模板使用说明

**（请在使用前详细阅读此说明！）**

1. 各部分内容可以直接打字输入，也可使用粘贴功能，但要保证粘贴内容格式正确。
2. 目录为自动生成模式，不要修改目录，论文完成后请右键单击目录区域，选择“更新域”即可。各级标题一定要使用相应标题格式，否则在目录里不能体现。
3. 论文中各种行间距请自行调整。
4. 如本模板和《哈尔滨理工大学本科毕业设计（论文）撰写规范》的规定不一至，以《哈尔滨理工大学本科毕业设计（论文）撰写规范》的规定为准。

**本模板仅供参考，请严格按照《哈尔滨理工大学本科毕业设计（论文）撰写规范》完成论文。如不会使用word的模板功能，请自行按《哈尔滨理工大学本科毕业设计（论文）撰写规范》完成论文。**

**（请在完成论文后，打印论文前，删除本页内容及后文批注！）**

基于iOS的党员管理系统

摘 要

为加强党员日常管理和党员会议以及党员之间的信息共享,我进行了基于iOS的党员管理系统的开发活动.本文描述了一个党员管理系统从业务需求到功能设计,再到功能实现,最后到测试的全部过程.结合国内外的党员管理系统的现状对系统做出需求分析,通过需求对系统进行功能设计,其功能主要有:注册,登录,新闻推送,党员之间即时通信,党员圈子分享,党员会议等.为确保系统能够正常运行,对其进行了详细的测试.确保系统的安全稳定.

随着时代的发展和科技的进步,党员管理系统的需求也会不断增加,为增强系统的可拓展性,系统开发采用的都是目前比较广泛和成熟的技术.界面的人性化设计,提高了党员管理的效率.本系统的设计实施,将会为党员管理提供很大的便利.

关键词　iOS；党员；管理系统

iOS based membership management system

**Abstract**

In order to strengthen the party members' daily management and information sharing between party members and members meeting, the party member management system based on iOS development. This paper describes a party member management system from business requirements to function design, and then to function realization, finally to test all the process. Combined with the status quo of the party member management system at home and abroad on the system requirements analysis, through the demand of the system function design, its main functions are: to register, login, news feeds, instant communication between party members, party members share circles, party conferences, etc. To ensure that the system can run normally, on the details of the test. To ensure the system security and stability.

With the development of The Times and the progress of science and technology, the demand of the party member management system will also continue to increase, to enhance the system can expand sex, system development is currently more extensive and mature technology. Humanized design of the interface, improve the efficiency of the party member management. The design and implementation of the system will provide great convenience for party member management.

**Keywords**Party member, iOS, management system

不要删除行尾的分节符

目 录

摘要 I

Abstract II

[第1章 绪论 1](#_Toc484080824)

[1.1 课题背景 1](#_Toc484080825)

[1.1.1 研究意义 1](#_Toc484080826)

[1.1.2 国内外研究现状分析 1](#_Toc484080827)

[1.1.3 论文研究的主要内容 2](#_Toc484080828)

[第2章 可行性分析及需求分析 4](#_Toc484080829)

[2.1 可行性分析 4](#_Toc484080830)

[2.1.1 技术可行性 4](#_Toc484080831)

[2.1.2 经济可行性 4](#_Toc484080832)

[2.1.3 操作可行性 4](#_Toc484080833)

[2.1.4 社会可行性 4](#_Toc484080834)

[2.2 需求分析 5](#_Toc484080835)

[2.2.1 流程控制 5](#_Toc484080836)

[2.2.2 数据流图 5](#_Toc484080837)

[2.3 开发工具与相关技术 5](#_Toc484080838)

[2.3.1 Xcode工具集 5](#_Toc484080839)

[2.3.2 Mysql数据库 6](#_Toc484080840)

[2.3.3 Php 6](#_Toc484080841)

[2.3.4 MVC 6](#_Toc484080842)

[第3章 解决方案分析 8](#_Toc484080843)

[3.1 方案对比研究 8](#_Toc484080844)

[3.1.1 Native App 8](#_Toc484080845)

[3.1.2 Web App 8](#_Toc484080846)

[3.1.3 Hybrid App 9](#_Toc484080847)

[3.2 本章小结 9](#_Toc484080848)

[第4章 系统设计 11](#_Toc484080849)

[4.1 总体设计原则 11](#_Toc484080850)

[4.2 功能设计 11](#_Toc484080851)

[结论 15](#_Toc484080852)

[致谢 16](#_Toc484080853)

[参考文献 17](#_Toc484080854)

[附录A 19](#_Toc484080855)

[附录B 20](#_Toc484080856)

[附录C 21](#_Toc484080857)

千万不要删除行尾的分节符。在目录上点右键“更新域”，然后“更新整个目录”。打印前，不要忘记把上面“Abstract”这一行后加一空行

# 绪论

## 课题背景

随着党员人数的不断发展,哈理工计算机学院的党员管理工作日益繁杂,面临着许多的困难,如何让广大党员发挥作用,进一步密切党群关系.建设基层服务型党组织,是计算机党支部管理的焦虑所在.目前,党员缴纳党费,需要通过现金,支付宝,微信等很多手段缴纳给党支部书记.党费工作变得繁琐,和无历史记录可寻.另外,现在每次召开党会,需要借助电话,短信,QQ等方式通知给每一个党员,开党会时又需要每个人在签到表上签字,会后需要每个人写一小段会议收获的感想,然后提交给党支部书记.然后党支部书记再挨个复制粘贴到一个文档里面,这样很繁琐.

鉴于目前党支部的党员日常管理非常繁琐,以及现在随着移动互联网的蓬勃发展,基本上所有人都在使用智能手机,而苹果手机因为其独特的操作系统iOS的流畅度,深得大众喜欢.所以决定在导师的指导下完成基于iOS的党员管理系统:党信.

### 研究意义

基于iOS的党员管理系统的研究与开发是为了给计算机党支部提供一个完善的党员管理系统.为党员开会,党费缴纳,党员新闻资讯,党内成员之间即时通讯提供便利.在手持设备上为广大党员提供了很多的便利.现在党员可以使用党信缴纳党费,会有截止日期之前的提醒,以及党费历年历史记录.党内会议在党信内签到,会后在党信内提交会议感谢和会议收获.党员管理者还可以给自己组织内的成员发送通知.新闻资讯模块,党员领导可以推送自己写的新闻,也可以推送国家时事政治,也可以推送校内新闻.最终的目的就是为党员组织提供这套完善的基于iOS操作系统的手机端党员管理系统.随着时代的发展,党信会做顺应时代发展的变化.以达到为广大党员提供便利的目的.

### 国内外研究现状分析

1.1.2.1国外研究现状

国内各相关高等院校在送方面也投入大量的人力、物力,建立了一些管理系.但大多都是自己内部使用,没有对外开放,所以广大高校只有一小部分可以使用.通过查阅资料和论文发现主要针对党员的基本信息管理,党员网上培训,党员信息化.但是大多都是桌面端,需要随手携带笨拙的笔记本电脑才可以操作和使用.而且现在他们的党员管理系统针对面比较小,都是根据自己学校的自身情况设计.对我们学校并不适用.而且也不会提供给其他学校去使用.

1.2.2.2国外研究现状

随着科技的发展,各国在党员管理方面的各项工作也都做了改变,传统方式以人工管理为主,有政党中的机密部门进行管理,并对这些信息进行保密管理,随着互联网的发展,人们对党员管理的观念发生了改变.在西方发达国家,由于互联网的发展,使得党员管理以信息化管理为主.

在发达国家对政党的信息化管理很流行.德国在2004年提出”网络党”的信息化需求.同时开启了”红色电脑与红色手机”的巨大工程.利用网络开展对政党组织的管理,包括对党员内部信息共享,实现数据安全控制.

同样在英国的党政管理工作页很先进,对党员组织实现了嘻嘻花,对党员工作的管理也基于网络.

总之在网络飞速发展的的大前提下,各国都在利用网络和互联网产品推进各自国家的党政建设管理.

### 论文研究的主要内容

#### 党信后台服务端

党信后台服务器需要为所有党信用户提供api接口.同时也需要存储所有党员数据.

党信后台服务器采用LAMP架构.LAMP（Linux- Apache-MySQL-PHP）网站架构是目前国际流行的Web框架，该框架包括：Linux操作系统，Apache网络服务器，MySQL数据 库，Perl、PHP或者Python编程语言，所有组成产品均是开源软件，是国际上成熟的架构框架，很多流行的商业应用都是采取这个架构，和 Java/J2EE架构相比，LAMP具有Web资源丰富、轻量、快速开发等特点，微软的.NET架构相比，LAMP具有通用、跨平台、高性能、低价格的 优势，因此LAMP无论是性能、质量还是价格都是企业搭建网站的首选平台.

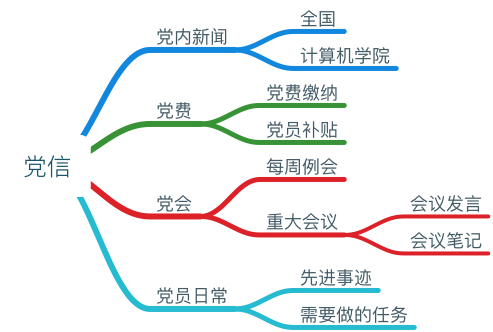
首页的新闻资讯,在党信后端开发一款可以发送新闻的web页面,可以让党员管理者在后台发送新的新闻内容.数据库里面存放着党员的个人信息,党费缴纳历史记录,日程会议出勤情况和党费一些选举的记录.

#### 党信客户端

党信客户端分为以下4个板块.

1. 新闻资讯模块
   1. 热点头条
   2. 党员必备
   3. 校内热讯
2. 通讯录
   1. 党组织
   2. 手机联系人
   3. 党信好友
   4. 我的群组
3. 消息
   1. 个人消息
   2. 群组消息
   3. 党内工作通知
4. 我的
   1. 党员空间
   2. 党员活动相册
   3. 党费钱包
   4. 时政要闻收藏
   5. 设置

以上是iOS客户端的框架概述,这些内容涵盖了党员管理的日常工作和日程管理.下面以思维导图呈现:



# 可行性分析及需求分析

## 可行性分析

可行性分析对于软件的开发非常重要，它实质上就是一次简化的软件整体分析和设计的过程，是较为抽象的分析和开发[2]。本软件主要是从技术、经济、操作和社会因素四方面进行分析和研究，确保软件的成功开发。可行性研究的目的就是用最小的代价在短时间内发现问题，制定应对措施，确保问题能够得到解决。

考虑到项目时间、资源等因素，在实际开发该管理系统时，常常要为资源不足和交付日期难以完成而苦恼，因而需要慎重地尽可能早的估计研制课题的可行性。

下面将分别从技术、经济、操作和社会因素四个方面进行分析和研究。

### 技术可行性

党信管理系统从技术上分为两大模块，一是后台数据管理技术，二是iOS端数据接收展示。

党信以Xcode编码工具IDE，以完全兼容C的Objective-C语言开发,

Objective-C的底层是C，是一门面向对象的语言，运行时语言，具有运行时才确定所调用方法的实例，具有自动内存管理机制，第三方框架众多，引入方便简单的特性。用Xcode工具进行iOS开发技术上可行。

后台采用php语言开发api接口,数据持久层采用Mysql.目前这些技术被广泛使用,是很成熟的应用技术,本系统实质是通过手机端人机界面友好交互实现对服务端api接口的调用,而服务端php代码又对mysql数据库里面的内容进行增删改查.对于硬件要求一般,主要考察对编程语言的掌握.

由以上分析可知,党信在技术上是可以实现的。

### 经济可行性

目前所有高校,以及党政工作机构都有应用本系统的基础,各种型号的计算机,大小配置不一的服务器，数据库和互联网是所有机构不可或缺的设备.至于iOS系统,现在基本上上班族都在使用苹果手机.即使没用苹果手机.ipad也很便宜.所以党信会有很多的收益群众。经济上的费用除了开发初期的软件开发费用还有上线运行后的维护费用和服务器的运行费用。开发成本并不高，由于数据量不大，数据结构不复杂，功能上争议少，故维护费用低，服务器的成本也不高，在经济上可行性很高。

由以上分析可知,党信管理系统在经济上是可行的。

### 操作可行性

党信在设计之初充分考虑的人机交互.在此基础上设计出了精简的人性化界面,无论是普通党员还是党员管理者,都可以在不使用任何帮助文档的前提下使用此系统。党信管理系统的管理人员操作系统简单可靠。从使用角度来说党信管理系统的用户操作方便，操作可行。用户交互上的设计与市场上主流交互设计吻合，各个app的交互方式相似性很大，本系统的交互会使得用户有亲切感又有熟悉感。

### 社会可行性

党信的社会可行性从法律角度来讲,不存在任何抄袭所以不会有任何违法，犯法行为,党信的用户大多是先进个人,不会做违反法律的事情.社会可行性是完全满足的。

从党员的角度分析，党员作为系统的用户，党费的收缴，党内新闻的获取是经常进行的行为，是作为党员必不可少行为。党信管理系统方便了党员对党内信息的获取，方便了党费的收缴，方便的获取到管理层的通知和活动使得党员的积极性大大提高。从党员本身的角度分析，社会可行性体现在党员的日常需求方面。

对党员管理层人员来说减轻了统计工作的工作量，减轻群体活动组织的工作量，使得会议的召开，群体活动的组织便利可靠。对于由于地理因素，天气因素，住院，生病等等诸多不能参加党员会议的党员可以通过ios端轻松地众多党员进行会议讨论，会议的实时性得到保证。

党员群体是一个联系紧密的群体，彼此之间的交流联系紧密，这为软件的使用普及提供了便利。软件的普及方便，使用面广泛使得社会可行性增高。

从以上各个人群的角度分析，党信管理系统的建立会方便广大党员群体。

## 需求分析

党信给计算机党支部提供一个完善的党员管理系统.为党员开会,党费缴纳,党员新闻资讯,党内成员之间即时通讯提供便利.在手持设备上为广大党员提供了很多的便利.现在党员可以使用党信缴纳党费,会有截止日期之前的提醒,以及党费历年历史记录.党内会议在党信内签到,会后在党信内提交会议感谢和会议收获.党员管理者还可以给自己组织内的成员发送通知.新闻资讯模块,党员领导可以推送自己写的新闻,也可以推送国家时事政治,也可以推送校内新闻.

为实现现代化,高效率的党员管理系统,设计党员管理系统党信要有如下几个重要的功能:

1. 能够完成管理员,普通党员的注册登录功能,注册时实现手机短信验证和对错误的登录进行报错和提醒。

2. 用户可以更改年龄.身高,体重等自身属性

3. 党员之间可以聊天。

4. 党员可以参与党内选举。

5. 党员可以在线缴纳党费。

6. 党员管理者可以发布会议。

7. 党员可以查阅时事政治新闻。

8. 党员可以查看历史事迹。

9. 可以随时登陆，随时退出登录。

## 开发工具与相关技术

### Xcode工具集

Xcode 是运行在操作系统Mac OS X上的集成开发工具(IDE)，由苹果公司开发。Xcode是开发OS X 和 iOS 应用程序的最快捷的方式,Xcode 具有统一的用户界面设计，编码、测试、调试都在一个简单的窗口内完成。

Xcode 可以编写 C、C++、Objective-C 和 Java 代码，可以生成 MacOS 支持的所有类型的执行代码，包括命令行工具、框架、插件、内核扩展、程序包和应用程序。Xcode 具有编辑代码、编译代码、调试代码、打包程序、可视化编程、性能分析、版本管理等开发过程中所有的功能。而且还支持各种插件进行功能扩展、具有丰富的快捷 键，有效帮助开发软件的开发人员提高效率。

Xcode的优势一在于集成多个开发所必须的工具，包括内存分析器，iOS系统模拟器，编码器等等。优势二在于支持多个语言的编码和运行，支持不同语言编码混编。优势三在于导入文件方便，支持打包成静态和动态库，可以在代码不可见的情况下导入可执行文件或者目标文件支持。优势四在于可以导入子工程，支持多工程协调运行，动态静态链接。

### Mysql数据库

MySQL是一个关系型数据库管理系统，由瑞典MySQL AB公司开发，目前属于Oracle公司。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。

它有如下优点:

1. Mysql是开源的，所以你不需要支付额外的费用。

Mysql支持大型的数据库。可以处理拥有上千万条记录的大型数据库。

2. MySQL使用标准的SQL数据语言形式。

Mysql可以允许于多个系统上，并且支持多种语言。这些编程语言包括C、C++、Python、Java、Perl、PHP、Eiffel、Ruby和Tcl等。

3. Mysql对PHP有很好的支持，PHP是目前最流行的Web开发语言。

4. MySQL支持大型数据库，支持5000万条记录的数据仓库，32位系统表文件最大可支持4GB，64位系统支持最大的表文件为8TB。

5. Mysql是可以定制的，采用了GPL协议，你可以修改源码来开发自己的Mysql系统。

6. Mysql使用的SQL语句简单便捷，操作简便，易读性好。

### Php技术简介

PHP（外文名:PHP: Hypertext Preprocessor，中文名：“超文本预处理器”）是一种通用开源脚本语言。语法吸收了C语言、Java和Perl的特点，利于学习，使用广泛，主要适用于Web开发领域。PHP 独特的语法混合了C、Java、Perl以及PHP自创的语法。它可以比CGI或者Perl更快速地执行动态网页。用PHP做出的动态页面与其他的编程语言相比，PHP是将程序嵌入到HTML（标准通用标记语言下的一个应用）文档中去执行，执行效率比完全生成HTML标记的CGI要高许多；PHP还可以执行编译后代码，编译可以达到加密和优化代码运行，使代码运行更快。对于小型的后台系统，php可以发挥其编码简单，操作简单的特长。

### MVC模式解析

从字面意思来理解，MVC 即 Modal View Controller（模型 视图 控制器），是 Xerox PARC 在 20 世纪 80 年代为编程语言 Smalltalk－80 发明的一种软件设计模式，至今已广泛应用于用户交互应用程序中。其用意在于将数据与视图分离开来。在 iOS 开发中 MVC 的机制被使用的淋漓尽致，充分理解 iOS 的 MVC 模式，有助于我们程序的组织合理性。

MVC 的几个明显的特征和体现：

View 上面显示什么东西，取决于 Model。只要 Model 数据改了，View 的显示状态会跟着更改。Control 负责初始化 Model，并将 Model 传递给 View 去解析展示。

1）Modal 模型对象：

模型对象封装了应用程序的数据，并定义操控和处理该数据的逻辑和运算。例如，模型对象可能是表示商品数据 list。用户在视图层中所进行的创建或修改数据的操作，通过控制器对象传达出去，最终会创建或更新模型对象。模型对象更改时（例如通过网络连接接收到新数据），它通知控制器对象，控制器对象更新相应的视图对象。模型并不是单独存在的数据集合，它要依赖于MVC中的C加以存储和改变，还要依赖于MVC中的V来加以数据展现。

2）View 视图对象：

视图对象是应用程序中用户可以看见的对象。视图对象知道如何将自己绘制出来，可能对用户的操作作出响应。视图对象的主要目的就是显示来自应用程序模型对象的数据，并使该数据可被编辑。尽管如此，在 MVC 应用程序中，视图对象通常与模型对象分离。

在iOS应用程序开发中，所有的控件、窗口等都继承自 UIView，对应 MVC 中的 V。UIView 及其子类主要负责 UI 的实现，而 UIView 所产生的事件都可以采用委托的方式，交给 UIViewController 实现。

3）Controller 控制器对象：

在应用程序的一个或多个视图对象和一个或多个模型对象之间，控制器对象充当媒介。控制器对象因此是同步管道程序，通过它，视图对象了解模型对象的更改，反之亦然。控制器对象还可以为应用程序执行设置和协调任务，并管理其他对象的生命周期。

控制器对象解释在视图对象中进行的用户操作，并将新的或更改过的数据传达给模型对象。模型对象更改时，一个控制器对象会将新的模型数据传达给视图对象，以便视图对象可以显示它。

对于不同的 UIView，有相应的 UIViewController，对应 MVC 中的 C。例如在 iOS 上常用的 UITableView，它所对应的 Controller 就是UITableViewController。

4）MVC三者整体关系分析



MVC三者关系如图所示，

使用MVC的目的是使业务和视图分离，让业务层独立发挥业务层的功能而不去考虑其他业务。数据层只负责数据的规范存储，刷新UI，响应事件等操作交给交给Controller进行。事件的获取和捕获是由View来实现的，View作为和用户交互的通道承载着数据的美化展示，数据的罗列方式不同给用户造成的感觉不同，以舒适流畅的方式展示数据更容易使用户接受，这就是View存在的最终意义，View的数据展示需要有实时性，View所展示的内容要随着数据的变化而变化，View监听数据变化的方式有KVO，消息中心等，不通过这两种方式也可以直接通过Controller处理外部事件时更新数据并更新UI。

# 解决方案分析

## 方案对比研究

设计系统之初，在iOS app的开发方式上考虑了原生性和h5开发。在当今移动端研发方式上大致有三套解决方案，分别是Native App(原生),Web App和Hybrid App(混合开发)。以下是三种方案的分析对比，及最终方案的选择。

### Native App的方案分析

Native App，原生APP，使用原生（即Android或iOS）开发的APP。两年多以前这非常流行，到现在为止，原生开发人员数量众多，一抓一大票，技术成熟.使用原生开发有其优势：应用的性能好，适配起来相对容易。学习成本要看人，个人觉得技术点不多，门槛相对稍高，但入门后学习起来就很轻松——网络资料实在是太多了。

但原生APP最头疼的有三个问题：

1、无法跨平台：Android和iOS都需要开发各自平台的版本——开发和维护需要两份人力，工作重复性高，成本高。Android和iOS在语言特性和实现方式上的差异使得两种app在功能或者用户体验上会造成很大的差异。

2、升级麻烦：每次升级都要下载安装包，Android还好，反正不需要审核，下载就下载吧，但iOS就麻烦了，发布每个版本还得经过App Store的审核，这导致第三个问题。原生的APP不能随时添加功能，对于线上bug的修改便捷性和实时性不强。

3、Android和iOS很难同步发布，这就导致Android和iOS的线上功能不同步，数据统计不同步，此起彼伏。

虽然有笨拙，不灵巧的缺点，但其优点也不少：

1、操作效率高，能为用户提供最佳的用户体验，能为用户提供优质的用户界面，能为用户提供强交互效果，有效提升营销的效果。

2、 开发项目独立，针对性强，因此开发设计能够针对不同平台、不同用户的需求进行设计出不同体验的产品；

3、能轻松地操控移动设备的底层功能，更契合操作系统，还可以实现本地的数据保存，在弱网条件下还可以实现部分功能。

### Web App的方案分析

Webapp说白了就是一个针对Iphone、Android优化后的web站点，它使用的技术无非就是HTML或Html5、CSS3、JavaScript，服务端技术涉及JAVA后台配置、PHP、ASP。

Web App有如下特点：

1、开发成本较低:使用HTML5 + CSS3 + js 等web开发技术就可以轻松的完成web app的开发。效果上面能够完全模拟传统应用程序效果。

2、升级较简单:由于不需要通过苹果商店发布，所以升级不需要通知用户，在服务端更新文件即可，用户完全没有感觉。升级灵活，发布灵活。

3、维护比较轻松:和一般的web一样，维护比较简单，它其实就是一个站点。

4、对于用户来说，没有安装包的Webapp更加节省手机硬盘空间。

虽然Webapp有诸多的优点，开发发版灵活，跨平台性强，可更改性强等灵活的特性，但Webapp也有其缺点。市场上诸多应用中，仅有Webapp而没有原生app的应用的用户量并不大，在用户的思想上人们会对原生的app有信赖心里，用户觉得原生app较为正式。在体验上，原生app的用户体验更好，滑动等手势操作更加流畅。页面之间的跳转更加流畅。

### Hybrid App的方案分析

Hybrid app从外观上来看是一个native app，实则只有一个UIWebView，里面访问的是一个web app，如新闻类和视频类的应用普遍采取该策略：native的框架加上web的内容。不同于native app需要针对不同的平台使用不同的开发语言（如使用Objective-C、Swift开发iOS应用，使用Java等开发Android应用，使用C#开发Windows Phone应用），hybrid app允许开发者仅使用一套网页语言代码（HTML5+CSS+JavaScript），即可开发能够在不同平台上部署的类原生应用 。由于hybrid app结合了native app良好用户交互体验和web app跨平台开发的优势，能够显著节省移动应用开发的时间和成本，hybrid app得到越来越多公司的青睐

按照网页语言和程序语言的混合，hybrid app通常可以分为三种类型：

1. 多View混合型：native view和web view独立展示，交替出现。 其应用主体通常是native app，web技术作为补充。即在需要的时候，将web view作为独立的view运行，在web view内完成相关的展示操作。开发难度与native app相当。举个例子：Instagram的timeline使用的是web view。

2. 单View混合型：在同一个view内，native view和web view为层叠关系，同时出现。开发成本较高，难度较大，但是体验较好。举个例子：百度搜索同时实现充分的灵活性和较好的用户体验。

3. Web主体型：应用主体是web view，穿插native功能，主要以网页语言编写。整体开发难度低，基本可以实现跨平台；而用户体验好坏，主要取决于底层中间件的交互与跨平台能力。举个栗子：项目管理工具 Basecamp使用web view呈现内容，调用系统原生API实现界面导航等功能来提高用户体验。



## 后台架构方案选择

后台架构的语言可以选择java语言也可以选择PHP语言。

1. 学习难易程度比较

语法上融合了多门语言的特点，和C语言很是相似，java要在学好语法的基础上熟悉很多核心类库，还要懂得面向对象的思想。PHP的在学习方面较为简单。

1. 编译执行的难度比较

PHP是一门脚本语言，具有解释执行的特点。Java的编译要先编译成.class文件，运行要在虚拟机上运行，也是以解释的方式在虚拟机上运行。中间的虚拟机层会对代码的运行速度造成影响。

1. 跨平台性的比较

Java因为虚拟机的缘故有很好的跨平台性，只要安装虚拟机，在任何平台都可以运行。PHP作为脚本语言同样具有很强的跨平台性。两者跨平台性良好。

4，设计架构的比较

Java的web开发技术用的是面向对象思想，面向对象思想的一大优点是封装强悍，不用知道内部细节，直接调用，功能强大。PHP的web开发是面向过程的编程思想，缺点是可维护性差，扩展性差，耦合性高。优点是效率高，开销小，不用建立对象，不用实例化，高性能。

综上所述，java的后台开发和PHP的后台开发各有优缺点，基于本系统的系统功能较为简单，结构层次较为清晰，不选择面向对象的java语言进行后台开发，选择简单易学的PHP语言进行后台开发。

## 本章小结

在hybrid app备受追捧的今天，我们也许会随大流优先考虑开发hybrid app。但是，每个产品各具特色，还没有哪一种解决方案可以完美地应万变。hybrid app的中庸让它大放异彩，也限制了它的应用场景 。在决定采用一种开发模式之前，有许多因素需要考虑，诸如：

你希望针对哪个平台做开发？

你希望通过应用商店发布应用么？

你希望利用手机特有的性能么？

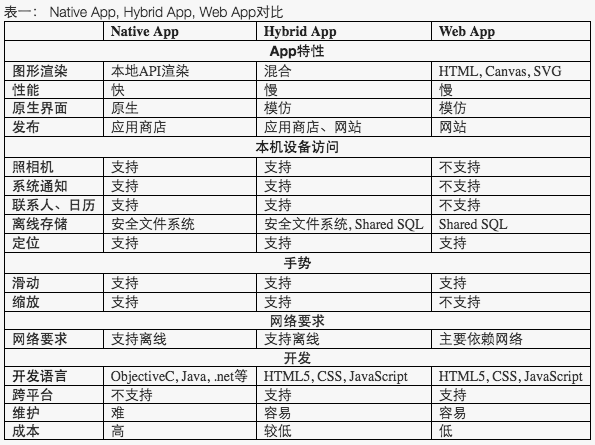
你的团队技术能力如何？

你的项目周期和预算是多少？

如果要设计Angry Birds一类对图形要求很高的游戏，那么在暂不考虑技术团队能力的情况下，native app是最优选择；如果要设计如Yelp一类内容导向的应用，那么hybrid app会是很好的选择；如果项目时间紧张，没有足够的人手和资金，对图形和系统原生特性没有要求，那么web app将是性价比最高的解决方案。

总结来说，一些重要的功能，访问率高的功能用原生方式实现，增强用户体验。一些内容频道，访问率较低的功能用H5实现，高效便捷，内容更改灵活。

表一对比了native app, hybrid app和web app在不同方面的表现，最终选择Hybrid App的解决方案，即所有功能，所有页面均以原生iOS页面的形式展示。



上述表格对三种实现方案各个方面做了整体分析，根据需要进行适当方案选择。当今主流方案是混合编程，即整个工程的主体部分用原生代码，一些内容频道用H5页面，便于改动，灵活便捷。

在后台开发语言的比较方面，从学习难易程度，编译执行的难度，跨平台性，设计架构的比较等方面对两门开发语言进行比较，得出如下结论：PHP的学习难度小，java和PHP都有很强的跨平台性，两门语言都是解释型语言，java的设计架构为面向对象架构，PHP的设计架构为面向过程的设计架构。

因系统功能较为简单，结构层次较为清晰，不选择面向对象的java语言进行后台开发，选择简单易学的PHP语言进行后台开发。

# 系统设计

## 总体设计原则

本系统为了设计、实现和后期维护的方便，以及系统用户使用的便利，所以必须采取一定的设计原则。其主要设计原则有：

程序简便：在软件开发过程中，尽量保持代码的简洁高效，这对于软件的后期维护是至关重要的。

1．可靠性 软件的可靠性是软件存在的基础，软件可靠性意味着该软件在测试运行过程中避免可能发生故障的能力，且一旦发生故障后，具有解脱和排除故障的能力。

2．可修改性 要求本系统具有良好的体系结构,web后端和iOS  
iOS客户端形结合，系统性能易于调整。

3．一致性 系统整体设计保持一致性，包括代码的MVC架构模式,和驼峰式命名规范.

4．先进性 本系统采用Objective-C、Mysql等成熟的开发技术和数据库，因此本设计具有很好的先进性，具体表现在其具有良好的可开发性，可扩展性。

## 功能设计

功能设计就是按照产品定位的初步要求，在对需求及现有产品进行功能调查分析的基础上，对所定位产品应具备的目标功能系统进行概念性构建的创造活动。功能设计是功能创新和产品设计的早期工作，是设计调查、策划、概念产生、概念定义的方法，也是产品开发定位及其实施环节，体现了设计中市场导向作用。可以采用用户设计和专业设计或二者结合的方式。是以使用者的潜在需求和功能成本规划为依据,设计产品的功能，经过功能的成本核算后,由专业人员进行产品设计并生产。功能设计的依据是市场细分和产品定位理论的深化，市场细分方法有好多种，但都是以功能细分的。

功能设计是把系统的需求进行实现的过程。通过对系统的实际调研和系统的分析，做出对科研成果管理系统的总体设计，总体设计作为整个项目开发的基础。在总体设计的基础上，结合前期做的逻辑控制分析，进行项目的开发。系统设计包括系统总体设计、系统详细设计、系统数据库设计、系统开发工具、开发技术和开发模式等。从思维导图出发，对数据流图进行分析，得出的层次化的模块结构图。党信理系统功能模块图如图4-1所示。

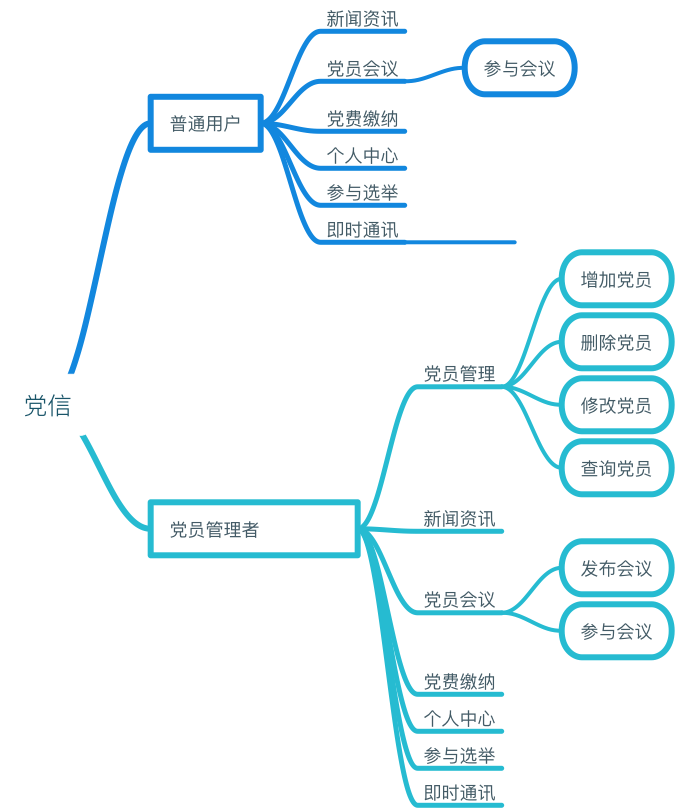


图4-1党信理系统功能思维导图

## 数据库设计

数据库设计(Database Design)是指对于一个给定的应用环境，构造最优的数据库模式，建立数据库及其应用系统，使之能够有效地存储数据，满足各种用户的应用需求（信息要求和处理要求）。在数据库领域内，常常把使用数据库的各类系统统称为数据库应用系统。

#### 概念结构设计

党信在设计和实现过程中,为确保后期软件可以容易的进行二次开发和软件的维护,尽量减少表的建立.只用用得上的字段建表.

#### 数据库表设计

数据库表所需要存储的主要有党员用户信息,新闻推送等.

用户表用于系统的所有党员用户信息,其中用户角色为1表示是普通党员,为0表示为管理员,表4-1为用户表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 说明 |
| u\_account | varchar | 32 | 账号（主键） |
| u\_password | varchar | 32 | 用户密码 |
| u\_role | varchar | 1 | 角色 |
| u\_name | varchar | 12 | 姓名 |
| u\_card | varchar | 18 | 身份证 |
| u\_phone | varchar | 12 | 电话号码 |
| u\_birthday | varchar | 32 | 出生年月 |
| u\_age | varchar | 32 | 年龄 |
| u\_unit | varchar | 32 | 党组织 |
| u\_desc | varchar | 32 | 描述 |

表4-1

## 本章小结

本章主要介绍了本课题设计的系统功能、实体属性、数据库详情等，让各个部分完全整合成一个总体，通过对每个部分的实现来完成对最后设计的完成。

# 系统实现

## 登录模块

登录模块主要是用户的登录,不包括用户的注册,因为党信是为党员组织提供的管理系统,所有用户都是由党员管理者插入到数据.而不是所有人都可以去注册.为了组织的安全和隐私考虑,手机端只做登录接口.在请求登录接口的时候,服务端会先判断数据库中是否存在,如果不存在账号则返回账号错误,如果存在单密码不正确就返回密码错误.如果账号和密码都正确,就返回登陆成功.

开 始

用户登录

登陆是否？

成功

进入

app

返回登

陆页

结束

N

Y

## 即时通讯

党员之间的交流采用XMPP协议, XMPPFramework是一个OS X/iOS平台的开源项目，使用Objective-C实现了XMPP协议（RFC-3920），同时还提供了用于读写XML的工具，大大简化了基于XMPP的通信应用的开发。具体流程如图5-2所示

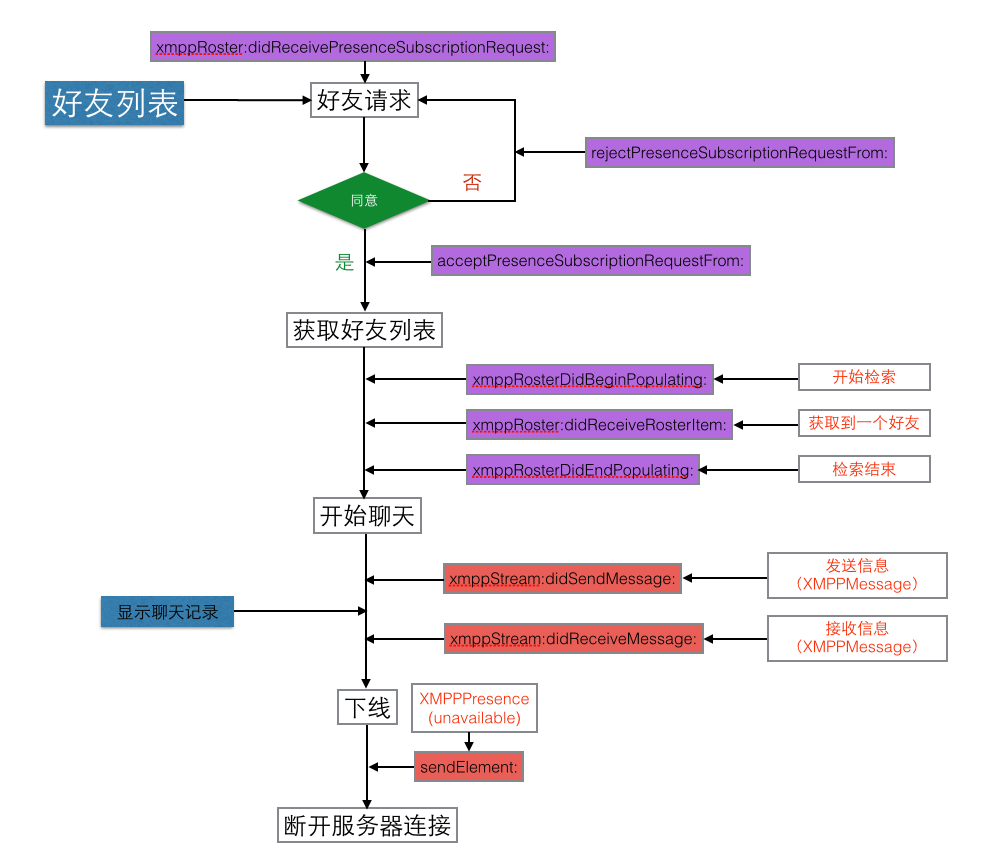


图5-2

### 登录和好友上下线

#### XMPP中常用对象们

XMPPStream：xmpp基础服务类

XMPPRoster：好友列表类

XMPPRosterCoreDataStorage：好友列表（用户账号）在core data中的操作类

XMPPvCardCoreDataStorage：好友名片（昵称，签名，性别，年龄等信息）在core data中的操作类

XMPPvCardTemp：好友名片实体类，从数据库里取出来的都是它

xmppvCardAvatarModule：好友头像

XMPPReconnect：如果失去连接,自动重连

XMPPRoom：提供多用户聊天支持

XMPPPubSub：发布订阅

#### 登录操作，也就是连接xmpp服务器

- (void)connect {

    if (self.xmppStream == nil) {

        self.xmppStream = [[XMPPStream alloc] init];

        [self.xmppStream addDelegate:self delegateQueue:dispatch\_get\_main\_queue()];

    }

    if (![self.xmppStream isConnected]) {

        NSString \*username = [[NSUserDefaults standardUserDefaults] objectForKey:@"username"];

        XMPPJID \*jid = [XMPPJID jidWithUser:username domain:@"lizhen" resource:@"Ework"];

        [self.xmppStream setMyJID:jid];

        [self.xmppStream setHostName:@"10.4.125.113"];

        NSError \*error = nil;

        if (![self.xmppStream connect:&error]) {

            NSLog(@"Connect Error: %@", [[error userInfo] description]);

        }

    }

}

connect成功之后会依次调用XMPPStreamDelegate的方法，首先调用

- (void)xmppStream:(XMPPStream \*)sender socketDidConnect:(GCDAsyncSocket \*)socket

然后

- (void)xmppStreamDidConnect:(XMPPStream \*)sender

在该方法下面需要使用xmppStream 的authenticateWithPassword方法进行密码验证，成功的话会响应delegate的方法，就是下面这个

- (void)xmppStreamDidAuthenticate:(XMPPStream \*)sender

### 接收消息和发送消息

#### 接收消息

通过实现

- (void)xmppStream:(XMPPStream \*)sender didReceiveMessage:(XMPPMessage \*)message;

当接收到 message 标签的内容时，XMPPFramework 框架回调该方法

根据 XMPP 协议，消息体的内容存储在标签 body 内

- (void)xmppStream:(XMPPStream \*)sender didReceiveMessage:(XMPPMessage \*)message {

    NSString \*messageBody = [[message elementForName:@"body"] stringValue];

}

#### 发送消息

发送消息，我们需要根据 XMPP 协议，将数据放到标签内，例如：

<message type="chat" to="xiaoming@example.com">

<body>Hello World!<body>

<message/>

- (void)sendMessage:(NSString \*) message toUser:(NSString \*) user {

    NSXMLElement \*body = [NSXMLElement elementWithName:@"body"];

    [body setStringValue:message];

    NSXMLElement \*message = [NSXMLElement elementWithName:@"message"];

    [message addAttributeWithName:@"type" stringValue:@"chat"];

    NSString \*to = [NSString stringWithFormat:@"%@@example.com", user];

    [message addAttributeWithName:@"to" stringValue:to];

    [message addChild:body];

    [self.xmppStream sendElement:message];

}

### 获取好友信息和删除好友

#### 好友列表和好友名片

[\_xmppRoster fetchRoster];//获取好友列表

//获取到一个好友节点

- (void)xmppRoster:(XMPPRoster \*)sender didRecieveRosterItem:(NSXMLElement \*)item

//获取完好友列表

- (void)xmppRosterDidEndPopulating:(XMPPRoster \*)sender

//到服务器上请求联系人名片信息

- (void)fetchvCardTempForJID:(XMPPJID \*)jid;

//请求联系人的名片，如果数据库有就不请求，没有就发送名片请求

- (void)fetchvCardTempForJID:(XMPPJID \*)jid ignoreStorage:(BOOL)ignoreStorage;

//获取联系人的名片，如果数据库有就返回，没有返回空，并到服务器上抓取

- (XMPPvCardTemp \*)vCardTempForJID:(XMPPJID \*)jid shouldFetch:(BOOL)shouldFetch;

//更新自己的名片信息

- (void)updateMyvCardTemp:(XMPPvCardTemp \*)vCardTemp;

//获取到一盒联系人的名片信息的回调

- (void)xmppvCardTempModule:(XMPPvCardTempModule \*)vCardTempModule

didReceivevCardTemp:(XMPPvCardTemp \*)vCardTemp

forJID:(XMPPJID \*)jid

#### 添加好友

 //name为用户账号

    - (void)XMPPAddFriendSubscribe:(NSString \*)name

    {

        //XMPPHOST 就是服务器名，  主机名

        XMPPJID \*jid = [XMPPJID jidWithString:[NSString stringWithFormat:@"%@@%@",name,XMPPHOST]];

        //[presence addAttributeWithName:@"subscription" stringValue:@"好友"];

        [xmppRoster subscribePresenceToUser:jid];

    }

#### 收到添加好友的请求

-(void)xmppRoster:(XMPPRoster\*)sender didReceivePresenceSubscriptionRequest:(XMPPPresence \*)presence

{

//取得好友状态

NSString \*presenceType = [NSString stringWithFormat:@"%@", [presence type]]; //online/offline

//请求的用户

NSString \*presenceFromUser =[NSString stringWithFormat:@"%@", [[presence from] user]];

NSLog(@"presenceType:%@",presenceType);

NSLog(@"presence2:%@ sender2:%@",presence,sender);

XMPPJID \*jid = [XMPPJID jidWithString:presenceFromUser];

//接收添加好友请求

[xmppRoster acceptPresenceSubscriptionRequestFrom:jid andAddToRoster:YES];

}

#### 删除好友

//删除好友，name为好友账号

- (void)removeBuddy:(NSString \*)name

{

XMPPJID \*jid = [XMPPJID jidWithString:[NSString stringWithFormat:@"%@@%@",name,XMPPHOST]];

[self xmppRoster] removeUser:jid];

}

## 个人中心

## 新闻推送

## 整体功能架构

千万不要删除行尾的分节符。“结论”以前的所有正文内容都要编写在此行之前。

结论

致谢

大学四年是人生的关键期，是学习能力最强的时候，是体力精力最好的时候，是学习专业知识最快的时光。快乐的大学四年伴随着专业知识的学习，社会实践能力的养成，专业动手能力的养成，学习兴趣的养成。感谢这四年来所有帮助我的人们。

感谢老师们对我的教导，大学四年来，无论是专业课的老师们还是公共基础课的老师们都对我们的知识体系的养成，应用实践能力的提升做出了很多贡献。老师们精心制作的教学ppt，在讲台上的生动讲解都为同学们展现了作为老师的育人为本。幽默生动的讲课方式，内容充实的课程内容都给我们留下了深刻的印象。感谢老师们四年来的默默付出。

感谢同学们对我的照顾和陪伴，在日常学习生活中都离不开同学们的身影。我们一起上课，一起参加社会实践，一起参加社团，一起紧张备考，一起参加比赛。相互帮助，共同进步。有了同学们的支持我才能越走越远，不断进取。

感谢学校的悉心培养，理工大学的浓厚学习氛围给了我很多学习的机会，同学们的各方面能力都有所提高，感谢这个展现自我，使我不断进步的平台。

大学四年青春激昂，感谢身边的人们，我们马上要走向社会了，验证四年奋斗成果的时候终于要来了，我一定不负老师和同学们的期望，知行统一，博厚悠远。

参考文献

1. 戈宝军，梁艳萍，温嘉斌．电机学[M]．第二版．北京：中国电力出版社，2013：1-3．
2. 主要责任者．题名：其他题名信息[文献类型标志]．其他责任者．版本项．出版地：出版者，出版年：引文页码
3. 余敏．出版集团研究[M]．北京：中国书籍出版社，2001：179-193．
4. 昂温G，昂温P S．外国出版史[M]．陈生铮，译．北京：中国书籍出版社，1988：89-102．
5. 全国文献工作标准化技术委员会第七分委员会．GB/T 5795-1986 中国标准书号[S]．北京：中国标准出版社，1986．
6. 作者.文章名.学术刊物名，年，卷（期）：引用部分起止页
7. 李晓东，张庆红，叶瑾琳．气候学研究的若干理论问题[J]．北京大学学报：自然科学版，1999，35（1）：101-106．
8. 叶志军，游佰强，J. Rosendahl，等．基于Flux 2D 的大型同步发电机在额定运行下定子绕组匝间短路故障特性研究[J]．中国电机工程学报，2013，33(3)：125-132．
9. LCASO A N，MARQUES CARDOSO A J．Remedial Operating Strategies for A 12-Pulse LCI Drive System [J]．IEEE Transactions on Industrial Electronics，2008，55(5)：2133-2139．
10. DE ARAUJO RIBEIRO R L，JACOBINA C B，DA SILVA E R C，et al．Fault Detection of Open-Switch Damage in Voltage-Fed PWM Motor Drive Systems [J]．IEEE Transactions on Power Electronics，2003，18(2)：587-593．
11. GAMEIRO N S，CARDOSO A J M．A New Method for Power Converter Fault Diagnosis in SRM Drives [J]．IEEE Transactions on Industry Applications，2012，48(2): 653-662．
12. 作者.文章名.编者名.会议名称，会议地址，年份：引用部分起止页
13. ALCASO A N，CARDOSO A J M．Power Supply Harmonic Filter Behavior in A Twelve-Pulse LCI Drive System Under Power Converter Faults[C]．Proceedings of the Power Electronics Specialists Conference， PESC '05 IEEE 36th，Recife，16-16 June，2005：2893-2897．
14. 研究生名.学位论文题目.学校及学位论文级别，答辩年份：引用部分起止页
15. 赵相宾．基于静止变频调速系统的抽水蓄能机组起动研究[D]．天津：天津大学，2007：13-27．
16. 张侃．同步电动机软起动装置的研究[D]．哈尔滨：哈尔滨工业大学，2008：8-21．
17. 张志祥．间断动力系统的随机扰动及其在守恒律方程中的应用[D]．北京：北京大学数学学院，1998：3-15.
18. CALM S R B. Infrared spectroscopic studies on solid oxygen [D]. Berkeley: Univ. of California. 1965：1-5.
19. 刘加林．多功能一次性压舌板：中国，92214985．2[P]．1993-04-14．
20. 河北绿洲生态环境科技有限公司．一种荒漠化地区生态植被综合培育种植方法：中国，01129210.5[P/OL]. 2001-10-24[2002-05- 28]. http:/ /211.152.9.47/sipoasp/zlijs/hyjs-yx-new.asp? recid=01129210. 5&leixin.

附录A

#include “stdio.h”

main()

{

for (;;)

printf(“成功到永远！”);

}

附录B

附录C