

2025年夏季《移动软件开发》实验报告

姓名：邓林 学号：23020007014

姓名和学号	邓林, 23020007014
本实验属于哪门课程?	中国海洋大学25夏《移动软件开发》
实验名称?	实验2: 天气查询小程序
博客地址?	写文章-CSDN创作中心
Github仓库地址?	https://github.com/xixiyhaha/2025Mobile-software-development.git

(备注：将实验报告发布在博客、代码公开至 github 是 **加分项**，不是必须做的)

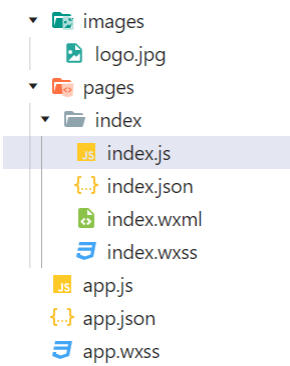
一、实验目标

- 1、掌握服务器域名 配置和临时服务器部署；
- 2、掌握 wx.request 接口的用法。

二、实验步骤

列出实验的关键步骤、代码解析、截图。

- 1、新建小程序，命名为weather；
- 2、清空文件并新建框架；



- 3、设置容器样式；
- 4、基本设计界面元素、排版，结果如图；



5、为地区设置picker地区选择器，并定义changeRegion函数，这里包括更新显示的地域名称、天气详细信息，增加getLocationID；

```

22 //更新地区信息
23 changeRegion: function(e){
24   //console.log(e.detail.value)
25   this.setData({region: e.detail.value});
26   this.getLocationID().then(result => {
27     this.getWeather();
28   })
29 },

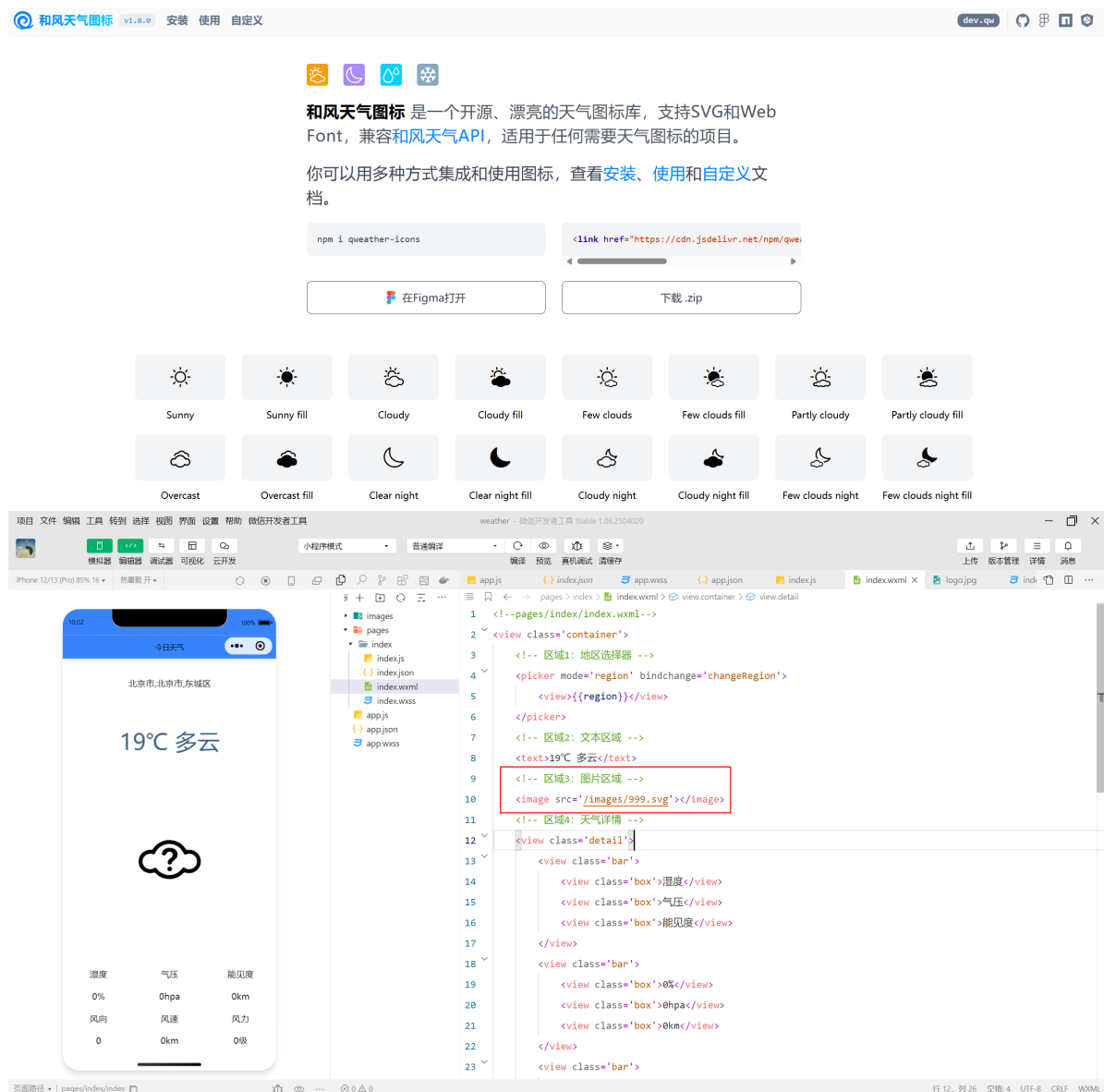
3   <!-- 区域1: 地区选择器 -->
4   <picker mode='region' bindchange='changeRegion'>
5     <view>{{region}}</view>
6   </picker>

```

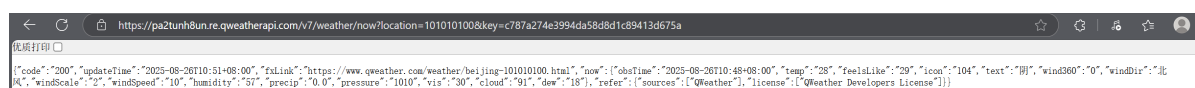
6、注册和风平台账号，新建天气项目，获取API接口来获取天气数据；



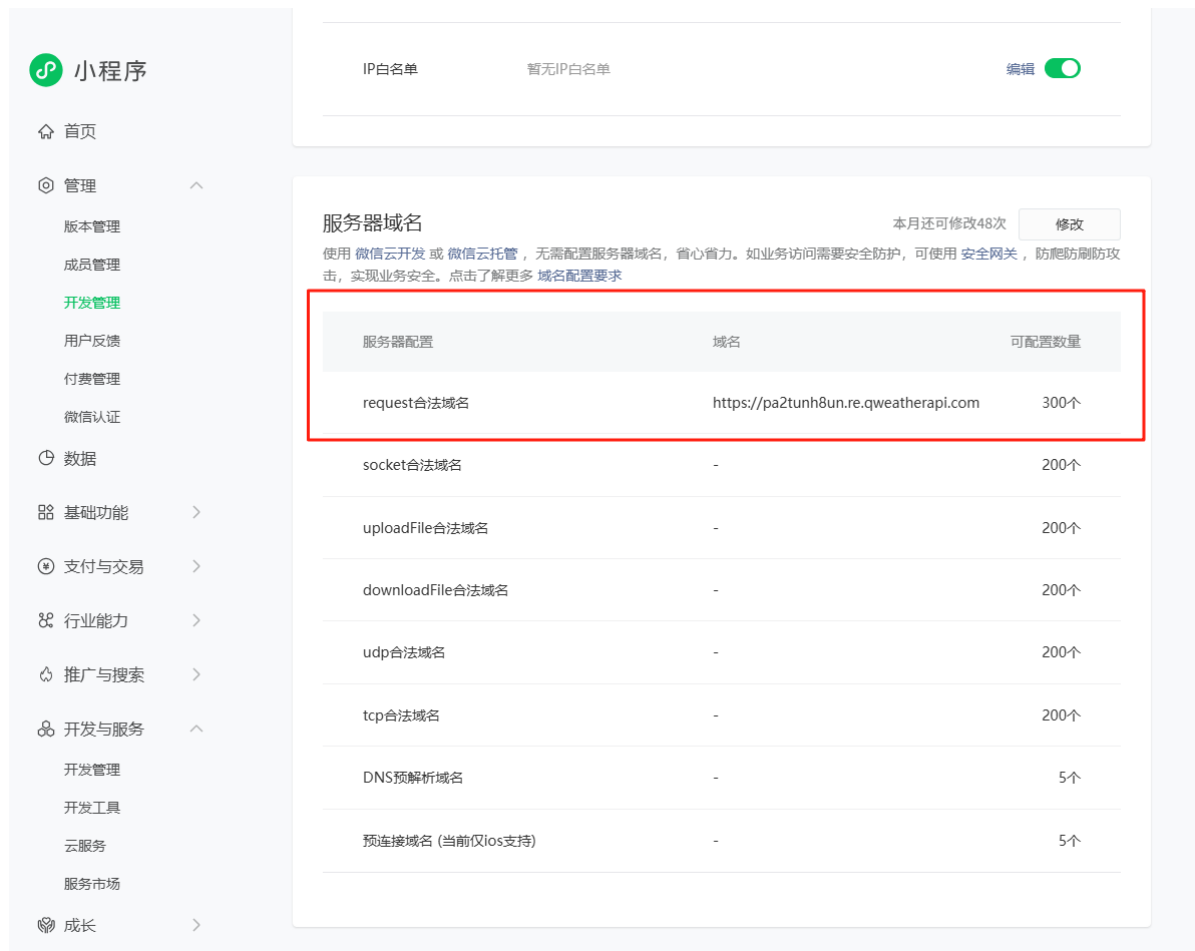
7、下载和风天气提供的天气图标并存入文件images中，将默认天气图标设为未知图标；



8、记录下自己的api host和api key,将他们替换到以下链接中: https://your_api_host/v7/weather/no-w?location=101010100&key=your_api_key; 在浏览器中访问链接应该可以直接获取到地区天气信息; 这个链接可以获取地区LocationID: https://your_api_host/geo/v2/city/lookup?location=beij&key=your_api_key;



9、在微信公众平台修改服务器域名为https://your_api_host;



10、定义天气信息获取函数getWeather、getLocationID;

getWeather: 用 getLocationID 获取到的“地理位置 ID”，调用天气 API 的“实时天气接口”，获取该地区的实时天气数据，并把数据存入页面，

getLocationID: 通过用户选择的地区名称，调用天气 API 的“城市搜索接口”，获取该地区对应的唯一地理位置 ID。



```

46  getLocationID: function(){
47      var that = this;
48      return new Promise(resolve => {
49          wx.request({
50              url: 'https://pa2tunh8un.re.qweatherapi.com/geo/v2/city/lookup?',
51              data:{
52                  location:that.data.region[1],
53                  adm:that.data.region[0],
54                  key:'c787a274e3994da58d8d1c89413d675a'
55              },
56              success: res => {
57                  console.log(res.data);
58                  that.setData({locationID:res.data.location[0].id})
59                  return resolve();
60              },
61          })
62      });
63  },

```

11、查看now数组中各元素，并修改对应的变量；

输出 终端 代码质量

Network Performance Memory AppData Storage Security >>

Tree Editing...

object {4}

locationID : 101010100

now {15}

cloud : 91

dew : 18

feelsLike : 29

humidity : 51

icon : 101

obsTime : 2025-08-26T11:32+08:00

precip : 0.0

pressure : 1009

temp : 29

text : 多云

vis : 30

wind360 : 45

windDir : 东北风

windScale : 3

windSpeed : 12

region {3}

0 : 北京市

1 : 北京市

2 : 东城区

__webviewId__ : 38

2 Page({

3 /* 页面的初始数据 */

4 data: {

5 region:['北京市','北京市','东城区'],

6 locationID:101010100,

7 now:{

8 temp:0,

9 text:'未知',

10 icon:'999',

11 humidity:0,

12 pressure:0,

13 vis:0,

14 windDir:0,

15 windSpeed:0,

16 windScale:0

17 }

18 },

7 <!--区域2: 单行天气信息-->

8 <text>{{now.temp}}°C {{now.text}}</text>

9 <!--区域3: 天气图标-->

10 <image src = '/images/{{now.icon}}.svg' mode = 'widthFix'></image>

```

11 <!--区域4: 多行天气信息-->
12 <view class = 'detail'>
13   <view class = 'bar'>
14     <view class = 'box'>湿度</view>
15     <view class = 'box'>气压</view>
16     <view class = 'box'>能见度</view>
17   </view>
18   <view class = 'bar'>
19     <view class = 'box'>{{now.humidity}} %</view>
20     <view class = 'box'>{{now.pressure}} hPa</view>
21     <view class = 'box'>{{now.vis}} km</view>
22   </view>
23   <view class = 'bar'>
24     <view class = 'box'>风向</view>
25     <view class = 'box'>风速</view>
26     <view class = 'box'>风力</view>
27   </view>
28   <view class = 'bar'>
29     <view class = 'box'>{{now.windDir}} </view>
30     <view class = 'box'>{{now.windSpeed}} km/h</view>
31     <view class = 'box'>{{now.windScale}} 级</view>
32   </view>
33 </view>

```

12、到这里最基本的选择地区更新地区天气的基本小程序就完成了。



13、这里我们也可以对小程序进行一些升级：当我们编译代码后会发现界面现实的是我们预设的“未知”天气，只有在我们重新选择地区后才会更新，我们可以添加onLoad函数使代码在编译使就进行数据更新；



14、我们也可以对页面进行一些美化。

```
40  /* 温度和天气状态文本样式 */
41  .temp-weather-container {
42    display: flex;
43    align-items: center;
44    margin-bottom: 0rpx;
45  }
46
47  .temp-text {
48    font-size: 110rpx;
49    color: #3C5F81;
50    font-weight: 400;
51    line-height: 2;
52  }
53
54  .weather-text {
55    font-size: 60rpx;
56    color: #3C5F81;
57    font-weight: 500;
58    margin-left: 50rpx;
59  }
```

```
13  /* 地区选择器样式 */
14  .location-container {
15    width: 100%;
16    margin-bottom: 50rpx;
17  }
18
19  .region-picker {
20    display: flex;
21    align-items: center;
22    justify-content: center;
23    padding: 40rpx 20rpx;
24    background-color: white;
25    border-radius: 10rpx;
26    box-shadow: 0 4rpx 12rpx rgba(0, 0, 0, 0.05);
27  }
28
29  .region-text {
30    font-size: 42rpx;
31    color: #333;
32    font-weight: 520;
33  }
```



三、程序运行结果

列出程序的最终运行结果及截图。



四、问题总结与体会

描述实验过程中所遇到的问题，以及是如何解决的。有哪些收获和体会，对于课程的安排有哪些建议。

问题：在和风天气获取API访问天气数据时，由于老版本的访问请求url和现在不一样，导致访问失败。

请求示例

解决：新版的访问链接

```
curl -X GET --compressed \  
-H 'Authorization: Bearer your_token' \  
'https://your_api_host/v7/weather/now?location=101010100'
```

也并不完

整，而是应该在后面加上key='your_api_key'，故完整的请求示例为：https://your_api_host/v7/weather/now?location=101010100&key=your_api_key；同时要将老版中的地名换为LocationID，故还需要访问：https://your_api_host/geo/v2/city/lookup?location=beij&key=your_api_key；

请求示例

```
curl -X GET --compressed \  
-H 'Authorization: Bearer your_token' \  
'https://your_api_host/geo/v2/city/lookup?location=beij'
```

收获和体会：通过这次小程序的制作，我掌握了通过wx.request接口访问外界链接的方法，也掌握了修改服务器域名的方法。通过wx.request接口，极大地丰富了小程序的玩法和功能，相当于给我们的小程序连上了网，能够进行更多使用地操作。同时，对天气界面的设计，我对flex排版的使用更加熟练。