



[行动校园]

后台说明手册

[基于微信开发的后台 web 系统]

编写:芸来软件[彭云]

2018 年 8 月 22 日

1. 需求说明.....	4
1.1 业务需求.....	4
1.2 功能需求.....	5
1.2.1 针对管理员.....	5
1.2.2 针对用户.....	5
1.3 系统总体说明.....	6
1.3.1 设计思路.....	6
1.3.2 前端结构.....	8
1.3.3 后台结构.....	8
2. 功能模块概述.....	9
2.1 登陆到后台管理系统.....	9
2.1.1 简单参数介绍.....	9
2.1.2 登录后台.....	9
2.1.3 登录信息错误.....	10
2.1.4 登录信息匹配.....	11
2.2 后台界面菜单.....	12
2.2.1 菜单简介.....	12
2.3 系统管理.....	15
2.3.1 系统设置.....	15
2.3.2 用户管理.....	15
2.3.3 城市设置.....	15
2.4 校购统计.....	15
2.5 兼职管理.....	15
2.5.1 兼职类型管理.....	15
2.5.2 兼职管理.....	15
2.5.3 兼职报名管理.....	15
2.5.4 企业管理.....	15
2.5.5 兼职投诉管理.....	15
2.6 校购管理.....	16
2.6.1 任务模板管理.....	16
2.6.2 跑腿任务管理.....	16
2.6.3 使用校购学校管理.....	16
2.6.4 校购投诉管理.....	16
2.7 用户提现.....	16
2.8 意见反馈.....	16
2.9 操作说明.....	16

摘要

开发环境:windows+linux

软件运行环境:linux[Centos 7]

使用数据库:mysql

使用编程语言:前端:微信+js 后台:PHP

使用框架:前端:微信 wxml 后台:laravel

版本维护工具:git

关键字:信息发布 校园跑腿 兼职 MVC 轻量级
自动化 视图分离

1. 需求说明

1.1 业务需求

基于微信平台定制的行动校园系统是一个相当于校园生活服务类的基于微信小程序的后台管理系统，为在校大学生在闲暇时间分配及主动寻找兼职的机会还有主动参与骑手送货跑腿的乐趣。行动校园系统基于实时定位可以在你所在的大学第一时间内获取到附近所在大学的同样使用该系统的小伙伴发布的兼职及任务，你可以在校园内主动发布一个兼职或者跑腿任务，然后等待兴趣相投的用户一起参加；完成活动参与，从而丰富自己的闲暇时间。由活动引发社交，从线上到线下。“不止于约，不期而聚。”通过线上的信息引导用户参与线下活动并进一步巩固社交关系。除此之外行动校园的跑腿模块还具备个人钱包功能，用户可以在完成跑腿任务后获得丰厚的打赏金额回馈并且可以直接提现到微信零钱钱包，从而进一步丰富了购物娱乐体验，当然不乏有那些信用差的同学屡次恶意使用系统做不正当的交易，因此我们可以使用该系统操作用户的信誉积分来主动冻结用户的使用权限以警示用户使用规范及约束行为。

1.2 功能需求

本系统的特点之一是基于微信平台，操作简单方便，界面友好。

系统功能大概分为以下几个方面：

1.2.1 针对管理员

主要有以下几大模块：登录模块、系统设置、用户管理、校购统计、兼职管理、兼职报名管理、企业管理、兼职投诉管理、校购管理、跑腿任务管理、校购投诉管理、校购学校管理、用户提现管理、意见反馈管理。

1.2.2 针对用户

主要两大板块：

兼职及校购、发布报名兼职及发布参与校购任务及提现等功能。

1.3 系统总体说明

1.3.1 设计思路

设计开发这个系统前按照传统的 MVC 开发模式进行开发, 先构建好路由 router 以及在控制器中书写逻辑和数据, 然后再封装好对应的 API 来供前端小程序读取数据, 这里大致把兼职和校购拆成两块来做然后微信认证登陆和一些公有数据采用统一模块, 其中兼职和校购的接口都是独立开发完成大致 API 路由如下:

公有:

```
$api->version('v1', function ($api) {  
    $api->group(['middleware' => 'api'], function($api){  
        //小程序公用部分接口  
        $api->group(['prefix' => 'mini_program'], function ($api) {  
            ## 用户登录  
            $api->get('login', 'App\Http\Controllers\API\CommonController@loginMiniprogram');  
            $api->group(['middleware' => 'zcjy_api_user'], function ($api) {  
                ### 图片上传  
                $api->post('upload_images', 'App\Http\Controllers\API\CommonController@uploadImage');  
                ### 获取用户信息  
                $api->get('me', 'App\Http\Controllers\API\CommonController@userInfo');  
                ### 发起意见反馈  
                $api->get('publish_feedback', 'App\Http\Controllers\API\CommonController@publishFeedBack');  
                ### 用户的信用记录  
                $api->get('credit_logs', 'App\Http\Controllers\API\CommonController@userCreditsLog');  
                ### 收到的/发起的投诉  
                $api->get('publish_and_receive_error/{platform_type}', 'App\Http\Controllers\API\CommonController@publishAndReceiveError');  
                ### 更改绑定手机号  
                $api->get('change_mobile', 'App\Http\Controllers\API\CommonController@changeMobile');  
            });  
        });  
    });  
});
```

兼职:

```

api.php
x
//兼职
$app->group(['prefix' => 'part_job'], function ($app) {
    #根据省份名称模糊获取id
    $app->get('get_province_id_by_name', 'App\Http\Controllers\API\PartJobController@getProvinceIdByName');
    #省份列表
    $app->get('provinces_list', 'App\Http\Controllers\API\PartJobController@getBasicProvince');
    #根据上一级id获取对应的子列表
    $app->get('cities_list', 'App\Http\Controllers\API\PartJobController@getCitiesList');
    #获取所有兼职类型
    $app->get('type_all', 'App\Http\Controllers\API\PartJobController@getAllJianZhiType');
    #获取兼职列表
    $app->get('list_all', 'App\Http\Controllers\API\PartJobController@getJianZhiList');

    /**
     * 需要登录后使用
     */
    $app->group(['middleware' => 'api_auth'], function ($app) {
        #获取兼职详情
        $app->get('detail/{id}', 'App\Http\Controllers\API\PartJobController@getJianZhiDetail');
        #个人中心的报名列表
        $app->get('auth_signs', 'App\Http\Controllers\API\PartJobController@userProjectSigns');
        /**
         * 高于系统信管积分可以使用
         */
        $app->group(['middleware' => 'api_credits_limit'], function ($app) {
            #用户确认收款
            $app->get('enter_project_price/{project_sign_id}', 'App\Http\Controllers\API\PartJobController@userEnterProjectPrice');
            #发起兼职报名
            $app->get('auth_publish_sign', 'App\Http\Controllers\API\PartJobController@publishProjectSign');
            #申请为企业用户
            $app->get('auth_apply_company', 'App\Http\Controllers\API\PartJobController@applyForCompanyUser');
            #用户投诉兼职
            $app->get('auth_error_company/{project_id}', 'App\Http\Controllers\API\PartJobController@userErrorProject');
            #用户删除自己的报名记录
            $app->get('auth_del_sign/{id}', 'App\Http\Controllers\API\PartJobController@userDelSelfSigns');
        });
    });

    /**
     * 企业用户才可以操作
     */
    $app->group(['middleware' => 'api_auth_company'], function ($app) {
        #企业用户发布的招聘信息
        $app->get('auth_publish_companys', 'App\Http\Controllers\API\PartJobController@companyPublishProject');
        #企业用户获取对应兼职招聘的报名名单
        $app->get('publish_project_sign/{project_id}', 'App\Http\Controllers\API\PartJobController@companyPublishProjectSign');
        /**
         * 高于系统信管积分可以使用
         */
        $app->group(['middleware' => 'api_credits_limit'], function ($app) {
            #用户完善企业信息
            $app->get('auth_complete_company_info/{id}', 'App\Http\Controllers\API\PartJobController@completeCompanyInfo');
            #企业用户更新单个招聘人员信息
            $app->get('auth_update_project_sign/{id}', 'App\Http\Controllers\API\PartJobController@companyUpdateProjectSign');

            #企业用户发布招聘兼职信息
            $app->get('auth_publish_project', 'App\Http\Controllers\API\PartJobController@companyActionPublishProject');
            #企业投诉个人
            $app->get('company_error_auth/{user_id}', 'App\Http\Controllers\API\PartJobController@companyErrorUser');
            #企业用户撤销兼职
            $app->get('company_cancel_project/{id}', 'App\Http\Controllers\API\PartJobController@companyCancelProject');
            #企业用户删除自己发布的兼职
            $app->get('company_del_project/{id}', 'App\Http\Controllers\API\PartJobController@companyDelProject');
            #企业用户删除用户的报名记录
        });
    });
});

```

校购:

```

114    });
115
116    //跑腿
117    $api->group(['prefix' => 'errand'], function ($api) {
118
119        #所有的任务模板列表
120        $api->get('all_tasks', 'App\Http\Controllers\API\ErrandController@allItems');
121        #对应学校的跑腿任务
122        $api->get('school_tasks', 'App\Http\Controllers\API\ErrandController@schoolTasks');
123        #跑腿任务详情
124        $api->get('task_detail/{id}', 'App\Http\Controllers\API\ErrandController@errandTaskDetail');
125
126        /**
127         * 需要登录后使用
128         */
129        $api->group(['middleware' => 'zcjy_api_user'], function ($api) {
130
131            #添加选择学校
132            $api->get('select_and_add_school', 'App\Http\Controllers\API\ErrandController@selectAndAddSchool');
133            #发布者/买手的任务列表
134            $api->get('tasks/{type?}', 'App\Http\Controllers\API\ErrandController@publisherTasks');
135            #我的钱包 收入/支出
136            $api->get('my_task_log', 'App\Http\Controllers\API\ErrandController@myPublishAndErrandLog');
137            /**
138             * 高于系统信誉积分可以使用
139             */
140            $api->group(['middleware' => 'zcjy_api_credits_limit'], function ($api) {
141
142                #发起提现
143                $api->get('publish_withdraw', 'App\Http\Controllers\API\ErrandController@publishWithdraw');
144                #发布跑腿任务
145                $api->get('publish_task', 'App\Http\Controllers\API\ErrandController@publishErrandTask');
146                #发布人删除任务
147                $api->get('del_task/{id}', 'App\Http\Controllers\API\ErrandController@delErrandTask');
148                #发布取消任务
149                $api->get('cancel_task/{id}', 'App\Http\Controllers\API\ErrandController@cancelErrandTask');
150                #发布人确认收货
151                $api->get('enter_receive_task/{id}', 'App\Http\Controllers\API\ErrandController@publishManEnterTaskRec');
152                #买手接单
153                $api->get('take_order_task/{id}', 'App\Http\Controllers\API\ErrandController@errandTakeOrderTask');
154                #买手确认送达
155                $api->get('enter_arrive_task/{id}', 'App\Http\Controllers\API\ErrandController@buyerEnterTaskArrive');
156                #买手确认物品费用
157                $api->get('enter_item_cost_task/{id}', 'App\Http\Controllers\API\ErrandController@errandEnterItemCostTask');
158                #发布者发起任务支付
159                $api->get('pay_task/{id}', 'App\Http\Controllers\API\ErrandController@payErrandTask');
160                #投诉发起者/买手
161                $api->get('error_task/{id}/{type?}', 'App\Http\Controllers\API\ErrandController@errorErrandTask');
162                #买手取消订单
163                $api->get('cancel_order_task/{id}', 'App\Http\Controllers\API\ErrandController@errandCancelOrderTask');
164
165            });
166        });

```

1.3.2 前端结构

1.3.3 后台结构

2. 功能模块概述

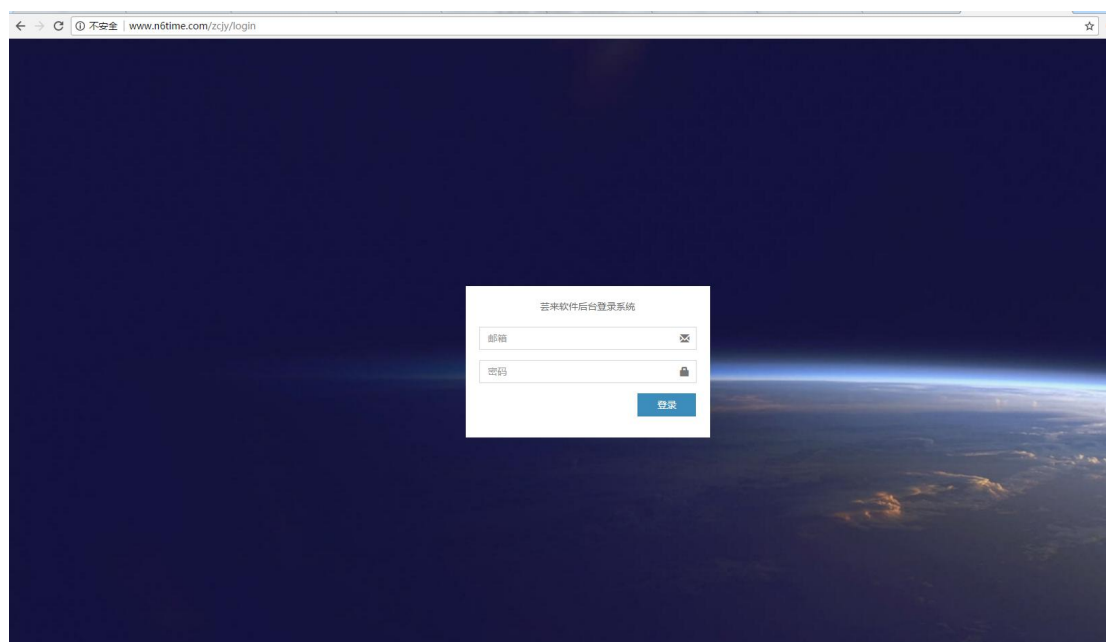
2.1 登陆到后台管理系统

2.1.1 简单参数介绍

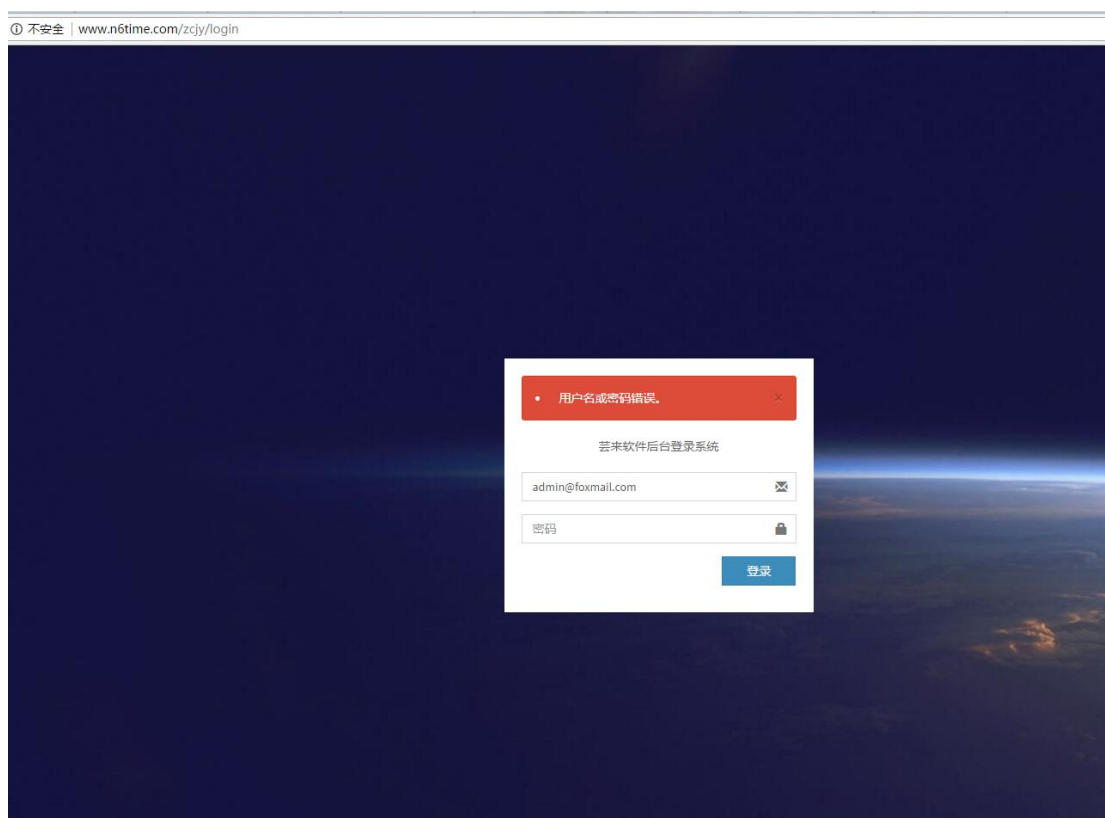
网站默认的后台地址是:<http://www.n6time.com/zcjy>, 如果之前 没有 保存 登陆 信息 系统 会 自动 识别 并 跳 转 到 <http://www.n6time.com/zcjy/login> 。

2.1.2 登录后台

浏览器中输入 <http://www.n6time.com/zcjy/login> 会跳转到如下图所示页面：

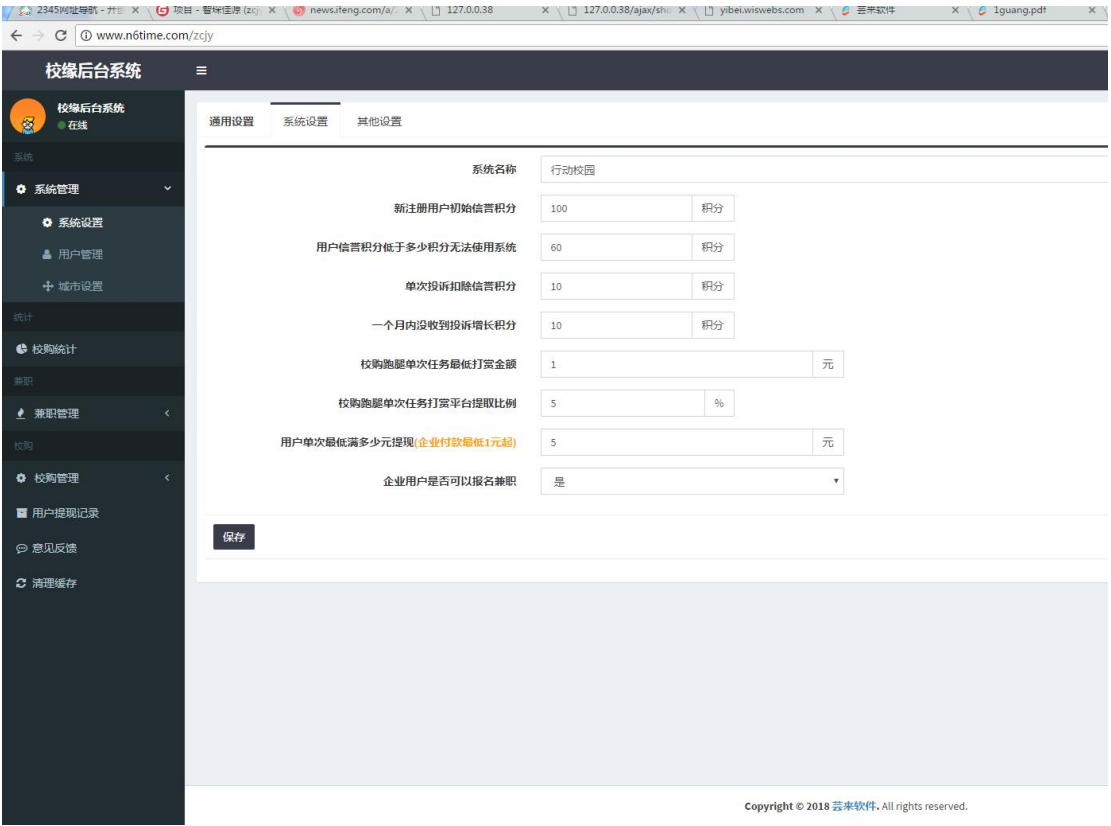


2.1.3 登录信息错误



系统会自动判定识别登陆信息并刷新页面指导用户重新输入登陆信息。

2.1.4 登录信息匹配



系统判定登陆信息匹配跳转到后台首页（跳转链接为 <http://www.n6time.com/zcjy>）

2.2 后台界面菜单

2.2.1 菜单简介



左侧导航依次显示系统管理, 校购统计, 兼职管理, 校购管理, 用户提现, 意见反馈, 操作说明等七大板块, 其中第一板块: 系统管理可以配置系统的一些配置规则如用户初始积分, 投诉扣除积分, 打赏金额, 提成比例及设置后台每页数据的显示条数等, 用户管理模块可以管理用户的信誉积分及账户余额, 还可以查看用户上一次使用的校购学校等; 城市设置可以配置扩展系统的省市区范围等; 第二板块: 校购统计会统计当天或者当月以及自定义时间段的校购任务金额统计及平台分佣; 第三板块: 兼职管理可以管理和添加兼职类型以及发布兼职, 可以代发布者管理处理报名列表以及处理前端用户发起的投诉还可以添加一些企业认证信息来完善信息; 第四板块: 校购管理可以管理查看小程序用户发布的跑腿任务及处理用户发起的投诉, 还可以查看所有正在使用校购的学校; 第五板块: 用户提现可以处理用户发起的提现记录, 根据用户提供的支付宝号来给用户打款并且可以操作提现状态, 第六板块: 意见反馈可以处理查看用户发起的反馈意见; 最后面的操作说明可在浏览器中在线预览系统说明文档。

2.3 系统管理

2.3.1 系统设置

2.3.2 用户管理

2.3.3 城市设置

2.4 校购统计

2.5 兼职管理

2.5.1 兼职类型管理

2.5.2 兼职管理

2.5.3 兼职报名管理

2.5.4 企业管理

2.5.5 兼职投诉管理

2.6 校购管理

2.6.1 任务模板管理

2.6.2 跑腿任务管理

2.6.3 使用校购学校管理

2.6.4 校购投诉管理

2.7 用户提现

2.8 意见反馈

2.9 操作说明

后台操作说明会自动在浏览器中打开 pdf 版的操作文档，方便阅读。