

毕业设计（论文）开题报告

（含文献综述、外文翻译）



题 目 基于python的个人网站设计与开发

姓 名 梁佳利

学 号 31501117

专业班级 计算机1502班

所在学院 计算机与计算科学

指导教师（职称） 彭彬（讲师）

二○一八 年 十二 月 三十一 日

毕业设计（论文）

开 题 报 告

（包括选题的意义、可行性分析、研究的内容、研究方法、

拟解决的关键问题、预期结果、研究进度计划等）

## 1. 选题的背景和意义

如今个人网站已经成为人们沟通交流展现自我发表接受信息的新的形式，不仅对传统交流工具和信息媒体产生了巨大的冲击，同时也带来了一场大规模的社会革命，作为社交媒体的一部分，对生活就业、政府和法律等方面都带来了巨大的影响。本次选题意义在于利用现今的网站开发技术，建设一个更加开放，安全，便捷使用以及更加完善的个人网站，使人们能够更好的享受互联网带来的便利。

### 1.1 选题的背景

随着互联网与人们生活日益密切以及网络技术的飞速进步，传统的交流媒介、交流方式还有交流对象都发生革命性的变化，个人网站飞速兴起并且逐渐成为了人们网络生活中不可或缺的一部分。个人网站是一个人们展现自我，与人交流的强大工具。作为社交网站的一部分，个人网站上课可以发布文章、图片和聊天，并且读者能够留下自己评论。随着信息的全球化，越来越多的人们希望通过个人网站了解相隔异地朋友的信息，如果能抓住这个时机，适时的推出新颖方便的个人网站系统，那一定能够使得国内个人网站企业提升一个台阶。

### 1.2 国内外研究现状

个人网站作为社交媒体的一部分在国外正在不同领域，发挥着越来越来大的作用。以国外的facebook，twitter为代表，发挥的作用已经远远超过了其本身，在政治与经济生活中都扮演了举足轻重的角色。个人网站的形式也十分多样化，人们除了分享文字图片外，人们还可以发布音频，视频，发起投票和调研等。因此个人网站每天都产生巨大的信息量。国内个人网站发展的势头很猛，一大批个人网站如雨后春笋般兴起，我曾经在移动应用开发课程中制作一款主题为新闻时事的个人博客app，主要功能为发表个人博客与图片，浏览当今新闻时事和好友发表的博客内容，并且对相关文本与图片进行评论修改于删除，本次毕业设计的想法基于这些内容，并做了功能上的扩充，最典型的就是增加了实时聊天的功能。在如今这个数字化的世界里，不管国内还是国外，人们的工作与生活都很大的依赖于网络，个人网站的出现正式随着这种变化而兴起，成为人们信息交流的新场所。

### 1.3 发展趋势

国外一些欧美国家中，在这方面有着更加长久的历史与积淀，因此个人网站发展更加成熟。而国内个人网站的出现与普及较落后，且加上一些政策的限制，发展不如发达国家，但如今国内的势头不输国外。个人网站的开发方式多种多样，目前html+JavaScript是主流的前端页面制作，不论国内国外都是如此。而服务端的开发则有所差异，国内的python制作的网站还在少数，国外在网站制作上使用python语言的不在少数，例如youtube就是用python编写的。目前python的使用已经非常普及，家喻户晓的pythonweb框架有Django，支持异步高并发的Tornado框架，短小精悍的flask,bottle，这些技术使得个人网站的开发更加便捷快速。如果在未来在web开发上，python能与Java更好的结合，那将会大大推动相关领域的进步。

**2．研究的基本内容**

本次课题主要研究基于python的个人网站开发，本课题主要针对用户发布个人信息，浏览他人信息，并且支持评论以及实时的聊天，设计并完成一个可靠的个人网站。当前个人网站功能繁杂但缺少部分实用功能，且很难保证用户隐私问题。本课题尝试开发一个功能全面且保证个人隐私的个人网站。本次网站设计在管理员模块与以往一些网站的不同之处或者新颖之处在于增加了一个审核模块的，用户发布的信息必须由管理员和一定的过滤算法审核过后经允许才能发布。文字内容方面采用敏感词过滤系统并建立反垃圾信息机制，图片内容方面主要依赖于人工智能的监制。本次个人网站的开发与设计与市面上的以及过去课程中制作的个人博客的设计与开发的不同之处或者说新颖之处在于本次个人网站开发与设计后端完全使用python语言编写，并选取Django或者其他框架使得开发更加便捷接口模块等更加清晰。在功能上与以往的个人网站的不同之处或者是新颖之处在于加入了实时聊天的功能。比如qq空间等个人网站只支持评论与发布信息等，并不支持两个用户之间聊天。这使得两个用户在同一个网站上的交流变得很不方便。

表1.1 用户基本内容

|  |  |  |
| --- | --- | --- |
| 模块 | 功能 | 描述 |
| 用户登陆注册注销模块 | 用户注册后登陆网站，成为正式用户，并可以对网站进行权限范围内的操作，注销后下线退出 | 注册时将用户账号密码等信息写入数据库权限为普通用户，登陆后将将该账户状态设置为登陆，注销后将状态设置为下线 |
| 用户搜索模块 | 输入关键字进行搜索，可以浏览用户发布的各种信息 | 通过将搜索关键字与数据库中的信息进行比较然后显示在页面中并且能够对查询结果按照一定的方式进行排序筛选。同时也能够针对用户名进行模糊查询。 |
| 个人相册功能模块 | 实现图片的上传分享与删除 | 上传的图片存储在服务器中，可供用户下载浏览。并且支持其他用户进行浏览与评论。删除将数据库中的图片的表项删除，然后删除服务器中的图片。 |
| 评论功能模块 | 用户可以针对其他用户发布的文章照片等信息发表自己的评论，同时用户能够删除的自己评论，并且用户具有删除其他人的信息的权限。 | 在数据库评论中插入删除一条评论记录 |
| 聊天功能模块 | 用户可以选择其他用户请求发起聊天，经同意后可以进行实时通讯。或受到聊天请求后同意即可进行聊天 | 使用websocket。 |

表1.2 管理员基本内容

|  |  |  |
| --- | --- | --- |
| 模块 | 功能 | 描述 |
| 管理员登陆注册注销模块 | 用户注册后成为管理员，登陆后可以进行全县范围内的操作，注销后下线退出 | 注册时将管理员账号密码等信息写入数据库权限为管理员，登陆后将将该账户状态设置为登陆，注销后将状态设置为下线 |
| 搜索模块 | 输入关键字进行搜索，可以浏览用户，用户发布的信息等 | 通过将搜索关键字与数据库中的信息进行比较然后显示在页面中并且能够对查询结果按照一定的方式进行排序筛选。同时也能够针对用户名进行模糊查询。 |
| 用户管理模块 | 管理员可以浏览用户的，并且可以对不是管理员的用户的信息进行增加修改删除等操作 | 在数据库相应表项中进行修改 |
| 信息管理模块 | 管理员可以浏览用户发布的内容与评论，并且可以对用户发布的内容与评论进行增加修改删除等操作 | 在数据库相应表项中进行修改 |
| 审核模块 | 用户发布的文章图片等信息首先通过某种算法决定该内容是否合法能够发布，之后显示在管理员的审核页面中，管理员可以选择审核通过和不通过决定该用户内容是否可以发表，保证网络环境的和谐与美好。 | 对用户发布的文章等由管理员或者一定的算法来检测是否由不合法的关键字。 |

本网站主要分为两方面的制作，主要是前端与后端的开发。本网站面向的用户主要是两类，一是用户，一是管理员。

### 2.1 基本框架

个人网站系统

用户注册登陆

查询用户和文章

管理员注册登陆

发布文章

发布图片

聊天

搜索

评论

修改用户信息

审核

图1.1 个人网站组织图

### 2.2 研究的重点和难点

本课题重点主要使用python与Django框架制作一个个人网站的服务端。主要分为普通用户与管理员两部分。主要在于用户功能模块的设计：用户登陆注册模块，搜索模块，个人相册功能，评论功能，聊天功能。本次课题重点和新颖的点主要在于：

审查系统：对内容敏感或者不合法的信息审查不通过。

用户之间可以通过网站进行聊天。

后端完全采用python编写使用Djangoweb框架。

难点主要解决一下几个方面：

1.用户要能够方便的编辑信息并且发布，对自己的文章相册等进行方便的修改删除，并且用户对其他人发布的信息进行评论操作。

2.如果分类管理存储用户的文章，相册等。

3.使用Django开发时要表现良好的兼容性，达到跨平台运行。

上述问题问题对本次网站的开发提出了更高的要求，本次开发的难点主要存在于以下几个方面：

1.在使用Django时如将降低程序的耦合度，这就要求设计出接口清晰，功能独立的模块

2.如何使用websocket技术实现及时通讯。

3.如何在用户，管理员操作网站时保证网站的安全，与及时响应。

4.如何在用户数，文件数和图片数增多时，进行分页显示。

### 2.3 拟解决的关键问题

内容审核管理属于关键问题，主要通过以下两个方面进行，一是在文字方面使用敏感词过滤系统和建立反垃圾信息机制。审核系统中首先预设一批关键词库并对词组进行排列组合，这批词库又会根据敏感性进行分类，系统会组织用户发布敏感词汇，获奖用户发出来的敏感词汇的内容全部删除，对于敏感性较低的词汇，发出来不会立即删除需要管理员经过二次审核。反垃圾系统比较简单，主要是对管理员认定的不合格的数据清理掉，将系统认为可疑的信息进行提示分类。二是在图片内容方面，主要依赖于用户的举报，管理员的审核以及第三方的接口辅助网站进行机器审核。

## 3．研究的方法及措施

毕业设计的论文研究方法大致有五种，1.调查法与实验法，2.规范研究与实证研究，3.定量分析与定性分析，4.文献综合法与个案研究法，5.跨学科研究法。本次课题主要采用一个方法有调查法，对国内外各大个人网站进行有目的有计划有系统的进行分析对比，通过搜集大量资料归纳总结各大网站的优点与缺点并且将调查结果在本次制作的网站中。另一个方法就是文献综合研究法，根据本次研究的课题，通过查阅大量文献来获得资料，从而全面的正确地掌握基于python的web开发以及Django的应用。

## 4．预期研究成果

对于用户方面的预期成果主要有以下几个方面：

内容充实主题明确，在网站风格上符合消费者的喜好

确实保障网络环境的和谐与美好

网站交互性好，使用便捷

获取信息及时并能随时和好友进行沟通

内容形式多样，包括文字图片等。

## 5．研究工作进度计划

第1周(第七学期)至第4周（第七学期）：查阅文献，确定选题；

第5周(第七学期)至第8周（第七学期）：完成文献综述和外文翻译；

第9周(第七学期)至第12周（第七学期）：修顶完善开题报告；

第13周(第七学期)至第15周（第七学期）：完成数据设计与搭建；

第16周(第七学期)至第7周（第八学期）：完成系统基本框架搭建，完成各功能模块编写；

第8周(第八学期)至第11周（第八学期）：毕业论文定稿；

第12周(第八学期)至第14周（第八学期）：完成毕业论文修改与最终版，审查系统准备答辩；

毕业设计（论文）

文 献 综 述

（包括国内外现状、研究方向、进展情况、存在问题、参考依据等）

浅谈基于python的个人网站开发

## 1. 国内外研究现状

在互联网技术日新月异飞速发展的今日，网络的使用与各种网络技术的应用已经渗透到了人们工作生活中的方方面面，如政治、经济、社交等，因此个人网站的建设页越来越受到人们的重视。同时也有越来越多的人渴望获得自己的个人网站，在网络中创造属于自己的世界。而互联网的科学技术飞速发展使得建立个人网站成为可能。

日前，福布斯网站发表文章，总结出了15个当今美国年轻人最常光顾的网站，其中，博客、视频、社交网站最受欢迎。这些网站虽然不如Yahoo、eBay或Google等有名，但个个鲜活有个性，也许从这些年轻人不一样的口味中，我们能欣赏到当今网络最流行和最具发展潜力的应用。它们也许就是明天的Yahoo！国内最有名的当然是微博，但说起微博不得不说twitter，人们可以通过它在网络发布个人信息，并通过评论，聊天等信息与他人产生关联，极大的满足了人们表达自我与他人交流的愿望。微博属于典型的个人社区网站，在社交网站多样化发展的当下，人们不在满足于简单的发送信息，而是朝着多元化个性化的方向发展。以微博为例，它支持用户编辑发表原创信息，允许他人对信息的转发、评论，以及多人对同一话题讨论。

网站开发主要分为前端设计和与后端服务器开发，如今在网站开发中已经有很多十分成熟的技术。前端主要使html与JavaScript结合使用。服务端开发语言主流为Java，其余还有JavaScript，python等。本次课题服务端将使用python+Django搭建自己的个人网站。Python属于解释型的面向对象语言，因其功能强大交互性好深受国内外人欢迎。Python作为一门动态，解释，面向对象的成熟的语言，因其简单易学，功能全面，语法简洁且在多数操作系统中都支持该语言在国外应用广泛，著名的搜索引擎Google，其核心代码就是用python完成的，迪士尼动画生成，许多unix版本都内建了python环境支持。国内知名的豆瓣网也是使用python建立的。Python与Java相比较，两者都是开源的跨平台语言，且后者能做到的前者页都能做到，且在基于web应用中具有明显优势。如知名的豆瓣网站就是使用python作为主题开发语言，进行全面网站架构以及相关应用的设计与开发，也成为国内较成功的python之web应用。[10]

Python之下的web框架多种多样，最具代表性则为Django。Django发布于2003年，是当前Python世界里最负盛名且最成熟的Web框架，最初被用来制作在线新闻的Web站点。Django的各模块之间结合得比较紧密，所以在功能强大的同时又是一个相对封闭的系统（依然是可以自定义的），但是其健全的在线文档及开发社区，使开发者在遇到问题时能找到解决办法。大量成功实践的网站都已Django为基础。Django乃开源应用框架兼采用MVC框架。其强大的数据库功能、自带的强大后台功能、优雅的网址、模板系统、缓存系统、国际化等优势使其迅速发展。[3]

## 2．研究方向

每个人的网上生活千差万别，对于个人网站的需求也不尽相同主要的目前主要的网站有以下几类。一是主题型个人网站——比如音乐个人网站，用户在这上面分享交流音乐等。二是社区型个人网站——主要用于向好友推荐自己感兴趣的内容。三是信息聚合类网站——如youtube，将新的信息及时的不间断的推送给用户。四是社交网站——如微博，让用户告诉圈子里的人自己的所思所想。

一般网站运行交互的过程机制如下，由用户发起网站访问请求即浏览器向服务器发起http请求，服务端根据请求的url找到用户请求的文件进行相应。这是网址运行的大致流程，实际上后台编程可以用任何编程语言实现，并不仅仅局限于Java、python等。由于python语言的种种优点和大多数服务器都支持python，这使得用python进行服务端编程成为可能。Python有很多方便简化web开发的框架，所谓web开发框架本质上就是socket服务端运行监听某个服务器端口，其包含两大部分：socket部分与业务逻辑处理。Python框架也随之被划分了两大类：一是两张都包含的，如tornado，二是只包含其中一个的，如Django。不能说Django是所有框架里最好的，但确实是使用率最高文档最全面的。Django基于MVC模式设计。MVC设计模式对服务端软件划分为三个分离的且完成一个独立任务的部分，主要分为：

控制器C——主要对请求转发处理工作

视图V——属于UI部分，有前端人员进行设计

模型M——是程序的实际的功能，对请求进行一系列处理相应

Django的MVC设计模式是使得网站开发有了明确的分层，更加易于维护。对开发人员更加重要的是它提供了全面完美的开发工具（缓存，session等）大大提升了开发效率。Django拥有功能全面的url配置，克服了参数设置的困难，设计出合理规整的url。在python应用于web开发的研究方向上还有另一个出名的Pylons。介绍它的原因就是它的设计理念于Django完全背道而驰，仅有2000多行的代码编写而成，附带有灵活好用的第三方库与模块。Pylons一大特点就是灵活与自由，完全依照开发者的意志进行定制。开发者可以自己选择与自己喜好对口的模板等。然而想要完全掌握pylons并不是这么容易，它灵活自由且以来第三方模块的的特点使得学习成本很大。Pylon的官方文档远不如DJango那么全面易懂，这一方面一直很受人诟病。与上述两者有着明显不同，tornado作为一个非阻塞式速度飞快的服务器框架，其有着许多特点，tornado主要由一套工具，vxwork系统，可自主选择连接主机的通信软件包。Tornado提供了一系列强大功能多样的工具，大大简化了开发人员的开发过程，不用去考虑一些底层细节的问题。同时它也支持可扩展和实时交互的操作系统。但是考虑到本课题个人网站只属于小型轻量级网站，显然Django更加适合本次课题的研究。[1]

接下来讨论个人网站的设计与开发。网站的核心必须明确，网站的所有设计制作必须仅仅围绕自己的主题，同时也要注重界面引导，除了功能外一个界面的好坏也很大程度上决定了一个网站的成功与失败。国内外知名个人网站目标都是表达自我与他人关系。如知名的豆瓣网站就是使用python作为主题开发语言，进行全面网站架构以及相关应用的设计与开发，也成为国内较成功的python之web应用。国内最有名的当然是微博，但说起微博不得不说twitter，人们可以通过它在网络发布个人信息，并通过评论，聊天等信息与他人产生关联，极大的满足了人们表达自我与他人交流的愿望。随着移动流量的飙升，传统的pc端与移动端的流量已经平分秋色。伴随着HTML5技术日趋完善，更多的中心也转移到了移动端的开发上。

## 3．进展情况

进展情况主要是关于python应用于网站开发的研究以及个人网站的设计与发展。首先关于python以及其在网站开发中的应用进展情况进行进一步探讨。Python作为一门动态，解释，面向对象的成熟的语言，因其简单易学，功能全面，语法简洁且在多数操作系统中都支持，而在各个领域中都有很多应用，例如系统编程，网络爬虫，人工智能与科学计算，大数据云计算，以及本次课题中的web开发。Python有众多优秀的框架，豆瓣，YouTube等大型网站主要部分的代码就是用python实现，其中较为实用的框架有Django，常应用于小型轻量级网站的开发。[10]

与此同时只需要小小的修改就能被移植到不同的平台中。现今进行web开发的java框架还是非常多，而在基于python的框架中最又名的就是Django了，Google App Engine甚至Erlang都有框架受它影响。众所周知，Django功能好，文档也较全面，主要基于MVC的模式，但是在Django中，用户的请求将由控制器接受然后交由框架自行处理，编程人员不必关心过多的细节，弱化了C这个部分，因此又被看作是MTV，其中T代表模板（template）。

## 4．存在问题

尽管python框架Django相关技术都已经非常完善，但由于是使用框架必然会带来一些代码改动与安全问题。Django的很多方法与函数都被过度分装，开发人员进行开发时若是想要一些特殊的功能，由于改动而付出的时间代价将会很大。并且还有Django框架本身带来的问题，Django模板实现了代码和样式完全分离，禁止在其中有python的语句，明显这样使得自由度大大下降了。除了模板本身的问题，python语言本身性能比较低不如C，Java。这也使得网站的性能有所下降，但是这对于其带来的好处来说还是微不足道的。随着网络的不断发展，网络安全问题也越来越受到重视，但是一下的一些情况可能给网站带来一些安全上的问题，例如由于设计不合理导致的逻辑问题会使得一些黑客绕过验证而获取一些不该获取的重要数据，还有DEBUG模式未关闭导致的敏感信息泄露，条件竞争漏洞等。还有一些框架本身无法克服的安全问题如跨站脚本攻击和低版本这些会使得用户账户密码以及其他重要信息被黑客盗取。在网站中sql注入风险属于较高的种类，通过将输入的语句编程可执行的恶意代码而绕过防火墙访问数据库，Django中的ORM虽然可以避免sql注入但并不是十分有效

参考文献（含开题报告和文献综述）

[1] Liu. Research of Web Real-Time Communication Based on Web Socket [J]. International Journal of Communications, Network and System Sciences.2012,Vol5(12):797.

[2] 黄栎桥, 陆鑫. 基于Struts框架的Web数据库分页技术[J]. 计算机应用.2008,Vol5(12):306.

[3] 郭力子. 基于开源的Web开发架构[J]. 计算机时代. 2008,Vol5(3):12.

[4] 聂玉成. 个人网站建设与实现[J]. 数码世界.2017,Vol32(1):115.

[5] Mathew Miles. Using web2py Python framework for creating data-driven web applications in the academic library [J]. Library Hi Tech.2016,Vol34(1):164.

[6] Mark Granovetter. The Strength of Weak Ties [J]. The American Journal of Sociology.1973,Vol78(6):136.

[7] 王连喜，蒋盛益，庞观松. 微博用户关系挖掘研究综述[J]. 情报杂志.2012,Vol31(12):91.

[8] 黄萱菁,张奇,吴苑斌. 文本情感倾向分析[J]. 中文信息学报.2011,Vol25(6):118.

[9] 徐伟明. ASP在网站开发中数据库连接方法及比较[J]. 电脑知识与技术.2010,Vol30(30):8448.

[10] 师德清. 基于Python、MongoDB和Red5的精品课程网站架构设计研究[J]. 电脑知识与技术.2011,Vol30(30):121.

[11] 王磊, 危建国. 基于Django的实验室设备管理系统设计与实现[J]. 软件导刊.2014,Vol2(02):103.

[12] 高昂. Django1.2新特性[J].程序天下事.2010,Vol12(12):26.

[13] 洪登. 基于Django实现用户点击图片更换头像功能[J]. 电脑编程技巧与维护.2017,Vol23(12):74.

[14] 江柳. 基于Django的博客系统开发研究[J]. 电脑编程技巧与维护.2016,Vol1(01):39.

[15] 董海兰 蔡祥宝. 基于Django的单文件法创建静态网站[J]. 电脑编程技巧与维护.2016,Vol5(01):6.

毕业设计（论文）

译文及原稿

译文题目 基于Web Socket实时通信研究

原稿题目 Research of Web Real-Time Communication Based on Web Socket

原稿出处 刘启刚，孙向阳 上海大学悉尼语言与商业研究所，中国上海

基于Web Socket的Web实时通信研究

概述

针对于传统Web实时通信解决方案（比如轮询，长轮询还有Flash插件）存在的局限性，提出了在Web实时通信领域使用新的Web Socket技术，介绍了Web Socket技术的特点，分析了Web Socket协议与HTTP协议的区别，提供了一种在客户端和服务器端实现Web Socket的方法，通过实验证明Web Socket可以大大地使得网络数据流量和网络延迟降低，使的Web Socket的在未来的应用前景更为广阔。

1.简介

在当今快速发展的信息时代，互联网已经成为了人们生活中不可或缺必不可少的一部分。人们对互联网的要求已经从Web 1.0时代的信息可访问性转变为Web 2.0时代的信息交互，以及在越来越多的定价系统，电子商务系统还有新闻发布系统中观察到的当前的即时性的交互。

目前，客户端浏览器和服务器之间的通信都是以基于超文本传输，一种基于请求 - 响应和无状态的应用层协议为基础的。HTTP客户端发起请求。它建立传输控制协议（TCP）连接。在收到客户端的请求消息后，服务器发送回消息作为响应并终止连接。在这样的模型下，服务器无法向客户端发送实时数据。因此，已经应用了诸如Flash，Comet和Ajax长轮询之类的技术来实现客户端和服务器之间的实时通信。但是，这些技术无法实现实时的通信，因为其中一些技术需要在浏览器上安装插件，他们之中得到一些插件可能会会导致服务器负载过重。

2.传统的Web实时通信解决方案

轮询，长轮询和HTTP流是Web开发人员过去用于完成浏览器和服务器之间实时通信的主要解决方案。轮询，长轮询和HTTP流是Web开发人员过去用于完成浏览器和服务器之间实时通信的主要解决方案。

轮询手动刷新页面的方法被自动运行程序取代，是浏览器中应用的实时通信的最早解决方案。该解决方案的一大优势是易于实施，无需客户端和服务器。但是，这个解决方案存在一些明显的缺点，很难弄清楚数据更新的频率，因此浏览器无法及时获取最新数据。另外，在一段时间内没有发生数据更新的情况下，浏览器的频繁请求将产生不必要的网络流量并给服务器带来不必要的负担。

为了使服务器随时与浏览器通信，Web开发人员设计了一种名为long-polling或Comet的新访问机制，服务器将通过该机制将新请求保存一段时间而不是立即发送响应。如果在此期间发生数据更新，服务器将使用新的数据响应浏览器，并且浏览器在收到响应时将发出另一个请求。通过这种机制，浏览器可以及时获取服务器端的最新数据。但是，如果发生大量并发，则通过维护这些实时HTTP连接将大大消耗服务器内存和计算容量。

开发人员还尝试了“HTTP Streaming”访问机制。它的主要区别是服务器永远不会关闭浏览器赞助的连接，服务器会随时使用此连接发送消息。在这种情况下，由于服务器不会发出连接完成的信号，因此服务器的响应可能会被网络中的防火墙和代理服务器缓冲，导致浏览器接收数据时发生一些错误。

3. Web Socket简介

Web Socket作为HTML5的一项新功能，被定义为一种技术，使网页能够使用Web Socket协议与远程主机进行全双工通信。它引入了Web Socket接口，并定义了一个全双工通信通道，它通过Web上的单个插槽进行操作。HTML5 Web Socket有效地提供了到Internet的套接字连接，而且开销最小。与Ajax轮询和Comet解决方案相比，它可以大大减少网络流量和延迟，这些解决方案通常用于通过维护两个HTTP连接来传输实时日期以模拟全双工通信。因此，它是构建可扩展，实时Web通信系统的理想技术。

要使用HTML5 Web Socket将一个Web客户端与另一个远程端点连接，应使用表示要连接的远程端点的有效URL初始化新的Web Socket实例。Web Socket将ws：//和wss：// scheme分别定义为Web Socket和安全Web Socket连接。在客户端和服务器之间的初始握手期间将HTTP协议更新为Web Socket协议时，建立Web Socket连接。

Web Socket连接使用标准HTTP端口（80和443），因此，它被称为“代理服务器和防火墙友好协议”。因此，HTML5 Web Socket不需要安装任何新的硬件。如果没有任何中间服务器（代理或反向代理服务器，防火墙，负载平衡路由器等），只要客户端和服务器都支持Web Socket协议，就可以成功建立新的Web Socket连接。

4. Web套接字连接和HTTP连接之间的比较

客户端和服务器之间的通信通常基于HTTP连接，这需要连接到客户端请求和服务器响应的头，根据HTTP协议定义，这些头包含一些传输控制信息，如协议类型，协议版本，浏览器类型，传输语言，编码类型，时间，Cookie和会话。在Firebug和Turn on Live HTTP Headers等软件的帮助下，可以清楚地观察到请求和响应的标题。一个请求和响应头的示例定义如下：

From client (browser) to server:

GET /long-polling HTTP/1.1

Host: [www.kaazing.com](http://www.kaazing.com)

User-Agent: Mozilla/5.0 (X11; U; Linux x86\_64; en-US; rv:1.9) Gecko/2008061017 Firefox/3.0

Accept: text/html,application/xhtml+xml,application/xml;q = 0.9, \*/\*; q = 0.8

Accept-Language: en-us,en;q = 0.5

Accept-Encoding: gzip,deflate

Accept-Charset: ISO-8859-1,utf-8;q = 0.7,\*;q = 0.7

Keep-Alive: 300

Connection: keep-alive

Cache-Control: max-age = 0

Referer: <http://www.example.com/>

From server to client (browser):

Date: Tue, 16 Aug 2008 00:00:00 GMT

Server: Apache/2.2.9 (Unix)

Content-Type: text/plain

Content-Length: 12

Hello world

由以上两个标题显示，除了数据“Hello World”之外，这些标题中的大多数数据对于最终用户在客户端和服务器之间的交互过程中是无用的，让Cookie和Session（这两个项目中包含的信息通常不仅仅是大多数网站标题中的控制信息）。此外，这些类型的标题将包含在每个交互中。因此，如果采用轮询和Comet解决方案，它必须浪费大量带宽，产生大量网络流量。此外，构建和分析标头将占用用于处理请求和响应的一些时间，并导致一定程度的延迟。polling和Comet的这些缺点表明，未来这两种技术必须被其他实时通信技术所取代。让我们转向Web Socket连接。

Web Socket使用HTTP Upgrade机制升级到Web Socket协议。Web Socket的握手机制与HTTP兼容。因此，HTTP服务器可以与Web Socket服务器共享默认的HTTP和HTTPS端口（80和443）。要建立新的Web Socket连接，在客户端和服务器之间的初始握手期间，HTTP协议将升级为Web Socket协议。建立连接后，Web Socket将基于全双工模型在客户端和服务器之间来回传输。初始握手的标题如下：

From client (browser) to server:

GET /text HTTP/1.1

Upgrade: WebSocket

Connection: Upgrade

Host: www.websocket.org

From server to client (browser):

HTTP/1.1 101 WebSocket Protocol Handshake

Upgrade: WebSocket

Connection: Upgrade

Hello World

清楚地表明，Web Socket连接头中包含的控制信息远远少于HTTP连接头中的控制信息。另一方面，根据Web Socket协议的规范，Web Socket连接的头部不允许使用Cookie和Session，首先，一旦连接成功建立，客户端可以自由地与服务器通信，只有两位控制信息附加到由UTF-8编码的最终用户所需数据，一位是位于开头的“\ x00”，另一位是位于末尾的“\ xFF”。Web Socket的这一定义大大减少了在Web Socket连接中处理标头所消耗的带宽和时间，然后导致更少的网络流量和更低的延迟。这些是Web Socket比polling和Comet更适合基于Web的实时通信的确切原因。从安全的角度来看，Web Socket协议和HTTP协议都可以实现安全传输。Wss和https是它们独立的安全传输协议。因此，Web Socket被认为是在网络流量，延迟和安全性方面进行实时通信的理想技术。

5. Web Socket技术的实现

A.客户端Web Socket的实现客户端的实现相对简单。下面是W3C工作组给出的Web Socket接口的定义：

[Constructor(in DOMString url, in optional DOMString

protocol)]

interface WebSocket {

readonly attribute DOMString URL;

// ready state

const unsigned short CONNECTING = 0;

const unsigned short OPEN = 1;

const unsigned short CLOSED = 2;

readonly attribute unsigned short readyState;

readonly attribute unsigned long bufferedAmount;

// networking

attribute Function onopen;

attribute Function onmessage;

attribute Function onclose;

oolean send(in DOMString data);

void close();

};

根据接口和构造函数的定义，新的Web Socket实例可以通过两个参数初始化，一个必要参数是有效的网络工作地址，另一个可选参数是协议类型。在浏览器中，Web Socket对象由JavaScript操作。可以通过一段简单的代码创建一个新的Web Socket实例：

var myWebSocket = new WebSocket

(“ws://www.websocket.org”);

前缀“ws”表示Web Socket连接，与此相关的是“wss”，它表示安全的Web Socket连接。在初始化之前，有必要检查客户端浏览器是否支持Web Socket技术：

if (“WebSocket” in window)

{var ws = new WebSocket

(“ws://example.com/service”);}

else

{alert(“WebSockets NOT supported here”);}

在发送消息之前，应该注册几个函数来处理Web Socket连接过程的连续事件，例如成功建立连接，接收消息，关闭消息。

myWebSocket.onopen = function(evt)

{alert(“Connection open ...”); };

myWebSocket.onmessage = function(evt) {alert( “Received

Message: “+ evt.data);};

myWebSocket.onclose = function(evt)

{alert(“Connection closed.”);};

要发送消息，只需调用post message方法，然后将消息内容作为其默认参数。发送消息后，请调用disconnect方法终止连接。

myWebSocket.postMessage(“Hello Web Socket! “);

myWebSocket. disconnect(); stockTickerWebSocket.

disconnect();

B.服务器端Web Socket的实现

与客户端的实现相比，服务器端非常复杂，客户端的大多数操作如生成头，分析头，提取有用数据都是由浏览器自动完成的，但这些都没有实现在服务器上，应该由开发人员手动完成。服务器端Web Socket实现主要依赖于Socket编程，这对C＃，Java和C ++来说很常见。在本文中，作者使用C＃实现服务器端Web Socket功能。首先，应该创建一个监听器来监视网络中的新请求。

private Socket serverListener = new Socket（AddressFamily.InterNetwork，SocketType.Stream，rotocolType.IP）;

accept函数负责监听新的请求，因此，它应该放在一个循环中，该循环一直运行以随时获取客户端请求。

while (true){Socket sc = serverListener.Accept();//get a new connection if (sc != null) { … }//process the request}

当收到新的连接请求时，类似地，需要注册几个服务器功能来处理诸如接收消息，发送消息，关闭通信期间发生的连接等事件。

ci.ReceiveData += new ClientSocketEvent (Ci\_ReceiveData);

ci.BroadcastMessage += new BroadcastEvent (ci. SendMessage);

ci.DisConnection += new ClientSocketEvent (Ci\_DisConnection)

然后调用BeginReceive方法接收客户端请求消息，并尝试与客户端浏览器握手，如果握手成功，则可以启动全双工通信。

ci.ClientSocket.BeginReceive（ci.receivedDataBuffer，0，ci.receivedDataBuffer.Length，0，new Async Call back（ci.StartHandshake），ci.ClientSocket.Available）;

在上面的代码中，StartHandshake方法负责根据客户端请求生成握手信息。在此方法中，将从请求标头中获取两个密钥“Sec-WebSocket-Key1”和“Sec-WebSocket-Key2”的值，并且将基于这两个值完成Web Socket协议中定义的MD5计算，并返回最后的结果是，MD5结果是在握手过程中保护数据的一种手段。如果握手成功，则新连接将被放入连接轮询中以供下次重用。

listConnection.Add（CI）;

这里要注意的是在发送消息时将字符“\ x00”放在消息的开头，将“\ xFF”放在消息的末尾，并在读取消息时删除这两个字符。此外，消息应在使用前由UTF-8编码或解码。

public void SendMessage(MessageEntity me){

ClientSocket.Send(new byte[] {0x00});

ClientSocket. Send(Encoding. UTF8. GetBytes (JsonConvert.SerializeObject(me)));

ClientSocket.Send(new byte[] { 0xff });

}

最后，调用DisConnection methd以在通信结束时关闭连接。