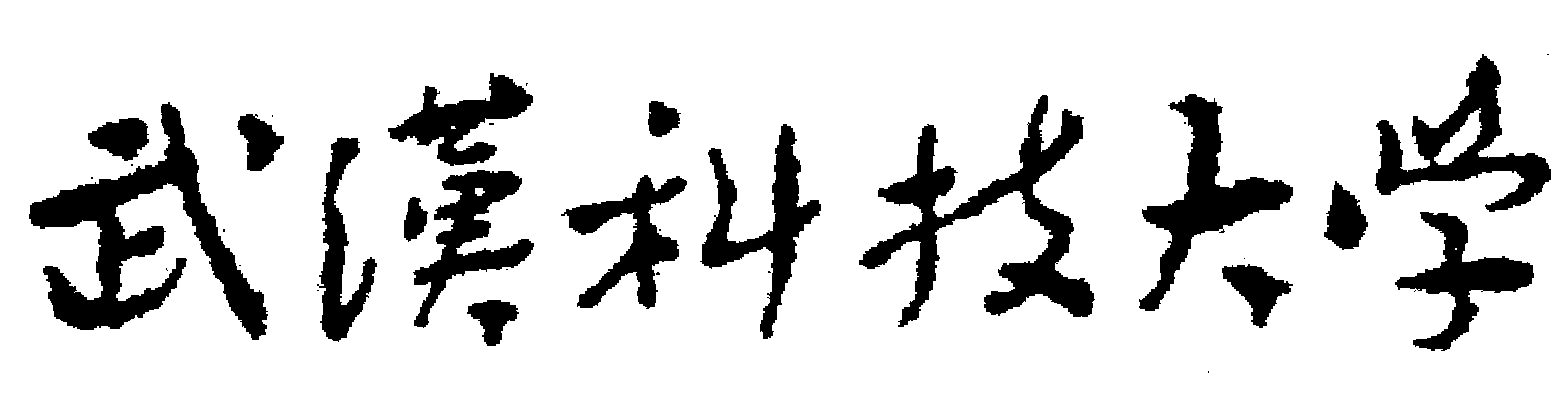
本科毕业论文

|  |  |
| --- | --- |
| **题目：** | 基于GIS 的地理社交软件 |

|  |  |
| --- | --- |
| **学 院:** | 计算机科学与技术学院 |
| **专 业:** | 网 络 工 程 |
| **学 号:** | 201113136053 |
| **学生姓名:** | 陈 灿 豪 |
| **指导教师:** | 杨 湘 |
| **日 期:** | 二〇一五年六月 |

摘 要

本地理社交软件是一个基于GIS，采用WEB APP模式，并根据现有的地理社交现状而为设计开发的社交平台。可以使用户根据自己的坐标定位而搜索附近用户所发的博文，状态，图片等社交元素，用户本身可以基于自己的坐标定位发布博文等；用户与用户之间可以互相交友，实现社交功能。

本文首先介绍了地理社交软件的现状和开发背景，然后论述了系统的设计目标，系统需求和总体设计方案，较详细的论述了系统的详细设计和实现。最后，本文地理社交软件设计难点并提出了还需要改进的地方和想法。本软件为主要为用户提供了会员注册，用户登陆，发布博文，查看博文，互相加为好友等功能。在设计方面，本系统采用了B/S架构，前后端分离设计模式，后端采用Node.js提供API，前端采用Angular框架构建单页面WEB APP。数据库采用Postgresql。

关键词： B/S结构； Angular； Node.js； Postgresql

**Abstract**

The online video system is based on the Internet, using B2C (Business to Consumers) model, and in accordance with the status of existing online video system to design and develop the e-commerce platform. It can not only expand the business scale and market influence, but also reduce operating costs and enhance efficiency.

This paper first introduces the status and the development background of online video system, and then discusses the system design objectives, system requirements and design of program, and more detailed discussion of the detailed design and realization of the system. Finally, the pager declare the difficulties of development of online video system and also raised the issue of the need to improve.

The system provides users with functions of member registration, video preview, user login, video upload**,** and watch video. It also provides administrators with credit management, personal information management, video management and other functions. In terms of design, the system uses the B / S three-layers structure, while using the JSP technique for dynamic page design. The system deploit the JavaBean to package the key code in consideration of the safety of system and the reusability of the code. And the database management system we choose is MySql.

**Key words**： B/S structure; JavaBean; JSP; MySql

目 录

[1 绪论 1](#__RefHeading___Toc20898)

[1.1视频点播系统概述 1](#__RefHeading___Toc14211)

[1.2系统开发背景 2](#__RefHeading___Toc14870)

[1.2.1网上视频点播的发展 2](#__RefHeading___Toc9384)

[1.2.2网上视频点播系统的现状 2](#__RefHeading___Toc8514)

[1.3系统实现的目标 3](#__RefHeading___Toc31992)

[1.4系统的开发意义 3](#__RefHeading___Toc8699)

[2 视频点播系统分析 4](#__RefHeading___Toc17519)

[2.1网站的设计目标 4](#__RefHeading___Toc15049)

[2.2网站的可行性分析 4](#__RefHeading___Toc18448)

[2.3网站的深入调查 4](#__RefHeading___Toc29418)

[2.4网站设计的特点 5](#__RefHeading___Toc20283)

[2.5网站开发的设计思想 5](#__RefHeading___Toc25819)

[2.6系统设计的总体规则 5](#__RefHeading___Toc8367)

[3 系统的概要设计及数据库设计 7](#__RefHeading___Toc16844)

[3.1网站的功能模块划分 7](#__RefHeading___Toc16543)

[3.2数据库设计 7](#__RefHeading___Toc15830)

[4 系统详细设计 10](#__RefHeading___Toc4869)

[4.1主要功能流程图 10](#__RefHeading___Toc32443)

[4.2功能总体说明 10](#__RefHeading___Toc31076)

[4.2.1目录结构说明 10](#__RefHeading___Toc637)

[4.2.2功能页面结构 11](#__RefHeading___Toc9386)

[4.3各功能模块的实现 11](#__RefHeading___Toc19368)

[4.3.1用户登陆模块的实现 11](#__RefHeading___Toc7244)

[4.3.2用户注册模块的实现 13](#__RefHeading___Toc20213)

[4.3.3 视频预览点播模块的实现 15](#__RefHeading___Toc10478)

[4.3.4视频上传的实现 15](#__RefHeading___Toc26135)

[4.3.5用户点播模块的实现 17](#__RefHeading___Toc10894)

[5 系统发布及性能分析 19](#__RefHeading___Toc4554)

[5.1系统运行环境的搭建 20](#__RefHeading___Toc1246)

[5.1.1 JAVA虚拟机的安装 20](#__RefHeading___Toc7476)

[5.1.2 Tomcat环境搭建 20](#__RefHeading___Toc31662)

[5.2系统程序的安装和加载 20](#__RefHeading___Toc13628)

[5.2.1文件的拷贝 20](#__RefHeading___Toc26447)

[5.2.2数据库的连接 20](#__RefHeading___Toc23505)

[5.2.3系统的性能分析 21](#__RefHeading___Toc10576)

[6 总 结 22](#__RefHeading___Toc7732)

[参考文献 23](#__RefHeading___Toc19476)

[致　谢 24](#__RefHeading___Toc28352)

# 1 绪论

## 1.1 地理设计软件概述

早在十年前，互联网已经以日新月覆的速度改变着人类的生活，无论是生活，行为习惯，无一没有发生质的变化。与此同时，互联网技术的发展也十分惊人，从最开始的WEB1.0时代到现在WEB2.0时代，甚至还在展望的WEB3.0.，都在响应着“科技改变世界”这一崇高的号召。如今，人们早就不满足在电脑上面浏览互联网，在方便小巧的手机上，也能体验互联网的精彩，某些方面，设置有着比电脑更好的体验，例如，通信，社交等。

2006年，WEB领域两大巨头Web Hypertext Application Technology Working Group和World Wide Web Consortium宣布合作，经过了8年的努力，在2014年完成了HTML5的定稿，从此，强大的HTML5诞生了，它强大的功能特性令所有的WEB 开发者为之振奋，利用HTML5，可以开发出十分完善的WEB APP。其中通过接口，可以在浏览器中获取到用户的地理定位坐标信息。而通过CSS样式表的优化，WEB APP能在全平台支持HTTP浏览器的环境运行，真正的跨平台。

而我国互联网的发展现状，同样能让人震惊，截至到2015年3月，中国网民已经超8亿，覆盖面更是覆盖了全行业，全年龄。面对这样庞大基数的人数，中国互联网的潜力不可小看。而近年来，互联网行业已经是中国的新兴行业。

站在这个新兴行业的浪潮里面思考，无处不是机遇和挑战。考虑到现在我国互联网的社交领域，形式和内容或多或少有些单调， 所以，地理社交这个课题是具有十分巨大的潜力的。

本次毕业设计的任务是，构建一个地理社交网站，运用SPA（单页面应用），也可以称之为WEB APP，采用的主要技术是基于服务器的Node.js平台， postgresql数据库，以及html5前沿技术，angular等。

Node.js是2009年Ryan Dahl开始开发的项目，旨在开发一个非阻塞的轻量级Web服务器，并提供一系列库，项目一经发布，十分火热，2010年，Node.js获得[云计算](http://baike.baidu.com/view/1316082.htm)服务商Joyent资助，创始人Ryan Dahl加入Joyent全职负责[Node.js](http://baike.baidu.com/view/3974030.htm)的发展。2011年7月，Node.js在微软的支持下发布[Windows](http://baike.baidu.com/view/4821.htm)版本。

说道非阻塞，Node.Js利用对谷歌浏览器Chrome的javascript 引擎V8的封装，V8本身的设计十分优良，使用了一些最新的编译技术，使得Node.js在其上面运行得十分迅速， 效率。同时，非阻塞的javascript简化了对慢资源的访问，减少了等待时间，将cpu效率发挥到了极点，使得Node.js 拥有了其他web服务器技术无法相比的优势。

Node.js作为服务器，有以下几点优点。

1、Restful Api：先进优良的设计模式，对rest（访问控制状态转移）是十分友好，利用Node.js原生的库类， 可以十分轻易得开发restful服务api。

2、单线程： Node.js可以在不新增额外线程的情况下，依然可以对任务进行并行处理 —— Node.js是单线程的。它通过事件轮询（event loop）来实现并行操作，对此，我们应该要充分利用这一点 —— 尽可能的避免阻塞操作，取而代之，多使用非阻塞操作。

3、非阻塞IO：Node采用一系列“非阻塞”库来支持事件循环的方式。本质上就是为文件系统、数据库之类的资源提供接口。向文件系统发送一个请求时，无需等待硬盘（[寻址](http://baike.baidu.com/view/1303626.htm)并检索文件），硬盘准备好的时候非阻塞接口会通知Node。

4、V8虚拟机： V8的开发团队的领导者正是JVM的领导者，他将JVM的开发经验带到了V8，使得V8这个引擎十分效率。

5、事件驱动：Node采用了一个称为“事件循环(event loop）”的架构，使得编写可扩展性高的服务器变得既容易又安全。提高服务器性能的技巧有多种多样。Node选择了一种既能提高性能，又能减低开发复杂 度的架构。这是一个非常重要的特性。并发编程通常很复杂且布满地雷。Node绕过了这些，但仍提供很好的性能。

1.2系统开发背景。

1.2.1 地理社交的发展。

近年来，随着电子设备软件和硬件的不断发展，无论是电脑主机，笔记本电脑，还是手机，平板都能Internet为人类所带来的便利。即使是足不出户，都能通过互联网尽知天下事。而新型的社交平台出现，更是不断冲击着传统的通信市场。

近年来的微博，微信，已经在社交领域打开了一片客观的市场，改变着许许多多人的生活，获得了许许多多人的承认。

设备的不断更新换代，人们已经不能满足与较为传统的社交方式，地理社交这个名词被搬上了互联网平台。通过地理信息来交友等社交操作，将会吸引不少人，同时，这种崭新的社交方式将会更大的丰富人们的网上冲浪体验。

1.2.2 地理社交的现状

在地理社交领域，国内，微信的附近的人，还有陌陌APP ，国外，google的google+ ，都对地理社交作出了不少开拓性的尝试。但是仍没有在社交领域获得突破性的进展，而国内，仍然处于基于地理位置的搜索用户，能进行创新的空间仍然很大，有很大的机遇。

1.3系统实现的目标

现在流行的网站和APP 不仅要有漂亮的UI设计，更是要注重用户的体验，人性化的设计，当然，还要有齐全的功能，不然，一切都是纸上谈兵。

要设计出一个功能齐全的系统，要有非常严谨的规划。有这样的一个观点，程序设计有80%是在设计，有20%是在编码。在系统的开发周期，早期有良好的规划是非常重要的，应该注重软件工程，善于利用UML为系统进行设计，详尽得列举用例等。

在这个系统中，使用HTML5构建系统的前端，用Node.js来搭建后台服务器，使用ajax 进行交互。Node.js使用Postgresql数据库构建GIS服务，Postgresql这个数据库是一个强大的对象数据库，关系型数据库，对地理信息的支持非常好，效率非常高。而Node.js的异步非阻塞，则为服务器的高并发奠定了坚实的基础，使系统可以提供非常稳定的服务。

1.4系统的开发意义

互联网的发展一波接着一波，技术日新月异，而人们的需求也急速上升。近年来，移动互联网的火爆，令人们急剧转变思维，号称互联网思维，想在互联网中开拓出点东西来。而地理社交是这股移动互联网浪潮中颇受关注的事物，利用地理信息作为基点进行社交活动，也许会冲击人们的生活方式，思考方式。

当今知名度最高的地理社交也许是微信中的一项功能“寻找附近的人”吧，还有“陌陌”这款APP吧。它们都是相当优秀的产品，都经过了市场的考验。设计中可以学习他们的优秀的思想和模式，借鉴它们的功能，适当注意一下整个系统的注意事项。

面对这样的一个地理社交网站，将会牵涉到许多技术上的问题，例如在postgresql中进行地理数据的操作，基于地理位置的查找，数据表的设计，跨表查询，前端页面的设计，单页面应用的设计等等，通过对整个系统的设计和实现，可以对技术上的问题进行一次实战，掌握或巩固上述技术的使用。

2地理社交系统分析

系统分析是项目开发中必不可少同时也是非常重要的一个环节，系统设计出来能不嫩更好得贴近实际，获得更好的运行效率，很大程度上就取决与系统分析得恰不恰当，合不合理，因此，对系统的功能要进行细致的分析，琢磨其中的业务，尽可能得了然于胸，才能够开发出完整实用的系统。

2.1软件的设计目标

本系统利用现在比较流行的Node.js后段技术，搭配Postgresql数据库实现的，系统主要分为两个部分。前端和后端，前端是用户界面，用于提供给用户操作使用。包括：注册，登陆，填写用户详细信息，用户上传和修改，查看附近所发的博文，查看博文，查看其他用户，请求添加好友等等。

后端则是服务器，提供给前端操作的rest api，同时开放一定的公共api 接口供外部程序使用。

2.2系统的可行性分析

时代在变化，计算机，互联网也不断变化，现在的互联网是变得越来越广播，越来越丰富。社交是现代人们使用互联网中必不可少的一部分，互联网上的社交虽然也曾经受到一些人的质疑，但是其中的魅力是无法阻挡的，社交为人们的交际提供了另一片天地，；另一片空间。使一些人打开了心扉，治愈了心灵。而现在，传统的社交方式已经满足不了人们的需求，互联网需要创新，需要多元化。

而地理社交方式正是迎合了这样的理念，在现在互联网中饱受关注，通过地理社交，人们可以丰富传统的社交方式，获得更好的用户体验，扩展了交际圈。

可行性研究使分析系统开发出来的项目是不能值得进行研究的过程，实际上也是一次系统分析简化的的过程，通过系统可行性分析研究，可以对系统的整体设计有更深的了解。因此，进行可行性分析使非常有必要而且使非常重要的。经过最初的设计目标和进行的市场调查得出以下四点的可行性分析：

1、经济可行性：由于完备系统的复杂性，而且仅仅使作为个人的毕业设计，加上时间仓促，个人精力有限，功能并不能像线上项目一样完善。所以没有经济效益可言。

2、技术可行性：从后端来说，虽然Node.js使新生的平台，但是经过了几年的火爆发展，社区的热情支持，技术上已经非常完善，成熟，网上的文档也足够丰富，库类也逐渐完善，所以使具备可行性的。而从前端来说，将阿javascript已经非常成熟，而我本身具有一定的单页面设计经验，所以也是具有可行性的。

3、运行可行性：一切皆运行在Linux系统上，系统需要搭建Postgresql，postgis服务，node.js，还有node的包管理器为NPM。上述需求皆有详细的文档，本人有比较丰富的linux 系统管理经验，具备运行可行性。

4、法律可行性：一切皆为开源，也与商业无关，没有盈利性质，同时为自主开发设计，不存在侵权的问题，因为在法律上可行的。

通过上述的可行性分析研究，我将采用Node.js + postgresql + postigs + angular.js + html5这个技术栈。

2.3网站的深入调查

经过以上的初步调查和研究，而且为确定了这个地理社交系统的可行性，并vibg决定开发本平台，为了能够对现行的平台内容进行充分了解，使所要开发的平台更具有实用性和适应性，通过两个月的实习调查和指导老师的耐心介绍，设计出该平台功能如下：

1.、用户能够完成用户注册，登陆功能。

2、用户能够查看地图，显示自己所在的位置，能够拖动地图，缩放地图。

3、用户能够填写关于自己的详细信息，上传到服务器。

4、用户能够上传自己的头像。

5、地图上能够显示用户周围为其他用户所发的博文，并能点击查看，再点击可以查看详细的信息。

6，用户能够查看其他用户的详细信息。

7、能够发送好友请求，对方若是同意，则可以加为好友。

2.4系统设计的特点

这个GIS地理社交系统使经过很长一段时间的调查和分析才具体开始实施的，它的目的是丰富现在互联网社交的模式，采用web技术，借助Internet的传播，实现地理社交广泛传播，使的人们的社交方式更为丰富，可以通过地理位置认识更广泛的朋友，丰富人脉圈，建立更为稳定的社区人脉关系，使现实生活中的社区更为团结，构建更好的内涵。

2.5系统开发的设计思想

基于GIS的基于地理位置的社交平台主要分为两个主要部分。

一是前端基于HTML5的客户端，用户可以通过客户端注册和登陆，查看显示出自己地理位置的地图，浏览附近为其他注册用所发的博文，并能对其博文进行详细浏览。也可以对博文发表评论，点击用户查看用户详细信息，若是觉得投缘，可以申请添加好友，对方若是接受，双方可以成为好友，之后可以进行互相发送信息。

二是构建在Node.Js平台之上的后台服务端，对前端客户端提供功能支持，使用Postgresql数据库进行信息储存，并使用Redis 对短生命周期数据进行缓存，以减轻对数据库的压力，提供更好的高并发服务。而提供的服务采用Rest Api提供，使地Api风格比较明朗和清晰，符合现代Api开发的直观，简单，表述性状态等要求。

2.6 系统设计的总体规划

高效率的平台都得要有自己的设计原则，开发模式，本平台贯彻的思想可以概括为以下几个方面。

一、简单，即所谓的Less is more！简单是指设计模式简单，在保持一定的可维护性和一定的可扩展性之后，没有采用复杂花俏的设计模式，保证系统的简单性，是因为考虑本平台的开发目的，系统架构的简单性是非常重要的。

二、功能性，本平台坚持代码所实现的功能比其他东西都重要，没有完成功能的代码只是一个空壳，根本没有丝毫用武之地。

三、实用性，一切皆从实际出发，任何的设计和功能都围绕功能展开，确保整个平台的完备，能实现所有应该实现的功能。

3平台的概要设计以及数据库设计

3.1平台的功能模块划分

经过前期的深入调查和研究，总结出该平台需要完成的一些具体功能如下：

（1）用户注册和登陆，功能实现用户录入基本信息完成注册流程，并能通过注册时填写的用户名或邮箱地址和密码进行登陆。

（2）用户详细信息录入和修改，功能实现用户录入详细信息和上传头像，修改后能进行修改。

（3）用户在自己所在位置上发表短篇幅的博文：客户端获取用户的地理位置信息，并随用户发表的博文内容一起上传到服务器上，服务器将地理信息和博文写入数据库，以地理信息的数据类型保存。

（4）在地图上显示博文数据给用户：用户在浏览拖拽地图时，向服务器检索地图博文数据并上传客户端地图上的中心点的坐标，服务器根据此坐标向Postgresql检索数据，并将数据发送到客户端，在客户端上的地图中显示出来。

（5）博文的评论：用户在地图上浏览博文上，可以查看更为详细的博文信息，并可以在博文下进行评论，上传到服务器保存在数据库中。

（6）

3.2 数据库设计

在搭建后端数据库之前，我们必须为系统设计数据库结构，一定要做到清晰明了，划分明确，适应系统的功能设计，同时又不能产生结构上的逻辑混乱，并要兼顾数据库的优化和性能。这一步，可以说是整个系统设计的最重要的一环。

本系统选择的数据库版本为Postgresql server 9.3版本。

以下是对本系统中各种数据信息进行考察分析后设计出来的数据库表

表3.1用户基本信息表(User)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段 | 字段名 | 类型 | 宽度 | 小数位 | 索引 | 可否为空 |
| id | id编号 | interger |  |  | 是 | 否 |
| username | 用户名 | character | 255 |  | 是 | 否 |
| Password | 密码 | character | 255 |  |  | 否 |
| email | 邮箱 | character | 255 |  |  | 否 |
| limit | 是否限制 | charater |  |  |  | 否 |
| created\_at | 创建时间 | Time |  |  |  | 否 |
| update\_at | 修改时间 | Time |  |  |  |  |

表3.2用户详细信息表(UserDetail)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段 | 字段名 | 类型 | 宽度 | 小数位 | 索引 | 可否为空 |
| user\_id | 用户编号 | bigint | 20 |  | 是 | 否 |
| nickname | 用户名称 | varchar | 50 |  |  | 否 |
| sex | 用户 | int | 11 |  |  | 否 |
| avatar |  |  |  |  |  |  |
| self\_description |  |  |  |  |  |  |
| city |  |  |  |  |  |  |
| school |  |  |  |  |  |  |
| interest |  |  |  |  |  |  |
| public\_email |  |  |  |  |  |  |
| photo\_nu,ber |  |  |  |  |  |  |
| qq |  |  |  |  |  |  |
| created\_at |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Markpost

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段 | 字段名 | 类型 | 宽度 | 小数位 | 索引 | 可否为空 |
| id | 用户编号 | bigint | 20 |  | 是 | 否 |
| type | 用户名称 | varchar | 50 |  |  | 否 |
| user\_id | 用户 | int | 11 |  |  | 否 |
| title |  |  |  |  |  |  |
| context |  |  |  |  |  |  |
| images |  |  |  |  |  |  |
| location |  |  |  |  |  |  |
| longitude |  |  |  |  |  |  |
| latitude |  |  |  |  |  |  |
| accuracy |  |  |  |  |  |  |
| valid |  |  |  |  |  |  |
| deadline |  |  |  |  |  |  |
| created\_at |  |  |  |  |  |  |
| update\_at |  |  |  |  |  |  |

MarkPostComment

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段 | 字段名 | 类型 | 宽度 | 小数位 | 索引 | 可否为空 |
| id | 用户编号 | bigint | 20 |  | 是 | 否 |
| Markpost\_id | 用户名称 | varchar | 50 |  |  | 否 |
| User\_id | 用户 | int | 11 |  |  | 否 |
| context |  |  |  |  |  |  |
| created\_at |  |  |  |  |  |  |
| updated\_at |  |  |  |  |  |  |

Friends

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段 | 字段名 | 类型 | 宽度 | 小数位 | 索引 | 可否为空 |
| id | 用户编号 | bigint | 20 |  | 是 | 否 |
| User\_id | 用户名称 | varchar | 50 |  |  | 否 |
| Friend\_id | 用户 | int | 11 |  |  | 否 |
| friendlines |  |  |  |  |  |  |
| isHidden |  |  |  |  |  |  |
| created\_at |  |  |  |  |  |  |
| updated\_ |  |  |  |  |  |  |

FriendRequest

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段 | 字段名 | 类型 | 宽度 | 小数位 | 索引 | 可否为空 |
| id | 用户编号 | bigint | 20 |  | 是 | 否 |
| requestUser\_id | 用户名称 | varchar | 50 |  |  | 否 |
| tarUser\_id | 用户 | int | 11 |  |  | 否 |
| status |  |  |  |  |  |  |
| message |  |  |  |  |  |  |
| created\_at |  |  |  |  |  |  |
| updated\_at |  |  |  |  |  |  |

4系统详细设计

4.1主要功能流程图

基于GIS地理社交主要功能流程图如图4.1

4.2功能总体说明

4.2.1目录结构说明

（1）客户端

1、templates用来存放html模板文件。

2、img用来存放客户端所需要的图片资源。

3、fonts用来存放客户端所需要的字体资源。

4、compents用来存放客户端所需要的第三方库。

5、scss用来存放scss文件。

6、css用来存放css样式表。

7、js用来存放javascript文件。

（2)服务器

1、models与数据库通信的ORM对象关系模型层。

2、services集成models层与服务器各种组件的服务层。

3、routes系统的url路由控制层。

4、socket系统与客户端进行web socket连接组件。

5、auth基于用户的认证组件。

6、databse数据库迁移和控制模块。

7、certs存放ssl证书。

8、utils通用工具组件。

9、node\_modules系统所需要的npm模块包目录

10、test系统测试程序。

4.2.2客户端页面结构

由于WEB客户端是单页面应用，页面结构以URL Hash值表示。

功能页面

主页面：index.html

注册：/#rsignin

登陆：/#login

填写详细信息：/#detail

上传或修改头像：/#avatar

地图页面：/#map

个人信息查看或修改：/#profile

查看详细的博文信息：/#markpost/:markpostId

查看其他用户详细信息：/#user/:userd

查看新信息页面：/#message

4.3各功能模块的实现

4.3.1用户登陆模块的实现

用户的认证是防止非法用户登陆的第一道防线，同时也能为各用户之间划分资源，方便管理，提供安全的访问权限控制。当用户通过客户端对服务器受保护资源进行访问或者上传时，系统会根据客户端发送的http报文中头部的sessionId来从redis中检索session，若是session则允许用户继续访问，若是失效或者不存在则拒绝用户访问并通知客户端重定向至登陆页面。

图4.2 用户登陆流程图

4.3.2 用户注册模块的实现

如果用户还没有注册的时候则提示用户可以注册，用户在填写表单并点击注册按钮之后，客户端判断用户注册信息是否有效，若是有效则发送给服务器，服务器判断用户名是否有效，是否重名等，若是有效，则在数据库中为用户创建数据，并将成功信息反馈给用户，提示用户注册成功。

4.3.3用户发表博文模块的实现。

做为一个基于GIS的地理社交系统，在地图上发表博文和非常重要的功能。

4.3.4在地图中显示博文信息模块的实现。

4.3.5用户在博文下评论功能模块的实现。

5 系统发布以及性能分析。

5.1系统运行环境的搭建。

5.1系统运行环境的搭建。

5.1.1 Node.Js平台的安装。

本系统运行在linux系统Centos7下，Node.js的安装步骤为：

（1）前往[http://nodejs.org/dist/](http://nodejs.org/dist/w)这个网页在其中选择0.12.1的版本下载，注意下载后缀名为.tar的包。

（2）解压tar包，输入命令./configure && make && sudo make install 进行编译并安装，至此Node.js与Npm已经安装完成。

5.1.2 postgresql数据库的安装

打开终端模拟程序，输入以下命令：

sudo yum install postgresql-server9.3 postgis-2.1

命令运行成功后，数据库即安装完成。

5.2系统程序的安装和加载。

5.2.1程序的放置

将系统文件放置在/user/local/下。

5.2.2数据库的连接。

本系统使用Postgresql作为数据库，系统所用到的表都在其中，这里的数据库名称为Map，用户名为map，密码设置为简单的开发强度密码admin123。

5.2.3系统的性能分析。

6总结

我的毕业设计论文具体阐述了基于GIS地理社交系统的主要功能和体系结构，以较重的笔墨讨论了系统的整天设计和可行性分析，数据库的设计，还有各模块的具体设计，前后端的通信和响应。

# 参考文献

1. Bruce Eckel, Think in Java. Prentice Hall.2006.2: 19~67
2. Brett McLaughlin, Java and XML. O’Reilly Media. 2006: 144~193
3. 清宏计算机工作室. JSP编程技巧[M]. 北京：机械工业出版社，2000：52~60
4. 周影．网络编程语言JSP实例教程[M]. 北京：电子工业出版社，2003：258~86
5. 陈云芳. 精通Strurt 2:基于MVC的Java Web应用开发实战. 北京：人民邮电出版社，2002：201~220
6. Donny. JSP与网站开发编程实战[M].北京：科学出版社，2001：150~163
7. Steven Holzner.Java技术内幕[M].北京：机械工业出版社，2002：412~420
8. Phil Hanna. JSP技术大全[M].北京：机械工业出版社，2002: 345~360
9. Karl Moss. Java Servlets(tm) Developer's Guide. McGraw-Hill/Osborne Media: 222~234
10. Chuck Canvaness, Brian Keeton. Jakarta Struts Pocket Reference. O'Reilly Media, 2003: 192~267
11. 弗拉.纳根．JavaScript权威指南.北京：机械工程出版社，2007：125~230
12. [杨中科](http://www.amazon.cn/mn/searchApp?searchWord=杨中科)．J2EE开发全程实录[M].北京：清华大学出版社，2007：354~375
13. Sangeeth. 深入浅出 J2EE 架构[M].北京：清华大学出版社，2006：232~260
14. Horstmann.C.S．Core Java Volume Ⅱ: Advanced Featuers, Eighth Edition. Prentice Hall, 2009.9: 57~87
15. 郑阿奇．MySQL实用教程[M].北京：电子工业出版社，2009：253~345
16. 何丽．精通DIV+CSS网页样式与布局.北京：清华大学出版社，2011：110~321

# 致　谢

经过两个多月的努力，自己终于完成了毕业设计和毕业论文的写作工作，在这里我要感谢我的设计指导老师杨老师，是他在论文的选题及技术方向上给我提出许多宝贵的设计意见，在最后的测试修改阶段又在百忙之中抽出时间为我提供了必要的帮助，这样使得我得以顺利的完成毕业设计开发工作，老师渊博的知识、敏锐的思路和实事求是的工作作风给我留下了深刻的印象，这将使得我终身受益，谨此向老师表示衷心的感谢。

感谢在校的老师们，在我大学四年里对我的谆谆教诲和无私帮助，在实际的开发工作中，我深深体会到了基础理论知识的重要性，在大学阶段只有好好学习，掌握好扎实的理论基础，这样才能在将来的工作中更快的学习，从而得心应手，有所成就，真正成为一个对社会和他人有用的人。

在这里还要班上的同学们，不论在系统的具体开发过程中，还是在论文的排版上，是他们给予我了精心的指导和热情的帮助，使我得以顺利完成毕业设计。

最后还要感谢我的同学们，在这次毕业设计及论文写作中给我的支持和帮助。