**[C#开发微信门户及应用(6)--微信门户菜单的管理操作](http://www.cnblogs.com/wuhuacong/p/3701961.html)**

前面几篇继续了我自己对于C#开发微信门户及应用的技术探索和相关的经验总结，继续探索微信API并分享相关的技术，一方面是为了和大家对这方面进行互动沟通，另一方面也是专心做好微信应用的底层技术开发，把基础模块夯实，在未来的应用中派上用途。本随笔继续介绍微信门户菜单的管理操作。

**1、菜单的基础信息**

微信门户的菜单，一般服务号和订阅号都可以拥有这个模块的开发，但是订阅号好像需要认证后才能拥有，而服务号则不需要认证就可以拥有了。这个菜单可以有编辑模式和开发模式，编辑模式主要就是在微信门户的平台上，对菜单进行编辑；而开发模式，就是用户可以通过调用微信的API对菜单进行定制开发，通过POST数据到微信服务器，从而生成对应的菜单内容。本文主要介绍基于开发模式的菜单管理操作。

自定义菜单能够帮助公众号丰富界面，让用户更好更快地理解公众号的功能。目前自定义菜单最多包括3个一级菜单，每个一级菜单最多包含5个二级菜单。一级菜单最多4个汉字，二级菜单最多7个汉字，多出来的部分将会以“...”代替。目前自定义菜单接口可实现两种类型按钮，如下：

click：

用户点击click类型按钮后，微信服务器会通过消息接口推送消息类型为event 的结构给开发者（参考消息接口指南），并且带上按钮中开发者填写的key值，开发者可以通过自定义的key值与用户进行交互；

view：

用户点击view类型按钮后，微信客户端将会打开开发者在按钮中填写的url值 （即网页链接），达到打开网页的目的，建议与网页授权获取用户基本信息接口结合，获得用户的登入个人信息。

菜单提交的数据，本身是一个Json的数据字符串，它的官方例子数据如下所示。

[复制代码](javascript:void(0);)

{

"button":[

{

"type":"click",

"name":"今日歌曲",

"key":"V1001\_TODAY\_MUSIC"

},

{

"type":"click",

"name":"歌手简介",

"key":"V1001\_TODAY\_SINGER"

},

{

"name":"菜单",

"sub\_button":[

{

"type":"view",

"name":"搜索",

"url":"http://www.soso.com/"

},

{

"type":"view",

"name":"视频",

"url":"http://v.qq.com/"

},

{

"type":"click",

"name":"赞一下我们",

"key":"V1001\_GOOD"

}]

}]

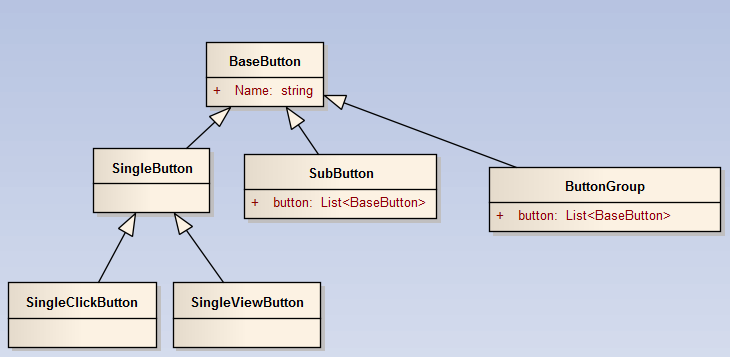
}

[复制代码](javascript:void(0);)

从上面我们可以看到，菜单不同的type类型，有不同的字段内容，如type为view的有url属性，而type为click的，则有key属性。而菜单可以有子菜单sub\_button属性，总得来说，为了构造好对应的菜单实体类信息，不是一下就能分析的出来。

**2、菜单的实体类定义**

我看过一些微信接口的开发代码，把菜单的分为了好多个实体类，指定了继承关系，然后分别对他们进行属性的配置，大概的关系如下所示。



这种多层关系的继承方式能解决问题，不过我觉得并不是优雅的解决方案。其实结合Json.NET自身的Attribute属性配置，可以指定那些为空的内容在序列号为Json字符串的时候，不显示出来的。

[JsonProperty( NullValueHandling = NullValueHandling.Ignore)]

有了这个属性，我们就可以统一定义菜单的实体类信息更多的属性了，可以把View类型和Click类型的菜单属性的url和key合并在一起。

[复制代码](javascript:void(0);)

/// <summary>

/// 菜单基本信息

/// </summary>

public class MenuInfo

{

/// <summary>

/// 按钮描述，既按钮名字，不超过16个字节，子菜单不超过40个字节

/// </summary>

public string name { get; set; }

/// <summary>

/// 按钮类型（click或view）

/// </summary>

[JsonProperty(NullValueHandling = NullValueHandling.Ignore)]

public string type { get; set; }

/// <summary>

/// 按钮KEY值，用于消息接口(event类型)推送，不超过128字节

/// </summary>

[JsonProperty(NullValueHandling = NullValueHandling.Ignore)]

public string key { get; set; }

/// <summary>

/// 网页链接，用户点击按钮可打开链接，不超过256字节

/// </summary>

[JsonProperty(NullValueHandling = NullValueHandling.Ignore)]

public string url { get; set; }

/// <summary>

/// 子按钮数组，按钮个数应为2~5个

/// </summary>

[JsonProperty(NullValueHandling = NullValueHandling.Ignore)]

public List<MenuInfo> sub\_button { get; set; }

.......

[复制代码](javascript:void(0);)

但是，这么多信息，不同的类型我需要指定不同的属性类型，那不是挺麻烦，万一我在View类型的菜单里面，把key属性设置了，那怎么办？

解决方法就是我们定义几个构造函数，分别用来构造不同的菜单信息，如下所示是对菜单不同的类型，赋值给不同的属性的构造函数。

[复制代码](javascript:void(0);)

/// <summary>

/// 参数化构造函数

/// </summary>

/// <param name="name">按钮名称</param>

/// <param name="buttonType">菜单按钮类型</param>

/// <param name="value">按钮的键值（Click)，或者连接URL(View)</param>

public MenuInfo(string name, ButtonType buttonType, string value)

{

this.name = name;

this.type = buttonType.ToString();

if (buttonType == ButtonType.click)

{

this.key = value;

}

else if(buttonType == ButtonType.view)

{

this.url = value;

}

}

[复制代码](javascript:void(0);)

好了，还有另外一个问题，子菜单也就是属性sub\_button是可有可无的东西，有的话，需要指定Name属性，并添加它的sub\_button集合对象就可以了，那么我们在增加一个构造子菜单的对象信息的构造函数。

[复制代码](javascript:void(0);)

/// <summary>

/// 参数化构造函数,用于构造子菜单

/// </summary>

/// <param name="name">按钮名称</param>

/// <param name="sub\_button">子菜单集合</param>

public MenuInfo(string name, IEnumerable<MenuInfo> sub\_button)

{

this.name = name;

this.sub\_button = new List<MenuInfo>();

this.sub\_button.AddRange(sub\_button);

}

[复制代码](javascript:void(0);)

由于只指定Name和sub\_button的属性内容，其他内容为null的话，自然构造出来的Json就没有包含它们，非常完美！

为了获取菜单的信息，我们还需要定义两个实体对象，如下所示。

[复制代码](javascript:void(0);)

/// <summary>

/// 菜单的Json字符串对象

/// </summary>

public class MenuJson

{

public List<MenuInfo> button { get; set; }

public MenuJson()

{

button = new List<MenuInfo>();

}

}

/// <summary>

/// 菜单列表的Json对象

/// </summary>

public class MenuListJson

{

public MenuJson menu { get; set; }

}

[复制代码](javascript:void(0);)

**3、菜单管理操作的接口实现**

我们从微信的定义里面，可以看到，我们通过API可以获取菜单信息、创建菜单、删除菜单，那么我们来定义它们的接口如下。

[复制代码](javascript:void(0);)

/// <summary>

/// 菜单的相关操作

/// </summary>

public interface IMenuApi

{

/// <summary>

/// 获取菜单数据

/// </summary>

/// <param name="accessToken">调用接口凭证</param>

/// <returns></returns>

MenuJson GetMenu(string accessToken);

/// <summary>

/// 创建菜单

/// </summary>

/// <param name="accessToken">调用接口凭证</param>

/// <param name="menuJson">菜单对象</param>

/// <returns></returns>

CommonResult CreateMenu(string accessToken, MenuJson menuJson);

/// <summary>

/// 删除菜单

/// </summary>

/// <param name="accessToken">调用接口凭证</param>

/// <returns></returns>

CommonResult DeleteMenu(string accessToken);

}

[复制代码](javascript:void(0);)

具体的获取菜单信息的实现如下。

[复制代码](javascript:void(0);)

/// <summary>

/// 获取菜单数据

/// </summary>

/// <param name="accessToken">调用接口凭证</param>

/// <returns></returns>

public MenuJson GetMenu(string accessToken)

{

MenuJson menu = null;

var url = string.Format("https://api.weixin.qq.com/cgi-bin/menu/get?access\_token={0}", accessToken);

MenuListJson list = JsonHelper<MenuListJson>.ConvertJson(url);

if (list != null)

{

menu = list.menu;

}

return menu;

}

[复制代码](javascript:void(0);)

这里就是把返回的Json数据，统一转换为我们需要的实体信息了，一步到位。

调用代码如下所示。

[复制代码](javascript:void(0);)

private void btnGetMenuJson\_Click(object sender, EventArgs e)

{

IMenuApi menuBLL = new MenuApi();

MenuJson menu = menuBLL.GetMenu(token);

if (menu != null)

{

Console.WriteLine(menu.ToJson());

}

}

[复制代码](javascript:void(0);)

创建和删除菜单对象的操作实现如下所示。

[复制代码](javascript:void(0);)

/// <summary>

/// 创建菜单

/// </summary>

/// <param name="accessToken">调用接口凭证</param>

/// <param name="menuJson">菜单对象</param>

/// <returns></returns>

public CommonResult CreateMenu(string accessToken, MenuJson menuJson)

{

var url = string.Format("https://api.weixin.qq.com/cgi-bin/menu/create?access\_token={0}", accessToken);

string postData = menuJson.ToJson();

return Helper.GetExecuteResult(url, postData);

}

/// <summary>

/// 删除菜单

/// </summary>

/// <param name="accessToken">调用接口凭证</param>

/// <returns></returns>

public CommonResult DeleteMenu(string accessToken)

{

var url = string.Format("https://api.weixin.qq.com/cgi-bin/menu/delete?access\_token={0}", accessToken);

return Helper.GetExecuteResult(url);

}

[复制代码](javascript:void(0);)

看到这里，有些人可能会问，实体类你简化了，那么创建菜单是不是挺麻烦的，特别是构造对应的信息应该如何操作呢？前面不是介绍了不同的构造函数了吗，通过他们简单就搞定了，不用记下太多的实体类及它们的继承关系来处理菜单信息。

[复制代码](javascript:void(0);)

private void btnCreateMenu\_Click(object sender, EventArgs e)

{

MenuInfo productInfo = new MenuInfo("软件产品", new MenuInfo[] {

new MenuInfo("病人资料管理系统", ButtonType.click, "patient"),

new MenuInfo("客户关系管理系统", ButtonType.click, "crm"),

new MenuInfo("酒店管理系统", ButtonType.click, "hotel"),

new MenuInfo("送水管理系统", ButtonType.click, "water")

});

MenuInfo frameworkInfo = new MenuInfo("框架产品", new MenuInfo[] {

new MenuInfo("Win开发框架", ButtonType.click, "win"),

new MenuInfo("WCF开发框架", ButtonType.click, "wcf"),

new MenuInfo("混合式框架", ButtonType.click, "mix"),

new MenuInfo("Web开发框架", ButtonType.click, "web"),

new MenuInfo("代码生成工具", ButtonType.click, "database2sharp")

});

MenuInfo relatedInfo = new MenuInfo("相关链接", new MenuInfo[] {

new MenuInfo("公司介绍", ButtonType.click, "Event\_Company"),

new MenuInfo("官方网站", ButtonType.view, "http://www.iqidi.com"),

new MenuInfo("提点建议", ButtonType.click, "Event\_Suggestion"),

new MenuInfo("联系客服", ButtonType.click, "Event\_Contact"),

new MenuInfo("发邮件", ButtonType.view, "http://mail.qq.com/cgi-bin/qm\_share?t=qm\_mailme&email=S31yfX15fn8LOjplKCQm")

});

MenuJson menuJson = new MenuJson();

menuJson.button.AddRange(new MenuInfo[] { productInfo, frameworkInfo, relatedInfo });

//Console.WriteLine(menuJson.ToJson());

if (MessageUtil.ShowYesNoAndWarning("您确认要创建菜单吗") == System.Windows.Forms.DialogResult.Yes)

{

IMenuApi menuBLL = new MenuApi();

CommonResult result = menuBLL.CreateMenu(token, menuJson);

Console.WriteLine("创建菜单：" + (result.Success ? "成功" : "失败:" + result.ErrorMessage));

}

}

[复制代码](javascript:void(0);)

这个就是我微信门户里面的菜单操作了，具体效果可以关注我的微信门户：广州爱奇迪，也可以扫描下面二维码进行关注了解。



菜单的效果如下：

