软件复用课程设计

client-server 应用程序

程序文档

小组： Team10

成员： 谢志杰 1352975

计鹏玥 1352914

王思尧 1352896

谢明玥 1352937

徐锦程 1353012

文档变更记录

|  |  |  |
| --- | --- | --- |
| 更改人 | 日期 | 更改内容 |
| 王思尧 | 2016年3月28日 | 创建文档 |
|  |  |  |
|  |  |  |
|  |  |  |

**1** 开发规划

**1.1** 开发人员

谢志杰 1352975

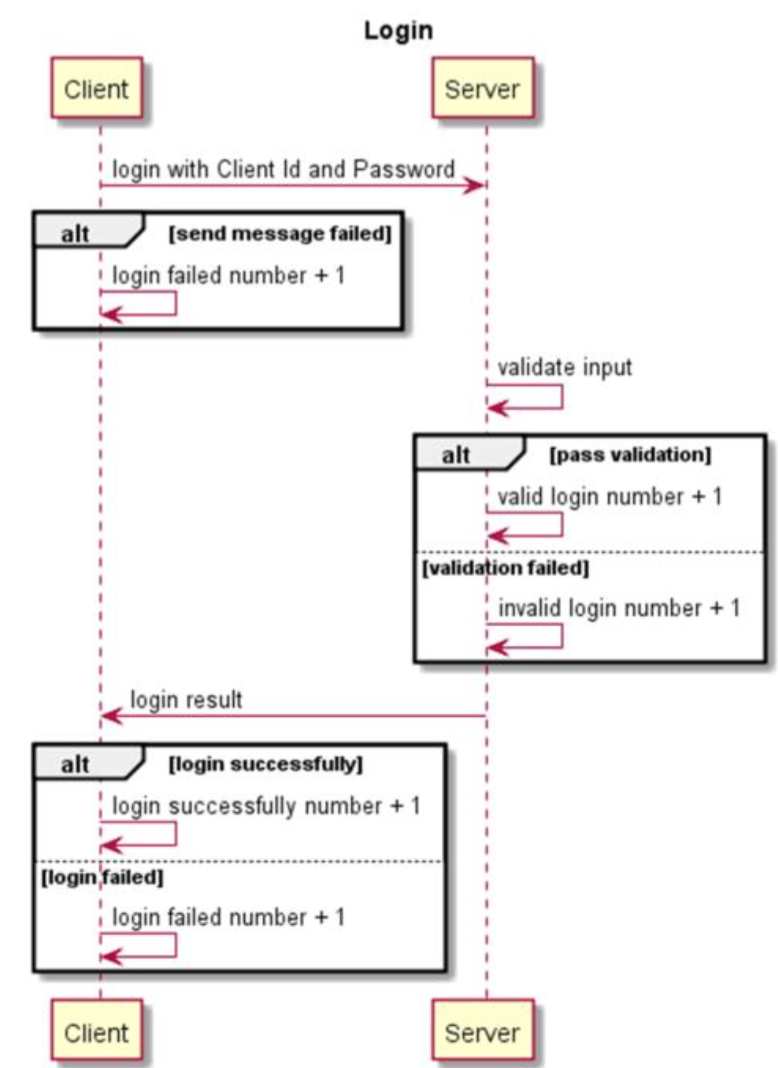
计鹏玥 1352914

王思尧 1352896

谢明玥 1352937

徐锦程 1353012

**1.2** 开发计划

项目起始于2016年3月16日，计划于2016年3月30日完成本次项目开发，实现计划初制定的全部需求，并完成各个阶段的测试，共历时两周时间。

**1.3** 开发环境和工具

Github － 版本管理

Java － 开发语言

Junit － 单元测试

**2** 总体设计

**2.1** 基本设计要求

程序时序图如下所示：

基本要求：

1.客户端登录服务器：

1.1.需要提供：用户id，用户密码，可配置的服务器地址

1.2.基本要求：登录之后保持连接

记录以下数据，每隔一分钟写入文件：

1.客户端：登录成功、登录失败

2.服务端：有效登录、无效登录

2.登录成功后客户端向服务端发送消息：

2.1.服务端验证接受的消息并发出应答信息：

2.1.1.忽略客户端登录之前发送的消息

2.1.2.不允许客户端一秒内发送超过5条消息（该条件需可配置）

－超过的消息可以直接忽略无需做出应答

2.1.3.不允许客户端每次登录发送超过100条消息（该条件需可配置）

－在客户端发送第100条消息后服务端要给出重新登录的应答信息

－客户端发送100条消息后，在客户端重新登录前忽略全部消息

－与客户端保持连接

－重新登录后，上一次登录发送的超过100条的消息不计入本次计数

2.1.4.验证通过需要作出OK应答

2.2.客户端收到重新登录应答后需要自动重新登录

2.3.记录以下数据，每隔一分钟写入文件：

1.客户端：发送消息的数量

2.服务端：接受／忽略消息的数量

3.服务端通知其他登录的客户端接收到的消息：

3.1.服务端向其他登录状态的客户端转发收到的有效消息

3.2.客户端告知服务端已经收到消息

3.3.记录以下数据，每隔一分钟写入文件：

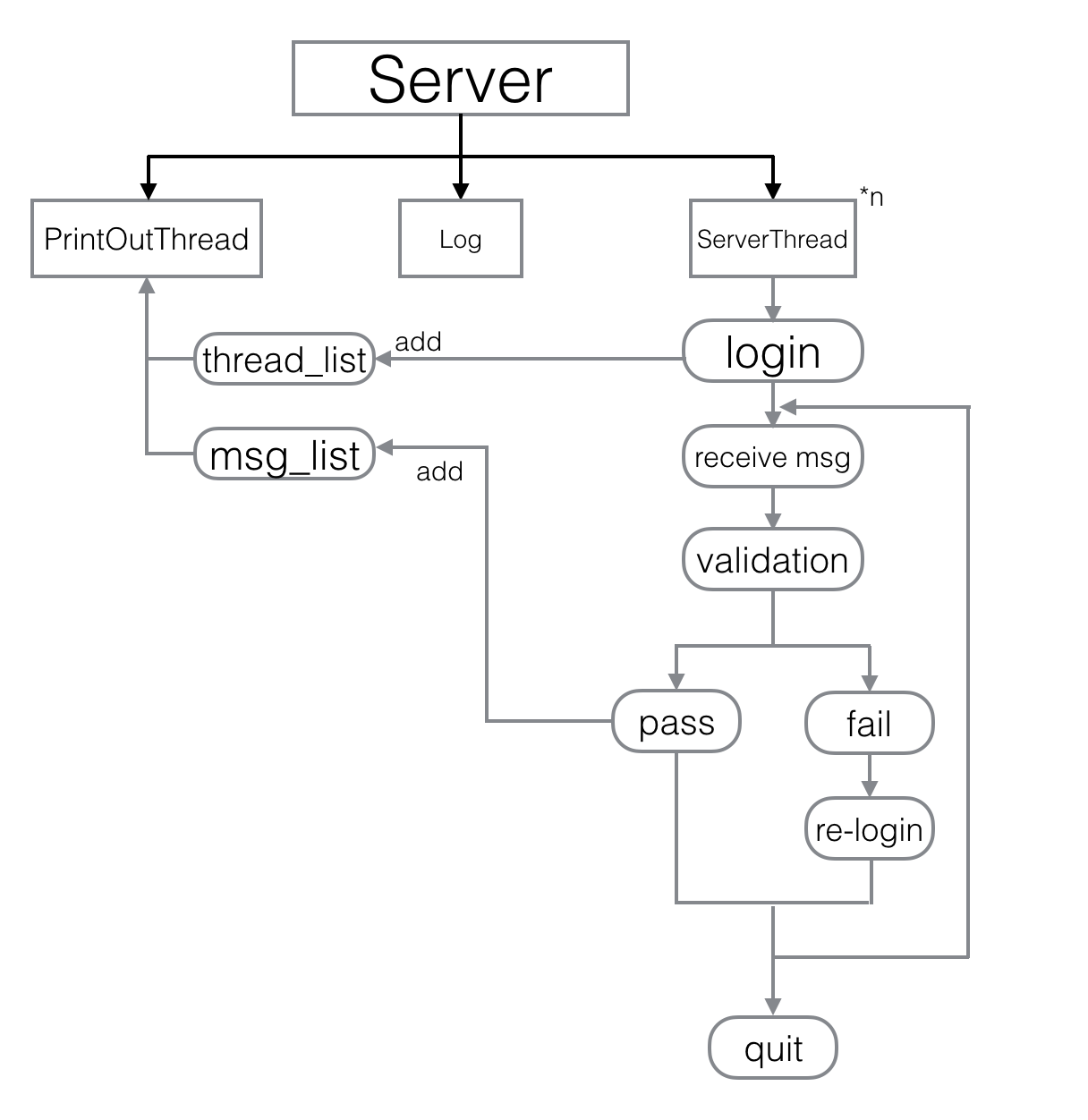
1.服务端：转发消息的数量，对于每一个客户端每一条消息＋1

2.客户端：收到的消息的数量

**2.2** 基本设计描述

考虑到上述提到的功能需求，该应用程序主要分成三个模块，即用户端、客户端以及工具类模块，客户端主要完成上述功能需求中有关客户端的部分，包括登录、发送消息、接受消息以及客户端日志记录等功能。服务端主要完成上述功能需求中有关服务端的部分，包括用户登录验证、消息确认及转发、消息应答以及服务端日志记录等功能。工具类包涵应用程序中需要用到的方法类，包括IO读写操作、消息生成与解析（json）等。

服务端基本设计如下所示：

****

**2.3** 主要界面流程描述

**//**待完成

**2.4** 模块列表

－客户端

－服务端

－工具类

－**IO**写操作

－**Message**

－数据库访问

**3** 接口规范

**3.1 IO**写操作**API**

**3.1.1. IOLog(String logAddr, boolean bAppend)**

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| logAddr | String | 写入文件地址 |
| bAppend | boolean | 是否在文件尾写入，true为在文件尾写入，false为覆盖原文件写入。 |

**3.1.2. IOWrite(String sWriten)**

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| sWriten | String | 写入文件内容，即写入文件的字符串。 |

**3.2** 消息**(Message)API**

**3.2.1. Message(String msg, long ownerThread)**

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| msg | String | 消息的内容，json格式的字符串。 |
| ownerThread | long | 消息所属线程的标识符。 |

**3.2.2. getOwner()**

|  |  |  |
| --- | --- | --- |
| 返回值 | 类型 | 含义 |
|  | long | 返回消息所属线程的标识符。 |

**3.2.3. setValue(String key, String value)**

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| key | String | 消息key |
| value | String | 消息value |

**3.2.4. setValue(String key, long value)**

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| key | String | 消息key |
| value | long | 消息value |

**3.2.5. getValue(String key)**

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| key | String | 通过key在消息中获取value。 |

|  |  |  |
| --- | --- | --- |
| 返回值 | 类型 | 含义 |
|  | String | 返回key对应的value值。 |

**3.2.6. toString()**

用于将JSONObject对象转换为json格式的字符串，无参数与返回值。

**3.3** 数据库**API**

**3.3.1. getPassword(String username)**

|  |  |  |
| --- | --- | --- |
| 参数 | 类型 | 含义 |
| username | String | 查找用户的用户名 |

|  |  |  |
| --- | --- | --- |
| 返回值 | 类型 | 含义 |
|  | String | 查找用户的密码，若不存在用户则返回“rootpassword”。 |

**5** 附录

**5.1** 第三方组件

**5.2** 参考资料