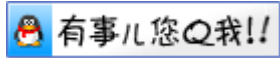


.NET快速开发框架

RDIFramework.NET, 基于.NET的快速信息化系统开发、整合框架, 给用户和开发者最佳的.Net框架部署方案。请关注官方博客: <http://blog.rdiframework.net/>

:: 首页 :: 新随笔 :: 联系 :: 订阅  :: 管理 336 Posts :: 79 Stories :: 2986 Comments :: 0 Trackbacks

公告



扫一扫上面的二维码图案，加我微信

个人简介

关于作者：专注于微软平台项目架构、管理和企业解决方案，多年项目开发与管理经验，曾多次组织并开发多个大型项目，在面向对象、面向服务以及数据库领域有一定的造诣。现主要从事基于RDIFramework.NET框架的技术开发、咨询工作，主要服务于金融、医疗卫生、铁路、电信、物流、物联网、制造、零售等行业。

系统架构师、信息系统项目管理师、DBA。

交流QQ：406590790

框架交流群：185568405



昵称：.NET快速开发框架
园龄：9年9个月
粉丝：1848
关注：43
[+加关注](#)

搜索

谷歌搜索

史上最全面的SignalR系列教程-2、SignalR 实现推送功能-永久连接类实现方式

1、概述

通过上篇史上最全面的SignalR系列教程-1、认识SignalR文章的介绍，我们对SignalR技术已经有了一个全面的了解。本篇开始就通过SignalR的典型应用的实现方式做介绍，例子虽然简单，但麻雀虽小五脏俱全。可以以此为基础，扩展到实际应用中。

SignalR 的实现机制与.NET WCF 或 Remoting 是相似的，都是使用远程代理来实现。在具体使用上，有两种不同通信模型：PersistentConnection 和 Hubs，其中 PersistentConnection 是实现了长时间的 Javascript 轮询（类似于 Comet），Hub 是用来解决实时信息交换问题，它是利用 Javascript 动态载入执行方法实现的。SignalR 将整个连接，信息交换过程封装得非常漂亮，客户端与服务器端全部使用 JSON 来交换数据。

| 通信模型 | 说明 |
|------------------------|---|
| Persistent Connections | Persistent Connections表示一个发送单个，编组，广播信息的简单终结点。开发人员通过使用持久性连接Api，直接访问SignalR公开的底层通信协议。 |
| Hubs | Hubs是基于连接Api的更高级别的通信管道，它允许客户端和服务端上彼此直接调用方法，SignalR能够很神奇地处理跨机器的调度，使得客户端和服务端能够轻松调用在对方端上的方法。使用Hub还允许开发人员将强类型的参数传递给方法并且绑定模型 |

对于这两种方式的讨论，感兴趣的朋友可以移步：
<https://stackoverflow.com/questions/9280484/signalr-why-choose-hub-vs-persistent-connection>

本篇主要介绍使用SignalR的永久连接类(Persistent Connections)的实现方式来实现消息推送（广播）功能，分别以Asp.NET MVC与控制台程序做实例演示。对于Hubs这种集线器的实现方式在下篇文章介绍。

2、SignalR的永久连接类Mvc实现

在ASP.NET MVC下实现SignalR的永久连接类非常的简单，只需简单4步就可以实现一个简单的群聊功能。

2.1、创建ASP.NET Mvc项目

常用链接

- 我的随笔
- 我的评论
- 我的参与
- 最新评论
- 我的标签

最新随笔

- 1..NET分离exe和dll在不同的目录让你的程序更整洁
- 2.最好用的.NET敏捷开发框架-RDIFramework.NET V3.6版全新发布 100%源码授权
- 3.RDIFramework.NET - .NET敏捷开发框架全新发布 V3.5版本 100%源码授权
- 4.史上最全面的SignalR系列教程-目录汇总
- 5.RDIFramework.NET敏捷开发框架 - 工作流程组件Web业务平台
- 6.RDIFramework.NET敏捷开发框架 - 工作流程组件介绍
- 7.RDIFramework.NET敏捷开发框架Web新增邮件中心实现便捷式的邮件收发
- 8.史上最全面的SignalR系列教程-6、SignalR 实现聊天室
- 9.史上最全面的SignalR系列教程-5、SignalR 实现一对一聊天
- 10.RDIFramework.NET敏捷开发框架通过SignalR技术整合即时通讯(IM)

我的标签

- RDIFramework.NET (158)
- C# (131)
- 快速开发框架 (76)
- Web前端 (51)
- Jquery EasyUI (50)
- 项目总结 (46)
- WinForm开发框架 (32)
- 实用数据库技术 (25)
- 项目与团队管理 (24)
- SQL SERVER (24)
- 更多

积分与排名

- 积分 - 596638
- 排名 - 375

随笔分类 (767)

- .NET技术 (145)
- Java (4)
- Oracle (30)
- RDIFramework.NET (151)

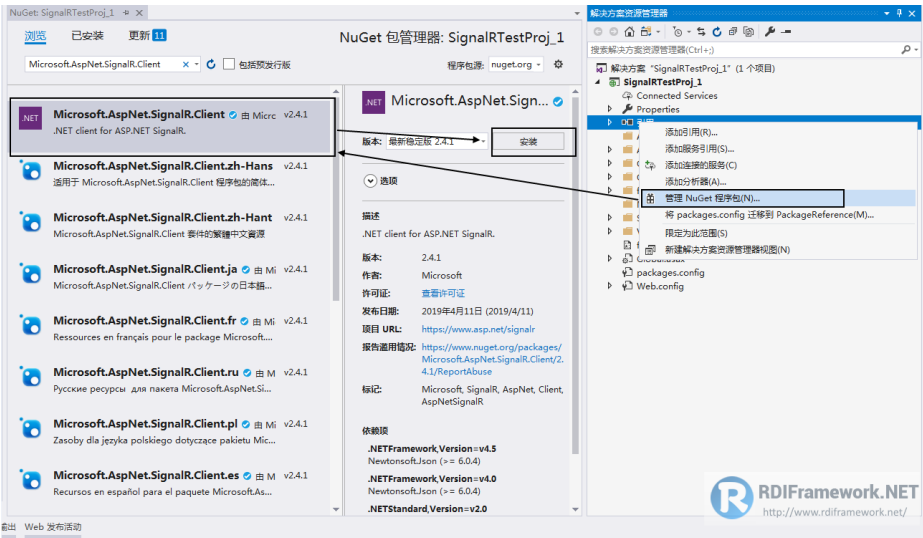


2.2、安装Nuget包

创建好项目后，要使用SignalR，需要先安装SignalR包，可以通过程序包管理控制台输入包安装命令进行安装。

```
Install-Package Microsoft.AspNet.SignalR
```

也可以使用界面方式，如下图所示。



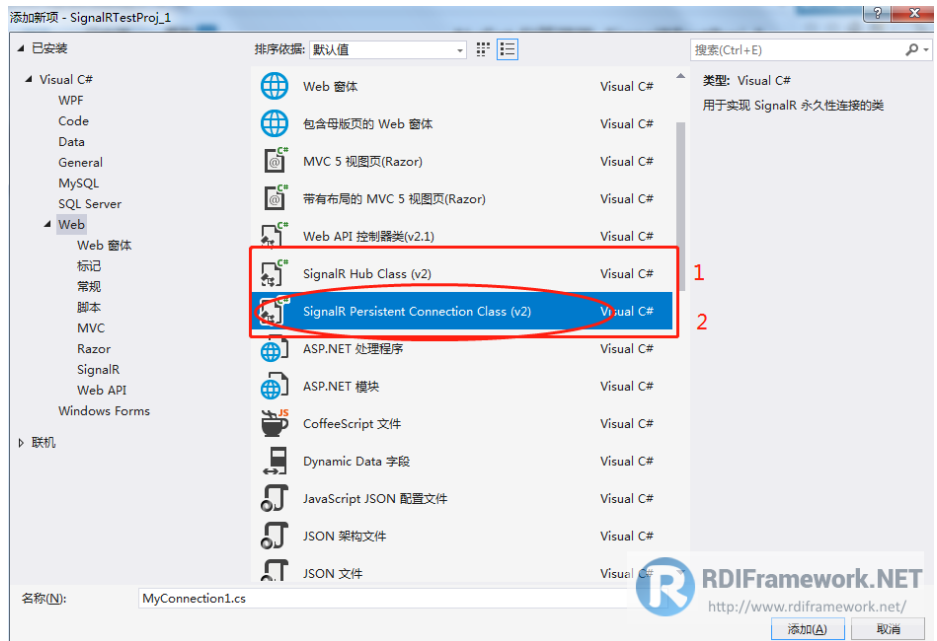
2.3、增加SignalR服务

规划适当的SignalR服务文件存放的文件夹，如我们放在Connections文件夹中，在此文件夹右键选择新建项，我们选择SignalR Persistent Connection Class永久连接类项。

RDIFramework.NET 工作流
(22)
SqlServer(25)
Web前端(77)
国土档案管理信息系统(9)
计算机常用知识(22)
开源(15)
企业级控件库(22)
商业资讯(7)
生活、情感(7)
数据库开发相关技术(85)
数据库开发应用(65)
数据库优化(17)
微信开发(17)
项目与团队管理(31)
消息交互(11)
娱乐(5)

随笔档案 (335)

2020年5月(1)
2020年4月(1)
2019年9月(4)
2019年8月(11)
2019年7月(1)
2019年6月(3)
2019年5月(1)
2019年4月(12)
2018年12月(6)
2018年11月(7)
2018年7月(2)
2018年3月(4)
2018年1月(1)
2017年11月(1)
2017年10月(2)
2017年9月(2)
2017年8月(2)
2017年7月(4)
2017年6月(2)
2017年5月(1)
2017年4月(1)
2017年3月(6)
2017年2月(1)
2017年1月(1)
2016年11月(1)
2016年10月(4)
2016年9月(1)
2016年8月(3)
2016年7月(1)
2016年6月(1)
2016年5月(9)
2015年12月(1)
2015年10月(1)
2015年8月(1)
2015年7月(1)
2015年6月(7)
2015年4月(3)
2015年3月(3)
2015年2月(12)
2015年1月(17)
2014年12月(2)
2014年6月(1)



确定后，VS会自动给我们生成对应代码，如下：

```
using System.Threading.Tasks;
using Microsoft.AspNet.SignalR;

namespace SignalRTestProj_1.Connections
{
    public class ChatConnection : PersistentConnection
    {
        protected override Task OnConnected(IRequest request,
string connectionId)
        {
            return Connection.Send(connectionId, "Welcome!");
        }

        protected override Task OnReceived(IRequest request,
string connectionId, string data)
        {
            return Connection.Broadcast(data);
        }
    }
}
```

我们对代码做适当修改，让业务更人性化，如下所示：

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading;
using System.Threading.Tasks;

namespace SignalRTestProj_1.Connections
{
    using Microsoft.AspNet.SignalR;

    public class ChatConnection : PersistentConnection
    {
        private static int _connections = 0;
```

2014年5月(2)
2014年2月(2)
2013年12月(1)
2013年11月(1)
2013年10月(2)
2013年9月(2)
2013年7月(5)
2013年5月(1)
2013年3月(1)
2013年2月(1)
2013年1月(1)
2012年9月(21)
2012年8月(4)
2012年7月(5)
2012年2月(1)
2012年1月(6)
2011年10月(1)
2011年8月(2)
2011年6月(20)
2011年5月(6)
2011年4月(8)
2011年3月(8)
2011年2月(2)
2011年1月(2)
2010年12月(4)
2009年9月(1)
2009年7月(19)
2009年6月(6)
2009年4月(2)
2009年3月(17)
2008年12月(11)
2008年10月(3)
2008年9月(12)
2008年8月(1)
2008年6月(8)

文章分类 (99)
.Net技术(8)
.NET快速开发框架(1)
Asp.Net(7)
CSS(2)
CSS/Jquery/EasyUI(4)
Excel(19)
Java(1)
JavaScript(4)
SQL SERVER(7)
常用操作技术(4)
开发总结(13)
企业级控件库(1)
数据库技术(14)
项目与团队管理(13)
娱乐、社会、生活(1)

最新评论
1. Re:史上最全面的SignalR系列教程-3、SignalR 实现推送功能-集线器类实现方式按博主的代码来会缺少用户名称,看了博主的源代码才知道,在控制器的HubChat中,有一句:

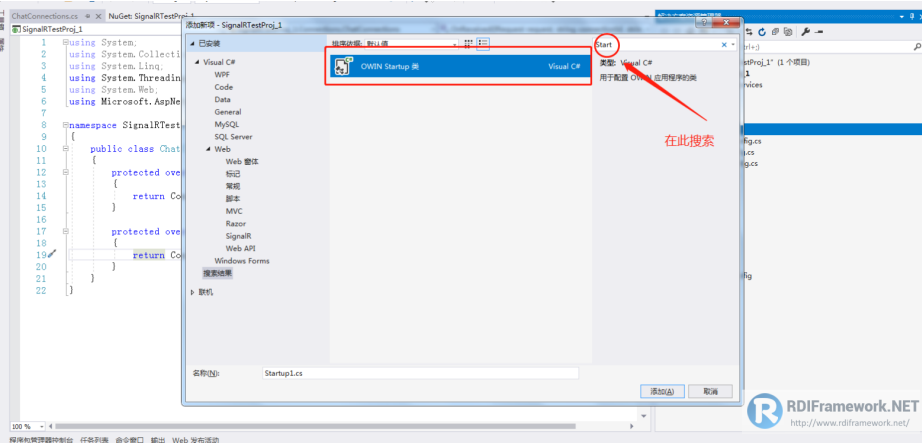
```
protected override Task OnConnected(IRequest request,
string connectionId)
{
    Interlocked.Increment(ref _connections);
    //广播消息
    Connection.Broadcast("新的连接加入, 连接ID: " +
connectionId + ", 已有连接数: " + _connections);
    return Connection.Send(connectionId, "双向连接成功,
连接ID: " + connectionId);
}

/// <summary>
/// 连接断开
/// </summary>
protected override Task OnDisconnected(IRequest
request, string connectionId, bool stopCalled)
{
    Interlocked.Decrement(ref _connections);
    return Connection.Broadcast(connectionId + "退出连
接, 已有连接数: " + _connections);
}

protected override Task OnReceived(IRequest request,
string connectionId, string data)
{
    var message = connectionId + "发送内容>>" + data;
    return Connection.Broadcast(message);
}
}
```

2.4、启动路由注册

要使Signal起作用,我们必须启动路由注册,要让客户端能找到,我们使用Owin。如下图所示,我们添加一个Owin Start启动类。



```
using Microsoft.Owin;
using Owin;
using SignalRTestProj_1.Connections;

[assembly:
OwinStartup(typeof(SignalRTestProj_1.App_Start.ChartStartup))]

namespace SignalRTestProj_1.App_Start
```

ViewBag.ClientName =
"用户-" +
Guid.NewGuid().ToString()
)...
--月光下的待宵草

2. Re:史上最全面的SignalR系列教程-2、SignalR 实现推送功能-永久连接类实现方式
感谢楼主干货，照着把栗子做了一遍，补充一下细节。1. 建议直接新建MVC模板项目，用空web项目会比较麻烦，反正这不是重点。2. 最好直接用安装命令安装，不容易选错，而且我用搜索方式会很慢。3. 添加...

--月光下的待宵草

3. Re:[推荐推荐][提供下载]
ORACLE SQL: 经典查询练手系列文章收尾(目录篇)

确实不错

--91洲际哥

4. Re:最好用的.NET敏捷开发框架-RDIFramework.NET V3.6版全新发布 100%源码授权
@Futuer 感谢提醒，正在续费中。...

--.NET快速开发框架

5. Re:最好用的.NET敏捷开发框架-RDIFramework.NET V3.6版全新发布 100%源码授权

官网域名到期了

--Futuer

6. Re:RDIFramework.NET敏捷开发框架 - 工作流程组件介绍

"少而好学，如日出之阳；壮而好学，如日中之光；志而好学，如炳烛之光。"【暖心良言】

--前方一片光明

7. Re:.NET分离exe和dll在不同的目录让你的程序更整洁
@Cfan1236 Dev...

--.NET快速开发框架

8. Re:.NET分离exe和dll在不同的目录让你的程序更整洁

软件UI不错！用的什么皮肤？

```
{
    public class ChartStartup
    {
        public void Configuration(IApplicationBuilder app)
        {
            // 有关如何配置应用程序的详细信息，请访问
            https://go.microsoft.com/fwlink/?LinkID=316888
            // 配置上文实现的ChatConnections
            app.MapSignalR<ChatConnection>
                ("/Connections/ChatConnection");
        }
    }
}
```

至此，我们服务端就算搭建起来了，现在我们增加一个页面来实现调用。

2.5、前端界面处理

前端js实现消息广播，并实时记录

```
<html>
    <head>
        <meta name="viewport" content="width=device-width" />
    </head>
    <title>SignalR永久连接类 Demo</title>
    <script src="~/Scripts/jquery-3.3.1.min.js"></script>
    <script src="~/Scripts/jquery.signalR-2.4.1.min.js"></script>
    </head>
    <body>
        <div>
            <h1>SignalR永久连接类 Demo</h1>
            <div>
                <input type="text" id="msgContext"
                    placeholder="请输入消息内容..." />
                <button id="sendMessage" class="btn-primary">Send</button>
                <ul id="messages"></ul>
            </div>
            <script>
                var connection =
                    $.connection("/Connections/ChatConnection");
                connection.logging = true;
                //客户端接收消息
                connection.received(function (data) {
                    $('#messages').append('<li>' + data +
                        '</li>');
                });
                //连接错误处理
                connection.error(function (err) {
                    alert('与服务器连接报错:' + err.message);
                });
                //连接成功
                connection.start().done(function () {
                    $('#sendMessage').click(function () {
                        var val = $('#msgContext').val();
                        //向服务器端发送消息
```


--Cfan1236
9. Re:最好用的.NET敏捷开发框架-RDIFramework.NET V3.6版全新发布 100%源码授权
@智通云狐 你好， 官网加微信详解! ...

--.NET快速开发框架
10. Re:最好用的.NET敏捷开发框架-RDIFramework.NET V3.6版全新发布 100%源码授权

你好，对你们这个框架很感兴趣，大概什么价格?

--智通云狐

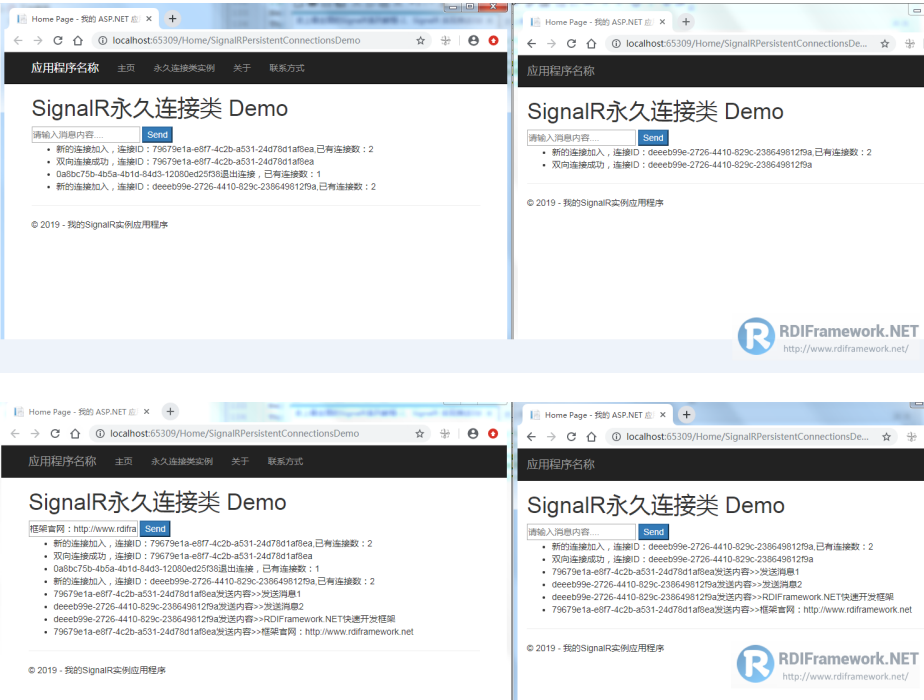
阅读排行榜

- 1. 微软良心之作——Visual Studio Code 开源免费跨平台代码编辑器(148575)
- 2. ORACLE PL/SQL编程之八：把触发器说透(123455)
- 3. [推荐]ORACLE PL/SQL编程之四：把游标说透(不怕做不到，只怕想不到)(73539)
- 4. 【推荐】使用Jquery+EasyUI进行框架项目开发案例讲解之一---员工管理源码分享(68351)
- 5. [推荐]ORACLE PL/SQL编程详解之一：PL/SQL 程序设计简介(千里之行，始于足下)(49021)
- 6. ORACLE PL/SQL编程之六：把过程与函数说透(穷追猛打，把根儿都拔起!)(44100)
- 7. Javascript中的apply与call详解(39090)
- 8. [顶]ORACLE PL/SQL编程详解之二：PL/SQL块结构和组成元素(为山九仞，岂一日之功)(37643)
- 9. 【强烈推荐】《ORACLE PL/SQL编程详解》全原创（共八篇）--系列文章导航(32938)
- 10. [推荐]ORACLE PL/SQL编程之五：异常错误处理(知己知彼、百战不殆)(29640)
- 11. Vue常用经典开源项目汇总参考-海量(26944)
- 12. [强烈推荐]ORACLE PL/SQL编程详解之七：程序包的创建与应用(聪明在于学习，天才在于积累!)(26537)
- 13. [推荐]ORACLE SQL：经典查询练手第一篇(不懂装懂，永世饭桶!)(25122)

```
connection.send(val);  
  
});  
  
});  
  
</script>  
  
</div>  
  
</body>  
  
</html>
```

OK，全面完成，F5运行看效果吧。

2.6、效果展示



3、控制台做SignalR服务端实现

除了Web实时应用之外，我们还可以用其他应用程序实现实时交互，如控制台。只需要Install-Package Microsoft.AspNet.SignalR.Client命令，示例代码如下：

```
using Microsoft.AspNet.SignalR.Client;  
using static System.Console;  
  
namespace ClientWithSignalRDemo  
{  
    class Program  
    {  
        static void Main(string[] args)  
        {  
            var connection = new  
Connection("http://localhost:65309/Connections/ChatConnection"  
);  
  
            connection.Received += WriteLine;  
            connection.Start().Wait();  
  
            string line;  
            while ((line = ReadLine()) != null)  
            {  
                connection.Send(line).Wait();  
            }  
        }  
    }  
}
```

14. 使用Jquery+EasyUI 进行框架项目开发案例讲解之四--组织机构管理源码分享(20935)
15. [推荐]ORACLE PL/SQL编程详解之三: PL/SQL流程控制语句(不给规则, 不成方圆)(20744)

评论排行榜

1. [推荐推荐][提供下载]ORACLE SQL: 经典查询练手系列文章收尾(目录篇)(420)
2. ORACLE PL/SQL编程之八: 把触发器说透(256)
3. (推荐) (提供下载) ORACLE常见问题一千问(不怕学不成、就怕心不诚!)(216)
4. (干货) .NET开发丰富的公共类库助你事半功倍 (供下载免费使用) (185)
5. [提供源码下载]在大型软件中用Word做报表: 书签的应用 (提供一种思路) (106)
6. 记大型商业软件<<国土档案管理信息系统>>之系统简介(89)
7. [推荐]ORACLE SQL: 经典查询练手第一篇(不懂装懂, 永世饭桶!)(86)
8. 【推荐】使用Jquery+EasyUI进行框架项目开发案例讲解之一---员工管理源码分享(81)
9. 推荐+1置顶+1 (分享、讨论、实现) 通用软件注册功能之建立有效的软件保护机制(73)
10. 原创企业级控件库之大数据量分页控件(73)

推荐排行榜

1. [推荐推荐][提供下载]ORACLE SQL: 经典查询练手系列文章收尾(目录篇)(142)
2. (干货) .NET开发丰富的公共类库助你事半功倍 (供下载免费使用) (124)
3. ORACLE PL/SQL编程之八: 把触发器说透(86)
4. 【强烈推荐】《ORACLE PL/SQL编程详解》全原创 (共八篇) --系列文章导航(71)
5. (推荐) (提供下载) ORACLE常见问题一千问(不怕学不成、就怕心不诚!)(58)

导航目录

- 1、概述
- 2、SignalR的永久连接类Mvc实现



4、代码下载

实例源码可以移步github下载, 地址:

<https://github.com/yonghu86/SignalRTestProj>

5、参考文章

- RDIFramework.NET敏捷开发框架通过SignalR技术整合即时通讯(IM)
- 史上最全面的SignalR系列教程-1、认识SignalR
- Real-time ASP.NET with SignalR
- HTML5 WebSocket
- WebSocket WebAPI
- 微信公众号开发系列-玩转微信开发-目录汇总
- RDIFramework.NET — 基于.NET的快速信息化系统开发框架 — 系列目录
- RDIFramework.NET — .NET快速信息化系统开发框架 — 工作流程组件介绍
- RDIFramework.NET框架SOA解决方案 (集Windows服务、WinForm形式与IIS形式发布) -分布式应用
- RDIFramework.NET代码生成器全新V3.5版本发布-重大升级

一路走来数个年头, 感谢RDIFramework.NET框架的支持者与使用者, 大家可以通过下面的地址了解详情。

RDIFramework.NET官方网站: <http://www.rdiframework.net/>

RDIFramework.NET官方博客: <http://blog.rdiframework.net/>

同时需要说明的, 以后的所有技术文章以官方网站为准, 欢迎大家收藏!

RDIFramework.NET框架由海南国思软件科技有限公司专业团队长期打造、一直在更新、一直在升级, 请放心使用!

欢迎关注RDIFramework.net框架官方公众微信 (微信号: guosisoft), 及时了解最新动态。

扫描二维码立即关注

- 2.1、创建ASP.NET Mvc项目
- 2.2、安装Nuget包
- 2.3、增加SignalR服务
- 2.4、启动路由注册
- 2.5、前端界面处理
- 2.6、效果展示
- 3、控制台做SignalR服务端实现
- 4、代码下载
- 5、参考文章

作者: **RDIFramework.NET**

出处: <http://www.cnblogs.com/huyong/>

Email: 406590790@qq.com

QQ交流: 406590790

框架官网: <http://www.rdiframework.net/>

框架官网博客: <http://blog.rdiframework.net/>

框架其他博客: <http://blog.csdn.net/chinahuyong>
<http://www.cnblogs.com/huyong>

RDIFramework.NET, 基于.NET的快速信息化系统开发、整合框架, 给用户和开发者最佳的.Net框架部署方案。

关于作者: 系统架构师、信息系统项目管理师、DBA。专注于微软平台项目架构、管理和企业解决方案, 多年项目开发与管理经验, 曾多次组织并开发多个大型项目, 在面向对象、面向服务以及数据库领域有一定的造诣。现主要从事基于RDIFramework.NET 框架的技术开发、咨询工作, 主要服务于金融、医疗卫生、铁路、电信、物流、物联网、制造、零售等行业。

如有问题或建议, 请多多赐教!

本文版权归作者和CNBLOGS博客共有, 欢迎转载, 但未经作者同意必须保留此段声明, 且在文章页面明显位置给出原文连接, 如有问题, 可以通过邮箱或QQ 联系我, 非常感谢。

分类: .NET技术, RDIFramework.NET, 消息交互

标签: RDIFramework.NET, C#, 快速开发框架, 项目总结, 即时通讯, WinForm开发框架

好文要顶

关注我

收藏该文



.NET快速开发框架

关注 - 43

粉丝 - 1848

+加关注

2

« 上一篇: 史上最全面的SignalR系列教程-1、认识SignalR

» 下一篇: 史上最全面的SignalR系列教程-3、SignalR 实现推送功能-集线器类实现方式

posted on 2019-08-12 11:26 .NET快速开发框架 阅读(940) 评论(2) 编辑 收藏

Feedback

#1楼 2019-08-12 11:52 其实我不笨

mark

支持(0) 反对(0)

#2楼 2020-06-10 14:49 月光下的待宵草

感谢楼主干货, 照着把栗子做了一遍, 补充一下细节。

- 1.建议直接新建MVC模板项目, 用空web项目会比较麻烦, 反正这不是重点。
- 2.最好直接用安装命令安装, 不容易选错, 而且我用搜索方式会很慢。

- 3.添加持久连接类不要在所有模板里搜索，直接在connection文件夹右键就有，既方便还可以避免一些小问题（见下面）。
- 4.Owin Start类用不着新建，MVC项目根目录自动有，直接在方法内添加那一句即可，注意引用。即使自己新建，也按我第二点说的，直接右键新建即可，否则可能会有版本问题。我开始通过所有模板搜索新建的start文件，owin版本是2.1，但是在global文件中却显示3.0，于是运行后报错。后来我删掉，右键直接添加，生成的版本就一致了。
- 5.视图文件中，注意jQuery版本和ChatConnection的路径。

支持(0) 反对(0)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问](#) 网站首页。

【推荐】了解你才能更懂你，博客园首发问卷调查，助力社区新升级

【推荐】超50万行VC++源码：大型组态工控、电力仿真CAD与GIS源码库

【推荐】《Flutter in action》开放下载！闲鱼Flutter企业级实践精选

Copyright © 2020 .NET快速开发框架
Powered by .NET Core on Kubernetes
Powered by: .Text and ASP.NET
Theme by: .NET Monster