I 摘 要  
  
HTML5是下一代互联网的Web标准，和以前的版本不同，HTML5并非仅仅用来表示Web 内容，它将使Web 进入一个成熟的应用平台，在这个平台上，视频、音频、图像和动画，以及同电脑的交互都被标准化。随着Adobe宣布停止Flash移动版开发，不少人预言HTML5的发展将给移动互联网产业带来革命性的深刻变化。作为新一代的网页语言，HTML5跨平台的支持得到不断增强，智能手机的迅速普及加上资费和网络速度的逐渐松绑，也为HTML5技术提供了良好的发展环境。 这其中有“Web 的 TCP”之称的 WebSocket 格外吸引开发人员的注意。WebSocket 的出现使得浏览器提供对 Socket 的支持成为可能，从而在浏览器和服务器之间提供了一个基于 TCP 连接的双向通道。Web 开发人员可以非常方便地使用 WebSocket 构建实时 Web 应用，开发人员的手中从此又多了一柄神兵利器。 本文首先分析国内外研究现状，然后介绍本文用到的关键技术的介绍以及本文实例开发中的环境配置的方法，接下来会详细说明如何利用WebSocket技术结合Golang实现一个简单的web聊天系统。

1 引言

1.1 课题背景

随着互联网的快速发展，HTML5逐渐成为了当代互联网web的标准，网络聊天工具作为一种重要的信息交流工具，受到越来越多的网民的喜爱。目前，出现了很多非常不错的聊天工具，其中应用比较广泛的有qq，微信，陌陌。无论个人还是企业等组织机构，对于沟通的需求也在不断发展，随着软件、网络和通信三大信息现代技术的发展，在沟通协作方面有着更多方便、快捷、事实等优势的即时通讯，成为了继电话、email之后有一个完全融入每个人生活的互联网工具。

最近，围绕HTML5 Web Sockets发表了大量热议，其中定义了一个通过Web上的单个套接字运行的全双工通信通道。HTML5 Web套接字不仅仅是对传统HTTP通信的另一种增量增强; 它代表了巨大的进步，特别是对于实时的，事件驱动的web应用程序。

1.2 国内外研究现状

目前国内聊天工具中层出不穷，不过老大当属腾讯的qq、微信这两款聊天工具，其中这两款聊天工具均有web版本。而且微信的web版本充分利用了HTML5的新特性，使得在浏览器中体验也不亚于客户端。

许多网站为了给客户带来更好的服务，纷纷在网站上添加了客户功能。例如：淘宝、天猫、京东的客户服务，还有三大营运商的客服服务。均在web上实现了即时通讯。

1.3 课题研究内容

本次课题研究为使用Go语言和HTML5实现一个web即时聊天系统。系统主要的模块有登录、注册模块，一对一聊天，群聊聊天，添加好友，添加群组等功能

1.4 课题研究方法  
本课题的研究方法为理论+实践。  
国内外搜集HTML5与GOlang相关的资料。  
使用go语言作为后端开发语言，HTML5做为前端开发语言，sublime text3开发环境。通过mysql数据库，

1. HTML5

2.1 实时Web应用程序的简史  
Web是围绕这样的想法构建的，客户端的工作是从服务器请求数据，服务器的工作是满足这些请求。这种模式多年来没有被挑战，但随着2005年左右引入AJAX，许多人开始探索在客户端和服务器之间建立双向连接的可能性。

Web应用程序已经长大了，现在比以前消耗更多的数据。阻止他们回来的最大的事情是客户端发起的事务的传统HTTP模型。为了克服这一点，设计了许多不同的策略以允许服务器将数据推送到客户端。这些策略中最流行的一种是长期轮询。这包括保持HTTP连接打开，直到服务器有一些数据推送到客户端。

所有这些解决方案的问题是它们携带HTTP的开销。每次发出HTTP请求时，一大堆标头和Cookie数据都会传输到服务器。这可能累加需要传输的相当大量的数据，这反过来增加延迟。如果你正在构建类似于基于浏览器的游戏，减少延迟是保持事务顺利运行的关键。最糟糕的部分是，很多这些头和cookie实际上不需要满足客户端的请求。

2.2 websocket是什么

Websocket是html5提出的一个协议规范，参考rfc6455。  
websocket约定了一个通信的规范，通过一个握手的机制，客户端（浏览器）和服务器（webserver）之间能建立一个类似tcp的连接，从而方便c－s之间的通信。在websocket出现之前，web交互一般是基于http协议的短连接或者长连接。

WebSocket是为解决客户端与服务端实时通信而产生的技术。websocket协议本质上是一个基于tcp的协议，是先通过HTTP/HTTPS协议发起一条特殊的http请求进行握手后创建一个用于交换数据的TCP连接，此后服务端与客户端通过此TCP连接进行实时通信。

2.3 WebSockets如何工作

WebSockets在客户端和服务器之间提供持久连接，双方可以随时使用它们来开始发送数据。客户端通过称为WebSocket握手的过程建立WebSocket连接。此过程从客户端向服务器发送常规HTTP请求开始。Upgrade此请求中包含一个标头，通知服务器客户端希望建立WebSocket连接。

以下是初始请求标头的简化示例。

GET ws://websocket.example.com/ HTTP/1.1  
Origin: http://example.com  
Connection: Upgrade  
Host: websocket.example.com  
Upgrade: websocket

WebSocket URL使用ws方案。还有wss对等的安全WebSocket连接HTTPS。  
如果服务器支持WebSocket协议，则它同意升级并通过Upgrade响应中的标头进行通信。

HTTP/1.1 101 WebSocket Protocol Handshake  
Date: Wed, 16 Oct 2013 10:07:34 GMT  
Connection: Upgrade  
Upgrade: WebSocket

现在握手已完成，初始HTTP连接将替换为使用相同底层TCP / IP连接的WebSocket连接。此时，任一方都可以开始发送数据。

使用WebSockets，您可以传输尽可能多的数据，而不会产生与传统HTTP请求相关的开销。数据通过WebSocket作为消息传输，每个消息由一个或多个包含您要发送的数据（有效载荷）的帧组成。为了确保消息可以正确重建，当它到达客户端时，每个帧都有4-12字节的有关有效负载的数据前缀。使用这种基于帧的消息系统有助于减少传输的非有效载荷数据量，从而显着减少延迟。

3、Go语言

Go编程语言是一个开源项目，使程序员更高效。

Golang是简洁、高效的。它的并发机制使得编写能够充分利用多核和联网机器的程序变得容易，而其新颖的类型系统实现了灵活的模块化程序构造。Go快速编译到机器代码，但具有垃圾收集的方便和运行时反射的力量。它是一个快速，静态类型，编译的语言，感觉像一个动态类型，解释语言。

轮询，就是你去车站上厕所。车站有很多人，你第一次去，发现厕所有人，于是你又回来等，过了几分钟，再去，发现还是有人。然后在回来等待，然后又去。以此往复。