

JAIN-SIP 聊天室系统

软件产品需求说明书

文件状态： [] 草稿 [√] 正式发布 [] 正在修改	文件标识：	JAIN-SIP 聊天室系统需求
	当前版本：	1.2
	作 者：	072668-尹乐 072649-赵振凯 072632-左文峰 072619-沙圣霖 072680-刘诗韵
	审 核：	
	完成日期：	2010/07/19

版本历史

版本/状态	作者	审核	起止日期	备注
最初的用户需求和系统需求，使用文字表述	本小组全体成员			头脑风暴得到
最终版	本小组全体成员			添加了

目录

1 综述.....	4
1.1 使用场景.....	4
1.1.1 普通聊天场景设计 1.....	4
1.1.2 内部会议场景设计 2.....	4
1.1.3 包含手机端的使用场景设计 3.....	5
1.1.4 手机短信聊天使用场景设计 4.....	5
1.2 任务与目的.....	5
1.3 计划与项目的执行.....	6
2 应用环境.....	6
2.1 每个用户都有一台 PC 机.....	6
2.2 存在用户应用手机客户端软件.....	8
2.3 存在用户通过短信聊天.....	8
3. 需求的抽象.....	9
3.1 服务端（Service）.....	10
3.2 客户端（Client）.....	10
3.3 客户端与服务器的通信协议.....	10
4. 功能分析.....	10
4.1 用户分类：.....	10
4.2 用例图：.....	11
4.3 用例：.....	11
4.3.1 viewUserInfo.....	11
4.3.2 modifyUserInfo.....	11
4.3.3 sendMessage.....	12
4.3.4 recieveMessage.....	12
4.3.5 manageUserInfo.....	12
4.3.6 manageMsgInfo.....	12
5. 非功能分析.....	13
5. 需要注意的问题.....	15
6. 约束条件.....	15
参考资料.....	16

1 综述

本系统是基于 JAIN-SIP 协议的聊天系统，应用于用户在线聊天，即时传送用户进行交流的信息，可视化地显示了双方或者多方用户的发言，方便了异地聊天的用户，同时也为多用户的会议提供了服务。

1.1 使用场景

1.1.1 普通聊天场景设计 1

- 1、用户登录聊天室
- 2、用户在聊天室界面的输入框中输入言论信息
- 3、用户点击发送按钮，将言论信息发送至聊天室服务器端，并在自己的显示域中显示信息
- 4、服务器端对聊天内容的合法性进行判断，然后进行屏蔽或者转发
- 5、若屏蔽，则向发送方提供错误提示信息。
- 6、若转发，服务器端为每个在聊天室内的非发送方用户发送该信息
- 7、其他用户接收到，在自己的显示域中显示信息。

1.1.2 内部会议场景设计 2

公司内部，学校内部或团体内部的会议聊天室

- 1、设置登录权限
- 2、内部人员登录到聊天室
- 3、内部人员进行会议讨论聊天
- 4、一个人员点击发送按钮，将言论信息发送至聊天室服务器端，并在自己的显示域中显示信息
- 5、服务器端对聊天内容的合法性进行判断，然后进行屏蔽或者转发
- 6、若屏蔽，则向发送方提供错误提示信息。

7、若转发，服务器端为每个在聊天室内的非发送方人员发送该信息

8、其他人员接收到，在自己的显示域中显示信息。

1.1.3 包含手机端的使用场景设计 3

聊天室的客户端中可以存在手机客户端，以下为手机客户端登录场景：

1、手机用户用自己的用户信息登录至聊天室

2、开始聊天，过程同上。

1.1.4 手机短信聊天使用场景设计 4

手机客户端用户不需要登录手机客户端，直接将信息通过短信的方式发送给服务器端即可：

1、用户若想了解聊天现状，发送查看聊天信息的短信至服务器端。

2、服务器端回送十条当前聊天内容。

3、用户将自己的发言通过短信形式发送给服务器端。

4、服务器端进行判断，随后屏蔽或转发。

5、服务器端按照一定的时间间隔将当前言论以短信的形式反馈给手机短信用户，
若该时间间隔内没有言论则不发送任何信息。

1.2 任务与目的

我们的任务是：

1) 开发基于 JAIN-SIP 的聊天室系统，包括：硬件、软件。完成系统的开发、测试和使用验证。

2) 从应用场景中，抽象出通用的软件部件或模块，以及相互间的通信与编码协议。

3) 依据不同的使用环境和系统，能够将软件部件进行配置，形成新的系统。

我们的目标是：

用 JAVA 语言以及 SIP 协议实现聊天室系统的开发。

1.3 计划与项目的执行

作为软件工程任务，需要经历需求、设计、构造、测试等过程。但是，作为学生项目，需要尽快迭代出一个结果。

学生要：

- 1) 讨论和定义过程阶段，
- 2) 给出预计的时间和人员分工
- 3) 按计划执行该过程
- 4) 验证计划与实际的是否一致？
- 5) 评审与检查活动：在计划中要明确每个阶段或活动的结束时，如何？何时进行评审？评审中发现的问题如何解决？

2 应用环境

考虑几种应用环境：

- 1) 每个用户都有一台 PC 机；
- 2) 存在用户应用手机客户端软件；
- 3) 存在用户通过短信聊天；

2.1 每个用户都有一台 PC 机

这种情况下，假定每个用户都已登录系统中。系统的示意图 2-1 如下：

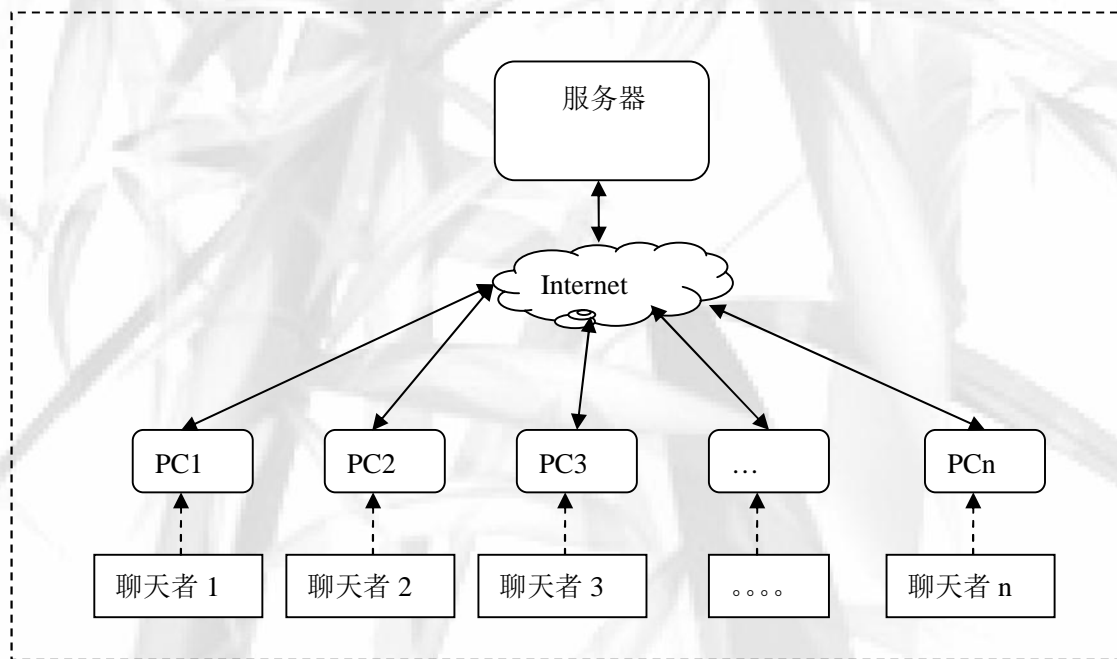


图 2-1 PC+ Internet 的聊天室系统

- 1) 一个聊天者通过面前的 PC 机向网络发送聊天信息
- 2) 网络将聊天信息发往服务器端
- 3) 服务器端进行判断
- 4) 可以发送时, 将该聊天信息组播或以单播实现组播的方式发送至其他聊天者的 PC 机上

2.2 存在用户应用手机客户端软件

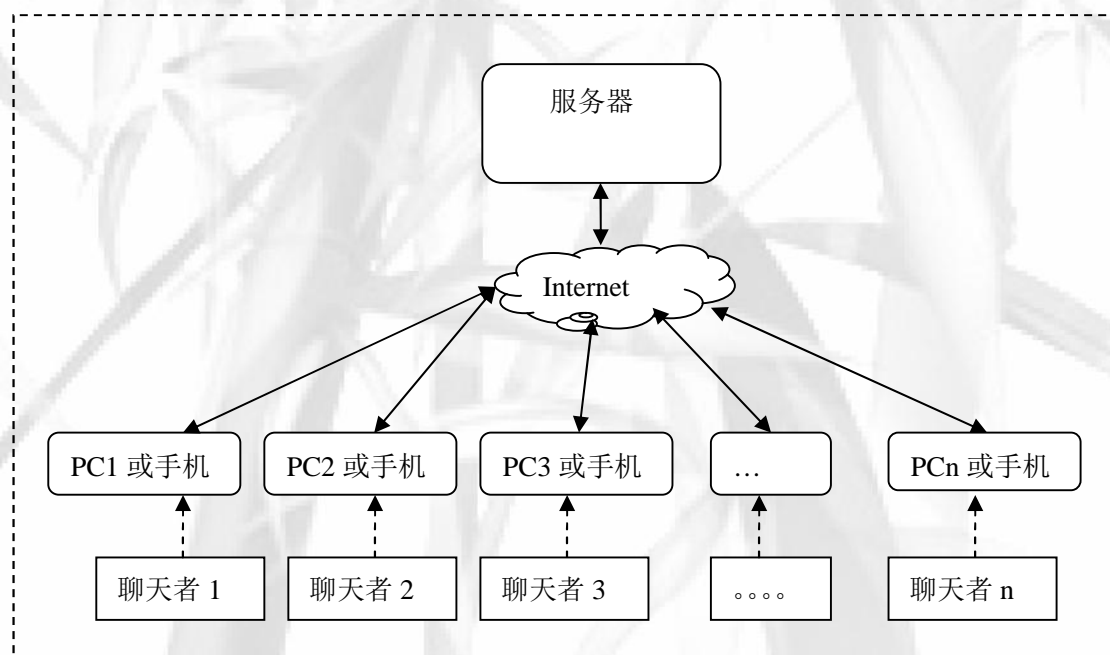


图 2-2 PC 或手机客户端+ Internet 的聊天室系统

- 1) 一个聊天者通过面前的 PC 机或手机客户端向网络发送聊天信息
- 2) 网络将聊天信息发往服务器端
- 3) 服务器端进行判断
- 4) 可以发送时，将该聊天信息组播或以单播实现组播的方式发送至其他聊天者的 PC 机上或手机客户端

2.3 存在用户通过短信聊天

在通信环境中，题目可以经通信通道以短信的形式传送到每个观众所拿的手机上或者通过单向机顶盒显示在观众面前的电视上。通过回传通道，观众在把答案以手机短信的形式将结果发送给“答案接收和处理服务器”。

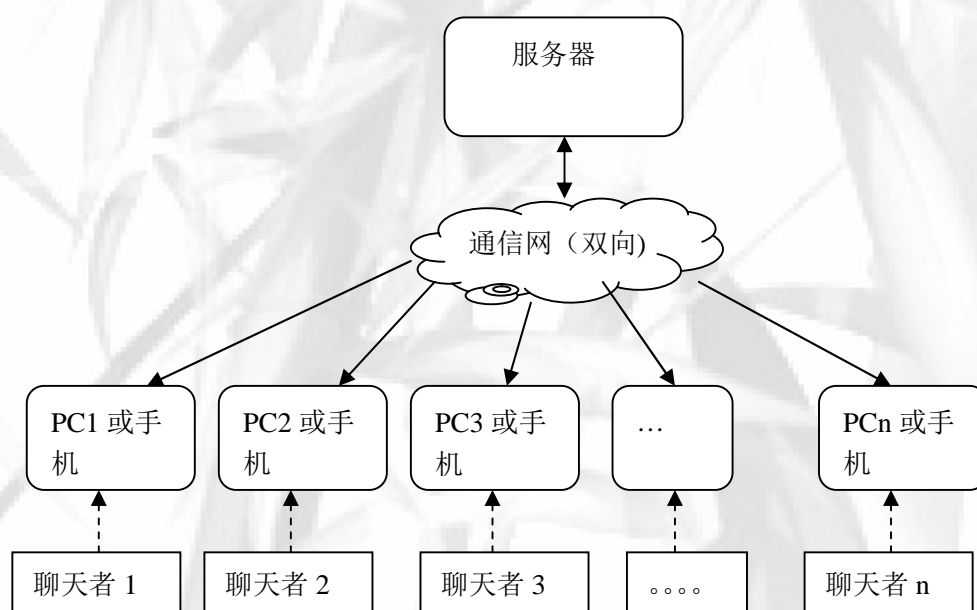


图 2-3 PC 或手机短信+ Internet 的聊天室系统

- 1) 手机用户将查看聊天记录信息发送至服务器
- 2) 服务器将当前十条聊天记录以短信的形式记录发送至用户手机
- 3) 用户可以将自己的需要发表的言论以短信的形式发送至服务器端
- 4) 服务器端进行判断
- 5) 若合法，则将该言论发送至其他用户当前使用模式的端处
- 6) 服务器定时（以固定的时间段）将当前的聊天记录以短信的形式发送给该手机用户。若该时间段中，聊天室内无聊天记录，则不发送任何信息。

3. 需求的抽象

对上面的几种情况，需要抽象出共性部件。显然，这是一个典型的服务/客户（service/client）模型。我们看到一下几个方面。

3.1 服务端（Server）

服务器端应具有：

- 1) **言论的判断**：判断该用户发表的言论是否合法，是应该屏蔽还是转发。
- 2) **信息的接收**：实时（系统先暂时不考虑离线接收的情况）接收用户发送的聊天信息。
- 3) **并发性问题**：即多个用户同时向服务器端发送聊天信息，此时服务器端需要考虑并提出解决方案和措施。
- 4) **信息的发送**：将合法的聊天信息转换成特定的数据格式（用 SIP 协议），发布到网上。
- 5) **发送信息过于频繁**：若一个用户发送信息过于频繁则不与发送并进行提示。
- 6) **用户长时间不发送信息**：若一个用户长时间（规定的时间段）不发表言论，则自动将其请出聊天室，以节省有效的连接数量。

3.2 客户端（Client）

客户端应具有：

- 1) **身份验证**：用户登录聊天室的身份验证，只有登录了才能在聊天室内发言。
- 2) **发送信息**：实时发送用户的聊天信息至服务器端。
- 3) **接收信息**：接收信息，并将其数据格式解析出来。

3.3 客户端与服务器的通信协议

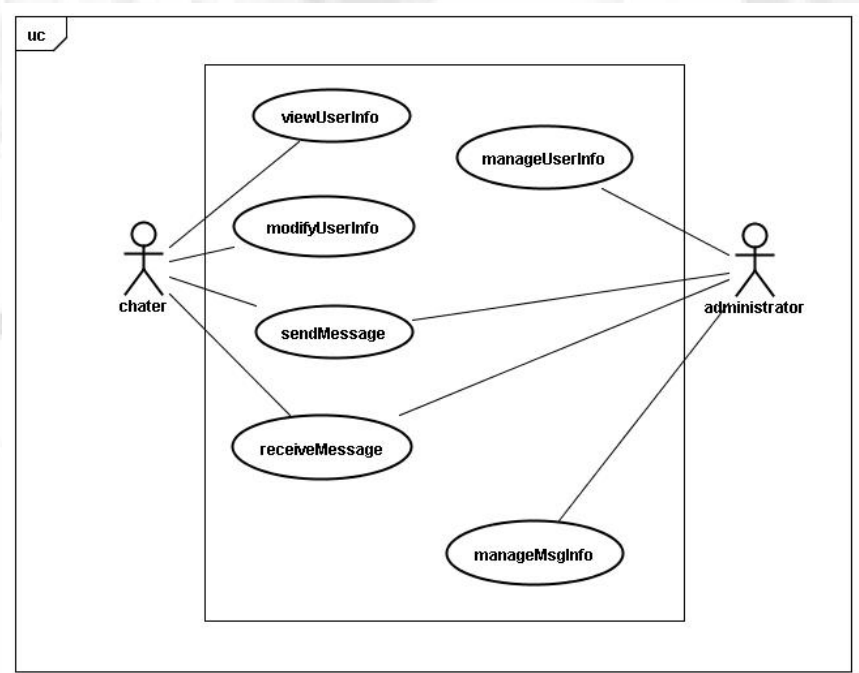
客户端和服务器之间的数据交换，构成了两者之间的通信协议。我们这次应用的是 JAIN-SIP 协议。

4. 功能分析

4.1 用户分类：

- 1) **聊天者**
主要功能：查看用户信息，修改用户信息，发送信息，接受信息。
- 2) **管理员**
主要功能：管理用户信息，管理聊天信息，创建聊天室。

4.2 用例图：



4.3 用例：

4.3.1 viewUserInfo

表 1 查看用户信息用例

Actors	Chater
Summary	A chater will see his or her informations.
Preconditions	Chater has logged in.
Detailed description	Chater press the “View” Button.
Postconditions	Chater will see his or her information.

4.3.2 modifyUserInfo

表 2 修改用户信息用例

Actors	Chater
Summary	A chater will change his or her information.
Preconditions	Chater has logged in.
Detailed description	1、 Chater press the “Modify” Button. 2、 Enter the infoemation the chater want to modify. 3、 Chater press the “Submit” Button.

Postconditions	The information of the chater changed.
----------------	--

4.3.3 sendMessage

表 3 发送消息用例

Actors	Chater. and administrator
Summary	Chater and administrator send message to all the others.
Preconditions	Chater and administrator have logged in.
Detailed description	1、 Enter the message into the input area. 2、 Press the “Send” Button.
Postconditions	The message will be sent and other chaters will receive the sent message.

4.3.4 recieveMessage

表 4 接收消息用例

Actors	Chater and administrator
Summary	Chater and administrator receive message.
Preconditions	Chater and administrator have logged in.
Detailed description	No action required by the receiver., the message will be automatically received by the receiver.
Postconditions	The message is received and diplayed in the show area.

4.3.5 manageUserInfo

表 5 管理用户信息用例

Actors	Administrator
Summary	Administrator can add,delete,modify,view users' informations.
Preconditions	Administrator have logged in.
Detailed description	deal with the informations of users by using add,delete,modify,view operations
Postconditions	User informations are changed.

4.3.6 manageMsgInfo

表 6 管理消息信息用例

Actors	Administrator
Summary	Administrator manage the chaters' messages
Preconditions	Administrator have logged in.
Detailed description	1、 Look at the message of chaters 2、 Judge whether the message is legal. 3、 If not legal , the message will be fobidden.

Postconditions	The sent messages are all legal.
----------------	----------------------------------

5. 非功能分析

(1) 可靠性

- 即使硬件出现故障，系统也可以可靠运行吗？
- 复制和故障转移方案是什么？
- 需要手动干预，还是系统可以自动进行故障转移？
- 实现可靠性会对性能造成负面影响吗？
- 实现可靠性的成本有多高？

可靠性需要考虑的一些具体方面是：

- **安全性：**假设攻击者就在外面。
如何知道系统用户就是他们所声称的，并只让他们访问经过授权的功能？
如何保护我的系统不受攻击？
考虑到网络攻击、机器攻击，甚至从您自己的系统内部发起的攻击。
- **事务性：**如何设计系统来保存工作单元的 **ACID** 属性？
如果在设计中涉及多个独立的子系统（**Web** 服务和 **SOA** 就是这种情况），则这一点就显得特别重要。不要假设始终可以进行两阶段提交（**two phase commit**）。

(2) 可用性

- 您是否为用户带来不适当的负担（例如，需要特殊的浏览器版本）？
- 系统是否根据模型-视图-控制器（**Model-View-Controller**）体系结构设计以使多用户界面成为可能？如果是这样，如何将它们绑定在一起？
- 是否界面本来就有状态而功能无状态（反之亦然）？

(3) 有效性

- **性能：**这个系统的运行情况有多好？
它只是平稳缓慢地运行吗？
系统可以达到其响应时间目标吗？
应用程序的设计是否符合性能要求？
您利用缓存了吗？
- **可伸缩性：**如果系统在小范围内运行看起来相当快，那么当扩展至每秒、每分钟或者每小时几千或成千上万个活动的时候呢？
它的设计是否达到吞吐量目标？
可以复制系统来实现线性扩展吗？
是否存在瓶颈（例如公共数据库）？

(4) 可维护性

您如何配置它？

如何监视它？

如果您一件事情需要执行很多次（例如，安装许多应用程序），那么会怎么做呢？

您是否有一个可复制的部署流程呢？

您是否可以使重复的任务自动化，使之在大范围内可行呢？

(5) **可移植性**

虽然列在最后，但它并非最不重要。例如：

如何采用标准来提供某种形式的平台中立性呢？

是否计划将应用程序迁移到您的最新和最高版本的应用服务器上呢？

如果不打算这样做，则当供应商撤消对该版本的支持时您要怎么做呢？

如果您的项目基于开放源代码，则也有类似的问题。如果每当某人有个更好的捕鼠器 (mousetrap) 您就必须重写整个应用程序，则没有人会问津。

还有一些分析：

1. 系统的完整性

系统的完整性指为完成业务需求和系统正常运行本身要求而必须具有的功能，这些功能往往是用户不能提出的，典型的功能包括联机帮助、数据管理、用户管理、软件发布管理和在线升级等。

2. 系统的可扩充性与可维护性

指系统对技术和业务需求变化的支持能力。当技术变化或业务变化时，不可避免将带来系统的改变。不仅要进行设计实现的修改，甚至要进行产品定义的修改。好的软件设计应在系统架构上考虑能以尽量少的代价适应这种变化，常用的技术有面向对象的分析与设计及设计模式。

3. 技术适应性与应用适应性

系统的适应性与系统的可扩充性和可维护性的概念相似，也表现产品的一种应变能力，但适应性强调的是在不进行系统设计修改的前提下对技术与应用需求的适应能力，软件产品的适应性通常表现为产品的可配置能力。好的产品设计可能要考虑到运行条件的变化，包括技术条件（网络条件、硬件条件和软件系统平台条件等）的变化和应用方式的变化，如在具体应用中界面的变化、功能的剪裁、不同用户的职责分配和组合等。

（以上非功能分析参考网络上的分析）

界面的美观性和人性化

界面的美观性和人性化也是非功能分析的重要方面。

美观性和人性化：

首先，界面的美观设计和人性化设计可使用户的视觉舒适感增强

其次，很好的视觉效果和人性化设计使用户对软件的喜爱度增高

最后，界面的美观性和人性化设计可以提升产品的品味和档次

5. 需要注意的问题

并发性：多个用户同时向服务器发送消息时，服务器应该如何处理，是缓存多条信息再一一发出，还是发送其中的一条，其余的丢失并反馈给用户信息。

身份认证：内部会议时登录聊天室需要权限。

发送过于频繁和久不聊天：用户在发送信息时，服务器的计时器会启动，并将发送时间记录在案，若发现某用户发送信息过于频繁时，暂时停止该用户发言一小段时间，并向该用户发送消息；若发现某用户就不说话，则向该用户发送消息，并将其踢出聊天室。

信号干扰：若是选用无线传输方式，则对信号的要求相对较高，但信号容易受到干扰，或者发生静电屏蔽的现象，从而导致传输出错或传输受阻，所以如何尽最大可能避免信号的干扰也是一个重要问题。

6. 约束条件

约束条件规定了实现本系统时需要遵循的原则或其它要求。

- 1) 服务器端用 Windows 平台；
- 2) 数据库：并不意味着用 DBMS，而是希望先用文件系统实现。这样可以摆脱本系统对 DBMS 的要求。也容易将本系统与现有的各种题库系统进行联接。
- 3) 开发环境：服务器端、客户端用 Eclipse。JAVA 语言进行编程。
- 4) 系统结构采用 C/S 结构。避免用 Web-Service，例如，Tomcat、We-Logic 等软件。其目的是，希望抽象出专门用于答题的服务器端软件。
- 5) 通信协议为 SIP 协议。

参考资料

1. 百度或者 google 搜索关键词：
2. 老师 ftp: //59.64.164.17 上放置的资料