

虚拟化校园食堂餐饮管理系统的设计与实现

张银霞,郭畅

(北京联合大学师范学院,北京 100011)

摘要:

微信和微信公众号的推出,让人们的生活发生了巨大的变化,智能手机在大学生群体中的使用率越来越高。本软件系统基于校园实际需求调研,实现利用手机微信扫码访问,综合在学校食堂点餐、评价、食堂餐饮管理等功能为一体,分为客户端和服务端。客户端基于.NET 微信公众平台开发;服务器端搭载 ASP.NET 动态网站和 SQL 数据库服务器,系统具有菜品信息推荐、论坛、交友、订餐配送和支付等强大的智能化信息处理能力;不同用户身份权限不同,虚拟校园餐厅功能可丰富大学生的就餐交流,弘扬中国传统餐饮文化。

关键词:

餐饮管理;虚拟化校园食堂;.NET 微信公众平台开发;ASP.NET 动态网站;SQL 数据库服务器

基金项目:

2018 年北京联合大学应用型课程建设项目(No.12205561107-102);2019 年北京市高水平人才交叉培养实培项目(创业类 No.137)

0 引言

智能手机在大学生群体中的使用率越来越高,他们很习惯在网络平台购物、点餐、交流。现在越来越多的学校门口在就餐时间内聚集了大量的送餐外卖,而外卖餐饮的制作环境、食材的安全性等方面都存在诸多问题。实际上,校园食堂的餐饮相对于校外的餐饮的优势还是很大的,安全卫生、绿色健康价格还亲民,但随着校园师生的逐渐增多,排队的情况日益增加,因各种原因无法来到食堂点餐的情况也有很多。结合这两大问题,反观我国手机微信市场、校园食堂订餐软件,目前还没有一款将校园食堂和外卖结合在一起的微信公众号。本系统基于校园实际需求调研,开发一款名为“虚拟化校园食堂餐饮系统”的微信公众号,利用手机微信扫码集学校食堂点餐、评价、食堂餐饮管理等多功能为一体的综合餐饮管理系统,面向不同的用户身份开放权限,虚拟校园食堂就餐功能,提供交友、论坛空间,丰富大学生的就餐交流方式,弘扬中国传统餐饮文化。

1 系统分析与设计

1.1 系统功能结构设计

本系统分为前台客户端和后台服务器端两部分,主要功能需求分为食堂订餐及配餐、虚拟化校园食堂空间和管理四大模块。本系统分为三种角色:教师和学生、食堂工作人员和管理人员。前台客户端是基于手机微信扫码在手机上使用,其中:食堂订餐、虚拟化校园食堂空间由教师和学生使用,可以进行查看学校食堂的配餐、菜品信息智能化筛选、下单订购、菜品评价、餐饮自主搭配、论坛交流及交友、查看积分和虚拟化食堂功能等;食堂配餐模块由食堂工作人员使用,可以进行菜品信息的发布。后台服务器端是 ASP.NET 动态网站在电脑端使用,主要是该系统的管理模块由食堂工作人员和管理员共同使用,可以实现对学生或教工的信息管理、学生菜品评价信息管理、食堂员工信息管理、公告管理、菜品信息发布和充值等内容,功能模块图如图 1 所示。

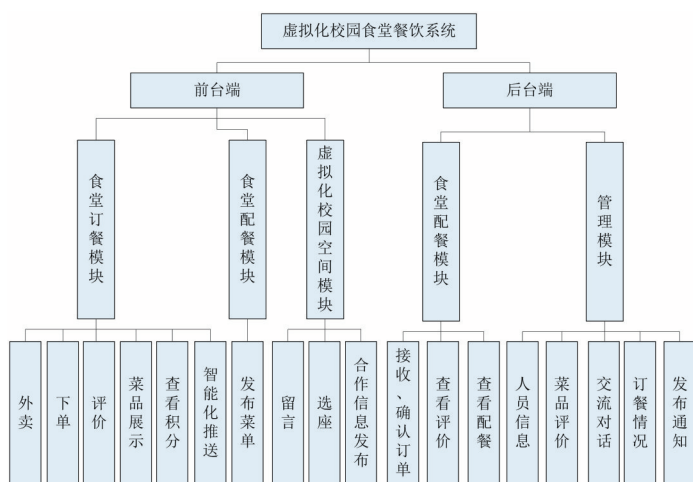


图1 系统功能模块图

1.2 开发技术

在实现技术方面,本软件的客户端基于.NET 的 MVC 模式开发微信公众平台的应用程序,选择“开发模式”的微信公众平台服务,开发账号的类型为测试账号,开发工具为 Visual Studio 2015,用到的微信前端开发技术有 HTML5、CSS3、JavaScript、jQuery Mobile、微信 JS-DK、微信浏览器私有接口 WeixinJSBridge 和 JSAPI;本软件的服务器端管理系统为 ASP.NET 动态网站,开发工具为 Visual Studio 2015,编程语言为 C#,使用 SQL Server 2008 进行数据存储和管理。

1.3 环境部署

(1)配置 ASP.NET 空间:本系统采用阿里云主机服务器作为存放 ASP.NET MVC 应用程序的主机空间,并申请域名。

(2)微信开发公众平台模式设计:本系统需要调用微信公众平台提供的开发接口 API,所以要申请成为开发者模式。登录微信公众平台官方网站,申请成为微信公众平台开发者,获取应用 ID(APPID)和应用秘钥(AppSecret)。

(3)申请接口测试号:本系统申请微信公众测试号,可以调用所有接口,不受限制。

2 获取 access_token

本软件系统的功能数据基本都来源于自建库,需要调用微信公众平台的接口,实现客户端微信和服务端动态网站数据交互。开发者在调用微信公众平台

的各个接口时多需要使用 access_token。在获取 access_token 时需要开发者的 appID 和 appsecret,通过在后台服务器端动态网站的页面中添加完成,获取的关键代码实现如下所示:

```

public string GetAccessToken(int accountId, out string error) {
    string access_token = string.Empty; error = string.Empty;
    try {
        //公众平台账户信
        Model.weixin_account accountModel = accountBLL.GetModel
(accountId);
        if (string.IsNullOrEmpty(accountModel.appid) || string.IsNul
lOrEmpty(accountModel.appsecret)) {
            error = "AppId 或者 AppSecret 未填写,请在补全信息! ";
            return string.Empty;
        }
        //没有找到该账户则获取 AccessToken 写入存储 1200 秒
        if (!tokenBLL.ExistsAccount(accountId)) {
            var result = Senparc.Weixin.MP.CommonAPIs.CommonApi.
GetToken(accountModel.appid,
            accountModel.appsecret);
            access_token = result.access_token;
            tokenBLL.Add(accountId, access_token);
            return access_token;
        }
        //获取公众账户的实体
        Model.weixin_access_token tokenModel = tokenBLL.GetAc
countModel(accountId);
        //计算时间判断是否过期
        TimeSpan ts = DateTime.Now - tokenModel.add_time;
        double chajunSecond = ts.TotalSeconds;
        if (string.IsNullOrEmpty(tokenModel.access_token) || chajun
Second >= tokenModel.expires_in) {
            //从微信平台重新获得 AccessToken
            var result = Senparc.Weixin.MP.CommonAPIs.CommonApi.
GetToken(accountModel.appid, accountModel.appsecret);
            access_token = result.access_token;
            //更新到数据库里的 AccessToken
            tokenModel.access_token = access_token;
            tokenModel.add_time = DateTime.Now;
            tokenBLL.Update(tokenModel);
        }
    }
    else {

```

```

    access_token = tokenModel.access_token;
}
}
catch (Exception ex) {
    error = "获取 AccessToken 出错:" + ex.Message;
}
return access_token;
}

```

3 系统核心功能的实现

本系统功能分支较多,此章节仅列出有代表性的系统核心功能的实现方法。

3.1 公告信息模块

本系统公告信息由管理员在后台服务器网页发布,实时推送到手机微信消息界面供教师和学生用户查看,公告信息模块的实现代码如下所示。

```

private bool RealseAnnouncemnet(int id) {
    CRMComm wx = new CRMComm();
    string errmsg = string.Empty;
    string ACCESS_TOKEN = wx.GetAccessToken(1, out errmsg);
    bool result = false;
    BLL.announcement bll = new BLL.announcement();
    Model.announcement model = bll.GetModel(id);
    BLL.user_oauth bll_user_oauth = new BLL.user_oauth();
    DataTable user_oauth_list = bll_user_oauth.GetList(0, "id>0",
    "id desc").Tables[0];
    foreach (DataRow user_oauth_dr in user_oauth_list.Rows) {
        string url = "https://api.weixin.qq.com/cgi-bin/message/tem-
        plate/send?access_token=" + ACCESS_TOKEN;
        string wxopenid = user_oauth_dr["oauth_openid"].ToString();
        string head_title = model.title;
        string release_time = model.create_time;
        string release_content = model.summary;
        string remark = model.remark;
        string notice_url = "http://" + new BLL.sites().GetModel(2).
        domain + "/aspx/wap/notice.aspx?action=" +
        DTEnums.ActionEnum.Edit + "&id=" + model.id;
        string data = notice_msg_data(wxopenid, notice_url, head_ti-
        tle, release_time, release_content, remark);
        Utils.HttpPost(url, data);
    }
    model.status = 1; //发布通知
}

```

```

if (bll.Update(model)) {
    result = true;
}
return result;
}

```

实现原理:访问根据微信的公众类 CRMComm,通过 wx.GetAccessToken 获得 ACCESS_TOKEN 值,并获得用户信息 users。根据 model.user_name 用户名拿到 oauth_openid 微信 openid,并通过微信推送信息的 URL,将通知标题、通知时间、通知简介进行发布,并且按照 send_msg_date 里的通过后台申请的微信模板信息(每个模板信息 id 不同),最后用 HttpPost 将 URL 的数据 post 到上面的 URL 地址上,最终推送到微信消息界面。并将通知的控件 model 的状态 status 由 bll.update 进行更新。

3.2 消息的智能推送

每当用户购买完毕,会根据用户的购买情况推送相关菜品到微信消息界面,调用微信接口推送消息的实现原理同公告信息模块,只是模板信息 id 不同,消息的智能推送代码的实现代码如下所示:

```

private bool send_caiping(int id, int item_id) {
    CRMComm wx = new CRMComm();
    string errmsg = string.Empty;
    string ACCESS_TOKEN = wx.GetAccessToken(1, out errmsg);
    bool result = false;
    BLL.users bll = new BLL.users();
    Model.users model = bll.GetModel(id);
    BLL.user_oauth bll_user_oauth = new BLL.user_oauth();
    string url = "https://api.weixin.qq.com/cgi-bin/message/tem-
    plate/send?access_token=" + ACCESS_TOKEN;
    string wxopenid = bll_user_oauth.GetModel(model.user_name).
    oauth_openid;
    string head_title = model.real_name;
    string release_time = DateTime.Now.ToShortDateString();
    string release_content = model.user_name;
    string remark = model.user_name;
    string notice_url = "http://" + new BLL.sites().GetModel(2).
    domain + "/aspx/wap/food_edit.aspx?action=" +
    DTEnums.ActionEnum.Edit + "&id=" + item_id; string data =
    send_msg_data(wxopenid,
    notice_url, head_title, release_time, release_content, re-
    mark);
    Utils.HttpPost(url, data);
}

```

```
model.status = 1; //发布通知
if (bll.Update(model)) {
    result = true;
}
return result;
}
```

3.3 食堂订餐模块

教师和学生用户通过主页点击“食堂订餐”按钮进入菜品列表页面,可预览食堂菜品名称、图片、简介、添加时间、价格。代码实现时,通过 StringBuilder 对象 templateBuilder,使用 templateBuilder.Append 方法以 HTML5 标记方式实现,菜品信息展示的核心代码如下所示:

```
Model.users model = GetUserInfo( );
int uid = model.id;
BLL.article bll = new BLL.article( );
DataTable noticeList = bll.GetList(2,20, "id>0 and brand_id=0", "id desc").Tables[0];
StringBuilder templateBuilder = new StringBuilder(220000);
foreach (DataRow noticedr in noticeList.Rows) {
    templateBuilder.Append("<li>");
    templateBuilder.Append("<a href=\"");
    templateBuilder.Append("food_edit.aspx?action=" + DTE-
nums.ActionEnum.Edit + "&id=" +
    Utils.ObjectToStr(noticedr["id"]);
    templateBuilder.Append("&id=\"");
    templateBuilder.Append(Utils.ObjectToStr(noticedr["title"]);
    templateBuilder.Append(">");
}
```

点击相应的菜品可选择数量并加入购物车,当点击确认按钮将执行 ShopCart.Add 自动将菜品加入购物车并跳转到购物车 food_cart.aspx,菜品添加购物车实现的核心代码如下所示:

```
protected void confirmButton_Click(object sender, EventArgs e) {
    int goods_id = model.id;
    int quantity = Utils.ObjToInt(txtCommoditySelectNum.Value);
    ShopCart.Add(model.channel_id, model.id, 0, quantity);
    Response.Redirect("food_cart.aspx");
}
```

3.4 食堂配餐模块

食堂人员可通过手机微信员工入口输入管理员设

置的特定账号登录,通过手机拍照、相册和 iCloud 方式上传菜品图片、输入菜品名称、简介、价格发布菜品,该功能是通过在页面的源代码中通过 JavaScript 脚本代码调用第三方插件完成,实现的核心代码如下所示:

```
<script type="text/javascript">
$(function ( ) {
    //初始化表单验证
    $("#form1").initValidform( );
    //初始化上传控件
    $(".upload-img").InitUploader({
        sendurl: "../tools/upload_ajax.ashx",
        swf: "../scripts/webuploader/uploader.swf"
    });
});
</script>
```

食堂人员可以输入菜品详情信息并上传到数据库中,通过 DoAdd 增加操作,将菜品的名称、价格、图片以及摘要上传,食堂配餐信息上传数据库实现的关键代码如下所示:

```
private bool DoAdd( ) {
    BLL.article bll = new BLL.article( );
    Model.article model = new Model.article( );
    model.channel_id = 2;
    model.site_id = 2;
    model.title = txtCaiMing.Value;
    model.tags = txtPrice.Value;
    model.img_url = txtAvatar.Text;
    model.zhaiyao = txtCaiPingJianJie.Value;
    model.cai_type = ddlCaiType.SelectedValue;
    if (bll.Add(model) > 0) {
        return true;
    }
    return false;
}
```

3.5 “虚拟化校园食堂空间”模块

由于学校统一对学生进行管理的特殊性,学生就餐时间会相对比较集中,导致放学时间段食堂人流量大、排队时间长、就餐人员用餐体验差等问题,同时现在的大学生使用手机上网的频率越来越高,在网络上他们可以畅所欲言、展示自己的个性,但由于时间、空间、场地等各种限制,学生交友范围有限,但校园的人

员身份相对比较安全可靠,可以提供一个交友的网络虚拟平台,让大学生们边吃边聊,就像置身于校园食堂、自主餐厅等现实环境中一样,在放松愉悦的同时,使学生们的交友范围可以跨专业、跨班级,结交更多的校园好友,也有利于学生们形成专业互补、组队比赛、考证帮扶等学习活动的开展,给他们的大学生活带来更多的乐趣。基于这样的一个目的在这款校园食堂餐饮管理系统中又开发出一个功能叫“虚拟化校园食堂空间”。

“虚拟化校园食堂空间”以大学生学习需求的种类为标准进行空间版块划分,由管理员在服务器端动态网站中进行空间组的创建和管理,以论坛的形式展示,教师和学生根据需要进行选择空间版块组,不同的空间组只能看到本组的内容。教师和学生留言后不是立刻显示而是需要后台管理员人工筛选审核后才会显示,以保证系统健康的交流环境。教师和学生可以通过上方的搜索框来进行关键词的搜索。

(1)“虚拟化校园食堂空间”组的创建和管理实现

该部分是在服务器端动态网页中实现,虚拟化校园食堂空间组的赋值和显示实现见下方 ShowInfo 事件过程处理,增加空间组的实现关键代码见下方 DoAdd 事件过程处理。

```
private void ShowInfo(int _id) {
    BLL.xt_virtual_group bll = new BLL.xt_virtual_group();
    Model.xt_virtual_group model = bll.GetModel(_id);
    txtTitle.Text = model.title;
    if (model.status == 0) {
        rbllsLock.Checked = true;
    }
    else {
        rbllsLock.Checked = false;
    }
}

private bool DoAdd() {
    bool result = false;
    Model.xt_virtual_group model = new Model.xt_virtual_group();
    BLL.xt_virtual_group bll = new BLL.xt_virtual_group();
    model.title = txtTitle.Text.Trim();
    model.status = 1;
    if (bll.Add(model) > 0) {
        result = true;
    }
}
```

```
}
return result;
}
```

实现原理:输入的组名和选择是否隐藏通过 ShowInfo 进行赋值,根据 bll.GetModel 里的 id 将输入的值赋值到控件 model 上。并且通过 DoAdd 进行增加到虚拟化校园空间列表内,进行对控件 model 的 bll.Add 插入操作。最后根据 bll.GetModel 里的 id 将输入的值赋值到控件 model 上。并且通过 DoEdit 进行对 status 的 model 空间进行修改操作,如果 rbllsLock.Checked 是开着的,那么将 status 改为 0,反之改为 1,执行 bll.update 将 model 进行修改。

(2)“虚拟化校园食堂空间”客户端页面实现

“虚拟化校园食堂空间”客户端页面在手机上运行,可以实现“预定座位”、预定和会员点击“预定餐位”三种方式进入,通过 HTML5 语言显示了虚拟化校园空间的分组名称和布局,实现的关键代码如下所示:

```
<div class="weui-grids">
<% DataTable dt = new DTcms.BLL.xt_virtual_group().GetList(
0, "status=1", "id asc").Tables[0]; %>
<% foreach (DataRow dr in dt.Rows) { %>
    <a href="xt_virtual_group_user.aspx?id=<%=dr["id"]%>"
class="weui-grid js_grid">
    <p class="weui-grid__label">
    <%=dr["title"] %>
    </p>
    </a>
    <% } %>
</div>
```

(3)“虚拟化校园食堂空间”组的论坛版块

虚拟化校园空间页面可以分别从首页点击“预定座位”按钮、预定和会员点击“预定餐位”三种方式进入。用户可以在对应版块发布消息、留言、讨论等内容,单击提交按钮后,可将姓名、发表内容和发布时间等信息上传到后台数据库的留言表中,等待管理员审核后显示,留言内容上传数据库实现的关键代码如下所示:

```
protected void btnSubmit_Click(object sender, EventArgs e) {
    this.id = DTRequest.GetQueryInt("id");
    BLL.xt_virtual_group bll = new BLL.xt_virtual_group();
    Model.xt_virtual_group model = bll.GetModel(id);
}
```

```

BLL.xt_virtual_group_comment bllt = new BLL.xt_virtual_group_comment();
Model.xt_virtual_group_comment modelt = new Model.xt_virtual_group_comment();
modelt.virtual_group_id = model.id;
modelt.virtual_group_title = model.title;
modelt.user_id = GetUserInfo().id;
modelt.user_name = GetUserInfo().user_name;
modelt.real_name = GetUserInfo().real_name;
modelt.content = txtContent.Value.Trim();
modelt.add_time = DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss");
if (bllt.Add(modelt) > 0) {
    Response.Redirect("xt_virtual_group_commnet.aspx?id=" + model.id);
}
}

```

用户在搜索栏中输入关键词后可以实现搜索功能,前台页面的关键词搜索实现的关键代码如下所示:

```

<div class="input-box">
<input id="keywords" name="keywords" placeholder="输入关键字"
onkeydown="if (event.keyCode==13) { SiteSearch ('/asp/wap/xt_virtual_group_commnet.aspx?id=<%=id%>&', '#keywords');
return false };"/>
<a href="javascript:;" onclick="SiteSearch ('/asp/wap/xt_virtual_group_commnet.aspx?id=<%=id%>&', '#keywords');">
<i class="icon icon-4 f25 weui-tabbar__icon"></i> </a>

```

```
</div>
```

其中,SiteSearch 函数代码如下所示:

```

function SiteSearch(send_url, divTgs) {
var str = $.trim($("#divTgs").val());
if (str.length > 0 && str != "输入关键字") {
    window.location.href = send_url + "?keyword=" + encodeURIComponent($("#divTgs").val());
}
return false;
}

```

4 程序运行界面效果

该系统的部分运行结果依据上述核心功能模块的顺序进行展示,公告信息实现效果如图 2 所示,消息的智能推送如图 3 所示,食堂订餐模块如图 4 所示,“虚拟化校园食堂空间”模块如图 5 所示。

5 结语

本软件系统实现时重点突破解决了如下问题:

(1)系统应用的便捷性,使用微信扫码即可访问本系统,不需额外下载 App,符合人们使用手机的习惯。

(2)菜品推荐的多样化标准的设定和信息的智能化处理,从系统平台的数据中提取用餐者以菜品的种类为标准的个人偏好和用餐规律,进行菜品推荐,并为食堂配餐倾向性,做好服务工作提供决策依据。

(3)平台的虚拟化校园餐厅功能,提供诸如交流、评价、论坛、交友等功能,模拟实际校园餐厅的空间和作用,给学生提供更多的相互认识、交流的机会,弘扬



图 2 公告信息结果



图 3 消息智能推送



图 4 食堂订餐



图 5 “虚拟化校园食堂”

中国传统用餐文化,丰富学生的大学生活,也为其各项学习活动提供互帮互助的机会,也有利于改善学生的就餐体验。

本软件系统还有以下几点先进性和独特性:

(1)现在市场上此类功能定位的软件系统还比较少,本系统是规范、安全、净化校园用餐环境,辅助学校进行餐饮管理的好助手。

(2)本系统自主开发的服务器端后台,搭载动态网站和数据库服务器,具有强大的智能信息处理能力,相对于市面上单纯的 App 和使用微信公众号、小程序创建起来的类似软件,对信息的处理能力更强大,可较好地服务于客户端定制信息的需求,给用户更好地使用感受。

参考文献:

- [1]刘瑞. 基于微信公众号的高校智能食堂建设的探索与畅想[J]. 电脑知识与技术, 2017, 13(31):282-283.
- [2]李淑娟,周伟强,宓詠. 高校智慧点餐系统的探索与实践[J]. 华中师范大学学报:自然科学版, 2017(S1):59-62.
- [3]杜欢. 基于 ASP 技术的“智慧食堂”网络系统的实现[J/OL]. 电子技术与软件工程, 2013(19):26-28.
- [4]邹珊珊. 基于微信公众平台的校园订餐系统设计与实现[D]. 扬州大学, 2018.
- [5]陈柳飞. 基于微信公众平台的连锁餐饮管理系统的设计与实现[D]. 华南理工大学, 2017.

作者简介:

张银霞(1979-),女,河南郑州人,硕士,讲师,研究方向为计算机网络与应用、信息安全

郭畅(1998-),男,北京人,在读本科

收稿时间:2020-05-12 修稿时间:2020-06-12

Design and Implementation of the Virtual Campus Canteen and the Restaurant Management System

ZHANG Yin-xia, GUO Chang

(Teacher's College of Beijing Union University, Beijing 100011)

Abstract:

The launch of WeChat and WeChat official has made great changes in people's lives. Smart phones are more and more used in College Students. Based on the actual requirement investigation in the campus, the software system has achieved Wechat scanning access by the mobile phone, and has integrated the ordering in the school canteen, the evaluating, the restaurant management and so on, and is divided into client and server. The client is developed on the base of the .Net Wechat public platform, and the server carries the Asp.net dynamic website and the SQL database server, and the system has the powerful intelligent information processing capability, and the different user has the different identity permission, and the virtual campus restaurant function can enrich the dining communication of the college students and carry forward the Chinese traditional catering culture.

Keywords:

Restaurant Management; Virtual Campus Canteen; .NET WeChat Public Platform; ASP.NET Dynamic Website; SQL Database Server