



西安电子科技大学
XIDIAN UNIVERSITY

XXX 学院

专业实验实践能力（B 级）达标测试报告

雾霾探测系统设计

姓名：XXX

学号：XXX

班级：XXX

2022 年 05 月 07 日

目录

1	任务描述	1
2	方案设计与论证	1
2.1	系统方案	1
2.2	系统实现	1
3	结果与展示	2
4	总结	4
	Appendices	5
附录 A	项目主要代码	5
A.1	app.py	5
A.2	第一部分	5
A.3	第二部分	8

雾霾探测系统设计

1 任务描述

雾霾的频繁出现已严重的影响到人们的出行，对人们的健康造成了重大影响。因此，能在出行前查看雾霾的指数，并采取相应的措施来把雾霾的影响降到最小就显得尤为重要。本系统在分析多种因子的影响下，设计一款手机端雾霾 app 探测系统。要求如下：

1. 定位功能：将定位城市保存在服务器端，并同时显示在客户端。
2. 界面设计：包含显示天气和空气质量指数的动态显示。
3. 天气详情和空气质量指数：定位后的城市在服务器端获取后，传给天气详情界面，通过所传城市用百度天气 api 获取对应的天气详情和空气质量指数，并保存在服务器端。
4. 完成报告。

2 方案设计与论证

2.1 系统方案

我计划用 Python Flask 框架设计网页版雾霾探测系统，static 文件夹用于存放包括 css 样式、背景图像和 javascript 脚本在内的各种静态文件，templates 文件夹用于存放网页主体 html 文件，用 flask 框架的 render_template 函数渲染 html 文件，运行 app.py 文件可在本地默认的 127.0.0.1:5000 查看渲染结果，可以手动更改 host 和 port。

整个 html 文件主体是两个部分，第一个部分包括时间、地理、天气和空气质量信息的获取与展示，考虑使用 jQuery 的幻灯片插件 OWL Carousel 对这一部分进行模板渲染；第二部分是未来一周的气温和天气预测，考虑用 echarts 绘制动态折线图进行展示，这一部分官网有现成的天气折线模板代码 <https://echarts.apache.org/examples/zh/editor.html?c=line-marker>，只需稍作修改加入即可。

2.2 系统实现

首先考虑如何获取位置信息，由于百度、高德地图等浏览器端的 js 定位接口需要支持 https 不是那么方便，故我选择使用 ip 定位，ip 定位精确度高，而且大部分天气数据 api 也支持 ip 来查询，本项目中我使用的 ip 查询接口为 <http://ip-api.com/json/>，获取 ip 得到 json 的文件如图 1 所示，我用到了其中的城市和经纬度信息用于后续信息查询。

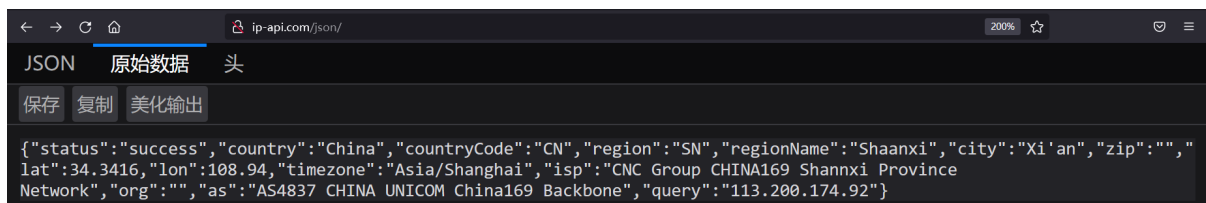


图 1: 获取 ip 得到的 json 文件

时间获取较为简单，可以直接用 js 内置函数获取，这里略去。接下来考虑获取天气和空气质量信息，可以用和风天气开发平台方便的获取到，天气信息获取官方文档见 <https://dev.qweather.com/docs/api/weather/weather-now/>，注册账号获取 KEY 后，利用经纬度信息查询到天气信息和空气信息 json 文件分别如图 2 和图 3 所示，

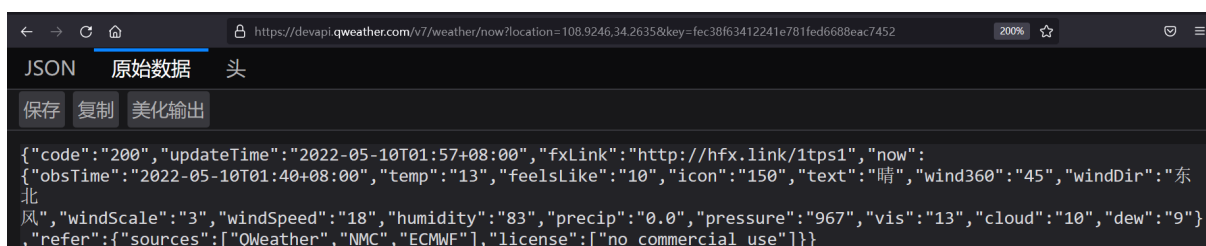


图 2: 天气信息 json 文件

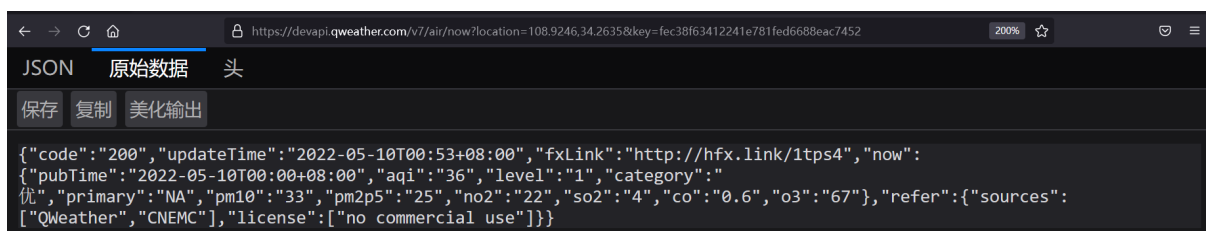


图 3: 空气信息 json 文件

关于第二部分一周天气预测信息获取考虑使用天气 API，官方文档见 <https://www.tianqiapi.com/index/doc?version=week>，获取账号和密钥后可通过城市 id、名称或 ip 查询天气信息，优先级递减，默认返回当前 ip 城市天气，我的项目中只使用 json 文件中的日期、气温和天气信息绘制折线图。这一部分直接参考 echarts 官网折线代码模板稍作修改填充信息即可。

包括 css 样式和 js 脚本在内的其他文件，只需调好颜色、字体和布局即可，不属于报告的主要部分，这里略去。

3 结果与展示

第一部分界面如图 4 所示，其中 4 个栏目信息如图 5 所示。

第二部分界面如图 6 所示。



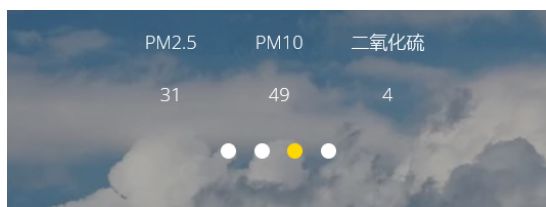
图 4: 第一部分界面展示



(a) 第一部分栏目 1 信息



(b) 第一部分栏目 2 信息



(c) 第一部分栏目 3 信息



(d) 第一部分栏目 4 信息

图 5: 第一部分栏目信息展示

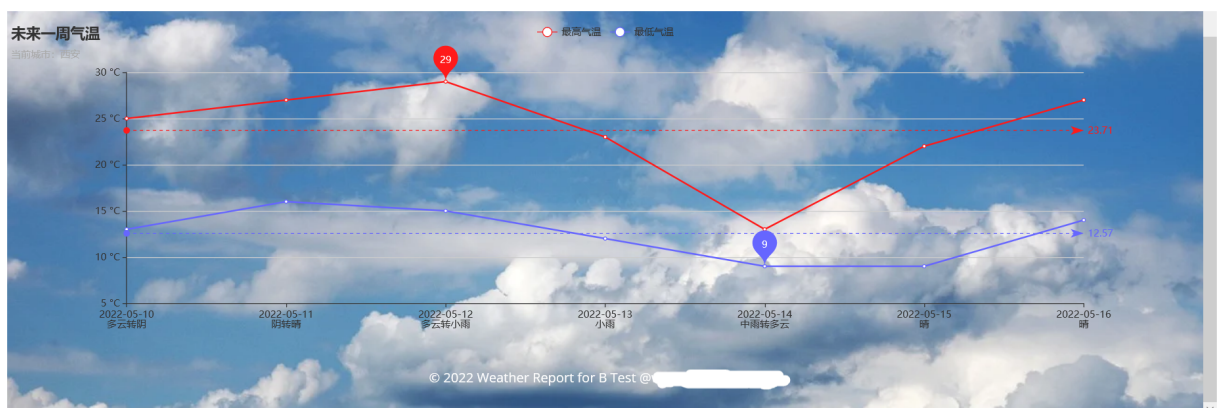


图 6: 第二部分界面展示

4 总结

通过本次项目实践，可以独立完成基础的 web 开发，整个项目由本人一人完成，我认为本次项目完成的较为完整美观且功能齐全，但不足是时间相关代码没有优化，需要手动刷新页面更新时间。项目主要代码放在附录，css 样式和 js 脚本等不是主要部分略去。

Appendices

附录 A 项目主要代码

A.1 app.py

```
1 from flask import Flask
2 from flask import render_template
3 from gevent.pywsgi import WSGIServer
4
5 app = Flask(__name__)
6
7 @app.route('/')
8 def info_handler():
9     return render_template('index.html')
10
11 if __name__ == '__main__':
12     # from waitress import serve
13     # serve(app, host="127.0.0.1", port=5000)
14     app.run()
15     # http_server = WSGIServer('', 4000), app)
16     # http_server.serve_forever()
```

A.2 第一部分

```
1 <div class="main-agileits">
2     <h1><b>雾霾探测系统(网页版)</b></h1>
3     <div class="main-wthree-row">
4         <div class="agileinfo-text">
5             <div class="date">
6                 <script type="text/javascript">
7                     var mydate=new Date()
8                     var year=mydate.getYear()
9                     if(year<1000)
10                         year+=1900
11                     var day=mydate.getDay()
12                     var month=mydate.getMonth()
13                     var daym=mydate.getDate()
14                     if(daym<10)
15                         daym="0"+daym
16                     var dayarray=new Array("Sunday","Monday","Tuesday",
17                                             "Wednesday","Thursday","Friday","Saturday")
18                     var montharray=new Array("January","February","
```

```
        March","April","May","June","July","August","
        September","October","November","December")
18     var h=mydate.getHours()
19     var m=mydate.getMinutes()
20     var s=mydate.getSeconds()
21     if (m<10) m="0"+m
22     if (s<10) s="0"+s
23     document.write(daym+" "+montharray[month]+" "+
        year+" <br> "+dayarray[day] + " <br> " +h+": "+m)
24     </script>
25 </div>
26 <h4 class="location"></h4>
27 <h2 class="temp">&nbsp;</h2>
28 <h6 class="weather"></h6>
29 <h6 class="wind"></h6>
30 </div>
31 <div class="w3layouts-slider">
32     <div id="owl-demo" class="owl-carousel owl-theme">
33         <div class="item agile-item">
34             <h6>相对湿度</h6>
35             <br>
36             <h6 class="humidity"></h6>
37         </div>
38         <div class="item agile-item">
39             <h6>能见度</h6>
40             <br>
41             <h6 class="vis"></h6>
42         </div>
43         <div class="item agile-item">
44             <h6>降水量</h6>
45             <br>
46             <h6 class="precip"></h6>
47         </div>
48         <div class="item agile-item">
49             <h6>大气压</h6>
50             <br>
51             <h6 class="pressure"></h6>
52         </div>
53         <div class="item agile-item">
54             <h6>空气质量</h6>
55             <br>
56             <h6 class="quality"></h6>
57         </div>
58         <div class="item agile-item">
59             <h6>质量指数</h6>
60             <br>
61             <h6 class="aq"></h6>
```



```
62         </div>
63         <div class="item agile-item">
64             <h6>PM2.5</h6>
65             <br>
66             <h6 class="pm25"></h6>
67         </div>
68         <div class="item agile-item">
69             <h6>PM10</h6>
70             <br>
71             <h6 class="pm10"></h6>
72         </div>
73         <div class="item agile-item">
74             <h6>二氧化硫</h6>
75             <br>
76             <h6 class="so2"></h6>
77         </div>
78         <div class="item agile-item">
79             <h6>二氧化氮</h6>
80             <br>
81             <h6 class="no2"></h6>
82         </div>
83         <div class="item agile-item">
84             <h6>一氧化碳</h6>
85             <br>
86             <h6 class="co"></h6>
87         </div>
88         <div class="item agile-item">
89             <h6>臭氧</h6>
90             <br>
91             <h6 class="o3"></h6>
92         </div>
93     </div>
94     <script>
95         var icons = new Skycons({"color": "#fff"}),
96             list = [
97                 "clear-night", "clear-day", "partly-cloudy-day",
98                 "partly-cloudy-night", "cloudy", "rain", "sleet",
99                 "snow", "wind",
100                 "fog"
101             ],
102             i;
103         for(i = list.length; i--; )
104             icons.set(list[i], list[i]);
105         icons.play();
106     </script>
107 </div>
```

A.3 第二部分

```
1      <div id="weather" class="col-lg-12 col-md-12" style="height:400px"></div>
2          <div class="get_weather_info">
3              <script>
4                  $.get("http://ip-api.com/json/", function(basic){
5                      // 旧版本 https://free-api.heweather.net/s6/weather
                      // now?location=" + basic['ip'] + "&key=
                      // b27e0f8b9678447c91a8edbcdded8b055
6                      $.get("https://devapi.qweather.com/v7/weather/now?
                      // location=" + basic['lon'] + "," + basic['lat'] +
                      // "&key=fec38f63412241e781fed6688eac7452",
                      function(param2){
7                          $(".location").text(basic['city']);
8                          $(".temp").text(param2['now']['temp'] + '°');
9                          $(".weather").text(param2['now']['text']);
10                         $(".wind").text(param2['now']['windDir'] + '('
                            + param2['now']['windSpeed'] + '级');
11                         $(".humidity").text(param2['now']['humidity'] +
                            '%');
12                         $(".vis").text(param2['now']['vis'] + 'km');
13                         $(".precip").text(param2['now']['precip'] + 'mm');
14                         $(".pressure").text(param2['now']['pressure'] +
                            'hPa');
15                     }, "json")
16                 }, "json");
17                 $.get("http://ip-api.com/json/", function(air){
18                     $.get("https://devapi.qweather.com/v7/air/now?
                        // location=" + air['lon'] + "," + air['lat'] + "&
                        // key=fec38f63412241e781fed6688eac7452", function(
                        // param2){
19                         $(".quality").text(param2['now']['category']);
20                         $(".aq").text(param2['now']['aqi']);
21                         $(".pm25").text(param2['now']['pm2p5']);
22                         $(".pm10").text(param2['now']['pm10']);
23                         $(".so2").text(param2['now']['so2']);
24                         $(".no2").text(param2['now']['no2']);
25                         $(".co").text(param2['now']['co']);
26                         $(".o3").text(param2['now']['o3']);
27                     }, "json")
28                 }, "json");
29             </script>
```

```
30         <script>
31             // 初始化天气折线图
32     var weather = echarts.init($('#weather').get(0));
33     option = {
34         title: {
35             text: '未来一周气温',
36             subtext: ''
37         },
38         tooltip: {
39             trigger: 'axis'
40         },
41         legend: {
42             data: ['最高气温', '最低气温', '湿度']
43         },
44         color: ['#ff1a1a', '#6666ff', '#61a0a8', '#d48265', '#91c7ae',
45             '#749f83', '#ca8622', '#bda29a', '#6e7074', '#546570', '#c4ccd3'],
46         xAxis: {
47             type: 'category',
48             boundaryGap: false,
49             data: []
50         },
51         yAxis: {
52             scale: true, // 纵坐标起始点根据最低值变化
53             type: 'value',
54             axisLabel: {
55                 formatter: '{value} °C'
56             }
57         },
58         series: [{
59             name: '最高气温',
60             type: 'line',
61             data: [],
62             markPoint: {
63                 data: [{
64                     type: 'max',
65                     name: '最大值'
66                 }]
67             },
68             markLine: {
69                 data: [{
70                     type: 'average',
71                     name: '平均值'
72                 }]
73             },
74         },
```

```
75         {
76             name: '最低气温',
77             type: 'line',
78             data: [],
79             markPoint: {
80                 data: [{
81                     type: 'min',
82                     name: '最小值'
83                 }]
84             },
85             markLine: {
86                 data: [{
87                     type: 'average',
88                     name: '平均值'
89                 }],
90             ]
91         },
92     }
93 ]
94 };
95 weather.setOption(option);
96 // 获取天气信息
97 $.get("https://www.tianqiapi.com/free/week?appid=81386558&appsecret
    =52jN6BzG&cityid=CN101110101",
98     function (res) {
99         //显示当前城市
100         {option.title.subtext = '当前城市: ' + res.city}
101         //给横坐标复赋值
102         option.xAxis.data = [res.data[0].date+'\n'+res.data[0].wea,
103             res.data[1].date+'\n'+res.data[1].wea,
104             res.data[2].date+'\n'+res.data[2].wea,
105             res.data[3].date+'\n'+res.data[3].wea,
106             res.data[4].date+'\n'+res.data[4].wea,
107             res.data[5].date+'\n'+res.data[5].wea,
108             res.data[6].date+'\n'+res.data[6].wea
109         ]
110         //由于温度返回的是xx℃ 而我们只需要数字 所以用parseInt截取数字
111         option.series[0].data = [parseInt(res.data[0].tem_day),
112             parseInt(res.data[1].tem_day),
113             parseInt(res.data[2].tem_day),
114             parseInt(res.data[3].tem_day),
115             parseInt(res.data[4].tem_day),
116             parseInt(res.data[5].tem_day),
117             parseInt(res.data[6].tem_day)]
118         option.series[1].data = [parseInt(res.data[0].tem_night),
119             parseInt(res.data[1].tem_night),
120             parseInt(res.data[2].tem_night),
```

```
121         parseInt(res.data[3].tem_night),
122         parseInt(res.data[4].tem_night),
123         parseInt(res.data[5].tem_night),
124         parseInt(res.data[6].tem_night)
125     ]
126     weather.setOption(option); // 使用刚指定的配置项和数据显示图
        表。
127 },
128 );
129     </script>
130     </div></div>
131     <div class="copy-rights wthree">
132         <p>© 2022 Weather Report for B Test    @name stu-id</p>
133     </div>
```