目录

[1 绪论 2](#_Toc525633444)

[1.1 研究背景及意义 2](#_Toc525633445)

[2 开发环境和相关技术介绍 3](#_Toc525633446)

[2.1 Mongo数据库介绍 3](#_Toc525633447)

[2.2 Node.js开发技术介绍 4](#_Toc525633448)

[3 系统需求分析 7](#_Toc525633449)

[3.1. 系统功能概述 7](#_Toc525633450)

[3.2. 系统功能模块介绍 7](#_Toc525633451)

[4 系统设计与实现 9](#_Toc525633452)

[4.1 数据库设计 9](#_Toc525633453)

[4.1.1 概念结构设计 9](#_Toc525633454)

[4.1.2 逻辑结构设计 9](#_Toc525633455)

[4.1.3 完整性设计 9](#_Toc525633456)

[5 已完成的功能模块 11](#_Toc525633457)

[5.1 登录注册模块 11](#_Toc525633458)

[5.2 首页模块 12](#_Toc525633459)

[6 剩余开发计划 13](#_Toc525633460)

# 绪论

## 研究背景及意义

个人博客是博客的一种。自建独立网站的博客主，有自己的域名、空间和页面风格。个人博客是个人持续性的日记或评论。它是最传统常见的博客。个人博客作者一般注重博客的内容，即使他们的博客从来没有被除了自己以外的人阅读过。博客经常不只是一个沟通的方法，它们成为了一种反映生活的方法或艺术创作。编写博客旨在沉淀技术与。很少个人博客出名和成为主流，但一些个人博客很快就聚集了许多读者。

# 开发环境和相关技术介绍

## Mongo数据库介绍

mongoDB是一个开源的，基于分布式的，面向文档存储的非关系型数据库。是非关系型数据库当中功能最丰富、最像关系数据库的。mongoDB由C++编写，其名字来源于"humongous"这个单词，其宗旨在于处理大量数据。mongoDB可以运行在Windows、unix、OSX、Solaris系统上，支持32位和64位应用，提供多种编程语言的驱动程序。mongoDB支持的数据结构非常松散，是类似json的BSON格式，通过键值对的形式存储数据，可以存储复杂的数据类型[7]。

mongoDB支持的数据类型有：null、boolean、String、objectId、32位整数、64位整数、64位浮点数、日期、正则表达式、js代码、二进制数据、数组、内嵌文档、最大值、最小值、未定义类型。

其中，内嵌文档我理解的并不是.doc.txt等文件，这里所指的文档是mongoDB的一个存储单元(相当于关系型数据当中的记录)，在mongoDB中的表现形式为{key1:value1，key2：value2}，而内嵌文档则是这样的形式{key1:value1,key2:{key2.1:value2.1,key2.2:value2.2}}。

mongoDB最大的特点是他支持的查询语言非常强大，其语法有点类似于面向对象的查询语言，几乎可以实现类似关系数据库单表查询的绝大部分功能，而且还支持对数据建立索引。

和其他数据库系统相比，Mongodb有点与众不同，它属于 NoSQL 类型的数据库，它的架构可以在多种不同场景中应用并发挥好的作用，但同时也会带来一点选择上的困难。MongoDB并不完美，却足够灵活，能够适应高要求的环境，例如Web类应用。同时，Mongodb既可以嵌入到应用程序中，也可以支持数据仓库、内容索引和部署软件、高可用的冗余系统、在线事务处理系统(OLTP)等各种应用类型。

**mongoDB的特性：**

1. 面向集合存储。数据被分组到若干集合，每个集合可以包含无限个文档，可以将集合想象成RDBMS的表，区别是集合不需要进行模式定义。

2. 模式自由。集合中没有行和列的概念，每个文档可以有不同的key，key的值不要求一致的数据类型。

3. 支持动态查询。mongoDB支持丰富的查询表达式，查询指令使用json形式表达式。

4. 完整的索引支持。mongoDB的查询优化器会分析查询表达式，并生成一个高效的查询计划。

5. 高效的数据存储，支持二进制数据及大型对象（图片、视频等）。

6. 支持复制和故障恢复。

7. 自动分片以支持云级别的伸缩性，支持水平的数据库集群，可动态添加额外的服务器。

## Node.js开发技术介绍

Node.js是基于Chrome JavaScript运行时建立的一个平台，实际上它是对Google Chrome V8引擎进行了封装，它主要用于创建快速的、可扩展的网络应用。Node.js采用事件驱动和非阻塞I/O模型，使其变得轻微和高效，非常适合构建运行在分布式设备的数据密集型实时应用[8]。

运行于浏览器的Javascript，浏览器就是Javascript代码的解析器，而Node.js则是服务器端JS的代码解析器，存于服务器端的JS代码由Node.js来解析和应用。

JS解析器只是JS代码运行的一种环境，浏览器是JS运行的一种环境，浏览器为JS提供了操作DOM对象和window对象等接口。Node.js也是JS的一种运行环境，node.js为JS提供操作文件、创建http服务、创建TCP、UDP服务等接口，所以Node.js可以完成其他后台语言能完成的工作。

Node.js提供了一个交互式运行环境 pepl，通过这个环境，可以立即执行JS代码，使用方法类似于Chrome浏览器中Firebug插件中的Console。

在Linux环境进入终端后，属于"node"或者“nodejs”进入Node.js的交互式运行环境，Ctrl+d可以退出此环境。

查看系统中安装的Node.js版本：node -v or nodejs -v

运行JS文件，eg:node file.js or nodejs file.js

Node.js官方提供了很多模块，这些模块分别实现了一种功能，如操作文件模块fs,构建http服务模块的http等，每个模块都是一个JS文件，当然也可以自己编写模块。

包可以将多个具有依赖关系的模块组织在一起，封装多个模块，以方便管理。Node.js采用了CommonJS规范，根据CommonJS规范规定，一个JS文件就是 一个模块，而包是一个文件夹，包内必须包含一个JSON文件，命名package.json。一般情况下，包内bin文件夹存放二进制文件，包内的lib文件夹存放JS文件，包内的doc文件夹存放文档，包内的test文件夹存放单元测试。package.json文件中需要包含的字段及包的使用。

npm是node.js的包管理工具，npm定义了包依赖关系标准，我们使用npm主要用来下载第三方包和管理本地下载的第三方包。

nodejs开发工具很多，比较常使用的IDE有VScode和webstrom。VScode比较流行，各种插件也多，当然用起来也不错；介绍这两个IDE主要是开源的，当然还有其它工具如IntelliJ、Sublime。这些IDE各有优势劣势，根据你开发的项目来选择了。本次系统的开发工具使用的是VSCode。

# 系统需求分析

## 系统功能概述

从总体上考虑，系统应该实现下列功能：

* + - 1. 登录注册
      2. 个人页面个性介绍
      3. 查看某篇具体博文
      4. 查看博客列表
      5. 修改某篇博文
      6. 删除某篇博文

## 系统功能模块介绍

本系统的功能模块如下 ：

* + 1. 登录模块，登录包括用户名和密码验证，密码采用MD5加密，登录账号密码错误会有相应的提示。

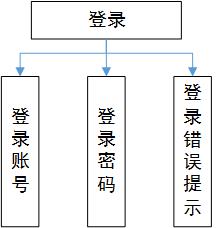


图3.3 系统登录模块

* + 1. 博客模块，用户登录后，拥有查看并管理相应的项目信息：

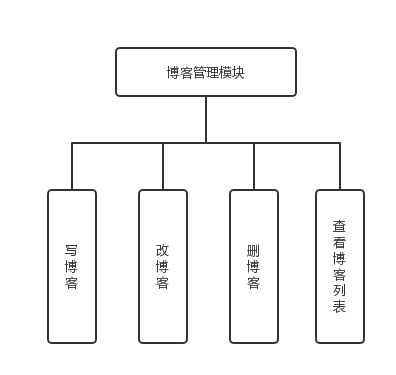


图3.4 博客模块

# 系统设计与实现

## 数据库设计

### 概念结构设计

E-R图也称实体-联系图(Entit y Relationship Diagram)，提供了表示实体类型、属性和联系的方法，用来描述现实世界的概念模型。构成E-R图的基本要素是实体型、属性和联系。

系统E-R图：

### 逻辑结构设计

在数据库的设计过程中，由E-R图向关系模型的转换是首要的工作。它将在概要设计阶段根据客户的业务和数据处理要求完成的E-R图按照3种完全不同的情况（1:1,1:n及m:n）转换成关系模型，从而最终在数据库管理软件中实现表的创建。

将全局E-R图转化为如下关系模型：

1. 用户表（用户编号（主键），账号名，密码）
2. 博文表（博客编号（主键），发布时间，归属目录，正文，标题）
3. 列表表（用户编号（主键）, 博客编号, 标题, 发布时间，归属目录）

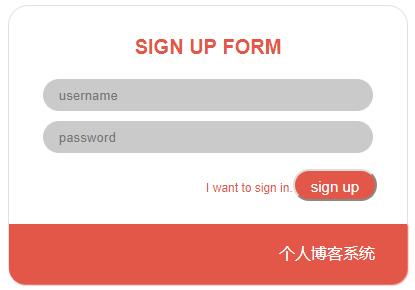
### 完整性设计

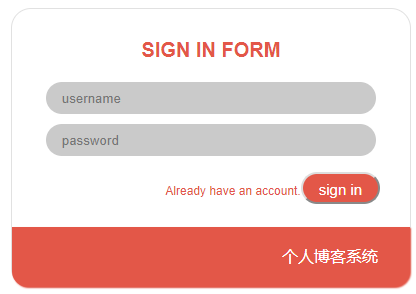
完整性设计主要是对数据库表的三种完整性进行设计：

1. 实体完整性：通过设置主键都已实现。
2. 参照完整性：用户角色对应表中的用户编号对应用户表中的编号、角色编号对应角色表中的编号。权限
3. 用户自定义完整性：在该系统中，设置了用户名唯一，用户组名唯一的约束。

# 已完成的功能模块

## 登录注册模块





## 首页模块



# 剩余开发计划

第四周（本周）完成列表模块和博文模块的前端后端开发

第五周完成数据库开发

第六周测试和完善报告