

Lethe Project

架构设计说明书

|  |  |
| --- | --- |
| 组长 | 15211105 于涌溢 |
| 成员 | 14211050 蒋松儒 15211105 于涌溢  15211092 闫涛 15211082 吴易泽  15211087 侯传嘉 14005025 黄邦靖  15211097 苟义钦 15211117 陈少华  15211123 吴鹏 15211076 练钊荣  15211035 刘希阳 15211093 何宇阳 |

2018年5月5日

目录

[1. 概述 3](#_Toc516321473)

[1.1. 编写目的 3](#_Toc516321474)

[1.2. 术语解释 3](#_Toc516321475)

[1.3. 参考资料 3](#_Toc516321476)

[2. 概要设计 4](#_Toc516321477)

[2.1. 系统架构图 4](#_Toc516321478)

[2.2. 系统功能结构图 7](#_Toc516321479)

[2.3. 部署结构图 7](#_Toc516321480)

[3. 主要业务流程 7](#_Toc516321481)

[3.1. 注册流程 7](#_Toc516321482)

[3.2. 登录流程 8](#_Toc516321483)

[3.3. 搜索联系人流程 9](#_Toc516321484)

[3.4. 添加联系人流程 10](#_Toc516321485)

[3.5. 删除联系人流程 11](#_Toc516321486)

[3.6. 发送消息流程 11](#_Toc516321487)

[3.7. 上传图片流程 12](#_Toc516321488)

[4. 数据库设计 13](#_Toc516321489)

[5. 开发工具 15](#_Toc516321490)

# 概述

## 编写目的

本说明书是即时通讯软件系统的开发概要设计，包括系统架构设计、数据结构设计等内容，用于指导后续的开发、测试等工作。

## 术语解释

集合（collection）：在MongoDB数据库中表示一组文档的集合，相当于关系型数据库中的表。

文档（document）：在MongoDB数据库中表示一组键值对形成的哈希表数据结构，是MongoDB中数据的基本单元，与JavaScript中的对象(Object)意义和结构相近，相当于关系型数据库中的记录。

对象唯一标识符(\_id)：MongoDB数据库为每个文档对象分配的不重复的id值。

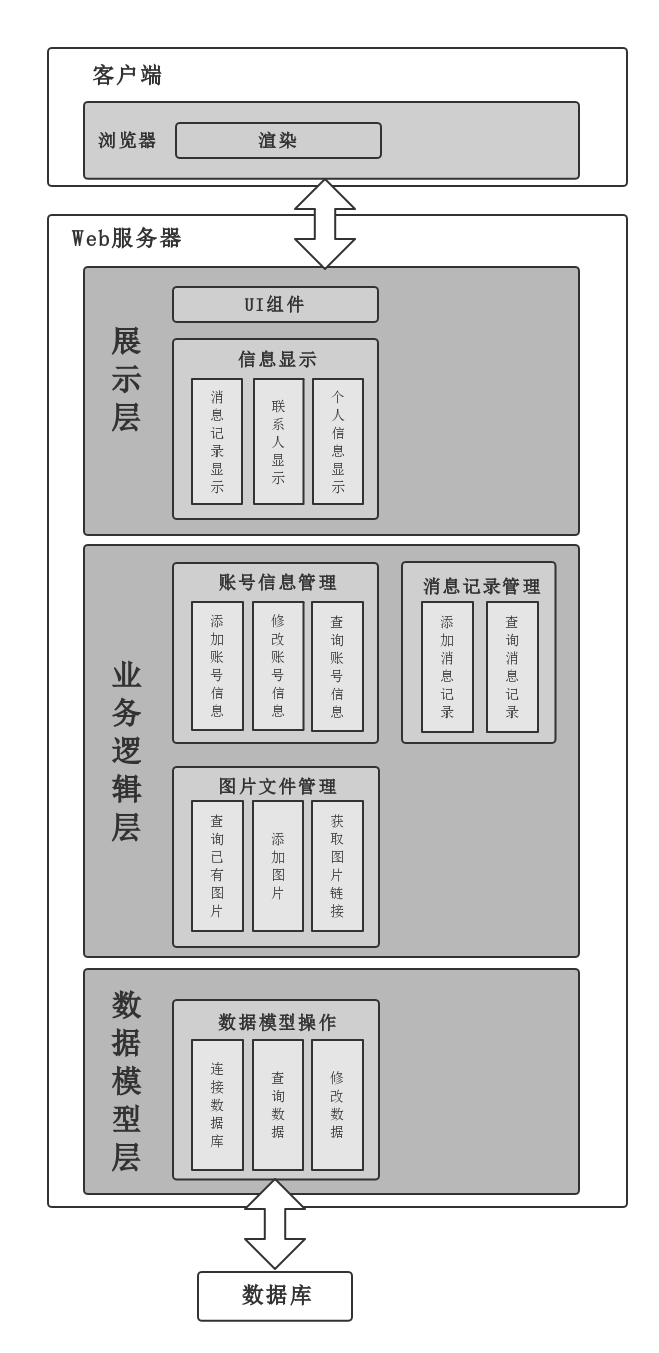
## 参考资料

<http://www.mongoing.com/articles/mongodb-terms>

<https://blog.csdn.net/u013830811/article/details/49180515>

# 概要设计

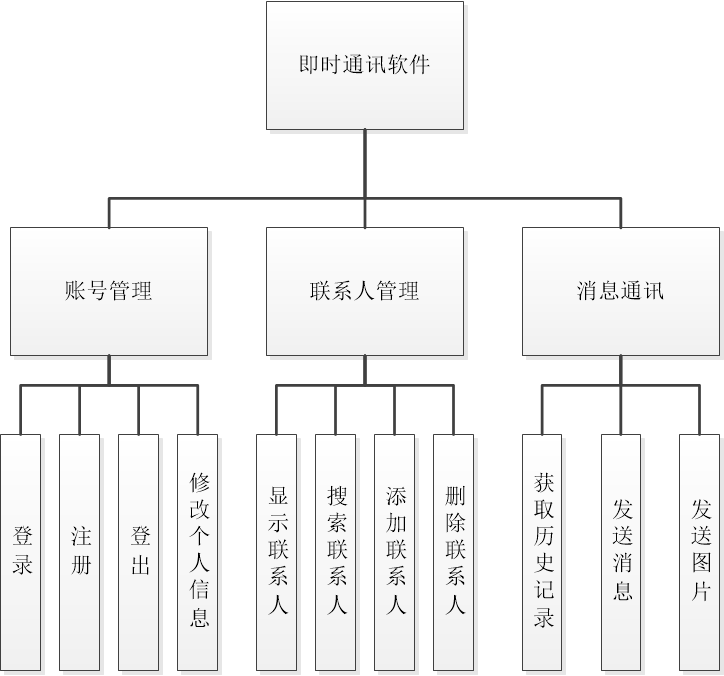
## 系统架构图



本系统采用典型的三层结构进行设计，即将整个系统分为展示层、业务逻辑层和数据模型层。

* 展示层即网页的前端部分，用于接受用户操作并向用户展示信息。该层通过调用业务逻辑层的接口获取用户的聊天记录、联系人信息、个人信息等数据并向用户展示。
* 业务逻辑层即网页的后端，其作用是为展示层提供接口，并通过数据模型层的接口与其进行交互以向展示层提供数据。该层定义了多个接口，包括账号信息的查询、添加、修改，消息记录的查询与添加和图片文件的查询、添加和获取。这些接口被调用后即会同数据模型层进行交互以进行相应的数据修改或查询。
* 数据模型层同样属于网页后端部分，但是该层专注于与数据库的交互，进行数据的存取操作。该层通过将数据库操作抽象为具体的功能，如增加图片、添加账号信息等，避免了数据库相关代码的重复编写。

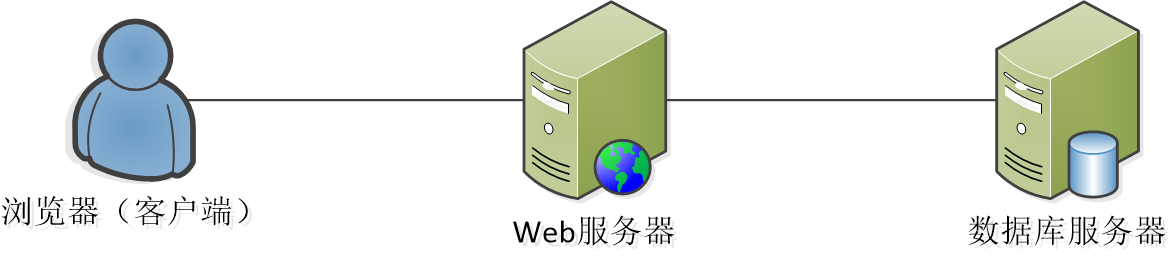
## 系统功能结构图



## 部署结构图

WebSocket服务器：需要配置和部署Web服务和node.js

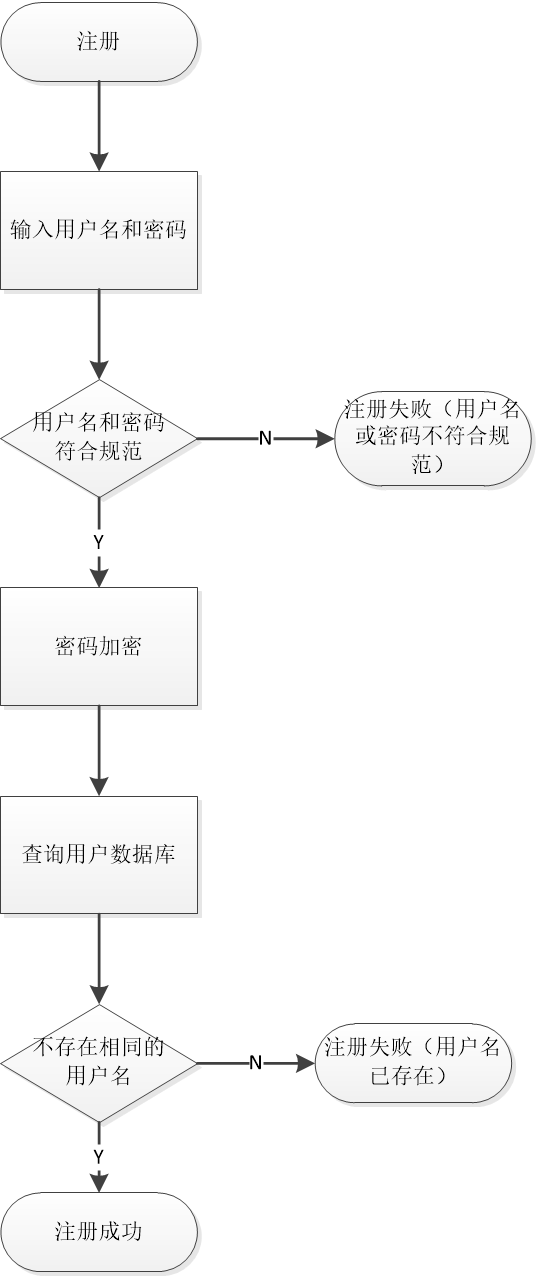
数据库服务器：需要配置和部署MongoDB



# 主要业务流程

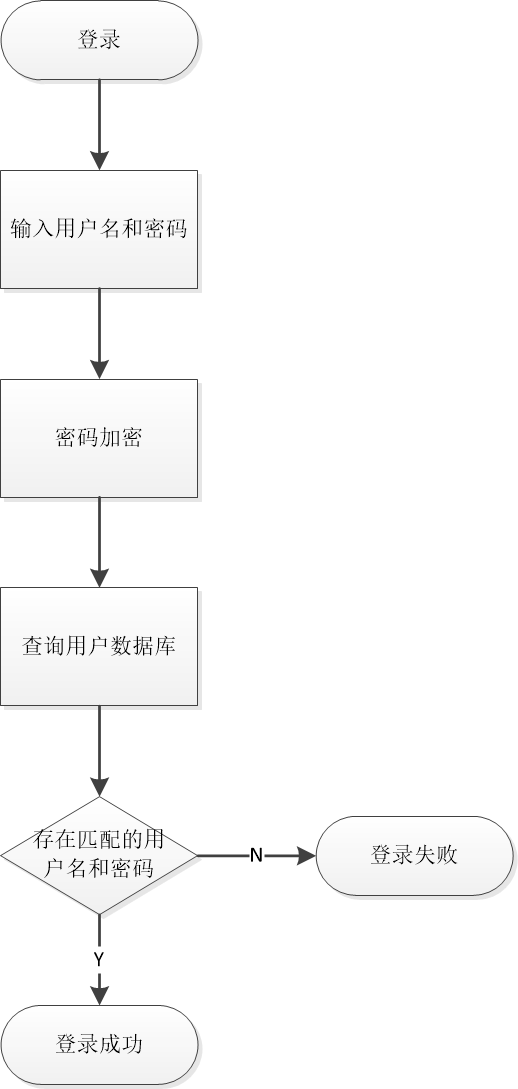
## 注册流程

该流程描述了用户通过用户名和密码建立新账号的功能，由于让用户的密码明文暴露在网络上是十分危险的，因此加入了密码加密步骤，其流程图如下所示：



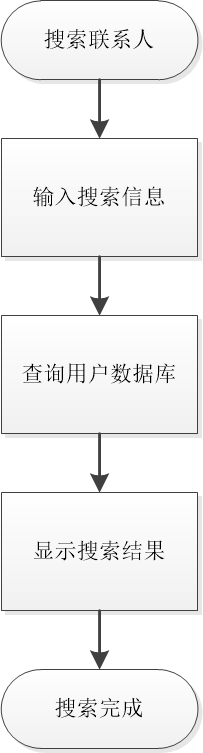
## 登录流程

该流程描述了用户通过用户名和密码登录已有账号的功能，其流程图如下所示：



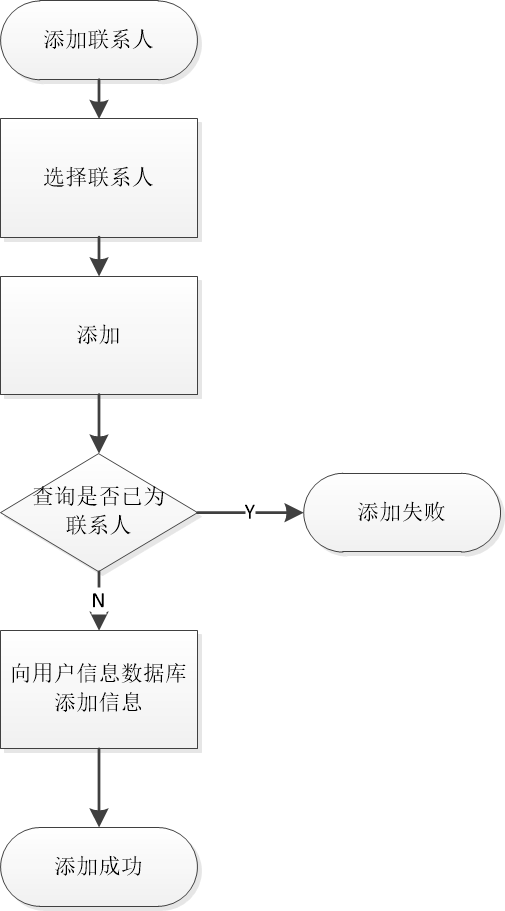
## 搜索联系人流程

该流程描述了用户通过关键词搜索当前系统中存在的其它与关键词匹配的用户的功能，其流程图如下：



## 添加联系人流程

该流程描述了用户通过搜索联系人的结果添加其它用户为联系人的功能，其流程图如下：

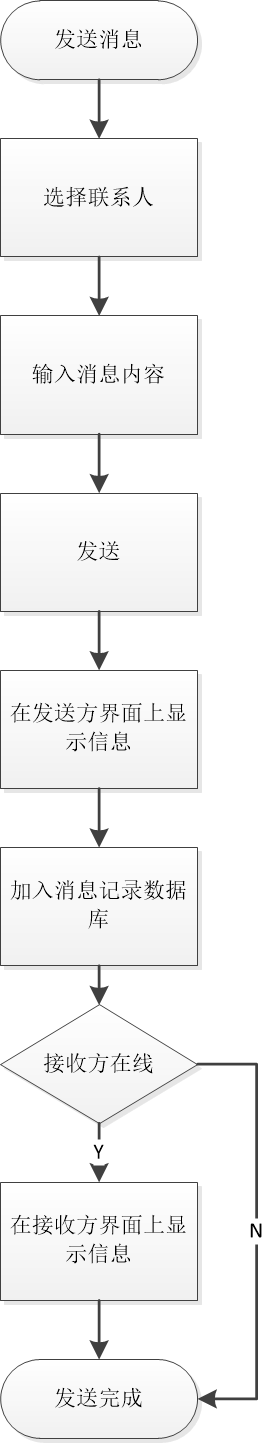


## 删除联系人流程

该流程描述了用户将已有的联系人删除以解除与该用户的联系人关系的功能，其流程与添加联系人相似，不再赘述。

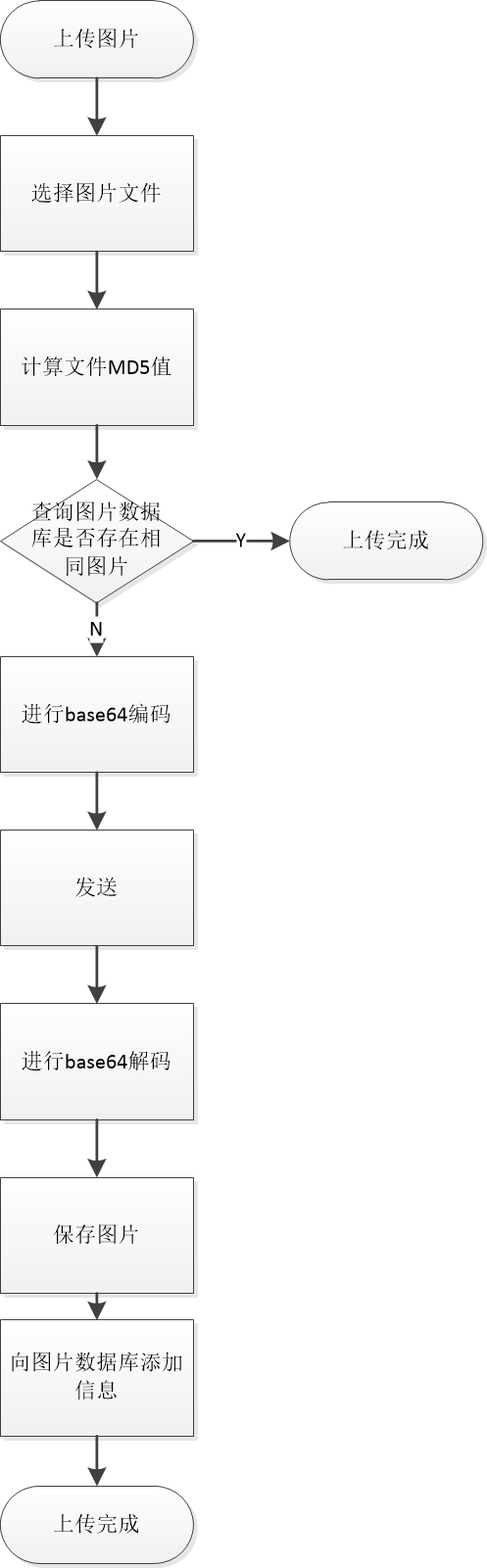
## 发送消息流程

该流程描述了用户向某一联系人发送消息的功能，其流程图如下：



## 上传图片流程

该流程描述了用户向服务器上传图片的功能，其流程图如下：



# 数据库设计

为能够更好的存取用户的聊天记录、联系人列表等数据，我们采用了非关系型数据库MongoDB，所定义的集合如下所示。

用户信息 user

|  |  |  |
| --- | --- | --- |
| 键 | 数据类型 | 说明 |
| username | str | 用户的用户名 |
| password | str | 用户的密码 |
| avatar | str | 用户的头像，保存图片的md5值 |
| friends | [@user] | 用户的联系人列表，是一个user集合文档的\_id的列表 |

消息 message

|  |  |  |
| --- | --- | --- |
| 键 | 数据类型 | 说明 |
| sender | @user | 消息的发送者，是一个user集合文档的\_id值 |
| content | str | 消息的内容 |
| timestamp | datetime | 消息的时间戳 |

消息记录 chat

|  |  |  |
| --- | --- | --- |
| 键 | 数据类型 | 说明 |
| members | [@user] | 消息记录的对象，是一个user集合文档id值，长度为2，代表发送方与接收方，\_id值较小的user在前 |
| messages | [message] | 消息列表，是一个message集合文档表 |

图片 image

|  |  |  |
| --- | --- | --- |
| 键 | 数据类型 | 说明 |
| md5 | str | 图片的MD5特征值 |
| suffix | str | 图片的后缀名 |

其中，user中的friends项会产生一定的冗余数据，但是避免了类似sql数据库中的join操作以提高查询的速度。同时，可以预见的，数据库中的主要数据应为用户间的消息，即message集合，因此我们认为friends项中的冗余信息可以接受。

# 开发工具

本系统预计使用Node.js和WebSocket技术开发，数据库使用MongoDB，开发所用IDE为JetBrains WebStorm。