**Blogit**

**项目需求规格说明书**

15小组

**2020年3月11日**

目录

[1文档介绍 2](#_Toc34840139)

[1.1 文档目的 2](#_Toc34840140)

[1.2 文档范围 2](#_Toc34840141)

[1.3 读者对象 3](#_Toc34840142)

[2综合描述 3](#_Toc34840143)

[2.1 功能介绍 3](#_Toc34840144)

[2.2 功能框架 4](#_Toc34840145)

[3功能性需求 4](#_Toc34840146)

[3.1 用户登录模块 4](#_Toc34840147)

[3.2 好友管理模块 4](#_Toc34840149)

[3.3 动态发布模块 5](#_Toc34840150)

[3.4 个人信息管理模块 5](#_Toc34840151)

[4接口需求 6](#_Toc34840152)

# 文档介绍

## 1.1 文档目的

1. 分析设计，以本需求规格说明书为标准完成总体设计和详细设计；
2. 代码实现，以本需求规格说明书为标准，并结合总体设计、详细设计完成代码编写
3. 测试，以本需求规格说明书为标准，结合分析设计完成单元测试用例和系统测试用例编写和测试。

## 1.2 文档范围

需求规格说明书对Blogit产品下的各种功能模块的功能定义、接口定义、UI设计、以及其他研发约束条件等研发需求做了详细定义。

## 1.3 读者对象

1. 项目经理：项目经理可以根据该文档了解预期系统的功能，并据此进行系统设计、项目管理；
2. 设计人员：对需求进行分析，并设计出系统，包括数据库的设计；
3. 开发人员：配合《详细设计说明书》，了解系统功能，进行系统编码设计；
4. 用户：了解预期系统的功能和性能，并与分析人员一起对整个需求进行讨论和协商。

# 综合描述

## 2.1 功能介绍

实现一个类qq空间的blog，目标功能包含以下：

（1）登录注册

（2）对自己的动态及留言进行操作

（3）对好友的管理包括添加，删除，拉黑，分组

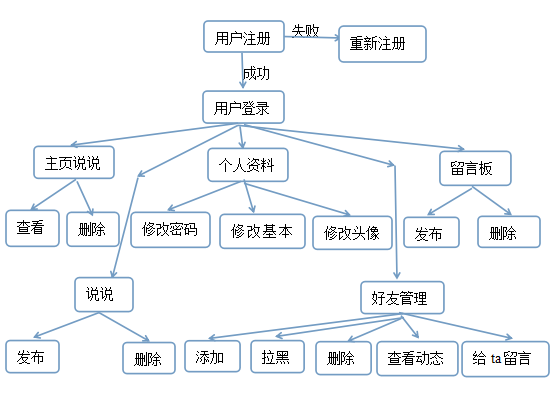
（4）根据登录的用户显示别人给自己的留言

（5）查看好友的动态以及留言

（6）个人资料的修改

（7）权限控制

## 2.2 功能框架



# 功能性需求

## 3.1 用户登录模块

### 3.1.1 新用户注册功能

通过用户名和密码进行注册，若有相同用户名，则提示用户名已存在，注册失败。

3.1.2 老用户登录功能

通过密码加密对用户进行注册以及判断登录，加强了对用户信息的安全强度。

## 3.2 好友管理模块

3.2.1 添加好友

在查询界面输入需要查询的用户名或账号，点击查询，若用户存在显示搜索结果，可点击添加好友。

3.2.2 拉黑好友

将好友添加进黑名单，移进黑名单分组。

3.2.3 删除好友

将好友移出好友分组。

3.2.4 查看动态

可以右击好友头像可进入好友blog，查看好友动态。

3.2.5 留言

在好友的blog留言板给好友留言，可留文字，图片，表情。

## 3.3 动态发布模块

3.3.1 心情说说发布

可发布文字，图片，表情。并且动态以及分条分页按时间顺序动态显示。

3.3.2 动态删除

可以删除指定动态。

## 3.4 个人信息管理模块

3.4.1密码修改

用户在个人资料界面点击修改按钮，出现修改密码和修改基本资料的界面。对密码进行修改，填写原始密码，新密码，重复新密码进行验证。

3.4.2 头像修改

在线预览功能选择上传文件，通过对文件后缀过滤只能上传指定格式的图片类型。选择后，实时预览上传的图片。点击上传，进行头像的修改，将头像名字保存在数据库。

3.4.3 基本信息修改

进入个人资料信息对生日、性别、签名等信息进行修改。

# 接口需求

Mybatis主要作为ORM层，负责与数据库交互相关的操作。 SpringMVC是REST协议在服务器端的一中封装方式，主要通过HTTP协议提供Web端数据处理的接口。

Spring做为中间层，主要使用依赖注入的方式耦合各层的对象。Controller层主要是封装服务器数据进行逻辑分析，做相关的数据处理，然后返回到Client端。Service层主要的功能是通过依赖注入的方式获取Mapper层的相关数据库对象的操作，一个重要的事情就是事务处理，保证多表同时操作时，数据库中的数据的一致性。同时Service层可以封装一些Controller层需要的操作，总而言之，Service层是对Controller层提供完善的数据处理的封装好的对象。

Mapper层主要是通过向数据库发送相应的sql语句，与数据库进行数据交换的接口。帮助Service层提供相关的数据交换的接口。