

# 软件工程课程设计

## 项目 目 设 计 文 档

小组成员：孙浩然、吕吉向、张藤原

谢晓辉、许芷毓、吴承旭

# 目 录

<b>第一章 应用概述</b>	<b>1</b>
1.1 编写目的	1
1.2 背景	1
1.3 定义	1
1.4 参考资料	2
<b>第二章 系统设计的原则</b>	<b>3</b>
2.1 实用性原则	3
2.2 简结性原则	3
2.3 数据实时性和真实性原则	3
2.4 系统模块化设计原则	3
2.5 安全性原则	3
<b>第三章 系统设计</b>	<b>4</b>
3.1 系统体系结构设计	4
3.2 设计优点	4
3.2.1 交互性强	4
3.2.2 安全性高	4
3.2.3 网络通信量少	4
3.2.4 数据处理能力强	4
3.3 设计模式	4
3.3.1 表现层	5
3.3.2 业务逻辑层	5
3.3.3 数据层	5
3.3.4 总体架构	5
3.4 系统功能架构设计	5
3.5 本章小结	5
<b>第四章 功能模块设计</b>	<b>6</b>
4.1 疫情地图查看功能设计	6
4.1.1 中国疫情地图模块	6
4.1.2 各地具体的疫情数据模块	7
4.1.3 数据爬取模块	10
4.1.4 中国疫情地图模块	11
4.1.5 海外疫情模块	12
4.2 疫情相关防控措施查看功能设计	16
4.3 热点新闻展示功能	16

<b>第五章 数据库设计</b> .....	17
5.1 数据安全性设计.....	17
5.2 数据安全性设计.....	18
<b>第六章 系统的结构化设计</b> .....	18
6.1 系统层次图.....	18
6.2 系统结构图.....	18
6.3 本章小结.....	19
<b>第七章 系统测试</b> .....	20
7.1 测试原则与方法.....	20
7.2 测试过程.....	20
7.2.1 模块测试.....	20
7.2.2 子系统测试.....	20
7.2.3 系统测试.....	21
7.2.4 验收测试.....	21
7.3 本章小结.....	21

# 第一章 应用概述

## 1.1 编写目的

2020 年的春节是一个特殊的春节，没有人潮拥挤；没有走街串巷；没有了以往的喧闹。疫情当前，大家选择了居家办公、学习，云拜年等保护自己也保护他人的方法。在特殊的时期我们不能给国家添乱，但疫情的发展也牵动着所有的人，不仅仅是某一个人，也是一个地区、一个国家甚至整个世界。

每一个人每天每刻都关注着疫情的发展，因为平台的多且杂可能会出现一些信息的查询不准确及效率偏低等情况。

因此，为方便公众及时了解疫情分布情况以及对疫情的发展信息有更加便捷、直观的获取，节省相关讯息的搜索时间，提高查阅效率等，我们开发该网站——疫情地图网站，当然也可为相关部门部署防控工作提供参考。

## 1.2 背景

1. 软件系统名称：疫情地图
2. 用户：公众
3. 联系

互联网覆盖及使用范围已触及生产、生活、工作、学习、娱乐的各个环节和领域，通过互联网进行信息的查阅已经逐渐成为一种习惯。与此同时，网络的讯息获取的要求不再只是全面，而是快捷、方便，技术的提升和方法的升级是网络讯息传播的竞争核心，如何在激烈的竞争中扩大使用率成为系统开发者努力追求的目标。

在疫情期间，公众更好地利用了“云系列”，学习、工作、娱乐等将互联网更好地利用起来，通过各种网络平台我们可成功获取疫情相关信息及学习疫情防控措施。

## 1.3 定义

(1) 总体设计——又叫概要设计，主要是确定系统的具体实施方案和确定软件结构。

(2) 详细设计，主要确定应该怎样具体地实现所要求的系统。详细设计阶段的任务不是具体地编写程序，而是要设计出程序的“蓝图”。其设计的结果基本上

决定了最终的程序代码的质量。

## **1.4 参考资料**

杨选辉 《信息系统分析与设计》 清华大学出版社 2007

韩润春，佟志臣，王汉新 《管理信息系统》 河北人民出版社 2003

温贤发 《Visual Basic 6 数据库程序设计高手》 北京科学出版社 2001

黎连业 《管理信息系统设计与实施》 北京清华大学出版社 1999

## 第二章 系统设计的原则

疫情地图网站系统是服务用户，满足用户需求而设计、解决实际问题而设计的，在新冠肺炎疫情全球大流行之际，用户需实时的了解到各个地区疫情的信息，学习对于新冠疫情的防护措施。因此，在系统开发过程中，不仅要以先进理论来指导、主流技术来支撑，而且要做到以用户为中心，满足用户的需求，开发出页面美观大方、功能较为强大完善且使用方便的系统。据此，通过综合的考虑本次系统设计的目标及需求，在设计室，因遵循以下设计原则。

### 2.1 实用性原则

本系统从实际的用户需求出发，满足用户了解疫情相关信息及学习相关防护措施的需求。

### 2.2 简结性原则

系统采用直观的方式来展现疫情的信息，主要通过地图及数据来展示疫情的信息，此外还有疫情数据的变化趋势图、每日新增病例等图形以便更直观的让用户了解当下疫情的变化趋势。

### 2.3 数据实时性和真实性原则

系统中的数据设计的来源是通过权威网站发布的真实的疫情的数据。在系统实现过程中，采用爬虫的技术，将真实有效的数据爬取下来。本系统也将每日更新数据，保证用户浏览到的数据信息是当下实时的疫情数据信息；

### 2.4 系统模块化设计原则

模块化是以功能块为单位进行程序的设计，可降低程序复杂度，使程序设计、调试及维护简单化。因此，采用模块化设计原则，增强了系统的灵活性和可扩展性。

### 2.5 安全性原则

目前只考虑了数据库的安全性方面，通过事务等实现数据的一致性；通过定义完整性规则实现数据的完整性。之后也会考虑数据信息的保密性、身份鉴别、访问控制、权限管理等。

## 第三章 系统设计

### 3.1 系统体系结构设计

本系统采用浏览器-服务器（B/S）体系结构的设计，B/S 体系结构图如图 3.1 所示。

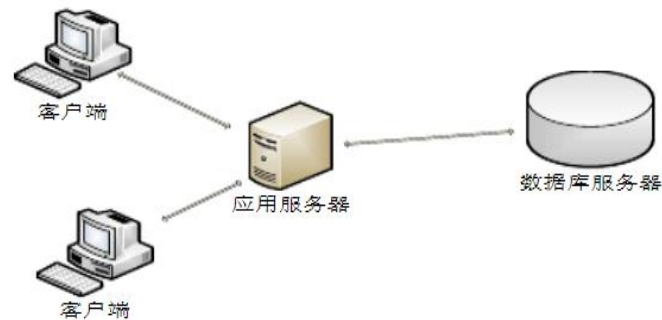


图 3.1 B/S 体系结构图

B/S 架构是浏览器/服务器交互模式，是 Browser/Server 的简称。这种架构的软件不需要在用户的电脑上安装任何客户端程序，只需要在用户的电脑上安装浏览器即可。用户可以使用浏览器通过 web 服务器和数据库做交互，交互的结果将会以网页的形式显示在浏览器端。

### 3.2 设计优点

#### 3.2.1 交互性强

在 B/S 模式中，客户端有一套完整的应用程序，有着强大的功能，充分发挥客户端的处理能力。

#### 3.2.2 安全性高

由于 B/S 是配对的点对点的结构模式，使用的是适用于局域网、安全性较高的网络协议，具有较高的安全性。

#### 3.2.3 网络通信量少

B/S 的网络通信量只包括客户端与服务器之间的通信量。

#### 3.2.4 数据处理能力强

B/S 的三层逻辑结构在处理数据时，处理速度较快。

### 3.3 设计模式

系统的设计采用三层设计模式，分别为表现层，业务逻辑层以及最底层的数据层。

### 3.3.1 表现层

表现层是用来用户进行交互，直接呈现给用户进行访问信息，展示界面。主要对用户的请求接受，以及数据的返回，为客户端提供应用程序的访问。

### 3.3.2 业务逻辑层

业务逻辑层是实现表现层的各个功能，控制页面的跳转，是系统业务逻辑的实现层。其主要负责对数据层的操作。也就是说把一些数据层的操作进行组合。

### 3.3.3 数据层

数据层处在三层中的最底层，用来访问数据库，为业务逻辑层提供访问的类；主要包含对数据库中的表进行增删改查的实现。其主要是对非原始数据的操作层，而不是指原始数据，即对数据库的操作，为业务逻辑层或表示层提供数据服务。

### 3.3.4 总体架构

系统采用三层的设计模式，在开发过程中，层次分明，有利于小组进行分工合作，增加了后期的可维护性。

## 3.4 系统功能架构设计

疫情地图网站系统面向所有用户，为用户提供查看当下有关新型冠状病毒所致疫情的基本信息，包括疫情感染情况变化、防控措施等。

系统分为地图查看功能，防疫知识普及功能和热点查看功能，如图 2.2 所示。其中，本系统的核心为疫情地图查看功能，同时本系统在实现过程中也注重该功能的实现。

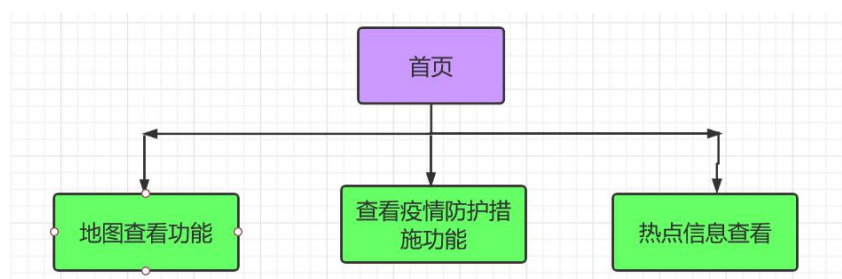


图 3.2 系统架构图

## 3.5 本章小结

本章介绍了系统架构和功能设计以及数据库设计三个部分，系统架构设计清晰地展示了本系统各层所使用到的技术及交互方式，数据库设计描述了表与表的关系以及表的详细信息。



## 第四章 功能模块设计

### 4.1 疫情地图查看功能设计

疫情地图查看功能的设计，主要用于向用户展示疫情数据信息。本系统将通过图表及数据等方式来展示疫情的数据信息，丰富且直观的展现出了当下疫情发展的信息及趋势，满足了用户的需求。

#### 4.1.1 中国疫情地图模块

1. 通过地图各地区的颜色深浅来表示确诊数量的程度，且点击‘累积确诊’或‘现存确诊’按钮，可以转换到相应的数据界面，同时在点击相关省份的时候会显示当前省份的‘累积确诊数量’或‘现存确诊数量’，在地图的实现上我们用 JS 文件引入地图文件，如图 4.1、4.2 所示。

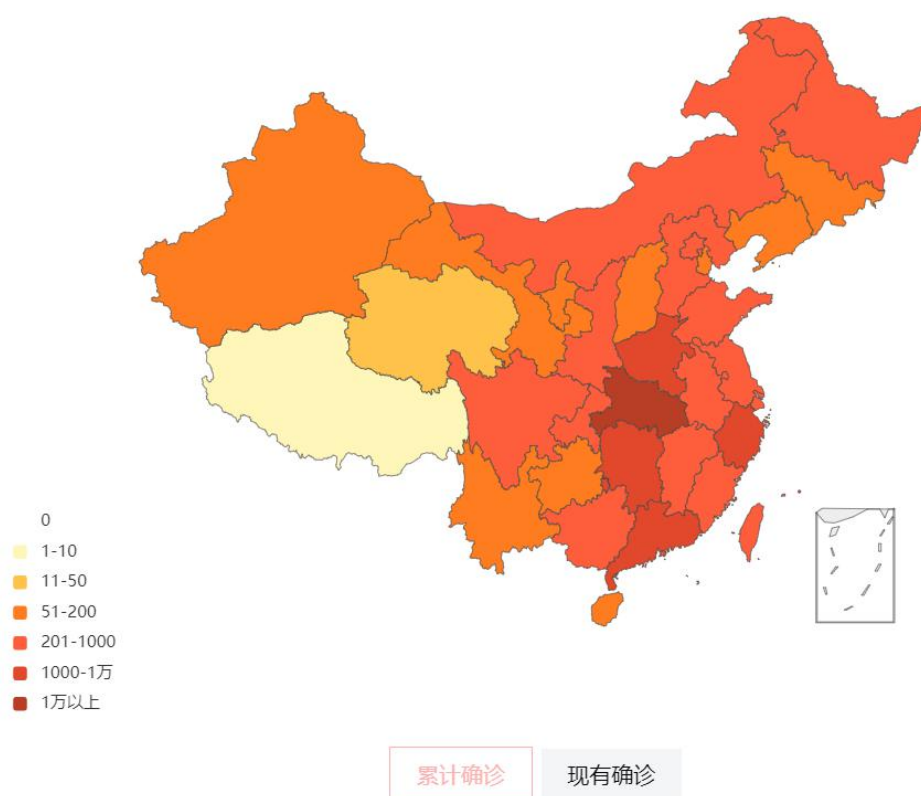


图 4.1 中国疫情地图累计确诊

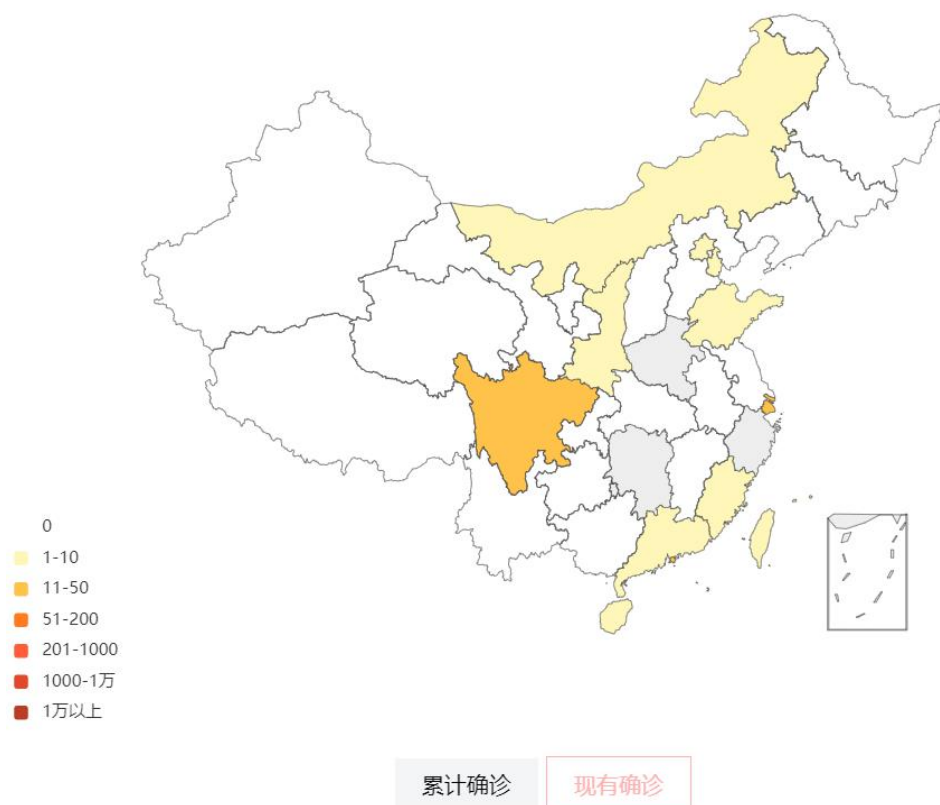


图 4.2 中国疫情地图现有确诊

2. 以颜色的深浅来表示确诊数量的程度，如 4.3（图例）所示。

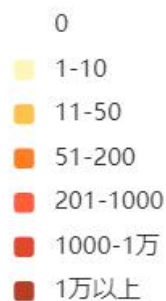


图 4.3 中国疫情地图图例

3. 我们可以在地图的上方看到总体的‘累积确诊’、‘现存确诊’、‘累积治愈’、‘累积死亡’，‘无症状感染者’和‘境外输入’的数据，如图 4.4 所示。



图 4.4 疫情总体信息展示

#### 4.1.2 各地具体的疫情数据模块

各个地区默认按照现有确诊人数的多少由人数从多到少排列，当点击某一地区（以内蒙古为例）的详情界面时，可以显示该地区的市级疫情地图和所有市的

疫情数据信息，如图 4.5-图 4.8 所示。

在数据的爬取上我们选择使用 Python，然后将数据存入数据库，然后再调出数据到相关界面。

国内病例 按现存确诊排序

地区	现存确诊	累计确诊	治愈	死亡	疫情
香港	44	1108	1060	4	
四川	18	583	562	3	<a href="#">详情</a>
上海	17	690	666	7	<a href="#">详情</a>
广东	10	1608	1590	8	<a href="#">详情</a>
北京	8	601	584	9	<a href="#">详情</a>
台湾	5	443	431	7	
内蒙古	5	237	231	1	<a href="#">详情</a>
山东	4	792	781	7	<a href="#">详情</a>

图 4.5 各省疫情具体数据



图 4.6 省内疫情总体数据和地图（内蒙古）

内蒙古疫情趋势图

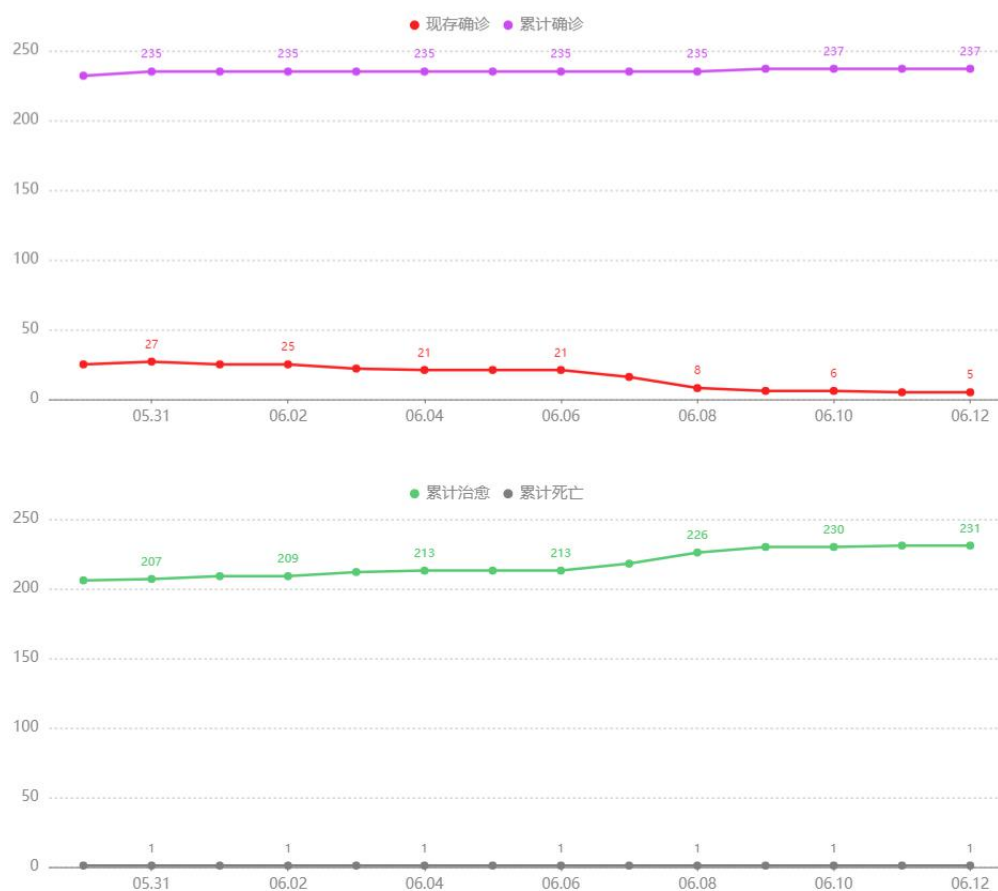


图 4.7 省内疫情趋势（内蒙古）

内蒙古病例（累计确诊排序）

地区	新增确诊	累计确诊	治愈	死亡
境外输入人员	0	160	155	0
鄂尔多斯市	0	11	11	0
包头市	0	11	11	0
赤峰市	0	9	9	0
呼伦贝尔市	0	9	9	0
锡林郭勒盟	0	9	9	0
巴彦淖尔市	0	8	7	1
通辽市	0	7	7	0
呼和浩特市	0	7	7	0
乌兰察布市	0	3	3	0
乌海市	0	2	2	0
兴安盟	0	1	1	0

图 4.8 省内疫情具体数据（内蒙古）

### 4.1.3 数据爬取模块

1. 导入包的操作的代码，如图 4.9 所示。

```
1 import requests
2 import time, json
3 import sys
4 import pymysql
```

图 4.9 导入包的操作

2. 发送请求并打印数据状态码的代码，如图 4.10 所示。

```
6 def get_wangyi_request():
7     url = 'https://c.m.163.com/ug/api/wuhan/app/data/list-total'
8
9     headers = {
10         'accept': '*/*',
11         'accept-encoding': 'gzip, deflate, br',
12         'accept-language': 'en-US,en;q=0.9,zh-CN;q=0.8,zh;q=0.7',
13         'origin': 'https://wp.m.163.com',
14         'referer': 'https://wp.m.163.com/',
15         'sec-fetch-dest': 'empty',
16         'sec-fetch-mode': 'cors',
17         'sec-fetch-site': 'same-site',
18         'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/37.36 (KHTML, like Gecko)'
19     }
20
```

图 4.10 发送请求并打印数据状态码

3. 存入数据库的相关操作的代码，如图 4.11 所示。

```
conn = pymysql.connect(
    host='localhost', # 我的IP地址
    port=3306, # 不是字符串不需要加引号。
    user='root',
    password='20000604',
    db='database',
    charset='utf8'
)
```

图 4.11 存入数据库的相关操作

4. 登陆界面的实现代码，如图 4.12 和图 4.13 所示。

```
2<html lang="en">
3<head>
4<meta charset="utf-8" />
5<link rel="shortcut icon" href="%PUBLIC_URL%/favicon.ico" />
6<meta
7  name="viewport"
8  content="width=device-width, initial-scale=1, shrink-to-fit=no"
9 />
10<meta name="theme-color" content="#000000" />
11<!--
12  manifest.json provides metadata used when your web app is installed on a
13  user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/
14 -->
15<link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
16<title>疫情信息查询系统</title>
17</head>
```

图 4.12 登陆界面的实现 1

```

1 import React, { Component } from 'react';
2 import { Form, Icon, Input, Button, message, Radio, Modal, Select, Cascader, notification } from 'antd';
3 import API from '../api/api';
4 import './adminLogin.less';
5
6 const Option = Select.Option;
7
8 class AdminLogin extends Component {
9   constructor() {
10     super();
11     this.state = {
12       modalVisible: false,
13       options: [],
14     }
15   }
16   //管理员登录方法
17   async _adminLogin(data) {
18     try {
19       let result = await API.Login({data});
20       const { status, msg } = result;
21       if (status === 200) {
22         notification.success({
23           message: '登录成功!',
24           description: '欢迎使用疫情信息查询系统',
25         });
26         this.props.history.push(`/app-${result.data.type}`);
27       } else {
28         message.error(msg || '登录失败');
29       }
30     } catch (err) {
31       message.error('登录失败', err);
32     }
33   }

```

图 4.13 登陆界面的实现 2

#### 4.1.4 中国疫情地图模块

(1) 通过地图各地区的颜色深浅来表示确诊数量的程度，且点击‘累积确诊’或‘现存确诊’按钮，可以转换到相应的数据界面，同时在点击相关国家的时候会显示当前国家的‘累积确诊数量’或‘现存确诊数量’，在地图的实现上我们使用 JS 文件引入地图文件，如图 4.14-图 4.15 所示。

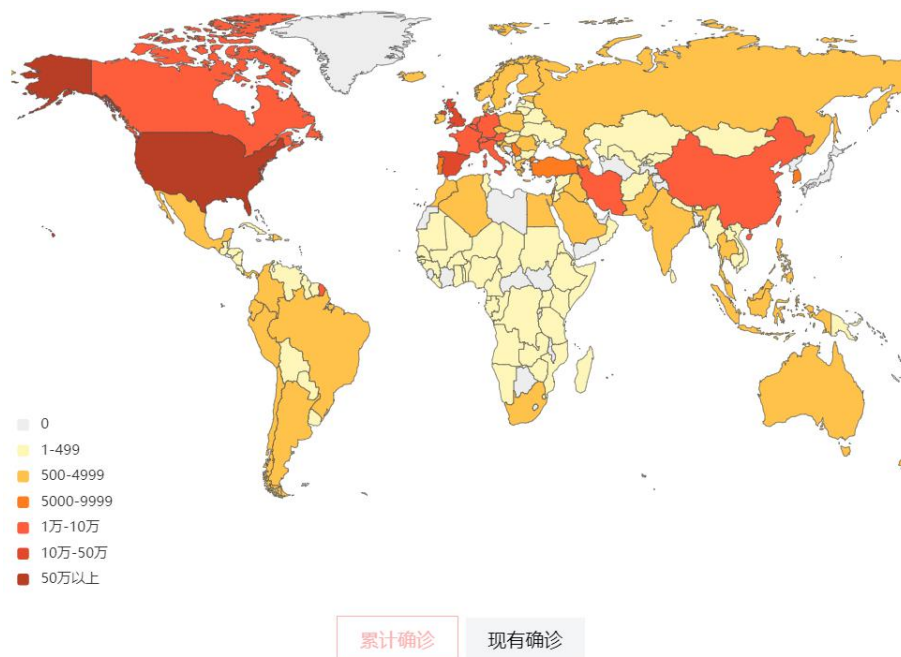


图 4.14 世界疫情地图累积确诊



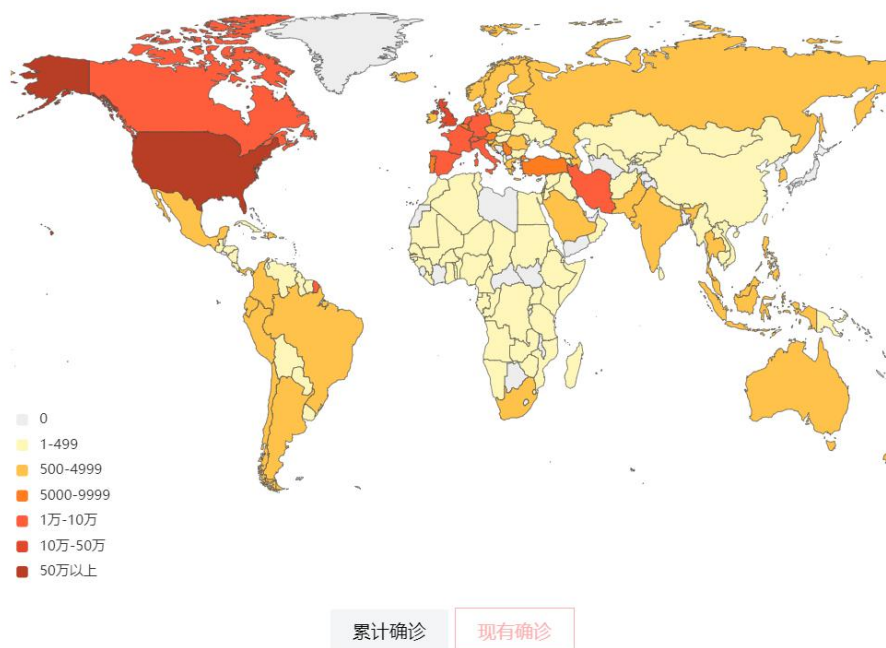


图 4.15 世界疫情地图现有确诊

(2) 以颜色的深浅来表示确诊数量的程度，如图 4.16（图例）所示。

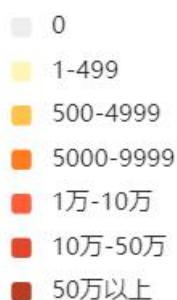


图 4.16 世界疫情地图图例

(3) 我们可以在地图的上方看到总体的‘累积确诊’、‘现存确诊’、‘累积治愈’、‘累积死亡’的数据，如图 4.17 所示。



图 4.17 疫情总体信息展示

#### 4.1.5 海外疫情模块

(1) 海外疫情模块

根据海外疫情的更新，绘制了多个图表，展示海外的疫情数据，通过曲线图

来显示本次疫情的一个变化趋势，可直观的向用户展示疫情发展的趋势，让用户更简单的了解到当下海外疫情的发展。如图 4.18-图 4.21 所示。

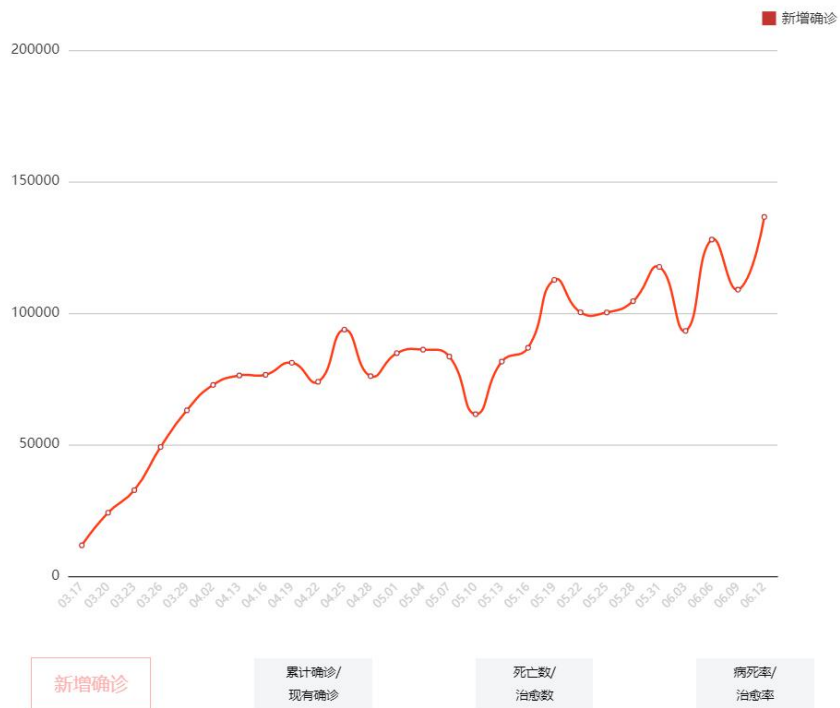


图 4.18 海外疫情趋势

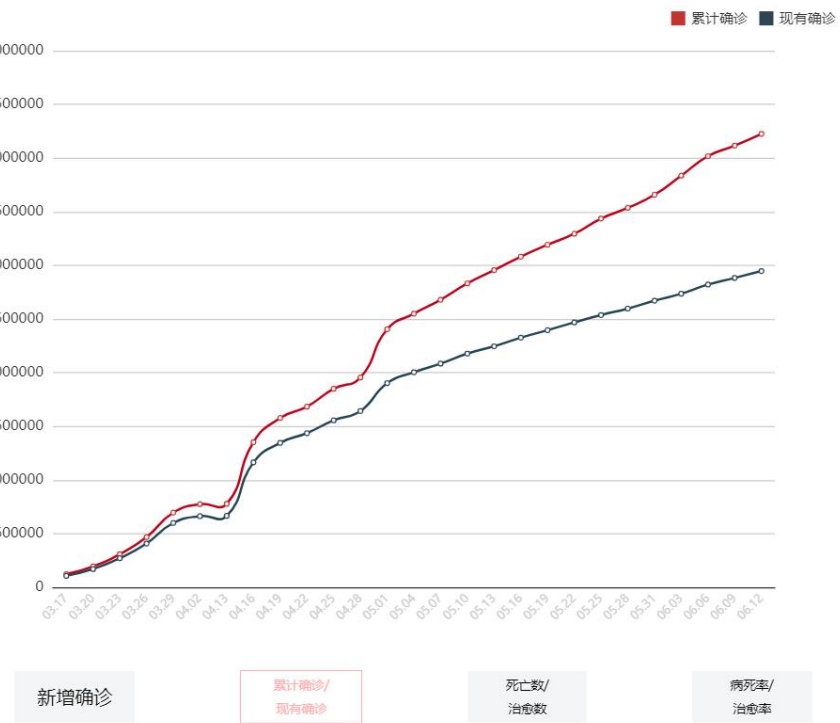


图 4.19 海外疫情趋势



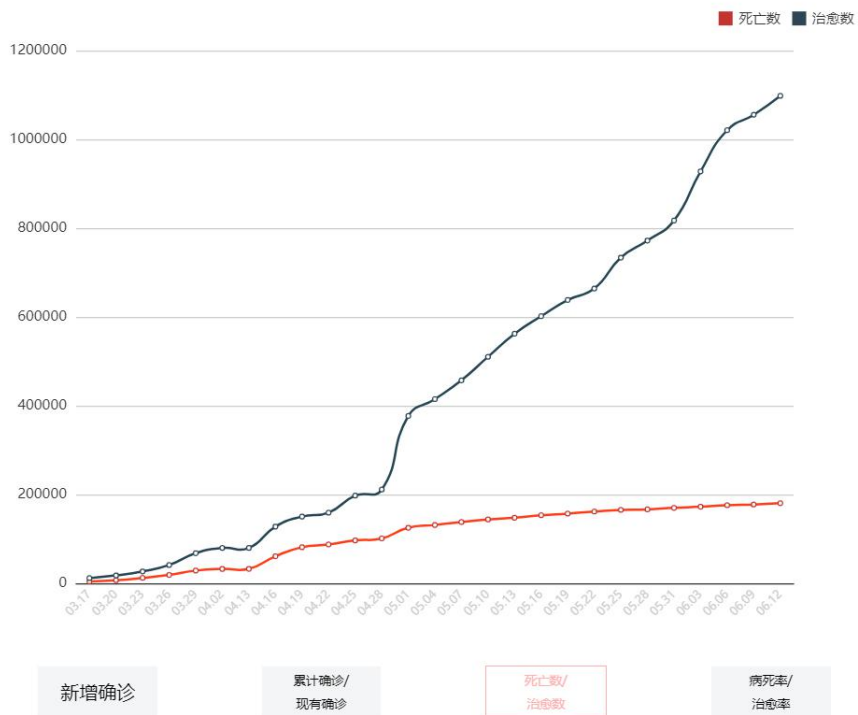


图 4.20 海外疫情趋势

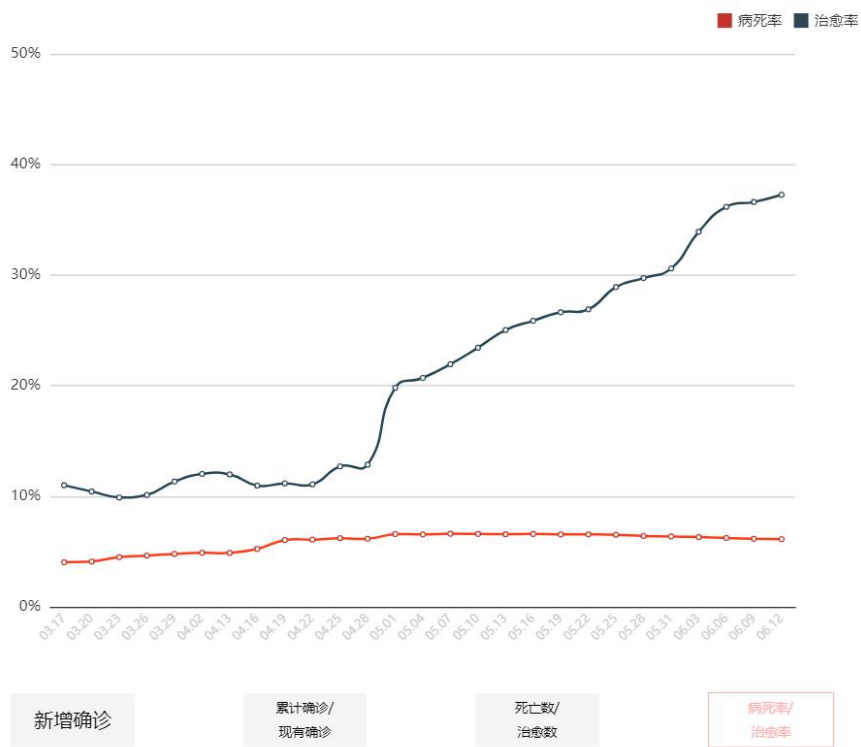


图 4.21 海外疫情趋势

## (2) 海外新增 Top10

海外疫情愈演愈烈，展示最近新增的 Top10 国家，如图 4.22 所示。

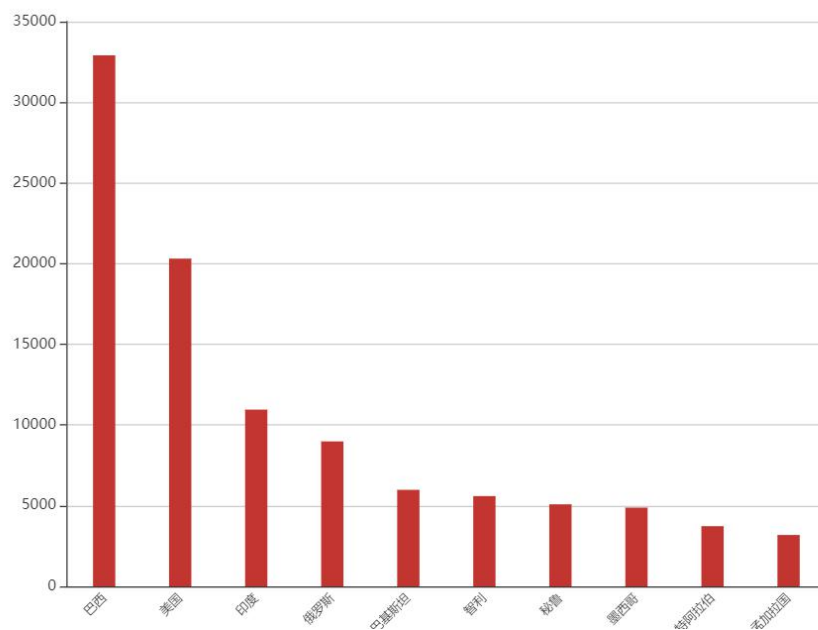


图 4.22 疫情新增 Top10 国家

### (3) 各地具体的疫情数据

各个地区默认按照累积确诊人数的多少由人数从多到少排列,可以看到一个国家大体的疫情数据,如图 4.23 所示。

在数据的爬取上我们选择使用 Python,然后将数据存入数据库,然后再调出数据到相关界面。

地区	新增确诊	累计确诊	治愈	死亡	疫情
美国	27221	2116922	841934	116825	
巴西	25982	828810	445123	41828	
俄罗斯	8706	520129	268862	6829	
印度	11432	308993	154330	8884	
英国	1452	294402	1282	41566	
西班牙	502	290289	196958	27136	
意大利	163	236305	173085	34223	
秘鲁	5961	220749	107133	6308	
法国	727	193220	72695	29377	
德国	330	187263	171904	8869	
伊朗	0	182545	144649	8659	

图 4.23 海外疫情具体数据展示

## 4.2 疫情相关防控措施查看功能设计

给用户提供可用作参考的且有效的新冠肺炎疫情的防控措施，在浏览当下新冠肺炎疫情的信息的同时，也让用户提高警惕性，加强自身的防控措施。

该功能的设计主要通过文字来展示，如图 4.24 所示。



图 4.24 防疫知识普及

## 4.3 热点新闻展示功能

疫情期间，我们足不出户，但是我们关注着世界各地的新闻热点，该功能给我们展示了疫情期间的新闻热点，如图 4.25 所示。

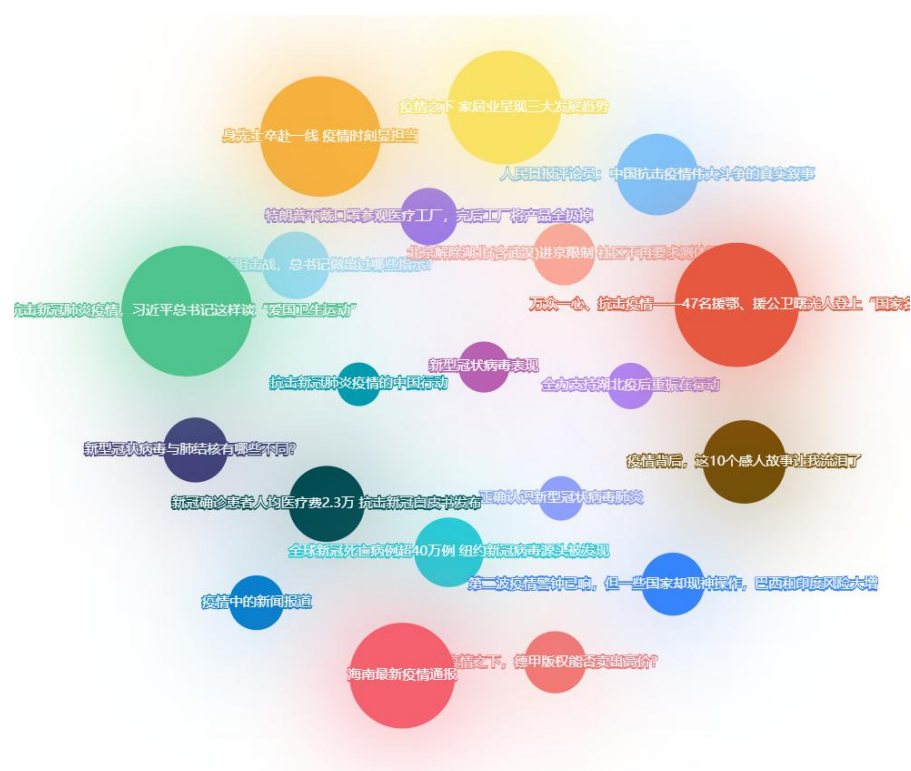


图 4.25 热点新闻

## 第五章 数据库设计

### 5.1 疫情数据信息设计

- (1) 全国现有确诊/疑似/累计人数。
- (2) 全国总新增确诊/新增境外输入确诊人数。
- (3) 全国死亡/治愈人数。
- (4) 确诊病例日增长变化数量。

### 5.2 数据安全性设计

在系统中，为了保证疫情数据的实时有效性，需要每日通过爬虫向数据库中定期更新数据信息。为了防止数据库被别人恶意篡改，将在数据库中进行一些安全措施来保证一定的安全性。

首先，对用户数据库的操作的权限进行限制，分为不同的角色，对不同的角色授予不通过的权限。此外，在数据库中可设置日志文件，对数据库的每一次操作都有一个准确的记录，在一定程度上保证了数据的安全。

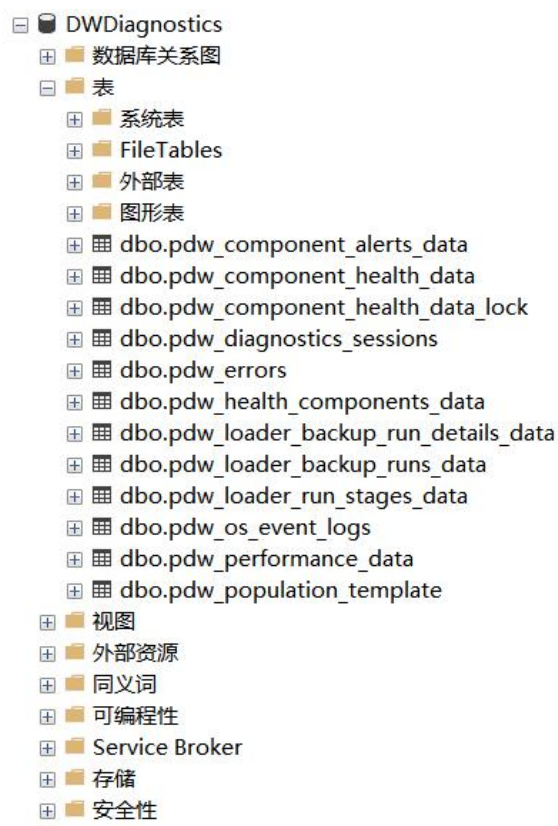


图 5.1 数据库表结构

## 第六章 系统的结构化设计

根据 SA 方法中的数据流图建立一个良好的模块结构图；运用模块化的设计原理控制系统的复杂性，即设计出模块相对独立的,模块结构图深度,宽度都适当的，单入口单出口的，单一功能的模块结构的软件结构图或软件层次方框图。

此方法提供了描述软件系统的工具，提出了评价模块结构图质量的标准，即模块之间的联系越松散越好，而模块内各成分之间的联系越紧凑越好。

结构化设计的目的:使程序的结构尽可能反映要解决的问题的结构。

结构化设计的任务:把需求分析得到的数据流图 DFD 等变换为系统结构图。

### 6.1 系统层次图

层次图用来描绘软件的层次结构。数据结构的层次方框图相同，但是表现的内容却完全不同。层次图很适于在自顶向下设计软件的过程中使用，该系统层次图如图 5.1 所示。

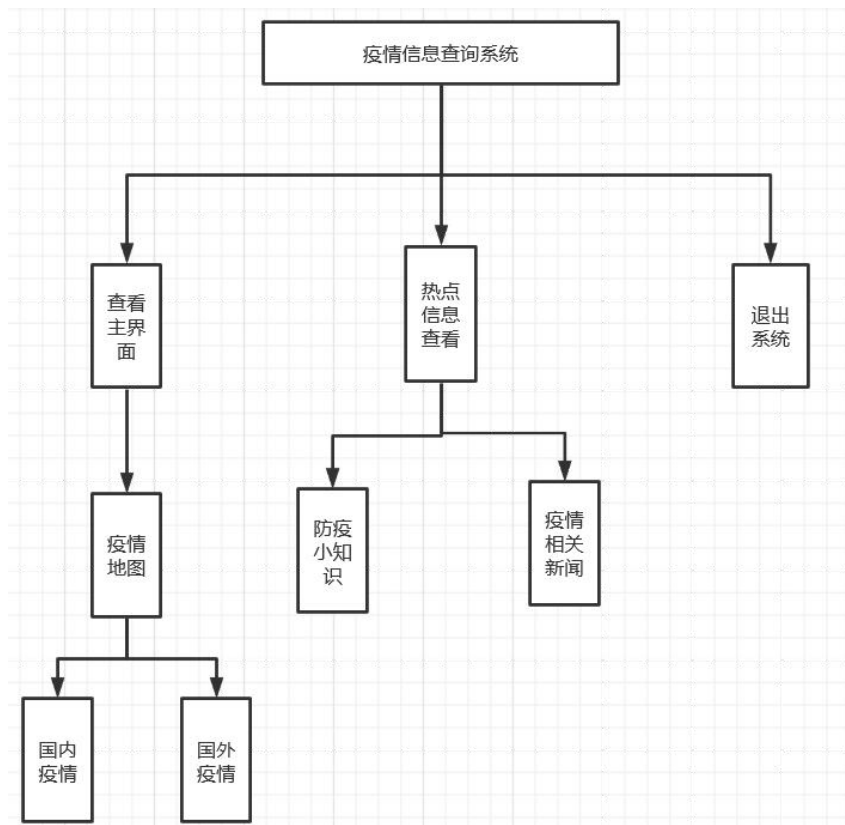


图 6.1 系统层次图

### 6.2 系统结构图

系统结构图是结构化设计方法使用的描述方式，也称结构图或控制结构图。

它表示了一个系统的层次分解关系, 模块之间的调用关系, 以及模块之间数据流和控制流信息的传递关系, 该系统结构图如图 6.2 所示。

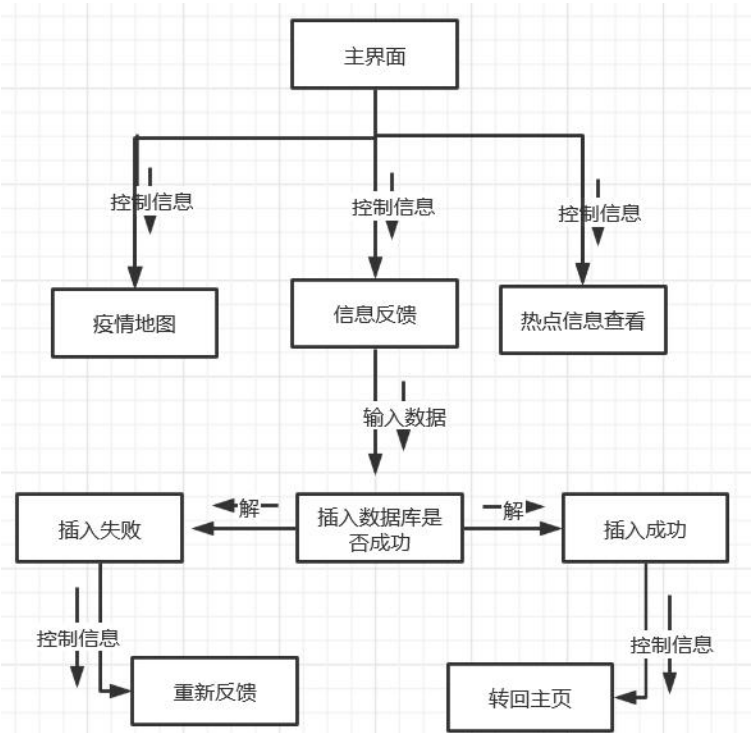


图 6.2 系统结构图

### 6.3 本章小结

本章说明了疫情信息查询系统的开发环境及开发工具, 展示了本系统的一些关键功能以及实现方法和流程。例如中国疫情信息展示, 国外疫情信息展示、防疫知识普及模块、新闻热点聚焦模块等, 以及系统的爬取处理数据的代码。体现了这些功能的实现逻辑和思路。

## 第七章 系统测试

### 7.1 测试原则与方法

测试的关键是测试用例的设计，其方法可分成两类：白盒测试和黑盒测试。

#### 7.1.1 白盒测试

白盒测试是把程序看成装在一只有透明的白盒子里，测试者完全了解程序的结构和处理过程。它根据程序的内部逻辑来设计测试用例，检查程序中的逻辑通路是否都按预定的要求正确地工作。

#### 7.1.2 黑盒测试

黑盒测试是把程序看成一只黑盒子，测试者完全不了解程序的结构和处理过程。它根据规格说明书规定的功能来设计测试用例，检查程序的功能是否符合规格说明的要求。

黑盒测试也称功能测试，它是通过测试来检测每个功能是否都能正常使用，在程序接口进行测试，它只检查程序功能是否按照需求规格说明书的规定正常使用，程序是否能适当地接收输入数据而产生正确的输出信息。黑盒测试着眼于程序外部结构，不考虑内部逻辑结构，主要针对软件界面和软件功能进行测试。

对于系统的测试是基于系统整体需求说明书的黑盒测试，覆盖系统所有联合的部件。系统测试是针对整个产品系统进行的测试，目的是验证系统是否满足了需求规格的定义，然后找出错误。

### 7.2 测试过程

测试的过程按以下四个步骤进行：

#### 7.2.1 模块测试

测试每个模块完成其定义的子功能，把模块作为单一的实体来测试，保证每个模块作为一个单元能正确运行，主要包括数据的测试，保存和调用以及对后台数据库的连接及存取和添加、删除、修改等功能。

#### 7.2.2 子系统测试

把经过单元测试的模块，放在一起形成一个子系统来测试，模块间的协调和通讯是这个测试功能中的主要问题。保证模块间的相互协调及数据一致。包括对数据库中已存关联信息的提取及查询，完成数据库的相关存储及相应功能。

### **7.2.3 系统测试**

把经过测试的子系统装配成一个完整的系统来测试，不仅能发现设计编码的错误，还可以验证系统是否完成其指定功能，对存在的遗漏进行调整。

### **7.2.4 验收测试**

通过运行本系统，以用户登陆，使用实际数据进行测试，验收测试目的是验证系统确实能够满足用户的需要，并在实际使用中检测设计中存在的缺陷，及对实际情况的考虑中存在的问题，进一步调整数据库功能，使其更加完备。

## **7.3 本章小结**

本章主要是对系统进行测试，希望能从中找出一些错误，并且进行改进，减少系统的错误。