

1.案例一:校园点餐案例

0.导入素材

1.在pages同级目录下创建imgJPG文件夹 2.在该文件夹下导入图片

1.页面与顶部和底部导航

1.编写项目所需页面

在app.json中编写pages

```
"pages": [  
  "pages/home/home",  
  "pages/category/category",  
  "pages/shopping/shopping",  
  "pages/collect/collect",  
  "pages/my/my"  
],
```

2.编写顶部导航

```
"window": {  
  "backgroundTextStyle": "light",  
  "navigationBarBackgroundColor": "#000",  
  "navigationBarTitleText": "贵州民族大学外卖",  
  "navigationBarTextStyle": "white"  
},
```

3.编写底部导航

```
"tabBar": {  
  "list": [  
    {  
      "pagePath": "pages/home/home",  
      "text": "首页",  
      "iconPath": "imgJPG/06.png",  
      "selectedIconPath": "imgJPG/01.png"  
    },  
    {  
      "pagePath": "pages/category/category",  
      "text": "分类",  
      "iconPath": "imgJPG/02.png",  
      "selectedIconPath": "imgJPG/07.png"  
    },  
    {  
      "pagePath": "pages/shopping/shopping",
```

```

        "text": "商城",
        "iconPath": "imgJPG/03.png",
        "selectedIconPath": "imgJPG/08.png"
    },
    {
        "pagePath": "pages/collect/collect",
        "text": "收藏",
        "iconPath": "imgJPG/04.png",
        "selectedIconPath": "imgJPG/09.png"
    },
    {
        "pagePath": "pages/my/my",
        "text": "我的",
        "iconPath": "imgJPG/05.png",
        "selectedIconPath": "imgJPG/10.png"
    }
}]

```

完整代码

```

{
  "pages": [
    "pages/home/home",
    "pages/category/category",
    "pages/shopping/shopping",
    "pages/collect/collect",
    "pages/my/my"
  ],
  "window": {
    "backgroundTextStyle": "light",
    "navigationBarBackgroundColor": "#000",
    "navigationBarTitleText": "贵州民族大学外卖",
    "navigationBarTextStyle": "white"
  },
  "tabBar": {
    "list": [
      {
        "pagePath": "pages/home/home",
        "text": "首页",
        "iconPath": "imgJPG/06.png",
        "selectedIconPath": "imgJPG/01.png"
      },
      {
        "pagePath": "pages/category/category",
        "text": "分类",
        "iconPath": "imgJPG/02.png",
        "selectedIconPath": "imgJPG/07.png"
      },
      {
        "pagePath": "pages/shopping/shopping",
        "text": "商城",
        "iconPath": "imgJPG/03.png",
        "selectedIconPath": "imgJPG/08.png"
      },
      {

```

```
    "pagePath": "pages/collect/collect",
    "text": "收藏",
    "iconPath": "imgJPG/04.png",
    "selectedIconPath": "imgJPG/09.png"
  },
  {
    "pagePath": "pages/my/my",
    "text": "我的",
    "iconPath": "imgJPG/05.png",
    "selectedIconPath": "imgJPG/10.png"
  }
],
"style": "v2",
"sitemapLocation": "sitemap.json"
}
```

效果



2. 页面布局

1. 搜索框

wxml

```

<!-- 搜索 -->
<view class="search">
  <input type="text" placeholder="搜菜谱、食材" />
</view>

```

WXSS

```

/* 搜索 */
input{
  text-align: center;
  height: 70rpx;
  line-height: 70rpx;
}
.search{
  border-radius: 20rpx;
  position: fixed;
  background-color:white;
  height: 70rpx;
  width: 90%;
  margin: 20rpx 5%;
}

```

2.轮播图

wxml

```

<!-- 轮播图 -->
<view class="search-img">
  <view style="height: 1px;"></view>
  <!-- 橱窗展示模块 -->
  <view class="img">
    <!--indicator-dots: 面板指示点 -->
    <swiper indicator-dots indicator-color="" autoplay>
      <swiper-item>
        <image src="../../imgJPG/s1.jpg" style="width: 100%; height: 350rpx;"></image>
      </swiper-item>
      <swiper-item>
        <image src="../../imgJPG/s2.jpg" style="width: 100%; height: 350rpx;"></image>
      </swiper-item>
      <swiper-item>
        <image src="../../imgJPG/s3.jpg" style="width: 100%; height: 350rpx;"></image>
      </swiper-item>
    </swiper>
  </view>
</view>

```

WXSS

```

/* 轮播图 */

```

```

.search-img{
  background-color: black;
}
.img{
  margin-top: 110rpx;
}

/* 搜索 */
input{
  text-align: center;
  height: 70rpx;
  line-height: 70rpx;
}
.search{
  border-radius: 20rpx;
  /*元素的位置相对于浏览器窗口是固定位置。即使窗口是滚动的它也不会移动*/
  position: fixed;
  background-color:white;
  height: 70rpx;
  width: 90%;
  margin: 20rpx 5%;
}

```

3.导航栏

wxml

```

<!-- 导航模块 -->
<view class="nav">
  <view class="nav-item">
    <view>
      <view>早中晚餐</view>
      <view>每日三餐推荐</view>
    </view>
    <view>
      <image src="../../imgJPG/title01.png" style="width:100rpx;height:100rpx" alt="" />
    </view>
  </view>
  <view class="nav-item">
    <view>
      <view>精选菜谱</view>
      <view>一周精选菜谱</view>
    </view>
    <view>
      <image src="../../imgJPG/title02.png" style="width:100rpx;height:100rpx" alt="" />
    </view>
  </view>
</view>

```

WXSS

```

/* 导航栏 */
.nav{
  display: flex;
  flex-direction: row;
  text-align: center;
}
.nav-item{
  display: flex;
  /*布局的排列方式 行*/
  flex-direction: row;
  width: 40%;
  margin-top: 40rpx;
  margin-left: 40rpx;
  font-size: 30rpx;
  background-color: #F3F3F3;
  text-align: center;
}

```

4.热门推荐

wxml

```

<!-- 热门推荐 -->
<view class="head">
  <view>热门推荐</view>
  <view class="hotimg">
    <image src="../../imgJPG/hot01.jpg"></image>
    <image src="../../imgJPG/hot02.jpg"></image>
    <image src="../../imgJPG/hot03.jpg"></image>
    <image src="../../imgJPG/hot04.jpg"></image>
  </view>
</view>

```

WXSS

```

/*热门推荐*/
.head{
  margin:30rpx;
  font-size: 40rpx;
  color:#333;
}
.hotimg{
  display: flex;
  flex-direction: row;
}
.hotimg image{
  height: 180rpx;
}

```

5.分割线

wxml

```
<!-- 分割线 -->
<view class="hr"></view>
```

WXSS

```
/* 分割线 */
.hr{
  height: 15rpx;
  background-color: #ccc;
  margin: 30rpx 0;
  /*透明度*/
  opacity: 0.3;
}
```

6.列表

wxml

```
<!-- 列表 -->
<view class="weekText">-本周佳作-</view>
```

WXSS

```
/*列表*/
.weekText {
  text-align: center;
  margin: 30rpx 0;
}
```

7.模板

1.在home目录下的home.js中编写initData函数

```
//自定义初始化数据的函数，用于获取并返回数据
initData:function(){
  let arr = [{
    img:"../imgJPG/week01.jpg",
    title:"锦上添花馒头",
    type:"林乡",
    collectNumber:"1089",
    browserNumber:"7.0万"
  },{
    img:"../imgJPG/week02.jpg",
    title:"快乐小鸡馒头",
    type:"林乡",
    collectNumber:"1206",
```



```
    browserNumber:"7.7万"
  },{
    img:"../imgJPG/week03.jpg",
    title:"快乐小鸡馒头",
    type:"林乡",
    collectNumber:"1206",
    browserNumber:"7.7万"
  },{
    img:"../imgJPG/week04.jpg",
    title:"快乐小鸡馒头",
    type:"林乡",
    collectNumber:"1206",
    browserNumber:"7.7万"
  },{
    img:"../imgJPG/week05.jpg",
    title:"快乐小鸡馒头",
    type:"林乡",
    collectNumber:"1206",
    browserNumber:"7.7万"
  },{
    img:"../imgJPG/week06.jpg",
    title:"快乐小鸡馒头",
    type:"林乡",
    collectNumber:"1206",
    browserNumber:"7.7万"
  },{
    img:"../imgJPG/week07.jpg",
    title:"快乐小鸡馒头",
    type:"林乡",
    collectNumber:"1206",
    browserNumber:"7.7万"
  },{
    img:"../imgJPG/week08.jpg",
    title:"快乐小鸡馒头",
    type:"林乡",
    collectNumber:"1206",
    browserNumber:"7.7万"
  },{
    img:"../imgJPG/week09.jpg",
    title:"快乐小鸡馒头",
    type:"林乡",
    collectNumber:"1206",
    browserNumber:"7.7万"
  },{
    img:"../imgJPG/week10.jpg",
    title:"快乐小鸡馒头",
    type:"林乡",
    collectNumber:"1206",
    browserNumber:"7.7万"
  },{
    img:"../imgJPG/week11.jpg",
    title:"快乐小鸡馒头",
    type:"林乡",
```

```

        collectNumber:"1206",
        browserNumber:"7.7万"
    },{
        img:"../imgJPG/week12.jpg",
        title:"快乐小鸡馒头",
        type:"林乡",
        collectNumber:"1206",
        browserNumber:"7.7万"
    },{
        img:"../imgJPG/week13.jpg",
        title:"快乐小鸡馒头",
        type:"林乡",
        collectNumber:"1206",
        browserNumber:"7.7万"
    }
    ];
    return arr;
},

```

2.在home目录下的home.js中的data中定义数组

```

data: {
    //创建数组
    array:[]
},

```

3.在home目录下的home.js中的中的onLoad函数中初始化数据

```

    //调用执行自定义函数
    let arr = this.initData();
    // 4、设置data中的数据变量
    this.setData({
        array:arr
    })

```

完整代码

```

// pages/home/home.js
Page({

    /**
     * 页面的初始数据
     */
    data: {
        //创建数组
        array: []
    },

    /**
     * 生命周期函数--监听页面加载
     */
    onLoad(options) {

```

```

//调用执行自定义函数
let arr = this.initData();
// 4、设置data中的数据变量
this.setData({
  array: arr
})
},

//自定义初始化数据的函数，用于获取并返回数据
initData: function () {
  let arr = [{
    img: "../../imgJPG/week01.jpg",
    title: "锦上添花馒头",
    type: "林乡",
    collectNumber: "1089",
    browserNumber: "7.0万"
  }, {
    img: "../../imgJPG/week02.jpg",
    title: "快乐小鸡馒头",
    type: "林乡",
    collectNumber: "1206",
    browserNumber: "7.7万"
  }, {
    img: "../../imgJPG/week03.jpg",
    title: "快乐小鸡馒头",
    type: "林乡",
    collectNumber: "1206",
    browserNumber: "7.7万"
  }, {
    img: "../../imgJPG/week04.jpg",
    title: "快乐小鸡馒头",
    type: "林乡",
    collectNumber: "1206",
    browserNumber: "7.7万"
  }, {
    img: "../../imgJPG/week05.jpg",
    title: "快乐小鸡馒头",
    type: "林乡",
    collectNumber: "1206",
    browserNumber: "7.7万"
  }, {
    img: "../../imgJPG/week06.jpg",
    title: "快乐小鸡馒头",
    type: "林乡",
    collectNumber: "1206",
    browserNumber: "7.7万"
  }, {
    img: "../../imgJPG/week07.jpg",
    title: "快乐小鸡馒头",
    type: "林乡",
    collectNumber: "1206",
    browserNumber: "7.7万"
  }, {

```

```

        img: "../../imgJPG/week08.jpg",
        title: "快乐小鸡馒头",
        type: "林乡",
        collectNumber: "1206",
        browserNumber: "7.7万"
    }, {
        img: "../../imgJPG/week09.jpg",
        title: "快乐小鸡馒头",
        type: "林乡",
        collectNumber: "1206",
        browserNumber: "7.7万"
    }, {
        img: "../../imgJPG/week10.jpg",
        title: "快乐小鸡馒头",
        type: "林乡",
        collectNumber: "1206",
        browserNumber: "7.7万"
    }, {
        img: "../../imgJPG/week11.jpg",
        title: "快乐小鸡馒头",
        type: "林乡",
        collectNumber: "1206",
        browserNumber: "7.7万"
    }, {
        img: "../../imgJPG/week12.jpg",
        title: "快乐小鸡馒头",
        type: "林乡",
        collectNumber: "1206",
        browserNumber: "7.7万"
    }, {
        img: "../../imgJPG/week13.jpg",
        title: "快乐小鸡馒头",
        type: "林乡",
        collectNumber: "1206",
        browserNumber: "7.7万"
    }
    ];
    return arr;
}
})

```

4.在pages目录下创建template文件夹,并在该文件夹下创建home-template.wxml文件

```

<template name='home'>
  <block wx:for="{{array}}" wx:key='item'>
    <view class="item">
      <image src="{{item.img}}"></image>
    </view>
    <view class="desc">
      <view class="title">
        {{item.title}}
      </view>
      <view class="count">
        <view class='type'>{{item.type}}</view>
      </view>
    </view>
  </block>
</template>

```

```

        <view class='collectNumber'>{{item.collectNumber}}收藏</view>
        <view class='browserNumber'>{{item.browserNumber}}浏览</view>
    </view>
</view>
<!-- 分隔线 -->
    <view class="hr2"></view>
</block>
</template>

```

5.在home目录下的home.wxml中引入模板 wxml

```

<!-- 导入模板 -->
<import src="../../template/home-template.wxml"></import>
<!-- 引用模板 is用来进行判断 data是要传入模板中的数据 -->
<template is='home' data='{{array}}'></template>

```

WXSS

```

/*模板*/
.item image{
    /* width: 85%; */
    /* width:280px;
    height: 169px; */
    width:672rpx;
    height: 408rpx;
    display: block;
    margin: 0 auto;
}
.desc{
    margin-left: 40rpx;
}
.title{
    font-weight: bold;
    font-size: 35rpx;
    margin: 30rpx 0;
}

.count{
    display: flex;
    flex-direction: row;
    color:#999;
    font-size: 24rpx;
}
.type,.collectNumber{
    margin-right: 20rpx;
}

```

3. springboot+mybatisplus整合小程序

利用springboot+mybatisplus实现小程序端的数据查询（本周佳作）。

1. 后端系统实现

1.1 实体类

```
@Data
public class Product {
    private Integer id; // 商品ID
    private String name; // 商品名称
    private String description; // 商品描述
    private Double price; // 商品价格
    private int inventory; // 商品库存
    private int categoryId; // 商品分类ID
    private String merchant; // 商家名称
    private Double rating; // 商品评分

    private String image; // 商品图片地址

    private Integer collectNumber; // 商品收藏数量

    private Double browserNumber; // 商品浏览数量
}
```

1.2 mapper

```
@Mapper
public interface ProductMapper extends BaseMapper<Product> {
    List<Product> findAll();
}
```

1.3 service

```
public interface ProductService extends IService<Product> {
}
```

实现类:

```
@Service
public class ProductServiceImpl extends ServiceImpl<ProductMapper, Product> implements
    ProductService {
}
```

1.4 controller

```
@RestController
@RequestMapping("/product")
public class ProductController {
    @Resource
    private ProductService productService;
}
```

```
@GetMapping("/list")
public ResultVO getAllProducts() {
    // 构造查询条件，按得分降序排序，取除前四个之外的所有商品
    QueryWrapper<Product> queryWrapper = new QueryWrapper<Product>()
        .orderByDesc("rating")
        .last("LIMIT 4, " + Integer.MAX_VALUE);
    List<Product> productList = productService.list(queryWrapper);

    ResultVO resultVO = new ResultVO();
    if (productList.size() > 0) {
        resultVO.setMessage("查询成功");
        resultVO.setCode(200);
        resultVO.setObject(productList);
    } else {
        resultVO.setMessage("查询失败");
        resultVO.setCode(500);
    }
    return resultVO;
}
```

测试结果：



```
{
  code: 200,
  message: "查询成功",
  object: [
    {
      id: 7,
      name: "蛋黄馒头",
      description: "金黄蛋黄，香气四溢",
      price: 3.5,
      inventory: 90,
      categoryId: 2,
      merchant: "张亮麻辣烫",
      rating: 4.7,
      image: "../../../imgJPG/week07.jpg",
      collectNumber: 1206,
      browserNumber: 7.6
    },
    {
      id: 13,
      name: "榴莲千层蛋糕",
      description: "以榴莲泥和千层蛋糕为主要原料，口感层次感强，味道浓郁。",
      price: 128,
      inventory: 20,
      categoryId: 3,
      merchant: "蛋糕之家",
      rating: 4.7,
      image: "../../../imgJPG/week12.jpg",
      collectNumber: 1206,
      browserNumber: 7.6
    }
  ]
}
```

2.小程序端实现

2.1 改写home.js

```
// pages/home/home.js
Page({
  /**
   * 页面的初始数据
   */
  data: {
    //创建数组
    array: []
  },

  /**
   * 生命周期函数--监听页面加载
```



```

    */
    onLoad(options) {
        //调用执行自定义函数
        this.initData();
    },

    //自定义初始化数据的函数，用于获取并返回数据
    initData:function(){
        wx.request({
            url: 'http://localhost:8080/product/list',
            success:resp=>{
                this.setData({
                    array:resp.data.object
                })
            }
        })
    }
})

```

2.2 改写template

```

<template name='home'>
    <block wx:for="{{array}}" wx:key='item'>
        <view class="item">
            <image src="{{item.image}}"></image>
        </view>
        <view class="desc">
            <view class="title">
                {{item.name}}
            </view>
            <view class="count">
                <view class='type'>{{item.merchant}}</view>
                <view class='collectNumber'>{{item.collectNumber}}收藏</view>
                <view class='browserNumber'>{{item.browserNumber}}浏览</view>
            </view>
        </view>
        <!-- 分隔线 -->
        <view class="hr2"></view>
    </block>
</template>

```

效果如图：

搜菜谱、食材

川菜

素菜

下饭菜

清淡

-本周佳作-



蛋黄馒头

张亮麻辣烫 1206收藏 7.6浏览

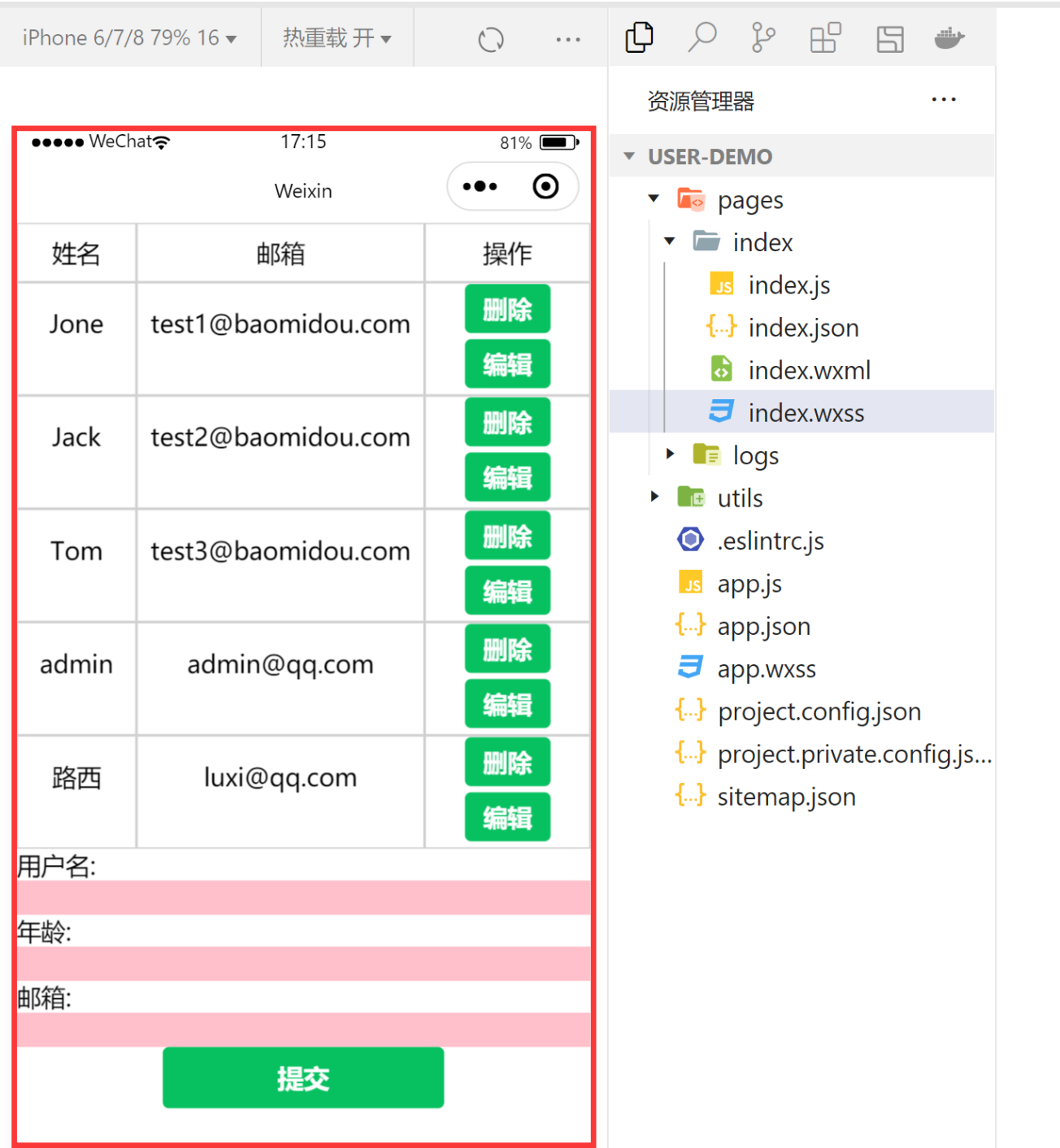




2.作业 利用springboot+mp整合小程序实现用户的增删改查

利用springboot+mybatisplus整合小程序实现用户的增删改查

实现效果：



3. 案例二 登录

1. 小程序端

1.1 个人中心-页面布局



登录 | 注册

查看个人主页

0
关注

0
粉丝

0
赞与收藏

0
好友动态



待支付



待支付



待支付



待支付



待支付

我的钱包

¥20

我的优惠券

暂无可用

领券中心

你的福利都在这里



官方微信客服



首页



分类



商城



收藏



我的

```
<!--pages/home/home.wxml-->
<view class="container">
  <view class="top-view">
    <!-- 登录注册 个人主页 -->
    <view class="user">
      <view class="row">
        <view class="name">
          <navigator url="/pages/auth/auth">登录</navigator>
          |
          <navigator url="/pages/auth/auth">注册</navigator>
        </view>
      </view>
      <view class="site">查看个人主页</view>
    </view>
    <!-- 关注粉丝 收藏 好友动态 -->
    <view class="numbers">
      <view class="row">
        <text>0</text>
        <text>关注</text>
      </view>
      <view class="row">
        <text>0</text>
        <text>粉丝</text>
      </view>
      <view class="row">
        <text>0</text>
        <text>赞与收藏</text>
      </view>
      <view class="row">
        <text>0</text>
        <text>好友动态</text>
      </view>
    </view>
  </view>
  <!-- 支付 -->
  <view class="middle-view">
    <view class="item">
      <image src="../../imgJPG/pay1.png"></image>
      <text>待支付</text>
    </view>
    <view class="item">
      <image src="../../imgJPG/pay2.png"></image>
```

```

        <text>待支付</text>
    </view>
    <view class="item">
        <image src="../../imgJPG/pay3.png"></image>
        <text>待支付</text>
    </view>
    <view class="item">
        <image src="../../imgJPG/pay4.png"></image>
        <text>待支付</text>
    </view>
    <view class="item">
        <image src="../../imgJPG/pay5.png"></image>
        <text>待支付</text>
    </view>
</view>
<!-- 我的钱包 -->
<view class="function-view">
    <view class="row">
        <view class="left">我的钱包</view>
        <view class="right">
            <text>¥20</text>
        </view>
    </view>

    <view class="row">
        <view class="left">我的优惠券</view>
        <view class="right">
            <text>暂无可用</text>
        </view>
    </view>

    <view class="row">
        <view class="left">领券中心</view>
        <view class="right">
            <text>你的福利都在这里</text>
        </view>
    </view>
</view>
<!-- 官方微信客服 -->
<view class="contact-view">
    <button open-type="contact">
        <image src="../../imgJPG/wechat.png"></image>
    </button>
</view>
</view>

```

1.2 样式

```

/* pages/home/home.wxss */
.top-view{
    background-color: #01ccb6;
    color: white;
}

```

```
padding: 40rpx;
}
.top-view .user{
display: flex;
flex-direction: row;
justify-content: space-between;
align-items: center;
}
.top-view .user .name{
display: flex;
flex-direction: row;
justify-content: space-around;
width: 200rpx;
}
.top-view .site{
background-color: rgba(0, 0, 0, 0.16);
padding: 20rpx;
border-top-left-radius: 32rpx;
border-bottom-left-radius: 32rpx;
}
.top-view .numbers{
display: flex;
flex-direction: row;
justify-content: space-between;
font-size: 28rpx;
padding: 40rpx;
padding-bottom: 0rpx;
}
.top-view .numbers .row{
display: flex;
flex-direction: column;
align-items: center;
}
.middle-view{
padding: 40rpx;
display: flex;
flex-direction: row;
justify-content: space-between;
font-size: 25rpx;
border-bottom: 18rpx solid #f5f5f5;
}
.middle-view .item{
display: flex;
flex-direction: column;
align-items: center;
}
.middle-view .item image{
width: 50rpx;
height: 50rpx;
margin-bottom: 20rpx;
}
.function-view{
padding: 40rpx;
```



```
    font-size: 28rpx;
  }
.function-view .row{
  padding: 30rpx 0;
  border-bottom: 1px solid #efefef;
  display: flex;
  flex-direction: row;
  justify-content: space-between;
}
.function-view .row .right{
  color: #8c8c8c;
}
.contact-view{
  padding: 40rpx;
  display: flex;
  flex-direction: row;
  justify-content: space-around;
}
.contact-view button::after{
  border: 0;
}
.contact-view image{
  width: 204rpx;
  height: 66rpx;
}
```

1.3 登录-页面布局



登录 注册

手机

请填写手机号码

验证码

请填写验证码

获取验证码

登录 | 注册

```
<!--pages/login/login.wxml-->
<view class="logo">
  <text>登录 注册</text>
</view>

<view class="form">
  <view class="row-group">
    <text>手机</text>
    <input placeholder="请填写手机号码" placeholder-class='txt' maxlength='11' />
  </view>
  <view class="row-group">
    <text>验证码</text>
    <input placeholder="请填写验证码" placeholder-class='txt' maxlength='4' />
    <view class="code">获取验证码</view>
  </view>
</view>
```

```
<view>
  <button class="submit">登录 | 注册</button>
</view>
</view>
```

1.4 样式

```
.logo{
  display: flex;
  flex-direction: column;
  align-items: center;
}

.logo image {
  margin-top: 140rpx;
  width: 216rpx;
  height: 100rpx;
}

.logo text {
  margin-top: 26rpx;
  margin-bottom: 50rpx;
  font-size: 24rpx;
  line-height: 24rpx;
  font-weight: 400;
  color: #8c8c8c;
  text-align: center;
}

.form{
  padding: 40rpx;
}

.form .row-group{
  padding: 20rpx 0;
  border-bottom: 1rpx solid #ddd;
  position: relative;
}

.form .row-group text{
  font-size: 28rpx;
  padding: 10rpx 0;
}

.form .row-group input{
  padding: 10rpx 0;
}

.form .row-group .txt{
  color: #ccc;
}
```

```
.form .row-group .code{
  position: absolute;
  right: 0;
  bottom: 26rpx;
  z-index: 2;

  width: 206rpx;
  height: 60rpx;
  border: 2rpx solid #00c8b6;
  border-radius: 12rpx;
  font-size: 26rpx;
  font-weight: 400;
  color: #00c8b6;
  display: flex;
  align-items: center;
  justify-content: center;
}

.form .submit{
  margin-top: 80rpx;
  color: #fff;
  border: 2rpx solid #00c8b6;
  background-color: #00c8b6;
  font-size: 32rpx;
  font-weight: bold;
}
```