过程结束后，用户可以点击屏幕右上角的“我的信息”按钮，弹出一个新的界面显示该用户设定的计划学习运动时间，和已经完成的学习运动时间。

该功能模块的核心是记录用户完成情况。本系统在地图的每个建筑上设定了触发点，当用户将主角移动到某个建筑上时，可以按下“触发”按钮，系统弹出新的一层显示建筑物的名字并且显示出对应建筑物的事件：如在教学馆的位置按下“触发”按钮，可以在新弹出的一层上看到教学馆的名字，并且出现“开始上课”和“开始自习”的按钮。用户点击开始按钮之后，系统将开始自动计时，直到用户点击“结束”按钮为止。之后系统将记录的时间存入本地数据库的对应用户的对列中。即完成了记录用户完成情况的功能。

4.4.3 添加好友和邀请好友功能

该系统除了本地的计时和计划功能以外，还设计了用户和好友的交互功能。用户可以在系统的主界面点击“好友”的按钮，此时系统会弹出一个新的图层提示用户输入好友的用户名，输入之后点击继续按钮，系统将获取到的好友用户名请求服务器进行验证，等待服务器回应。若服务器端好友用户名已经成功找到，则在好友关系数据表中记录，并且给客户端一个成功的响应。手机客户端接受到成功的响应之后在界面上显示“添加好友成功”的提示。

在添加了好友之后，用户就可以向自己的好友发送邀请信息了。当用户将主角移动某个建筑物前并点击触发按钮，在新弹出的一个图层中点击“邀请好友”按钮，系统接受用户的请求，整合时间、地点、事件等成为一个字符串，并且向服务器发送请求，将处理好的字符串更改到数据库的好友关系表中。服务器收到请求之后开始处理请求，当请求处理完毕之后发送给手机客户端一个成功的响应，客户端收到成功的响应之后，显示“邀请好友成功”的提示。

当用户想查看自己的好友发送给自己的邀请信息时，在系统的主界面点击“信息”的按钮，系统会弹出一个新的图层并向数据库请求获得信息接受人为用户名的邀请信息，当成功获取服务器的响应之后，系统手机客户端的部分会显示接受到的邀请信息。用户可以根据获得的邀请信息找到自己好友所在的位置，一起进行学习/运动的活动。

4.6 服务器设计

该系统在处理用户注册登录功能以及好友模块功能时需要与服务器进行信息的交互。所以该系统设计 Java Servlet+Tomcat 的基于 MVC 框架的服务器，用以实现相应的功能，MVC 全名是 Model View Controller，是模型(model)，视图(view)，控制器(controller)的缩写。该框架的目的是将业务逻辑、数据、界面显示分离，从而实现在改进用户交互界面的同时，不需要重新编写业务逻辑。同时也简化了分组开发，给不同的开发人员同时开发视图、控制器和业务逻辑提供了可能。

4.6.1 实体层设计

实体(Entity)层中分别将不同的数据表：用户表，好友关系表封装成不同的对象，并且加入对应属性的 get/set 方法。是服务器层次中最底层，也是最接近数据库的一层，目的是为了实现在数据访问层服务层对数据表中的数据进行操作：在把属性项封装成 java 对象之后，对数据的操作可以通过对类的操作去实现。

4.6.2 数据访问层设计

数据访问（DAO）层是基于 SQL 语句对数据库中数据进行操作的一层，是数据处理的核心层，它处于实体层和服务层的中间，提供了数据操作传递中介的作用。针对不同的实体层，该系统设计了不同的数据访问层。在用户登录的数据访问层中，定义了添加用户，修改用户，检查用户的账号密码是否正确的操作；在好友信息的数据访问层中，定义了添加好友关系，修改好友信息的功能。这些操作均作为各自数据访问层的成员函数，这样只有数据访问层实现了对数据库的连接和使用，服务层就可以在服务层中直接调用数据访问对象进行不同功能的操作而不需要对数据库数据进行操作。

4.6.3 服务层设计

服务（Servlet）层是为了处理 HTTP 请求和响应的一层，是服务器的顶层，也是直接和客户端程序进行交互的一层。针对客户端的不同的功能请求，服务层设计了不同的类处理请求：注册新用户服务，用户登录验证服务，查看用户信息服务，更改用户信息服务，添加好友服务，发送邀请信息服务，查看好友邀请信息服务。每个服务类针对 HTTP 协议中的 get 和 post 方法进行了处理。当客户端发送 request 给服务器时，服务层解析 request 中的数据、解码，之后调用数据访问层中的相关操作进行处理，处理完毕之后封装 response 返回给客户端，完成一次请求的处理和响应。

4.7 本章小结

本章介绍了系统的总体设计，之后介绍了该系统数据库的设计。再根据客户端程序和服务器程序分成两个部分进行了系统的设计介绍：在客户端部分介绍了系统的 UI 设计，包括了每个界面与用户的交互的设计和业务逻辑跳转的设计，之后阐述了客户端部分系统的详细功能设计，包括了用户注册登录模块，学习生活计划功能模块和好友交互功能模块。最后介绍了服务器的设计，包括 MVC 模式的分层设计，实体层、数据访问层和服务层的功能和设计。

第五章系统实现

根据系统的需求和设计，本章对系统中相关部分的实现进行了详细的介绍。

5.1 地图的制作和显示

在本系统中，加载了东北大学的鸟瞰视角地图。东北大学的鸟瞰地图是由 Tiled Map Editor 制作的 TMX 类型地图。Tiled Map 可以支持鸟瞰视角地图、等距斜视地图（斜 45° 地图）和六边形地图。

在 Tiled Map Editor 中新建一个长度和宽度分别为 80 和 50 图块的地图，之后添加图块素材，本系统中的地图使用 32 像素*32 像素的图块作为素材。每个像素图块长宽属性相同，但是图块的内容均不同。添加图块素材之后，将图层命名为 **bkground**，表示地图总体的背景，在此背景图层上利用图块素材拖拽画出东北大学的主要道路。之后新建一个 **buildings** 的图层，在此图层上利用图块素材拖拽画出东北大学的各个主要建筑。完毕之后新建一个 **events** 图层，在地图上加上事件触发的地点标志。分层设计和制作地图有利于有序的管理地图信息，并且方便进行后续的事件处理。

保存并导出 TMX 格式的地图，就可以在 Cocos2d-x 引擎中使用了。以下列出 TMX 地图类的常用函数。

表 5-1-1 瓦片地图集类 TMXTiledMap 的常用函数

函数名	返回类型	介绍
getMapSize	尺寸	获得地图尺寸，单位为像素格
setMapSize	空	设置地图尺寸，单位为图素格
getTileSize	尺寸	获得图素格尺寸
setTileSize	空	设置图素格尺寸
getObjectGroups	数组	获得对象数组
setObjectGroups	空	设置对象数组
getProperties	值	获得地图属性
setProperties	空	设置键值对数组
layerNamed	键值对集合	根据层次名称获得层次
objectGroupNamed	对象组	根据对象组名获得对象组
propertyNamed	键值对集合	根据属性名获得属性值
propertiesForGID	键值对集合	根据 GID 获得键值对集合
getObjectGroup	地图对象组	返回特殊层的地图对象组

表 5-1-2TMXLayer 类的常用函数

函数名	返回类型	介绍
releaseMap	空	删除地图
tileAt	精灵类	根据图素横纵坐标索引获得图素精灵
tileGIDAt	整型	根据图素横纵坐标获得素图的 GID 值
setTileGID	空	设置图素横纵坐标上的图素 GID
removeTileAt	空	根据图素横纵坐标索引删除该图素
positionAt	点坐标	根据图素横纵坐标索引获得图素位置
setupTiles	空	创建设置图素
propertyNamed	字符串	根据属性名获得属性值

将地图资源放入项目的 res 文件中，使用以下代码在程序中显示制作好的地图资源：`_tileMap=TMXTiledMap::create("NeuMap.tmx"); this->addChild(_tileMap);`因为地图的长宽像素远大于手机屏幕的像素，所以需要移动去显示全部地图的功能。在 Cocos2d-X 引擎中，TMXTiledMap 类可以作为程序中的精灵（Sprite）类操作。所以通过 `this->setPosition()` 的成员函数可以实现移动地图的效果，其中函数的参数单位为像素。如果不进行位置属性的设置，在手机屏幕上将会显示地图的左下角部分。

5.2 角色的移动和碰撞检测

该系统是以主角为中心视角的，用过操作角色进行移动获取整个地图的信息。接下来介绍地图和主角跟随命令进行移动功能的实现：

根据系统的需求，当用户点击上下左右四个按钮时，角色能够对应的进行上下左右的移动操作，而当角色的位置已经在屏幕的边界的时候，用户再下达移动命令的时候，则需要判断角色是否会移动出边界，如果将要移动出屏幕边界时，则不移动角色的位置，转而在相反位置移动地图，通过相对的位移产生继续移动角色的效果，算法的流程图如图所示：

`Sprite* player = Sprite::create("Player.png"); _tileMap->addChild(player);`此代码可以把主角的图片初始化为一个精灵，并作为一个子元素加载到地图之上。精灵类具有获得当前位置的成员函数，使用 `getPosition()` 可以得出精灵所在的当前位置的坐标。为了方便角色和地图的移动，需要先把精灵位置坐标的单位转化为地图图块的坐标：在 Cocos2d-X 中，坐标系原点在屏幕的左下角，从原点向右作为 X 轴的正半轴，从原点向上作为 Y 轴的正半轴。而在 Tiled Map 系统中坐标的原点定在屏幕左上角，原点向右作为 X 轴的正半轴，原点向下作为 Y 轴的正半

轴，对比如图 5-2 所示：

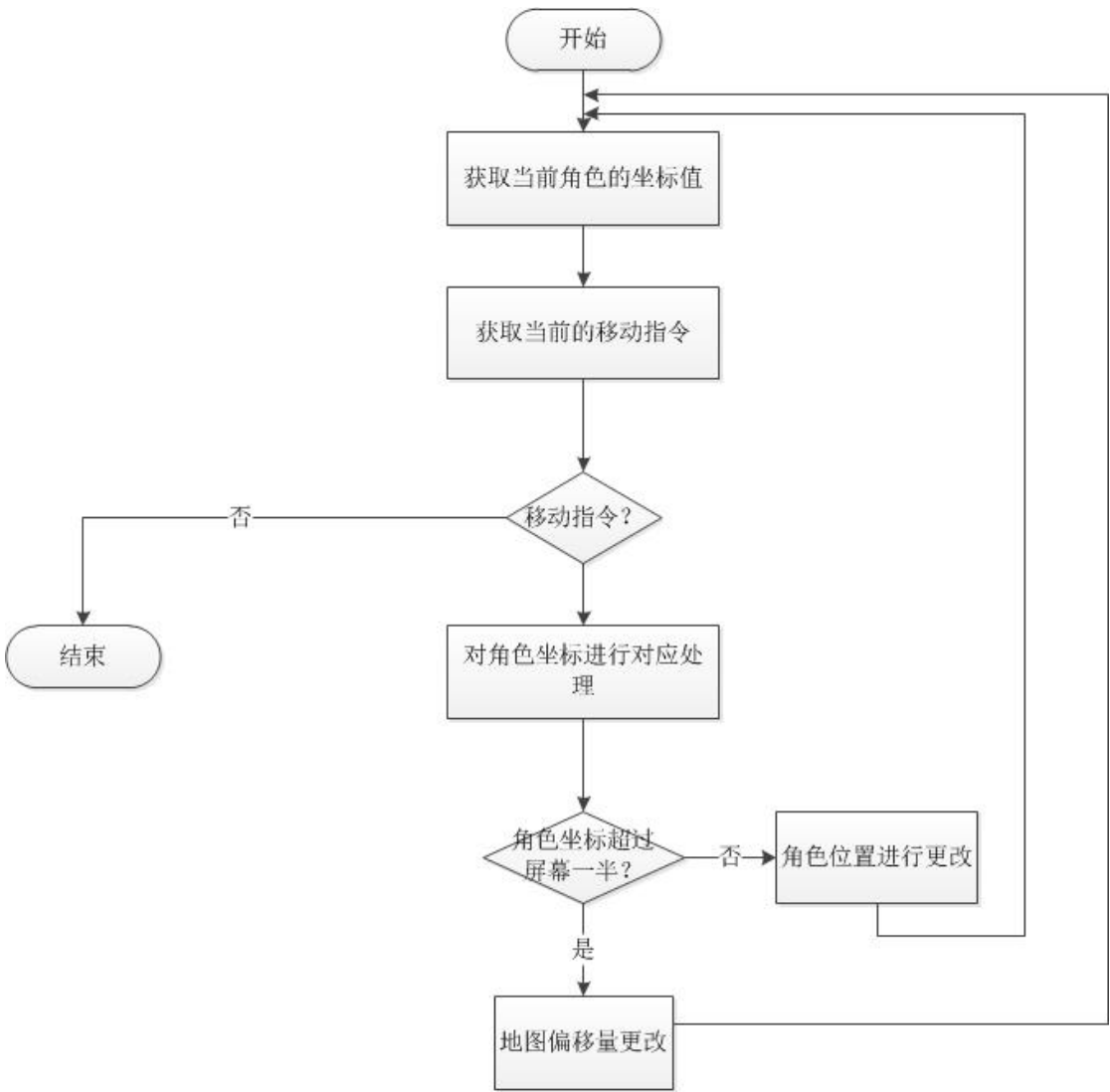


图 5-1 地图移动算法流程图

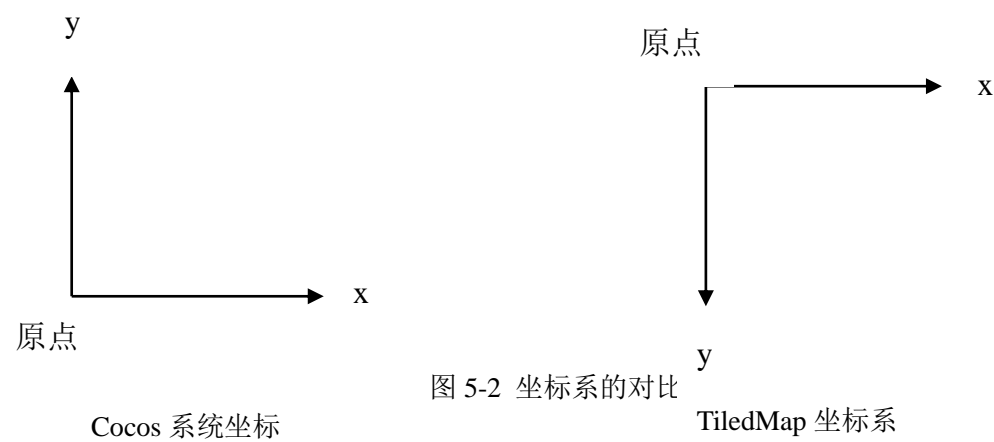


图 5-2 坐标系的对比

所以在进行移动之前，需要进行坐标的转化。由图可知，两种坐标系的 x 轴定义是一样的，y 轴的方向正好相反，所以在进行坐标转化时只要把角色 x 轴的位置取出并且除以图块的像素大小，就可以得出角色在地图中对应的 x 轴坐标。

而 y 轴坐标用整个地图 y 轴图块的数量减去 1 再减去角色当前的 y 轴坐标和图块大小的商即完成了坐标的转化。公式表达为： $(\text{int})\text{Player} \rightarrow \text{getPositionX}() / \text{MAPTILESIZE}, \text{map} \rightarrow \text{getMapSize}().\text{height} - 1 - (\text{int})(\text{Player} \rightarrow \text{getPositionY}() / \text{MAPTILESIZE})$;

完成角色的移动控制功能之后，系统还需要完成碰撞检测的功能。Cocos2d-X 中对于 TMX 地图封装的成员函数里包括了 `getTileGIDAt()` 这样的函数。该函数的参数是一个坐标值，返回值是整形的数值，功能是返回参数坐标值对应的瓦片地图块的 GID 值。在制作地图的时候，把地图背景和建筑分离成两个图层，这样在做碰撞处理的时候，使用 `map->getLayer("buildings")` 函数获得建筑层的相关信息，因为在建筑层上放置图层之后，图块信息都会有一个非零 GID 的值，而道路图块因为不在建筑图层之上，所以 GID 值为 0。在做碰撞检测之前使用 `getTileGIDAt()` 判断主角的下一个位置 GID 是否为零，如果为零则说明没有建筑物的遮挡，可以通过。如果非零则说明下一个位置有建筑物的遮挡，主角不可以通过。以下是该算法的流程图。

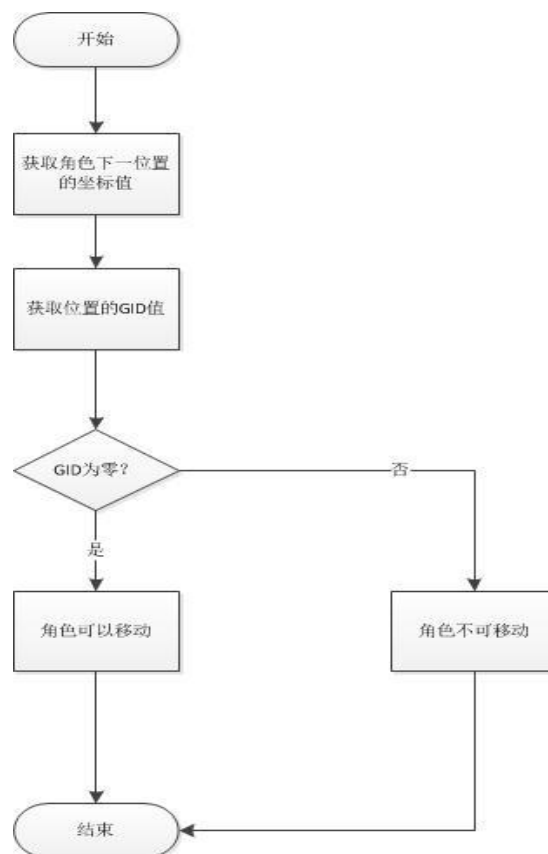


图 5-2-2 碰撞检测算法流程图

5.3 客户端界面的设计与实现

5.3.1 系统开发环境的语言、搭建和配置

该系统客户端采用苹果公司开发的 Xcode 作为编译器，因为在 Xcode 中没有内嵌 Cocos2d-X 3.0 中的模板，所以需要使用 Cocos 3.2 资源包中的 python 安装文件创建新项目，在命令行中 cd 到 Cocos 资源包下的 bin 目录中，使用 `python new cocos -p project -l cpp` 创建一个名称为 project 的，编译语言为 C++ 的 Cocos2d-x 项目。找到创建项目的根目录之后，点击生成的 Xcode 项目文件即可打开刚刚新建的空白项目。空白项目中含有 AppDelegate 类文件和一个 HelloScene 类文件。在 AppDelegate 类文件中自动声明了程序的导演类和入口场景，在使用命令行方式创建的程序中入口场景默认为 HelloScene 类。可以在 AppDelegate 中的 `director->runWithScene(scene);` 这句话中对 scene 进行更改从而改变入口程序。

系统服务器端采用 MyEclipse+Tomcat+Mysql 搭建。下载 Mac 版的 MyEclipse 和 Tomcat 7 分别进行安装，并且在 MyEclipse 中的 preference 中设置 Tomcat 的安装路径。下载 Mac 版本的 Mysql 进行安装，并且配置环境变量。因为在 Mac OS 中中文编码是 GBK 模式，所以需要在 MyEclipse、Tomcat 和 Mysql 配置中设置中文编码集为 UTF-8，避免出现中文编码不同而出现的乱码问题。之后就可以通过 MyEclipse 中启动 Tomcat 部署本地的服务器。

至此客户端和服务端端的配置环境已经搭建完毕，可以在此环境上进行系统的开发和编码。

5.3.2 UI 控件的实现

Cocos 引擎程序中的 UI 控件由两个方式实现：通过在源代码中编写和在 CocoStudio 中编辑。通过代码编辑的 UI 控件具有定位准确，设置属性方便，与事件触发代码更容易整合的优点。而用 CocoStudio 编辑 UI 控件可以省去一些设置属性和声明的代码，并且不需运行就可以看到控件的位置和效果，方便开发者进行快速的 UI 设计。该系统采用两者方式结合编写的编写方式，充分利用两种设计方法的优点。

首先设计的是用户登录注册的界面。新建 EntryScene 的 C++ 类，并在类的初始化方法中新建了一个 Scene 类，并在 Scene 类中添加 Layer 类用以显示 UI 的控件。其他的 UI 控件选择在 CocoStudio 中制作，打开 CocoStudio，并新建一个 Layer 层，之后将背景图片（东北大学北门）作为一个 Sprite 类放置在屏幕的中间位置，之后在背景上加入两个 EditTextField 类，作为用户输入账号和密码的

UI 控件。之后加入两个 Button 类控件，将控件的文字分别更改为登录和注册。这样用户登录注册的页面 UI 设计基本完成。将 UI 界面作为 csb 类型导出资源，并放置在 Xcode 项目文件中的资源文件中。在 EntryScene 类的实现方法中使用 CSLoader::createNode()方法，可以把 csb 资源文件作为一个 Node 类导入到程序中使用。在 csb 资源文件中的 UI 控件可以通过 getChildByName()函数取出对应的对象。考虑到添加的两个 Button 类控件需要添加触摸监听，在该系统中使用 addTouchListener()的函数注册监听事件，此函数的参数即绑定的事件触发的回调函数。添加完监听事件之后，用户登录注册界面的 UI 设计完成。

之后设计的是系统的主界面，此界面因为需要加载学校的地图资源文件，考虑到兼容的问题，所以不使用 CocoStudio 创建 UI 界面。在加载完地图文件之后，需要加载与用户交互的 UI 控件，拿用户点击操作的上、下、左、右按钮为例，该系统使用 Menu 类作为容器统一管理。每个单独的按钮则使用 MenuItemImage 类的 create 方法制定图片和点击之后的回调函数创建。其中向上按钮的代码实现如下：
`MenuItemImage * pTopItem = MenuItemImage::create("top.png", "top_press.png", this, menu_selector(NeuMap::menuCallback));`

构建菜单项完成之后，使用 Menu 的 create 函数将菜单项添加到菜单容器之中，之后在 Layer 类中使用 addChild 函数将菜单容器添加至程序页面上。最后需要使用每个菜单项从 Sprite 类中继承下来的 setPosition()成员函数设定菜单项在屏幕上的显示位置。

系统主界面设计完之后，还需要设计各个弹出层的 UI 控件。考虑到弹出层的 UI 比较简单，并且重复性比较高，采用 CocoStudio 工具辅助设计。实现的步骤与实现用户登录注册界面的基本相同，需要打开 CocoStudio 软件并根据每个弹出层的实际设计需求选择新建场景类或图层类，在新建的图层上拖拽 UI 控件，并给它们分别命名 ID，保存之后导出 CSB 格式的资源文件并将其放入项目的资源文件夹中，即可在程序中使用代码加载已经设计好的 UI 界面。图 5-3 为使用 CocoStudio 构建 UI 界面的流程图。

根据以上的流程分别构建设定学习界面图层、设定运动界面图层、查看信息图层、教学馆图层、运动场图层、图书馆图层、添加好友图层之后，整个系统的 UI 界面实现工作完成。

5.3.3 UI 界面跳转的实现

在该系统中，用户经常需要 UI 界面的跳转，例如用户点击“我的信息”按钮之后弹出一个新的 UI 层，显示出关于用户的信息；用户点击“学习”按钮之

后弹出一个新的 UI 层，允许用户对时间进行设置等等。在 Cocos2d-X 引擎中，使用单例模式的导演类控制场景的切换，而一个场景可以控制自己层次的切换。

在用户登录注册界面中，用户成功登录之后会进入程序的主界面，此时先使用 `Director::getInstance()` 获取导演类的单例。之后调用成员函数：`replaceScene()` 进行切换场景功能的实现，其中的参数为即将要运行的场景类的构造函数。



图 5-3 构建 UI 界面流程图

`replaceScene()`的实现的功能是完全取代当前场景，将程序转入一个新的场景。但是当用户点击“学习”或者“运动”按钮时，系统应弹出一个新的 UI 层，而背景仍应是系统的主界面。所以为了实现这样的功能，该系统采用了在原来的层上添加一个新层的办法。并且通过添加触碰等级高的监听事件，实现弹出层的模态化，即弹出一层之后不允许用户点击别的层上的触碰事件。实现功能的关键代码为 `eventDispatcher->addEventListenerWithSceneGraphPriority(listener, stuLayer)`。

5.4 用户登录注册功能模块的实现

用户使用系统之前需要进行登录的操作，第一次使用系统的用户则需要先进行注册，注册成功之后才可以登录进入系统主界面。用户进入系统的入口程序按照提示输入账号和密码之后，点击“注册”按钮，会触发注册的事件，系统获取用户输入的账号和密码并向服务器端请求注册的服务，等待服务器响应，并根据服务器响应提示用户注册成功或者失败。图 5-4-1 为用户注册功能实现的流程图。

其中当用户点击“注册”按钮之后，触发注册在按钮上的监听事件，系统获取用户输入的信息，进行 HTTP 传输协议的封装，以 post 方式向服务器请求 AddUserServlet 服务。当服务器端收到请求之后，在 Servlet 层进行 request 的解析，包括中文的转码，参数的设置。之后调用 DAO 层中的 addUser()成员函数，在 DAO 层中使用 preparedStatement 封装 insert 的 sql 语句，完成对服务器端数据库的操作。完成插入的操作之后，由 DAO 层向 Servlet 层和客户端逐层传递返回值。在客户端收到 HTTP 的相应之后，开始调用在请求发送时定义的回调函数 onHttpRequestCompleted()，在此函数中进行对服务器返回的响应进行处理，若解析出的服务器返回值为 1，则在本地创建一个以注册时输入的账号为用户名的个人档案数据项，否则在界面显示注册失败。

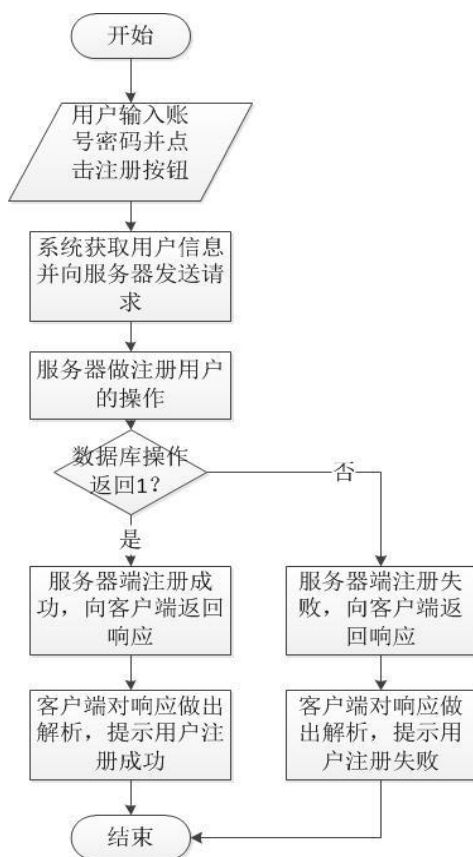


图 5-4-1 注册用户流程图

用户注册完毕之后可以在原界面输入账号和密码进行登录，系统获取用户输入的信息之后向服务器发送登录请求，若服务器返回登录成功则跳转至系统主界面，反之则提示登录失败，继续停留在登录界面。图 5-4-2 为用户登录流程图。

其中用户登录的实现和用户注册模块的实现基本相同，不同的在于封装好参数之后请求服务器端的 SearchUserServlet 服务，另外在客户端解析服务器返回值成功之后，将已成功登录的用户名封装成一个键值对存储在本地，供系统的其他场景获得当前登录的用户名。

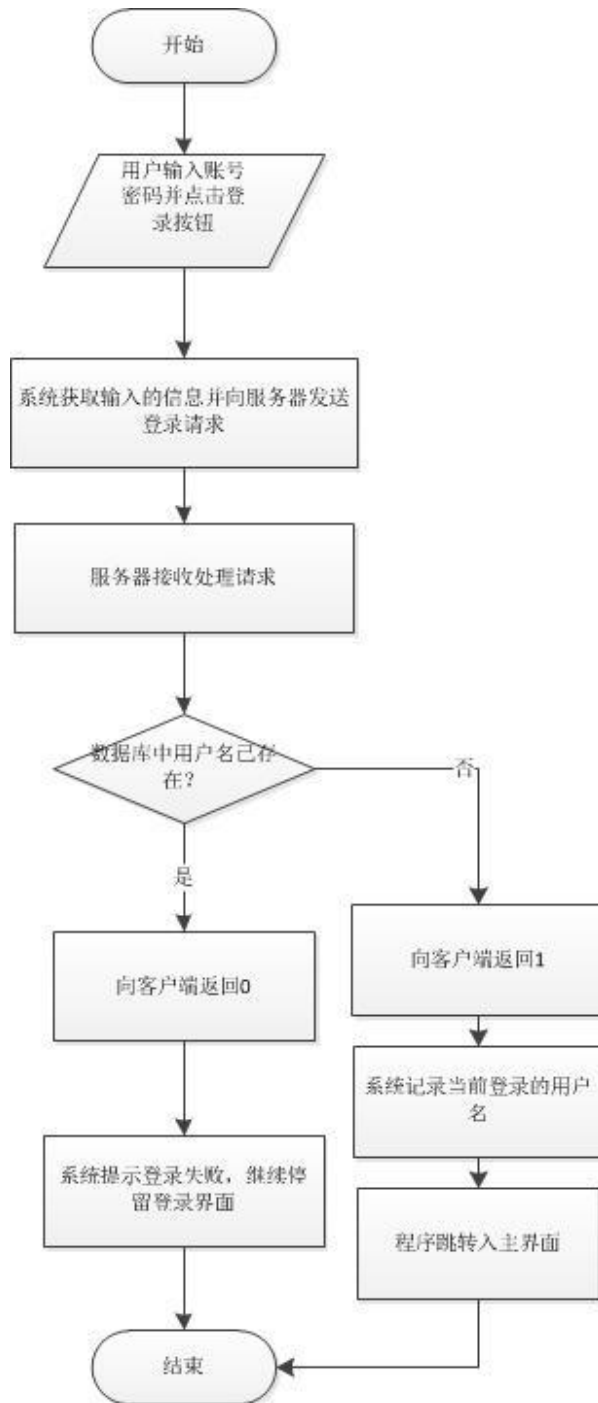


图 5-4-2 用户登录流程图

5.5 制定计划和完成功能的实现

已登录的用户可以通过点击系统主界面的“学习”和“运动”按钮实现制定学习和运动计划的功能。当用户点击按钮之后会触发 `eventListener` 注册的监听事件，并选择对应的回调函数进行处理。在回调事件里调用新的 UI 层并将新的层添加到场景的最上层进行处理。当用户在输入框输入时间数字并点击确认之后，会触发确认按钮的回调函数，建立与本地数据库的连接，系统开始找寻匹配当前用户名的数据库信息，之后进行数据库的存储工作。图 5-5-1 是制定计划的功能流程图。

其中在用户输入计划时间时，控件 `TextField` 只能获取到 `String` 类型的值，但在数据库中要求以数字的形式存进去。系统利用 C++ 的 `iostream` 类，定义了 `StringToNum()` 函数解决 `String` 转化为 `int` 类型的问题；在连接本地数据方面，该系统自定义了 `SqlLite3Util` 类实现了数据库的连接，增、添、改、查等基本功能。在需要进行数据库操作的时候，调用 `SqlLite3Util` 的类方法就可以完成相应的操作。

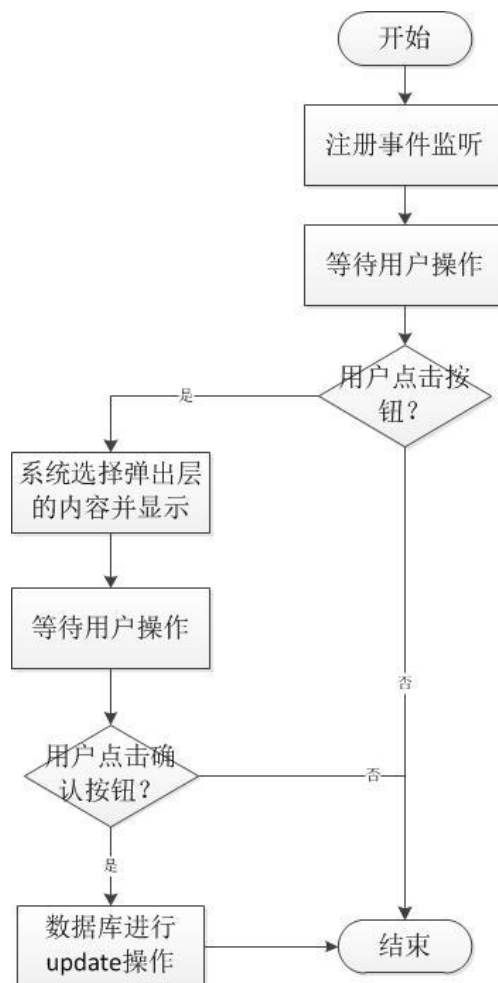


图 5-5-1 制定计划功能流程图

当用户制定计划之后便可以记录自己的完成情况。用户将角色的位置移动到相应的建筑物的出发点位置，点击“触发”按钮，便可以触发对应建筑的事件，用户可以根据 UI 界面的提示选择开始某项活动，点击之后系统开始自动计时，直到用户点击“结束”按钮之后，系统开始统计用户花费的时间，并且将数字分类的存储在用户对应的数据库之中。图 5-5-2 是活动完成情况功能的流程图。

其中在用户点击“开始”按钮之后，系统使用 `Text` 类的成员方法 `setText()` 将按钮的文字改为“结束”，用户可以不用点击新的按钮，而在原本的按钮位置进行操作即可。关于系统计时功能的实现，系统调用了 `time.h` 头文件，在程序中使用 `localtime()` 方法获得当前系统的时间，之后定义秒、分、时的变量存储取出的系统时间，当用户点击“结束”按钮之后，该系统将两次获得的时间进行对比取差，得出用户进行活动所花费的时间。最后调用 `SqlLite3Util` 中的 `updateDataFromDBBase()` 完成对数据库的更改操作。

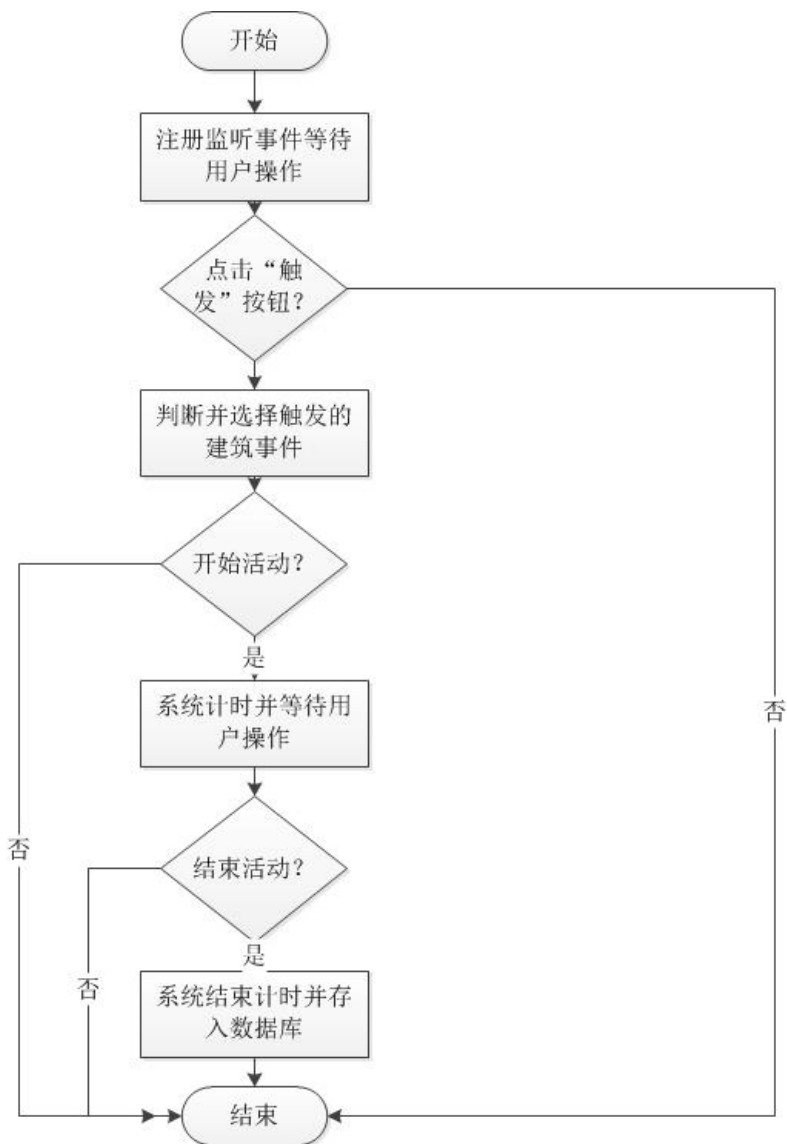


图 5-5-2 完成计划功能流程图

5.6 好友功能模块实现

该系统中好友模块的功能包括添加好友、给好友发送邀请信息以及查看好友信息的功能。以下将分别介绍其实现过程。

5.6.1 添加好友功能实现

已登录的用户可以添加好友并且邀请好友与自己一起进行计划的完成，以达到相互督促的效果。当用户选择添加好友的功能之后，可以根据系统的提示输入好友的用户名，点击“确认”按钮之后，系统根据获得的用户名向服务器请求添加好友的服务，根据服务器的返回值提示添加好友成功或者失败。图 5-5-1 是添加好友功能的流程图：

其中客户端系统会将请求的参数封装在 HTTP 的协议中，通过 post 的方法向服务器请求 AddFriendServlet 服务。服务器通过在 Servlet 层中的 doPost 方法获得 request 中的参数，考虑到在 HTTP 中传输的字符编码为 ISO8891 编码，所以在服务器中采用 getBytes 方法进行转码，将传输的字符改为 UTF8 格式的编码。之后调用 DAO 层定义的 addFriend()成员函数，在 DAO 层中使用 preparedStatement 封装 insert 的 sql 语句，完成对服务器端数据库的操作。完成插入的操作之后，由 DAO 层向 Servlet 层和客户端逐层传递返回值。在客户端收到 HTTP 的相应之后，开始调用在请求发送时定义的回调函数 onHttpRequestCompleted()，在此函数中进行对服务器返回的响应进行处理，包括分辨不同响应和返回值的字符解析等，做完一系列的解析工作之后，在客户端上提示用户操作成功。

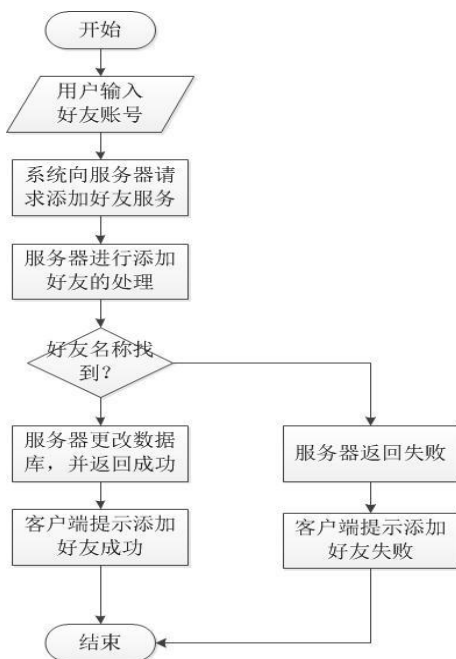


图 5-6-1 添加好友功能流程图

5.6.2 发送好友邀请信息和查看信息功能

用户在添加好友之后可以向好友发送邀请信息，邀请好友与自己一起进行学习或者运动活动。用户将角色的位置移动到特定的建筑物前，并且点击“触发”按钮之后，在新的弹出层之上点击“邀请好友”的按钮，系统向服务器请求发送好友邀请信息的服务，服务器端接受到请求之后做出处理，向用户所有的好友发出邀请信息。图 5-6-2 为发送邀请信息的功能流程图。

客户端在接受用户的操作之后会将请求的参数以 HTTP 的 post 格式向服务器请求 SendMessageServlet 服务。服务器端通过 Servlet 层中的 doPost 方法获取 request 的参数，考虑到传输的过程中牵涉到中文的编码问题，在接收到参数之后需要把 HTTP 传输时采用的 ISO8891 编码转化成 UTF8 的编码。之后服务器 Servlet 层调用对应的 DAO 层中的 sendMessage()函数，在 DAO 层中完成对数据库的连接和 update 的 sql 语句的执行。完成数据库的更新操作之后由 DAO 层开始逐层往回传递返回值，客户端部分在接收到服务器传回的响应之后会调用发送相应 post 时定义的回调函数 onHttpRequestCompleted()，并在此函数中处理不同响应的分辨和返回值字符串的解析处理，最后客户端根据解析的结果提示用户是否操作成功。

用户在有好友给自己发送邀请信息之后，可以在系统的主界面中点击“我的信息”按钮，显示自己收到的所有好友发送的邀请信息。系统会将当前用户名的账号作为参数打包向服务器请求 GetMessageServlet 服务，在客户端的回调函数 onHttpRequestCompleted()中解析和分解服务器返回的字符串，并在客户端中显示用户收到的邀请信息。

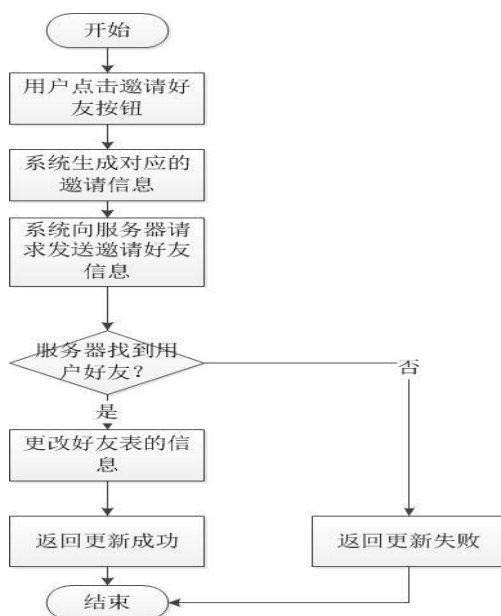


图 5-6-2 发送邀请信息流程图

5.7 本章小结

本章介绍了系统的实现。由于本系统实现分为客户端和 Web 服务器端两个部分，所以在本章开始介绍了在客户端部分实现 TMX 地图的制作和显示，然后描述了系统如何实现地图的视角如何跟随角色的移动和碰撞检测的过程。鉴于本系统是在 Cocos2d-x 引擎平台上开发，所以跟 object-c 编写的 UI 界面过程不同，本章中间部分着重介绍了系统 UI 界面的制作和跳转的实现过程。

介绍完客户端部分之后，本章重点介绍了客户端和服务端交互的功能部分，先描述了用户登录和注册的业务逻辑以及过程中关键的代码实现，在实现用户登录和注册之后，本章介绍了制定计划和完成计划的功能模块的实现，在此部分中介绍了系统如何与服务器之间进行通信，并且详细的介绍了服务器端的业务逻辑处理过程。本章最后介绍了好友功能模块的实现，包括添加好友、发送好友信息、查看自己接受的信息功能。按照从客户端处理到服务器端数据操作的顺序详细的介绍了实现过程。

第六章系统测试

由于软件开发复杂性的原因，软件开发过程中不可避免的会产生各种错误。为了在软件开发的顺利完成以及在软件投入实际运营之前，尽可能多地发现软件当中的错误，需要尽可能全面地对软件进行测试。目前软件测试仍然是保证软件质量的关键步骤，它是对软件需求分析、设计以及编码的最后的检查。

对应于软件开发的各个过程，软件测试也分为几种不同的层次。每一个层次都是下一层次的继续。

(1)单元测试，单元测试对应于详细设计部分，从软件的详细设计中导出。软件的独立单元将在与程序的其他部分相隔离的情况下进行测试。

(2)集成测试，集成测试对应于系统的概要设计，主要用来的是各个模块之间的接口的正确性。

(3)系统测试，系统测试的目的是通过测试发现是否需求分析中的功能相符。

6.1 单元测试

单元测试是在软件开发过程中要进行的最低级别的测试活动，在单元测试活动中，软件的独立单元将在与程序的其他部分相隔离的情况下进行测试。

本系统中的服务器部分采用 JUnit 对每个类进行单元测试。JUnit 是由 Erich Gamma 和 Kent Beck 编写的一个回归测试框架。JUnit 测试属于白盒测试，因为程序员知道被测试的软件如何完成功能和完成什么样的功能。JUnit 是一套框架，实际应用中只要继承 TestCase 类，就可以用 Junit 进行自动测试了。它包括以下特性：用于测试期望结果的断言（Assertion）、用于共享共同测试数据的测试工具、用于方便的组织和运行测试的测试套件、图形和文本的测试运行器。

在 MyEclipse 中的 java build path 下的 libraries 中加上 JUnit 就可以使用了，在项目中专门设置测试包，用来保存测试类。测试类都是继承自 TestCase，运行顺序为：测试开始，setUp 方法，testXXX 方法，tearDown 方法，测试结束，其中 setUp 与 tearDown，这两个函数是 JUnit 框架中提供初始化和反初始化每个测试方法的。setUp 在每个测试方法调用前被调用，负责初始化测试方法所需要的测试环境；tearDown 在每个测试方法被调用之后被调用，负责撤销测试环境。testXXX 方法是用来测试相关类的自定义方法。

在 MyEclipse 中运行指定的测试用例，会自动显示出各个测试方法的结果，对于没有通过的用例给出错误代码。在单元测试的中，发现了一些问题。其中比

较重要的问题有中文在传输的过程中出现的乱码问题。以及数据库操作中出现的 Null 指针异常的问题。

解决方法：针对中文乱码的问题，由于 mac 系统、HTTP 传输、Myeclipse 和 Tomcat 以及 Mysql 默认的字符集都不一样，所以导致问题的出现，分别在各个环境中设置中文的编码集可以解决一部分问题，因为 HTTP 传输过程中不依赖任何开发环境，所以必须在 Myeclipse 中的 java 代码中对传输过来的参数进行编码的转化处理才能完全解决问题；为了避免空指针异常的问题，在直接调用 DAO 层进行数据库操作之前进行初始化和赋值，从而解决了异常导致的不能操作的问题。

本系统的客户端采用对每一个 C++ 类文件进行测试的方法进行单元测试，通过更改 AppDelegate 文件中的入口程序，可以选择让程序直接进入某一个类文件进行测试。同时在类的实现文件中适当位置加入 Log 函数，可以在程序的输出窗口看到希望进行测试参数的值。这样通过对类文件逐一的测试，完成对客户端部分的单元测试。

6.2 集成测试

集成测试，也叫组装测试或联合测试。它是在单元测试的基础上，将所有模块按照设计要求组装成为子系统或系统，进行的测试。集成测试应该考虑以下问题有：数据在模块接口之间传递时是否会丢失；各个子功能组合起来，能否达到预计要求的父功能；模块之间是否会有不良的影响；单个模块中的误差是否会在其他模块中进行放大。合理地组织集成测试，直接影响到模块测试用例的形式、所用测试工具的类型、模块编号和测试的次序、生成测试用例和调试的费用。通常，有两种不同的组装方式：一次性组装方式和增量式组装方式。

本系统对生成邀请信息和发送邀请信息两个子模块也进行了集成测试，发现生成邀请信息在任何情况下可以正常运行，而发送邀请信息功能并不成功。经过对代码的分析之后发现，因为发送邀请信息的功能传递的信息参数格式和生成邀请的信息格式不一致，导致在数据库操作中发生了参数不足不能操作的情况。在统一了信息格式之后，问题解决。

6.3 系统测试

系统测试是将已经确认的软件、计算机硬件、外设、网络等其他元素结合在一起，进行信息系统的各种组装测试和确认测试，其目的是通过与系统的需求相比较，发现所开发的系统与用户需求不符或矛盾的地方。

因为 IOS 设备的价格偏高，所以系统测试时采用 Xcode 中自带的 Iphone 模拟器进行测试。这种情况下可以模拟用户基本所有的操作，并且可以顺利的和服务器进行通信测试，其测试内容如表 6-3 所示。

表 6-3 系统测试记录

操作名称	操作流程	操作结果和输出
注册系统	打开系统，输入账号密码点击注册按钮	点击按钮到系统提示操作完成花费大约 2 秒，基本满足系统的要求
登录系统	打开系统，输入账号密码点击登录按钮	点击按钮到系统进入主界面花费大约 3 秒，基本满足系统的要求
移动角色	用户点击移动按钮	角色的移动在用户点击之后 0.5s 左右执行，满足要求
碰撞检测	用户移动角色到建筑附近	用户不能将角色移动到建筑之中，碰撞检测满足要求
设定计划	用户点击设定按钮，输入计划的学习/运动时间	系统将用户设定的时间更新至数据库
完成计划	用户将角色移动到建筑旁，点击开始按钮	系统记录用户花费的时间，统计的时间误差保持在 1 分钟以内，满足要求
添加好友	用户点击添加好友按钮，输入好友用户名	从用户操作到系统提示操作完成花费 2 秒左右，满足要求
查看信息	用户点击查看信息按钮	在界面上显示用户收到的消息，总花费 1 秒左右
发送邀请	用户将角色移动到建筑旁，点击邀请好友按钮	系统向用户所有好友发送邀请信息，总花费 1 秒左右

打开系统程序，在第一个显示的界面输入错误的账号和密码，会在原页面上提示登录失败如图 6-1。



图 6-1 用户登录失败运行截图

输入正确的账号和密码，会进入程序的主界面，如图 6-2，说明用户登录模

块功能正常。

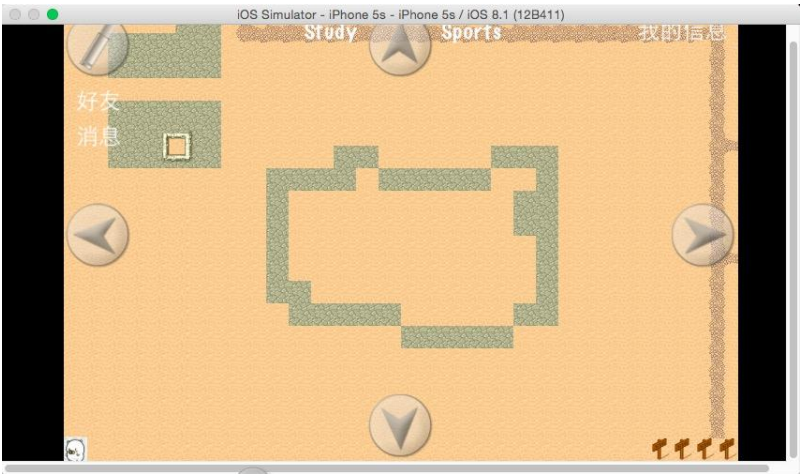


图 6-2 系统运行主界面

用户在主界面点击“我的信息”按钮，可以看见弹出的界面上显示用户设定的计划时间和完成的时间百分比，如图 6-3 显示，若用户第一次登录，则均显示为 0。



图 6-3 显示用户计划信息

用户点击“学习”按钮，设定计划学习的时间；点击“运动”按钮，设定计划运动时间。设定之后再次点击“我的信息”按钮，可以看见之前设定的时间已经显示在个人的信息之中。说明设定计划和查看信息模块正常。

之后用户将角色移动到“教学馆”的位置，并且点击开始“开始自习”，如图 6-4 所示。过 5 分钟之后点击“结束”按钮，再次点击“我的信息”按钮，可以看到自习的时间已经加了 5 分钟，说明完成计划模块正常。

用户再点击“添加好友”功能，输入好友的用户名，继续在“教学馆”前点击邀请好友。更换之前新添加的好友的用户名进行登录，可以看到消息之中有用户刚刚发送来的邀请信息，如图 6-5 所示。说明添加好友和发送邀请信息功能模块正常。



图 6-4 教学馆事件触发



图 6-5 系统消息中心

6.4 本章小结

本章主要介绍了软件测试的知识和对系统进行测试的信息。随着系统的开发，分别对系统进行了单元测试、集成测试和系统测试。并借助表格和截图阐述了系统对实现的各个功能做出的测试。以及对测试过程中出现的问题进行了分析和解决。

第七章结论

7.1 本文工作总结

本文对系统以用例建模的方式进行了功能性的需求分析，明确了系统需要实现的各个功能，并且根据功能需求将系统分成客户端和服务端两个部分。之后根据实际情况补充了非功能性需求。之后对系统的功能进行了总体的设计，包括数据库设计，系统 UI 设计、系统功能设计和服务器设计四个方面。紧接着根据实际情况对地图的制作、角色的移动、客户端界面以及各个功能模块分别对客户端和服务端进行了具体的实现。最后对整个系统分成单元测试、集成测试、和系统测试三个模块进行了测试。通过测试检验了系统实际的功能实现情况和运行效果。

本文的工作主要包括下面几个方面：

（1）先简单介绍了 Cocos2d-X 引擎相关内容的特点和基本概念。包括 Cocos 的发展历史、Cocos2d-X 引擎的特点、Cocos2d-X 引擎中的核心基本类。分析了这些核心基本类的使用方法。之后介绍了 CocoStudio 的特点和简单使用方法。最后介绍了选择的数据库软件、Tomcat 和 Servlet 的基本概念知识和特点。

（2）利用用例建模的方法获取并分析了系统的功能性需求，并根据实际情况补充部分需求，提出了系统的非功能性需求。描述了系统开发所需要的服务器和客户端的软件支持，并说明了采用的开发工具。

（3）介绍了系统的总体设计，之后阐述了该系统数据库的设计。再根据客户端程序和服务器程序分成两个部分进行了系统的设计介绍：其中在客户端部分程序中详细的对系统 UI 和具体的功能进行了设计，并且对服务器分三层：实体层、数据访问层、服务层分别进行了设计。

（4）分三个大模块介绍了系统的实现：首先介绍了系统如何实现地图的制作和显示，以及角色在地图上的移动和碰撞检测的功能。接着针对客户端界面的设计对 UI 界面进行了实现。最后针对不同的功能模块分别详细的介绍了这些功能的实现过程。

（5）对系统进行了单元、集成和系统测试，从单元到模块到系统验证了系统功能性的正确性和有效性。

7.2 进一步展望

本论文的开发给计划性的软件和游戏软件的结合提供了思路，并且编写了代码复用性比较高的工具类代码和服务端服务代码。但由于开发时间和本人能力有限，系统还有很多可以得到改进和优化的地方：

（1）系统的 UI 还可以有很多优化的地方。由于时间比较紧，地图的素材块图片没有找到很能与学校真实情况所匹配的，所以地图有的建筑只能用比较抽象的图片去代替。还有一些按钮控件，背景图片都需要用 photoshop 等软件去修改，所以很多界面还需要美工人员的帮忙才能表现的更好。

（2）系统的某些功能还不够完善。例如在消息显示页面，目前系统只能显示有限的 10 条信息，因本人能力有限，未能掌握滚动列表的使用方法，将显示信息的容器改成滚动显示效果和用户体验应该更好。另外向好友发送邀请信息的时应添加一个列表显示好友的姓名，并且选定某些特定好友发送邀请信息。但也是因为时间问题，导致此功能不是很完善。

（3）服务器还有很多问题没有考虑。在搭建服务器时，没有考虑到大量并发的问题，数据库的操作并发数量也没有考虑到，如果有多个用户同时对服务器和数据库进行操作时，为避免出现问题应加入事务锁会比较稳妥。

如果有更多的时间，笔者希望也能逐步完善本文中的系统。随着手机软件的发展和开发引擎平台的发展，这种融合性的系统会更多的受到开发者的关注，希望本文中的系统可以对类似的系统软件有微薄的借鉴作用。

参考文献

- [1] 霍常伟. 基于 cocos2d-x 引擎的移动游戏 UI 系统设计及应用[D]. 北京交通大学, 2012.
- [2] 周飞龙. Cocos2d-X 引擎中 MVC 框架的设计与实现[D]. 华中科技大学, 2013.
- [3] 钟迪龙, Cocos2d-x 3.x 游戏开发之旅[M], 北京: 电子工业出版社, 2014.
- [4] 管蕾, iOS8 开发指南[M], 北京: 人民邮电出版社, 2014.
- [5] 张照, iPhone 开发入门很简单[M], 北京: 清华大学出版社, 2012.
- [6] Jayant Varma, Lua 实战: iOS 游戏开发[M], 北京: 人民邮电出版社, 2013.
- [7] 满硕全, cocos2dx 权威指南[M], 北京: 机械工业出版社, 2013.
- [8] 李华明, ios 游戏编程之从零开始[M], 北京: 清华大学出版社, 2013.
- [9] 达尔林普尔, 纳斯特, Objective-C 基础教程[M], 北京: 人民邮电出版社, 2013.
- [10] 王浩, ios 应用开发最佳实践[M], 北京: 电子工业出版社, 2013.
- [11] 刘建卓, Cocos2D-X 游戏开发技术精解[M], 北京: 人民邮电出版社, 2013.
- [12] 翁永翔, 李阳德, 郑顺航. 基于 Cocos2d 的 iPhone 游戏的开发方法概述[J]. 电子技术与软件工程, 2014, (3):69-69.
- [13] 张筱毅. COCOS2D-X 强交互网络游戏架构设计与实践[D]. 西安电子科技大学, 2014.
- [14] 陈立. 基于 Cocos2d-x 引擎手机游戏实训平台的设计及应用[D]. 南昌大学, 2014.
- [15] 张晓龙. CocoStudio, 游戏开发新工厂[J]. 程序员, 2013, (7):74-76.
- [16] 徐威龙. Cocos2d-X3. 0 带来了什么[J]. 程序员, 2014, (4):86-87.
- [17] 李宁. 用 Cocos2d-x 实现射击类游戏: 星际大战[J]. 电脑编程技巧与维护, 2014, (13):76-86.
- [18] Xie X A, Wang S D, Zeng X L, et al. Analysis of Genetics Diversity of *Poria cocos* by ISSR Marker[J]. Hubei Agricultural Sciences, 2008, 47(10):1111-1113.
- [19] 安毅生, 刘卫方. 基于 Cocos2D 框架的交互式迷宫游戏设计与实现[J]. 计算机与数字工程, 2011, 39(12):154-157.
- [20] 路颜铭. 从跨平台到 3D 技术, Cocos2d-x 的进化之路[J]. 程序员, 2014, (9):88-89.
- [21] 赵东. 利用 Apache 和 Tomcat 构建支持 Servlet 的动态网站[J]. 计算机应用研究, 2000, 17(11):47-49.

-
- [22] 胡莉萍. Tomcat+JSP+MySQL 整合配置初探[J]. 中国科技信息, 2010, (5):102-103.
- [23] Brittain J, Darwin I F. Tomcat: The Definitive Guide[J]. Tomcat the Definitive Guide, 2007, 29(1):1 - 4.
- [24] 谢春明. 基于 Java 的 HTTP Servlet 的创建[J]. 西南民族大学学报: 自然科学版, 2002, 28(1):109-113.
- [25] Moss K. Java Servlets[J]. Object-Oriented Programming and Java, 1999, (1):259-277.

致谢

经过快半年的努力，我终于完成了我的毕业设计——同时也是大学学习中的最后一份作业。毕业设计的完成，也预示着我的大学生活即将结束了。在大学四年里，我从学校学到的不仅仅是计算机的专业知识，同时还有自主学习的方法和为人处世的道理。在此，我要感谢每一位在大学生活中给予我帮主的亲人，老师、同学和朋友。

非常感谢计算机专业的各位老师在大学四年中给予我的教导，你们不仅教会了我计算机的专业知识，还教会了我如何管理自己的学习，授人以鱼不如授人以渔，感谢你们在我遇到困难时给予我的帮助。

还有，我要感谢在我大学四年和毕业设计期间给予我帮助的小伙伴们，在我遇到设计问题的时候，是你们及时的给我提出了宝贵的意见，并在技术上和知识上同我不断的交流探讨，是我们双方都能够取得进步。

我要特别感谢我的毕业设计指导老师——孙永佼副教授。孙老师从毕业设计的开题报告开始一直到论文的定稿，都给予了我很大的帮主。及时指出我的论文中的不足，在我做毕业设计遇到技术性的问题的时候，老师您主动的给我提供指导帮助，一步一步的引导我学会新的技术和知识，使我受益匪浅。我要向尊敬的孙老师，说一声：“谢谢！”

最后，我谨向所有关心和帮助过我的老师、同学、领导、同事和家人致以真诚的谢意。

