

编号： CACR2021BIRNJC



作品类别： ☒ 软件设计   ☐ 硬件制作   ☐ 工程实践

## 2021 年第六届全国密码技术竞赛作品设计报告

---

题目： 基于国密的智慧医疗分诊系统

2021 年 10 月 20 日

中国密码学会

## 基本信息表

编号：CACR2021BIRNJC

作品题目：基于国密的智慧医疗分诊系统

作品类别：☒软件设计    ☐硬件制作    ☐工程实践

作品内容摘要：

随着信息技术的发展，人们对电子医疗服务质量的需求越来越高。为了给患者带来更好的就医体验，提高患者就诊的效率，降低医护人员的工作强度，本作品设计实现了一种智慧医疗分诊系统，它可以帮助患者根据系统所提供的人体平面图精准的判断自身的病情，并选择相应的诊室和医生进行挂号预约。同时，该系统提供了智慧分诊功能，减少患者排队等候的时间，提高医护人员的工作效率等。本系统使用国密算法SM2对患者和医护人员的密码信息进行加密。对医生的基本信息，如姓名、医龄、简介等，以及科室名称和公告信息使用国密算法SM9进行加密。通过使用不同的加密算法，提高医患双方信息的安全性，防止其被敌手进行非法窃取和篡改。基于此设计思想，使用Python语言开发了一套嵌入国密算法的智慧医疗分诊系统，并对系统的功能和性能进行了相关的检验和测试。

关键词（五个）：

国密 SM2，国密 SM9，数据加密，智慧分诊，隐私保护

## 1.作品功能与性能说明

### 1.1 功能说明

#### 1.1.1 医疗分诊功能

为了方便就医人员看病就诊，在网上进行预约排队挂号，我们设计了一种医疗分诊系统。该系统可以在就医人员选择科室进行挂号时，展示人体平面图界面，可以帮助就医人员根据自身的情况更加准确的选择要挂的科室，就医人员也可以选择自己的医生。本功能不仅提供了就医人员选择医生的权力，实现公平、公正的分诊，同时也大大节约了就医人员的时间，提高了就医效率，避免就医人员间的纠纷等。提升了医疗服务水平，给人们带来更好的就医体验和服务。

#### 1.1.2 信息加密、解密功能

信息涉及到每个人的财产安全甚至是人身安全，因此保护信息的安全尤为重要。本作品使用国密算法 SM2 和 SM9 对不同的信息进行加密，以保证就医者信息和医生信息为重要目的，确保攻击者无法获取到有效的个人信息，实现对个人信息的有效保护。如今在医院中还存在黄牛倒卖号的现象，这会使得本已挂好号排好队的就医人员受到不公平的对待，破坏看病就医的秩序。除此之外，也存在对就医人员信息盗取的非法行为，不怀好意的人将信息卖给其他人来获利，侵犯了就医人员的隐私，甚至还可能威胁到就医人员的财产安全和人身安全。因此，用国密算法进行信息加密可以有效的保护信息的安全，给就医人员带来一定的安全感。

### 1.2 性能说明

(1) 保护就医人员的隐私（个人信息、就医情况）

(2) 保护医生的个人信息

(3) 对就医人员及医生的关系进行保密

(4) 使用不同的算法对就医人员和医生的信息分别进行加密，提供了更好的安全性

(5) 设立了人体平面图，便于就医人员根据自身情况进行诊室选择，也可提供自助分诊功能，提高了就诊效率，为医生等工作人员提高了工作效率

(6) 可以有效抵抗攻击者对数据的篡改和伪造，攻击者无法获取就医人员和医生的真实信息。

## 2.设计与实现方案

### 2.1 实现原理

本作品首先对就医人员和医生的信息分别选用不同的加密算法进行加密并存储在数据库中，就医人员在系统中根据自身病情选择诊室，发送请求获取医生信息，选择医生进行挂号预约，流程图如下：

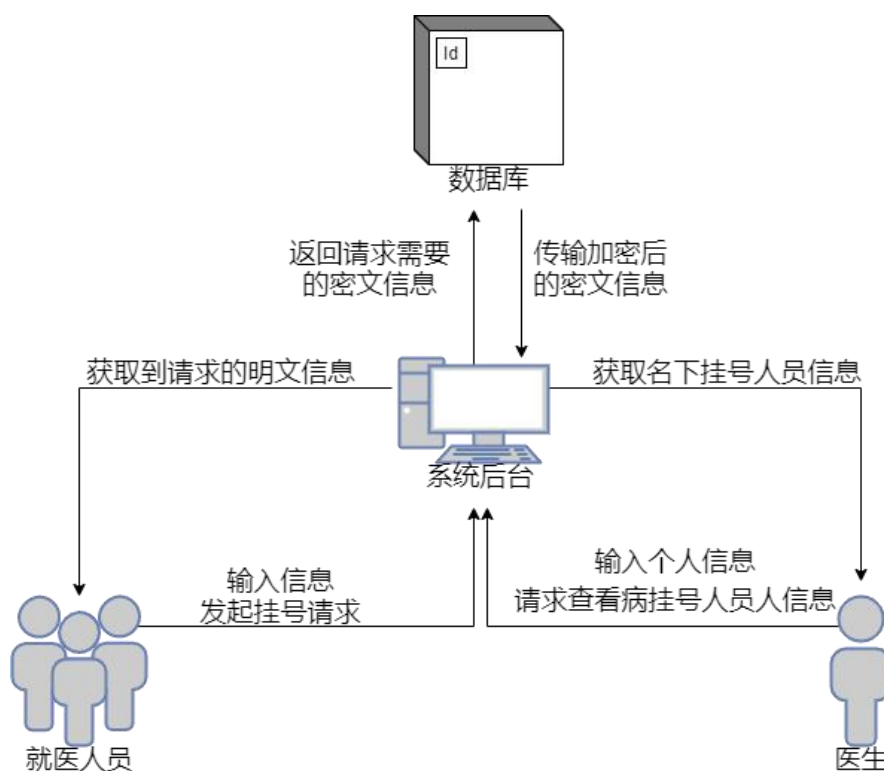


图 2.1 基于国密的智慧医疗分诊系统流程图

如图 2.1 所示，具体步骤如下：

(1) 医生进入医疗分诊系统进行注册，填写个人信息（基本信息、科室信息、简介）等，将数据传送给系统后台，使用 SM9 国密算法对医生基本信息加密并存储在数据库中，用 SM2 国密算法对医生的登陆密码进行加密并存储在数据库

中，完成登录。医生将个人信息传入到医疗系统中，便于就医人员预约挂号的选择。

(2) 就医人员进入医疗分诊系统，进行注册，填写个人基本信息，完成登录。系统将信息输入后台并使用 SM9 国密算法进行加密，存储在数据库中。

(3) 就医人员进入人体平面图系统界面，根据自身疼痛情况选择需要就诊科室，向系统后台提供请求获取医生信息进行挂号预约，系统后台从数据库中查询获取对应诊室的医生信息进行解密并返回给就医人员，就医人员可根据医生的信息选择医生进行挂号预约。若就医人员没有指定的医生，有智能分诊功能，可帮助自动选择医生进行挂号预约。

(4) 医生通过医疗分诊系统向系统后台请求其名下就医人员基本情况，后台从数据库中调取数据并进行解密返回给医生，医生可看到自己名下的就医人员信息及预约挂号情况。

### 2.1.1 SM2 算法实现原理

#### (1) 加密原理

设需要发送的消息为比特串  $M$ ， $klen$  为  $M$  的比特长度。

为了对明文  $M$  进行加密，作为加密者的用户 A 应实现以下运算步骤：

A1：用随机数发生器产生随机数  $k \in [1, n-1]$ ；

A2：计算椭圆曲线点  $C_1 = [k]G = (x_1, y_1)$ ，( $[k]G$  表示  $k * G$ ) 将  $C_1$  的数据类型转换为比特串；

A3：计算椭圆曲线点  $S = [h]P_B$ ，若  $S$  是无穷远点，则报错并退出；

A4：计算椭圆曲线点  $[k]P_B = (x_2, y_2)$ ，将坐标  $x_2$ 、 $y_2$  的数据类型转换为比特串；

A5：计算  $t = KDF(x_2 \parallel y_2, klen)$ ，若  $t$  为全 0 比特串，则返回 A1；

A6：计算  $C_2 = M \oplus t$ ；

A7：计算  $C_3 = Hash(x_2 \parallel M \parallel y_2)$ ；

A8：输出密文  $C = C_1 \parallel C_2 \parallel C_3$ 。

#### (2) 解密原理

设  $klen$  为密文中  $C_2$  的比特长度。

为了对密文  $C = C_1 \parallel C_2 \parallel C_3$  进行解密，作为解密者的用户 B 应实现以下运算步骤：

B1: 从  $C$  中取出比特串  $C_1$ ，将  $C_1$  的数据类型转换为椭圆曲线上的点，验证  $C_1$  是否满足椭圆曲线方程，若不满足则报错并退出；

B2: 计算椭圆曲线点  $S = [h]C_1$ ，若  $S$  是无穷远点，则报错并退出；

B3: 计算  $[d_B]C_1 = (x_2, y_2)$ ，将坐标  $x_2$ 、 $y_2$  的数据类型转换为比特串；

B4: 计算  $t = KDF(x_2 \parallel y_2, klen)$ ，若  $t$  为全 0 比特串，则报错并退出；

B5: 从  $C$  中取出比特串  $C_2$ ，计算  $M' = C_2 \oplus t$ ；

B6: 计算  $U = Hash(x_2 \parallel M' \parallel y_2)$ ，从  $C$  中取出比特串  $C_3$ ，若  $U \neq C_3$ ，则报错并退出；

B7: 输出明文  $M'$ 。

### (3) 安全参数设置

随机数  $k$  和私钥  $d_B$  最好大点， $2^{**}50$  以上比较安全。

## 2.1.2 SM9 算法实现原理

### (1) 加密算法

设需要发送的消息为比特串  $M$ ， $klen$  为  $M$  的比特长度， $K_1\_len$  为分组密码算法中密钥  $K_1$  的比特长度， $K_2\_len$  为函数  $MAC(K_2, Z)$  中密钥  $K_2$  的比特长度。

为了加密明文  $M$  给用户 B，作为加密者的用户 A 应实现以下运算步骤：

A1: 计算群  $l$  中的元素  $Q_B = [H_1(ID_B \parallel hid, N)]P_1 + P_{pub-e}$ ；

A2: 产生随机数  $r \in [1, N-1]$ ；

A3: 计算群  $l$  中的元素  $C_1 = [r]Q_B$ ，将  $C_1$  的数据类型转换为比特串；

A4: 计算群  $T$  中的元素  $g = e(P_{pub-e}, P_2)$ ；

A5: 计算群  $T$  中的元素  $w = gr$ ，按将  $w$  的数据类型转换为比特串；

A6: 按加密明文的方法分类进行计算：

a) 如果加密明文的方法是基于密钥派生函数的序列密码算法，则

1) 计算整数  $klen = mlen + K_2\_len$ ，然后计算  $K = KDF(C_1 \parallel w \parallel ID_B, klen)$ 。

令  $K_1$  为  $K$  最左边的  $mlen$  比特， $K_2$  为剩下的  $K_2\_len$  比特，若  $K_1$  为全 0 比特串，则返回 A2；

2) 计算  $C_2 = M \oplus K_1$ 。

b) 如果加密明文的方法是结合密钥派生函数的分组密码算法，则

1) 计算整数  $klen = K_1\_len + K_2\_len$ ，然后计算  $K = KDF(C_1 \parallel w \parallel ID_B, klen)$ 。  
令  $K_1$  为  $K$  最左边的  $K_1\_len$  比特， $K_2$  为剩下的  $K_2\_len$  比特，若  $K_1$  为全 0 比特串，  
则返回 A2；

2) 计算  $C_2 = Enc(K_1, M)$ 。

A7: 计算  $C_3 = MAC(K_2, C_2)$ ；

A8: 输出密文  $C = C_1 \parallel C_3 \parallel C_2$ 。

## (2) 解密算法

设  $mle$  为密文  $C = C_1 \parallel C_3 \parallel C_2$  中  $C_2$  的比特长度， $K_1\_len$  为分组密码算法中  
密钥  $K_1$  的比特长度， $K_2\_len$  为函数  $MAC(K_2, Z)$  中密钥  $K_2$  的比特长度。

为了对  $C$  进行解密，作为解密者的用户 B 应实现以下运算步骤：

B1: 从  $C$  中取出比特串  $C_1$ ，将  $C_1$  的数据类型转换为椭圆曲线上的点，验证  $C_1$  是  
否成立，若不成立则报错并退出；

B2: 计算群  $T$  中的元素  $w' = e(C_1, de_B)$ ，将  $w'$  的数据类型转换为比特串；

B3: 按加密明文的方法分类进行计算：

a) 如果加密明文的方法是基于密钥派生函数的序列密码算法，则计算

1) 计算整数  $klen = mle + K_2\_len$ ，然后计算  $K' = KDF(C_1 \parallel w' \parallel ID_B, klen)$ 。  
令  $K_1'$  为  $K'$  最左边的  $mle$  比特， $K_2'$  为剩下的  $K_2\_len$  比特，若  $K_1'$  为全 0 比特串，  
则报错并退出；

2) 计算  $M' = C_2 \oplus K_1'$ 。

b) 如果加密明文的方法是结合密钥派生函数的分组密码算法，则计算

1) 计算整数  $klen = K_1\_len + K_2\_len$ ，然后计算  $K' = KDF(C_1 \parallel w' \parallel ID_B, klen)$ 。  
令  $K_1'$  为  $K'$  最左边的  $K_1\_len$  比特， $K_2'$  为剩下的  $K_2\_len$  比特，若  $K_1'$  为全 0 比特串，  
则报错并退出；

2) 计算  $M' = Dec(K_1', C_2)$ 。

B4: 计算  $U = MAC(K_2', C_2)$ ，从  $C$  中取出比特串  $C_3$ ，若  $U$  不等于  $C_3$ ，则报错并  
退出；

B5: 输出明文  $M'$ 。

## 2.2 功能实现

基于国密的智慧医疗分诊系统，软件总体框架图如图所示：

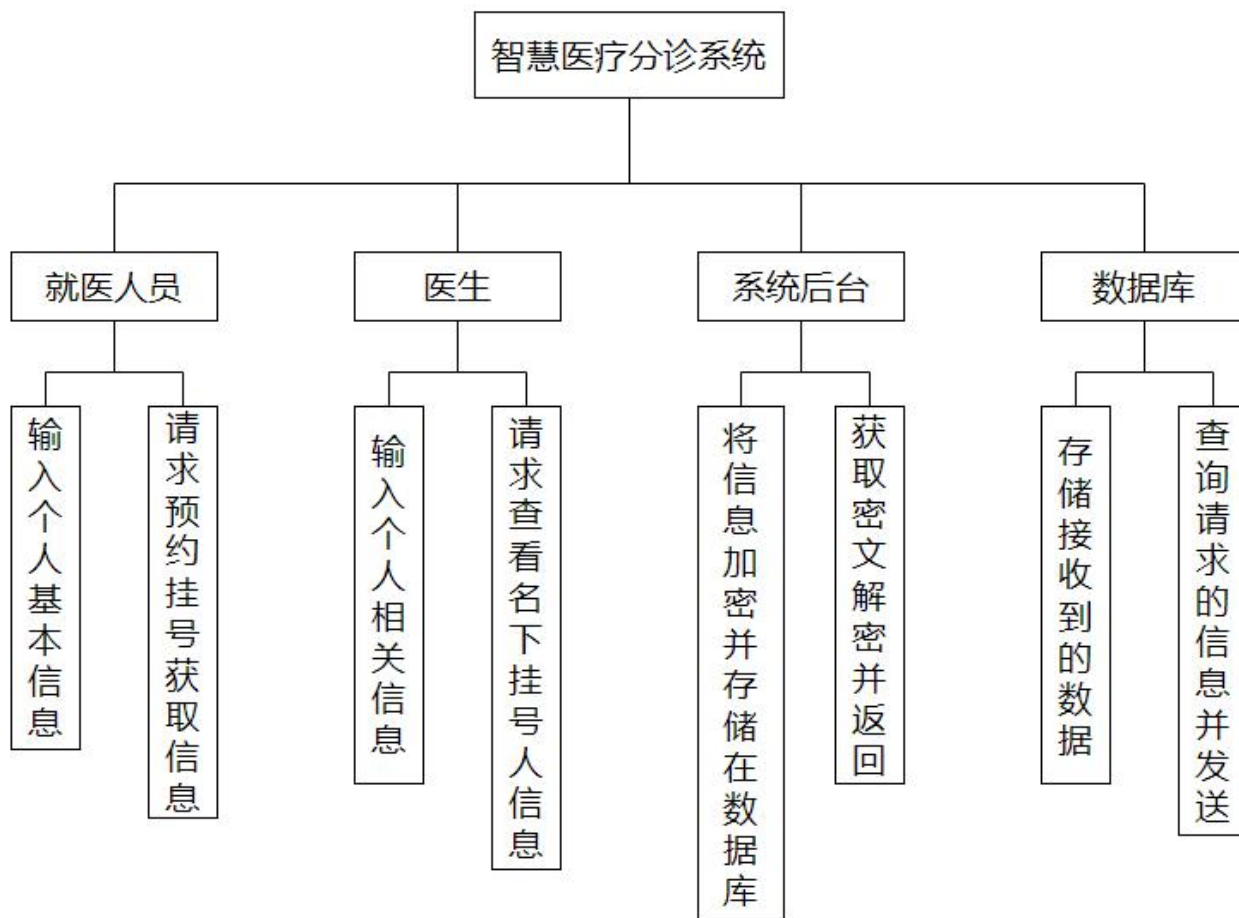


图 2.2 软件框架图

1.数据库：存储加密后的所有信息，包括就医人员和医生的个人基本信息情况，科室信息及公告信息等。

①数据库存储总数据加密信息情况，如图 2.3 所示

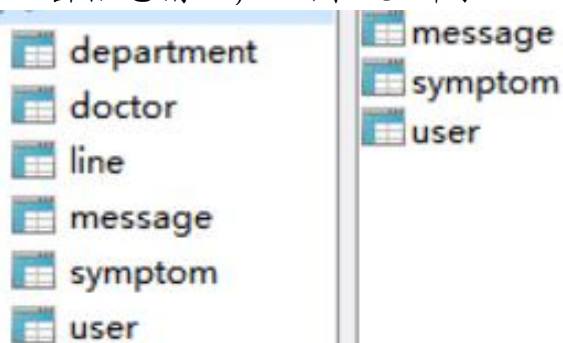


图 2.3 数据库信息

②医院内所有科室加密后的密文信息，如图 2.4 所示



department @doctor (test) - 表

id	name
1	403203754471932522426
2	133455083830869169632
3	475318953511492900369
4	535270485514048077453
5	237092892158385368459
6	237428478466438856025

图 2.4 医院内所有科室密文信息

③单个医生个人信息加密密文信息，如图 2.5 所示

doctor @doctor (test) - 表

id	user	name	years	introduction	dp
1	2	860859745543522256090	5	1641852960372C	1

图 2.5 单个医生个人信息加密情况

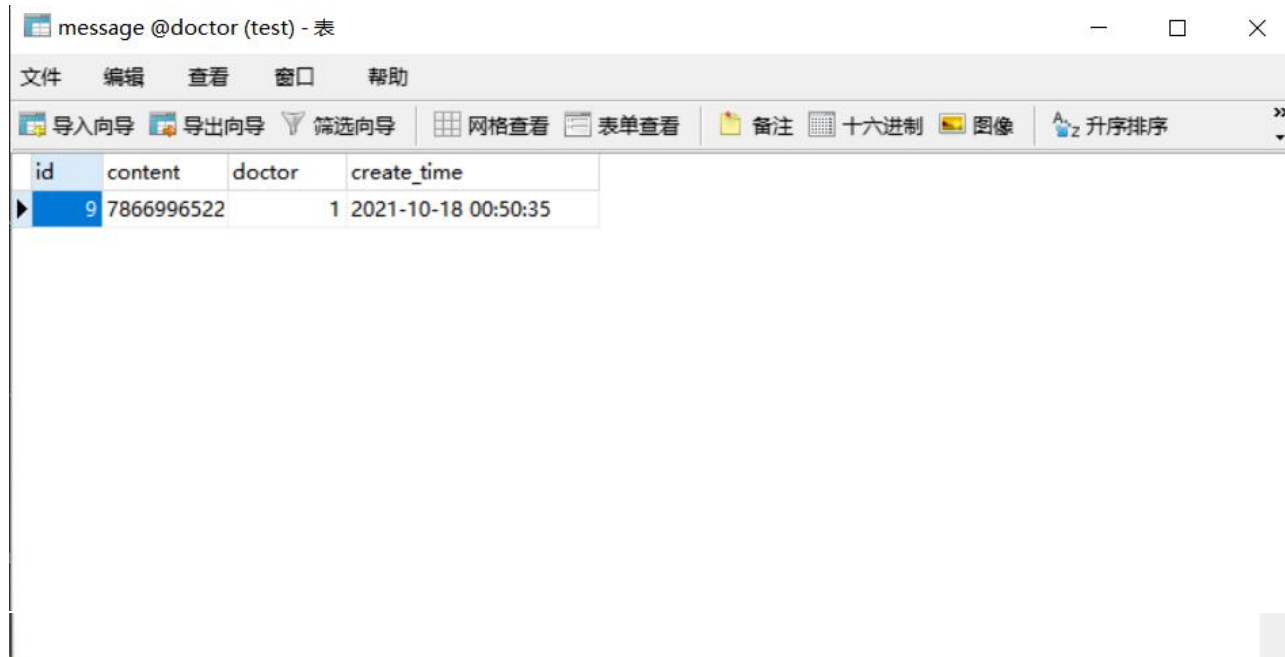
④医生名下所属患者信息情况，如图 2.6 所示

line @doctor (test) - 表

id	user	doctor	create_time
13	1	1	2021-10-16 00:28:16
14	5	1	2021-10-18 00:48:28

图 2.6 医生名下患者信息

## ⑤科室内公告信息通过密文存储，如图 2.7 所示

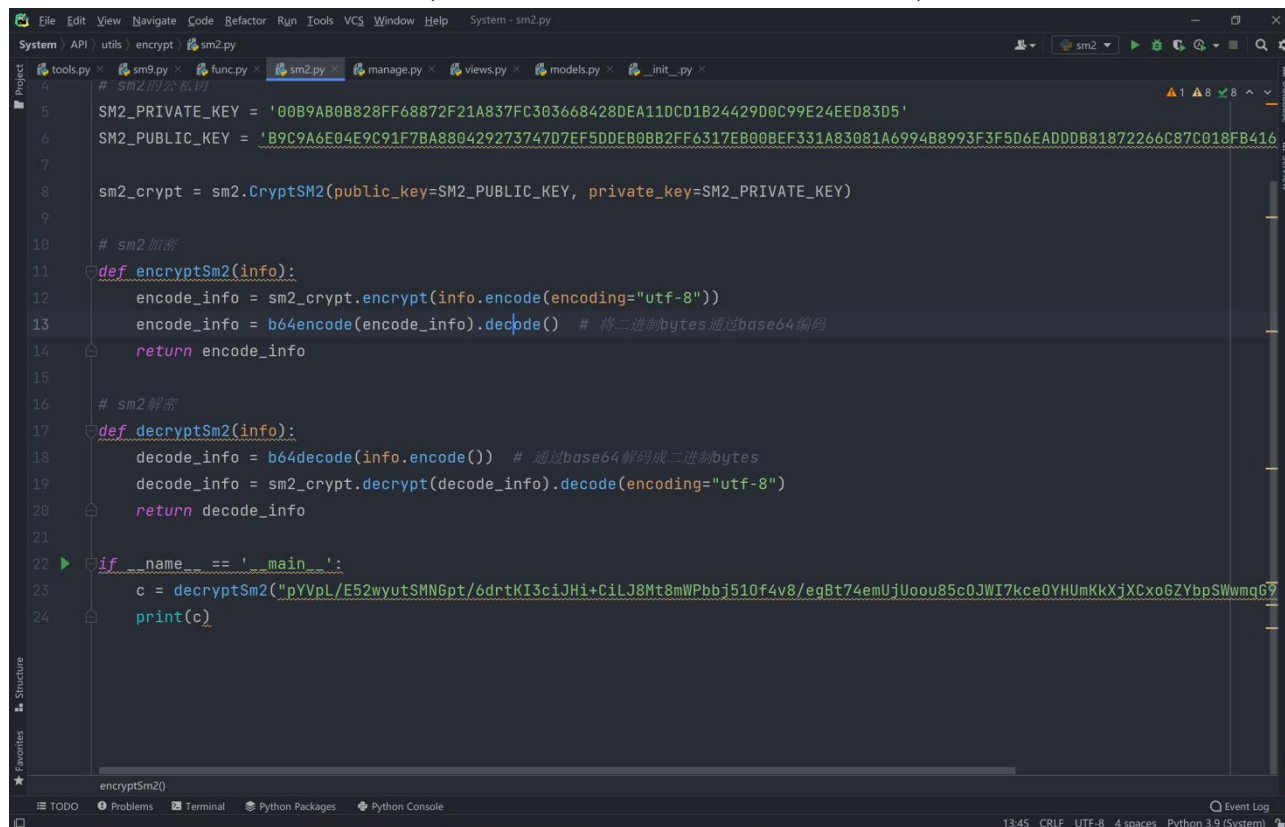


id	content	doctor	create_time
9	7866996522	1	2021-10-18 00:50:35

图 2.7 科室内公告信息密文

2.系统后台：实现对信息的加密和解密，包括 SM9 加解密算法和 SM2 加解密算法，向数据库初入和获取密文信息。

## ①SM2 加密解密算法实现，主要针对登陆密码进行加密，如图 2.8 所示。



```

4
5 SM2_PRIVATE_KEY = '00B9AB0B828FF68872F21A837FC303668428DEA11DCD1B24429D0C99E24EED83D5'
6 SM2_PUBLIC_KEY = 'B9C9A6E04E9C91F7BA880429273747D7EF5DDEB0BB2FF6317EB00BEF331A83081A6994B8993F3F5D6EADDD8B1872266C87C018FB416'
7
8 sm2_crypt = sm2.CryptSM2(public_key=SM2_PUBLIC_KEY, private_key=SM2_PRIVATE_KEY)
9
10 # sm2加密
11 def encryptSm2(info):
12     encode_info = sm2_crypt.encrypt(info.encode(encoding="utf-8"))
13     encode_info = b64encode(encode_info).decode() # 将二进制bytes通过base64编码
14     return encode_info
15
16 # sm2解密
17 def decryptSm2(info):
18     decode_info = b64decode(info.encode()) # 通过base64解码成二进制bytes
19     decode_info = sm2_crypt.decrypt(decode_info).decode(encoding="utf-8")
20     return decode_info
21
22 if __name__ == '__main__':
23     c = decryptSm2("pYVpL/E52wyutSMNGpt/6drTKI3ciJHi+CiLJ8Mt8mWPbbj510f4v8/egBt74emUjUoou85c0JWI7kce0YHUmKkXjXCxo6ZYbpSWmq69")
24     print(c)
    
```

图 2.8 SM2 算法实现

②SM9 加密解密算法实现，对医生个人信息、科室信息、公告信息等进行加密，如图 2.9 所示。

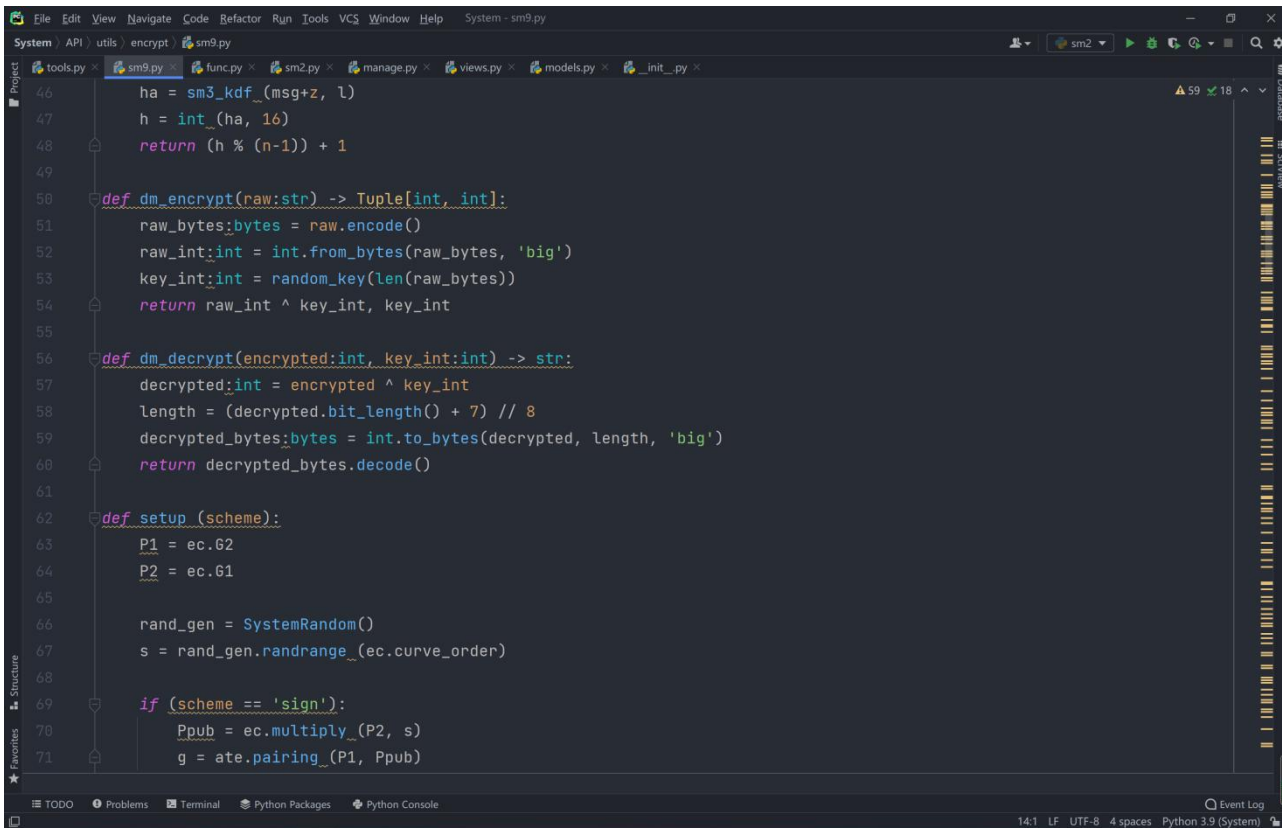


图 2.9 SM9 算法实现

③程序 views.py 用来实现系统路由，当系统访问指定路径时，程序框架显示页面，实现代码如图 2.10，2.11，2.12 所示。

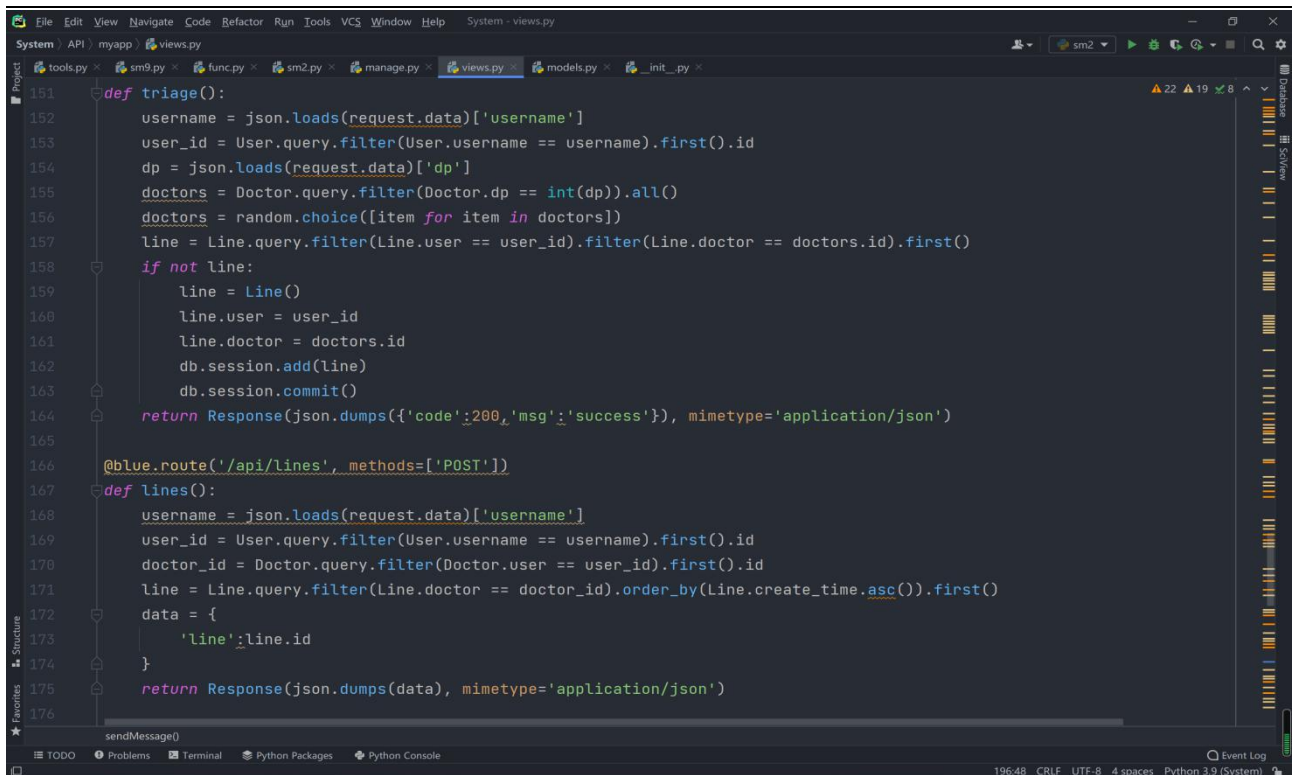


图 2.10 病患医生建立关系实现

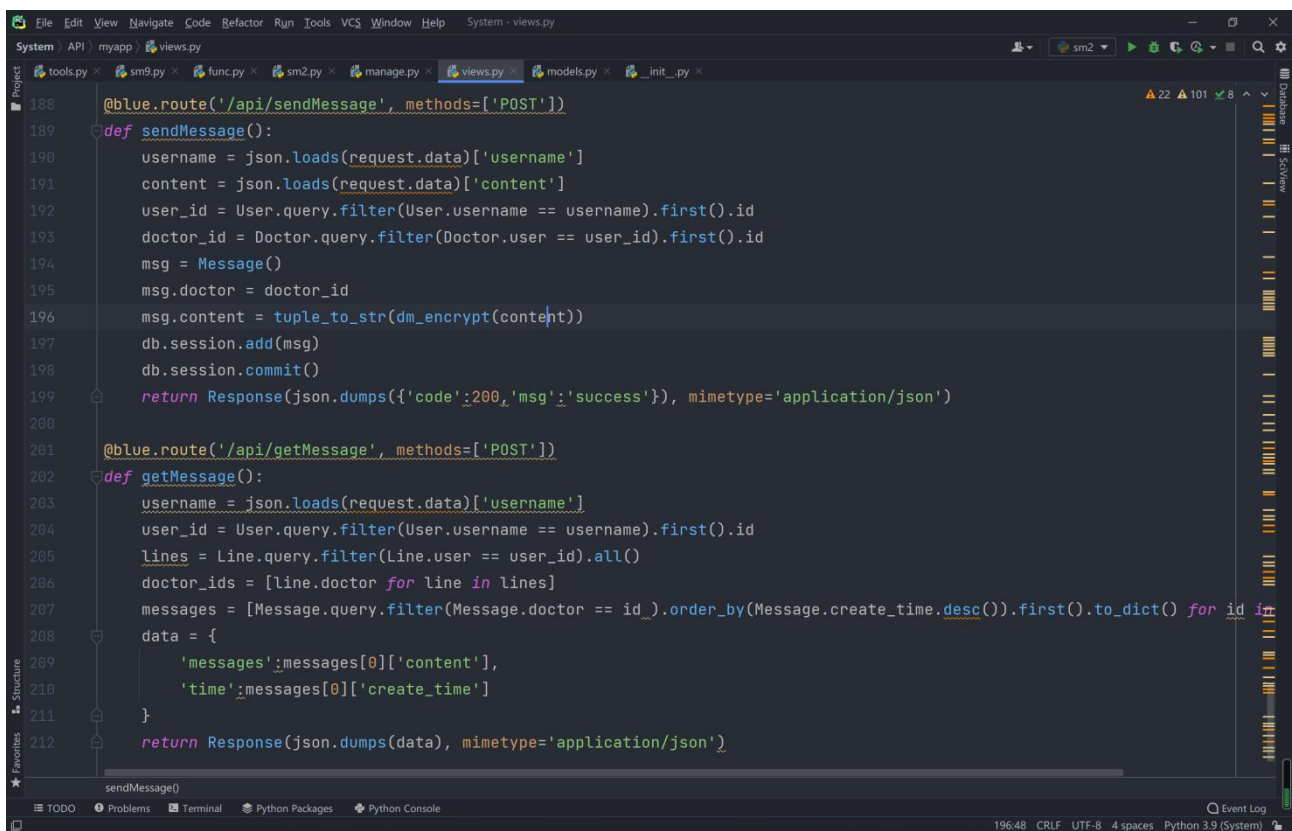


图 2.11 信息发送接收



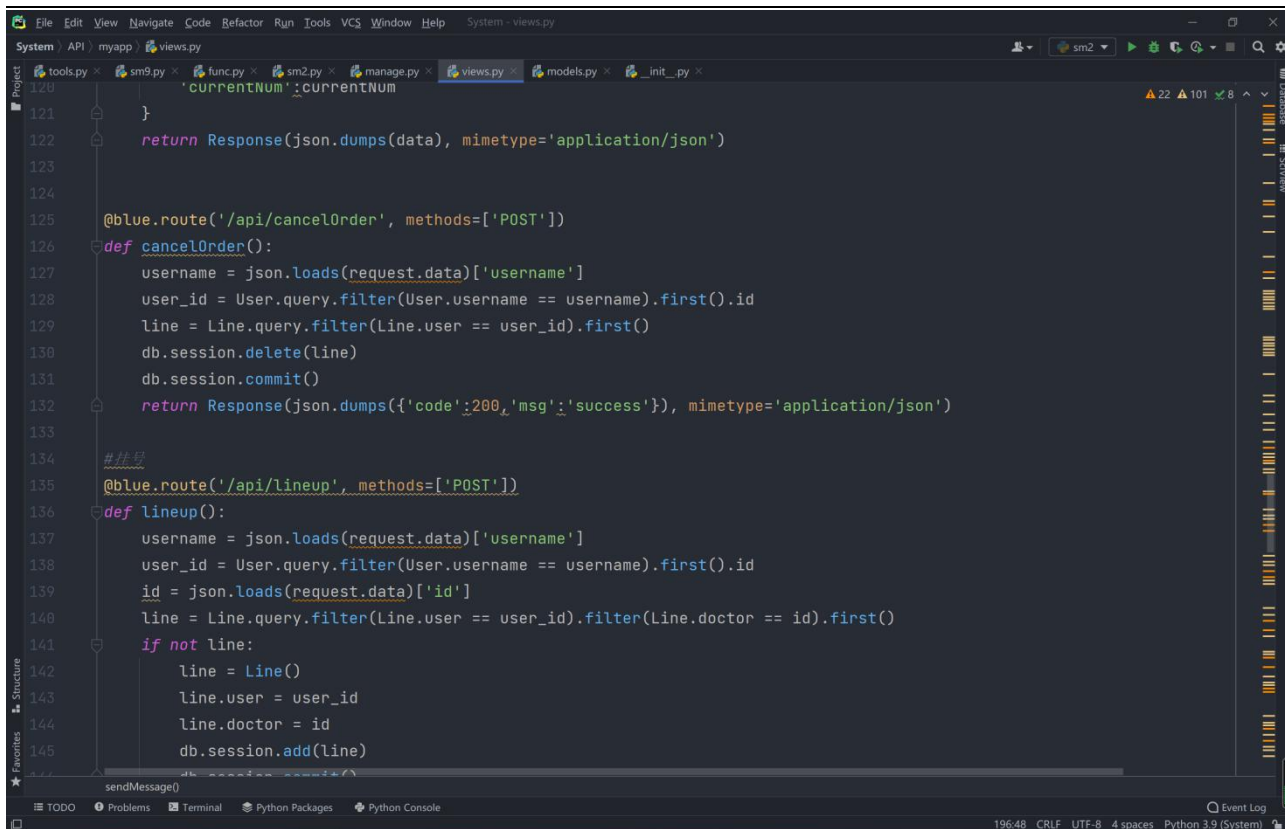


图 2.12 取消预约功能实现

④程序 models.py 用来将系统与数据库之间建立联系，方便系统在数据库中的数据读写，具体代码如图 2.13 所示。

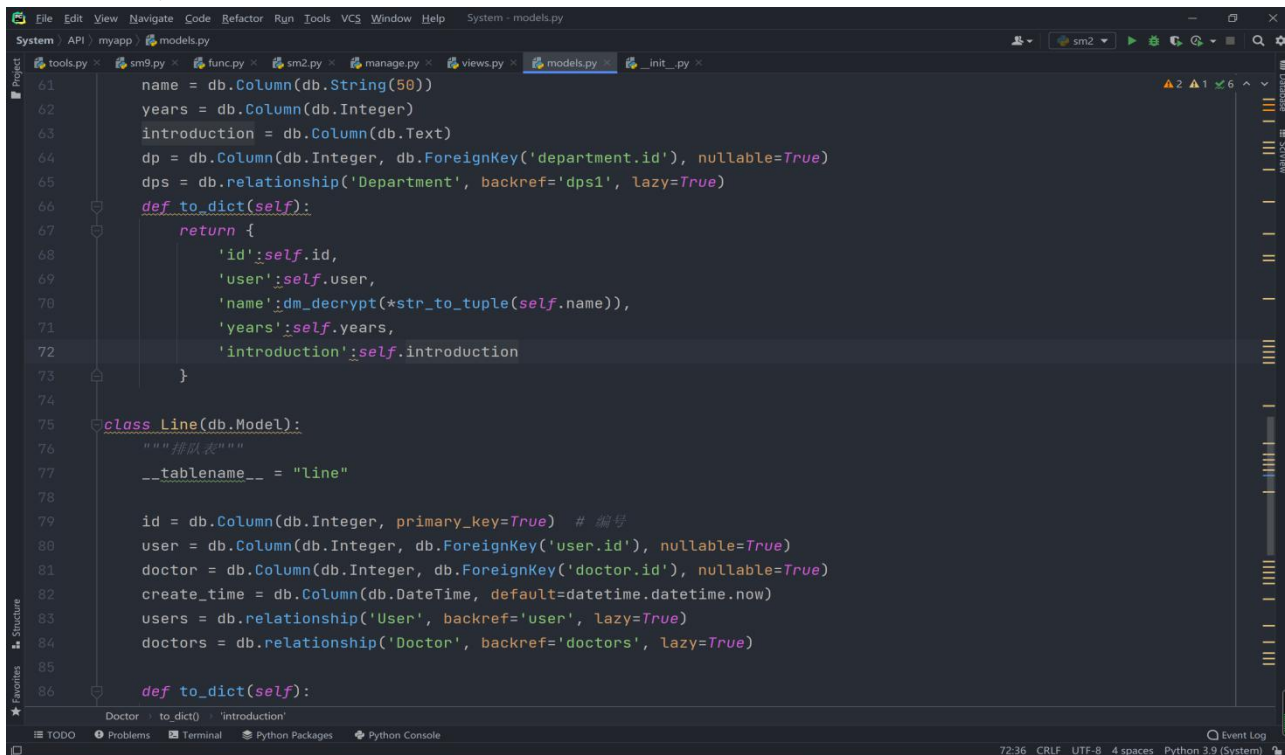


图 2.13 models 模板实现

### 3. 医生：输入个人相关信息，如图 2.14 所示

```
class Doctor(db.Model):
    """ 医生表 """
    __tablename__ = "doctor"

    id = db.Column(db.Integer, primary_key=True) #编号
    user = db.Column(db.Integer, db.ForeignKey('user.id'), nullable=True)
    name = db.Column(db.String(50))
    years = db.Column(db.Integer)
    introduction = db.Column(db.Text)
    dp = db.Column(db.Integer, db.ForeignKey('department.id'), nullable=True)
    dps = db.relationship('Department', backref='dps1', lazy=True)

    def to_dict(self):
        return {
            'id': self.id,
            'user': self.user,
            'name': dm_decrypt(*str_to_tuple(self.name)),
            'years': self.years,
            'introduction': self.introduction
        }
```

图 2.14 输入医生个人信息

### 4. 就医人员：进行注册、登录，输入个人信息，如图 2.15 所示

```
class User(db.Model):  
    """用户表"""  
    __tablename__ = "user"  
  
    id = db.Column(db.Integer, primary_key=True) #编号  
    username = db.Column(db.String(500), unique=True) #用户名  
    password = db.Column(db.String(500)) #密码  
    user_type = db.Column(db.Integer)  
  
    def verify(self, pwd):  
        try:  
            if decryptSm2(self.password) == pwd:  
                return True  
            else:  
                return False  
        except:  
            return False
```

图 2.15 注册用户信息

## 2.3 运行结果

1. 就医人员进入系统进行注册，如图 2.16 所示

The screenshot shows a web application titled "智慧医疗分诊系统" (Smart Medical Triage System). The main content area is titled "用户注册" (User Registration). It contains three input fields: "用户名" (Username), "密码" (Password), and "确认密码" (Confirm Password). Below these fields is a link "去登陆" (Go Login) and a green button labeled "注册" (Register).

图 2.16 注册界面

2. 就医人员注册成功后进行登录，如图 2.17 所示

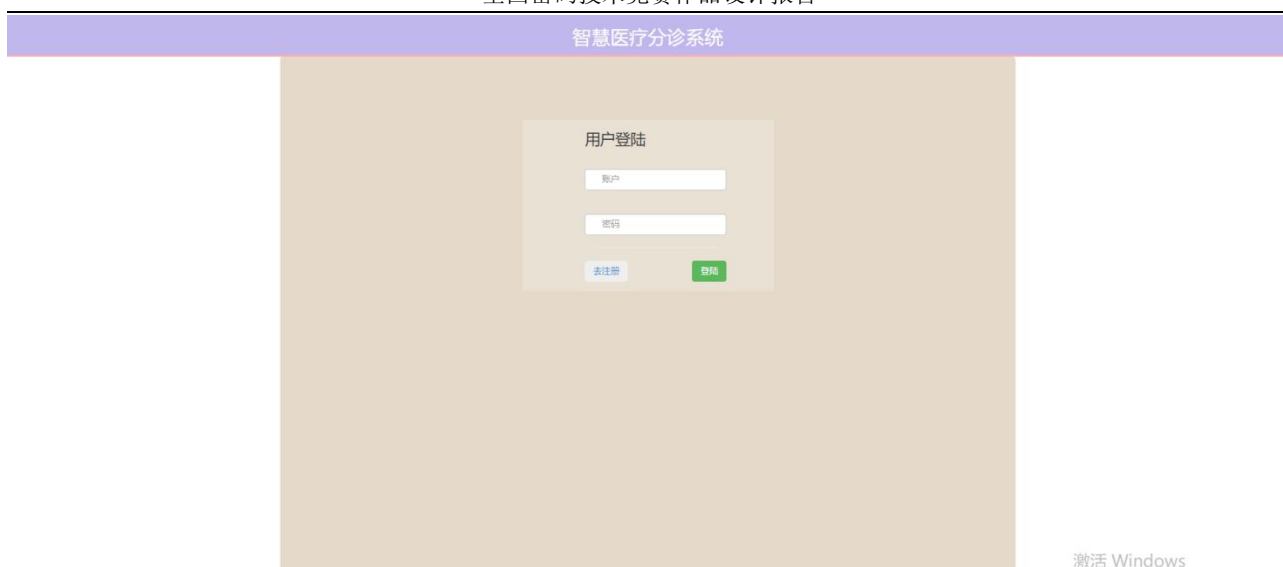


图 2.17 登陆界面

3.登陆后，进入用户选择页面，可以进行预约挂号，查看已经预约的信息和科室消息通知，如图 2.18 所示

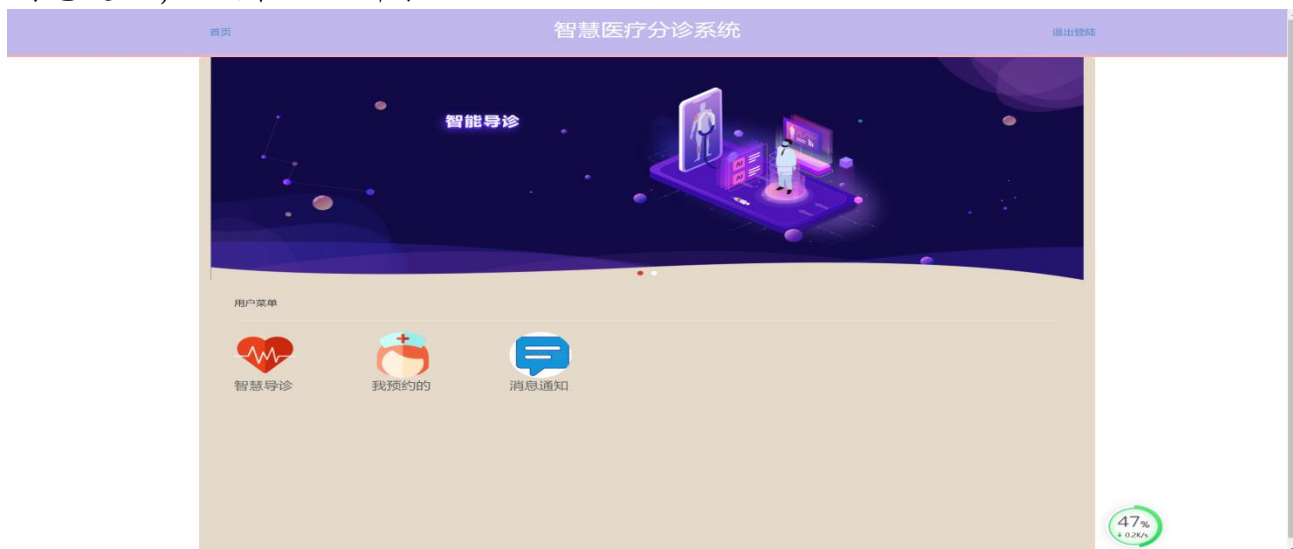


图 2.18 用户选择界面

4.选择智慧导诊后进入根据人体平面图界面，可以点击疼痛部位，查看所属科室，选择诊室挂号，帮助人们进行诊室的选择，提供分诊功能，如图 2.19 所示



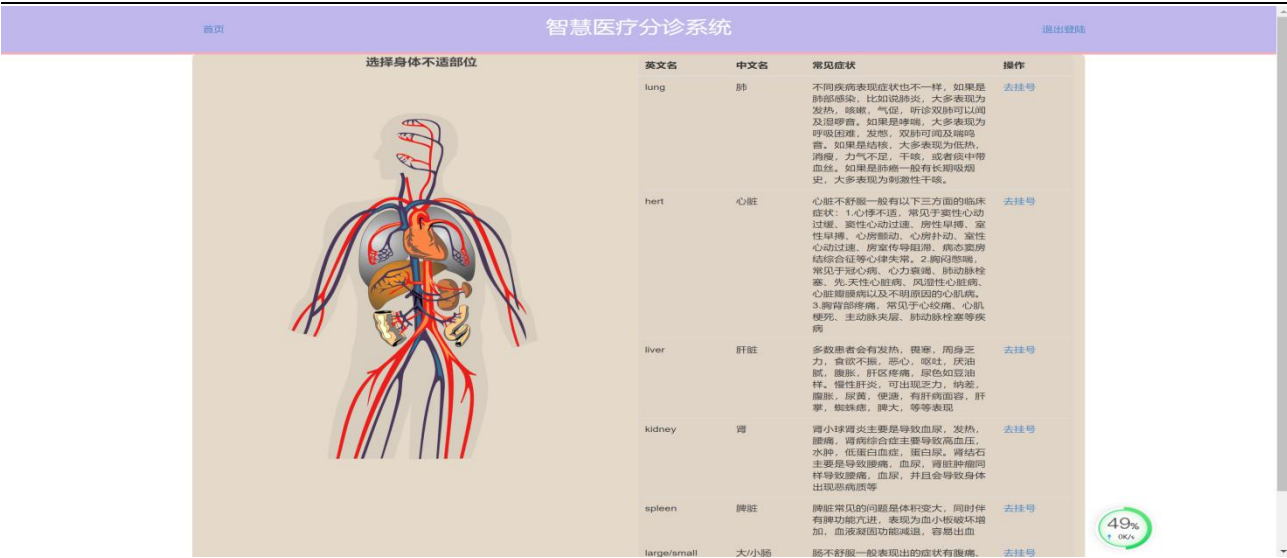


图 2.19 选择诊室界面

5.选择诊室后，可查看所属科室的医生信息，如：医生姓名、医龄、当前预约人数、医生简介等。若就医人员对医生无要求，可点击智能分诊，系统可自动选择医生并进行预约挂号，如图 2.20 所示



图 2.20 查看医生信息

6.预约成功界面，如图 2.21 所示

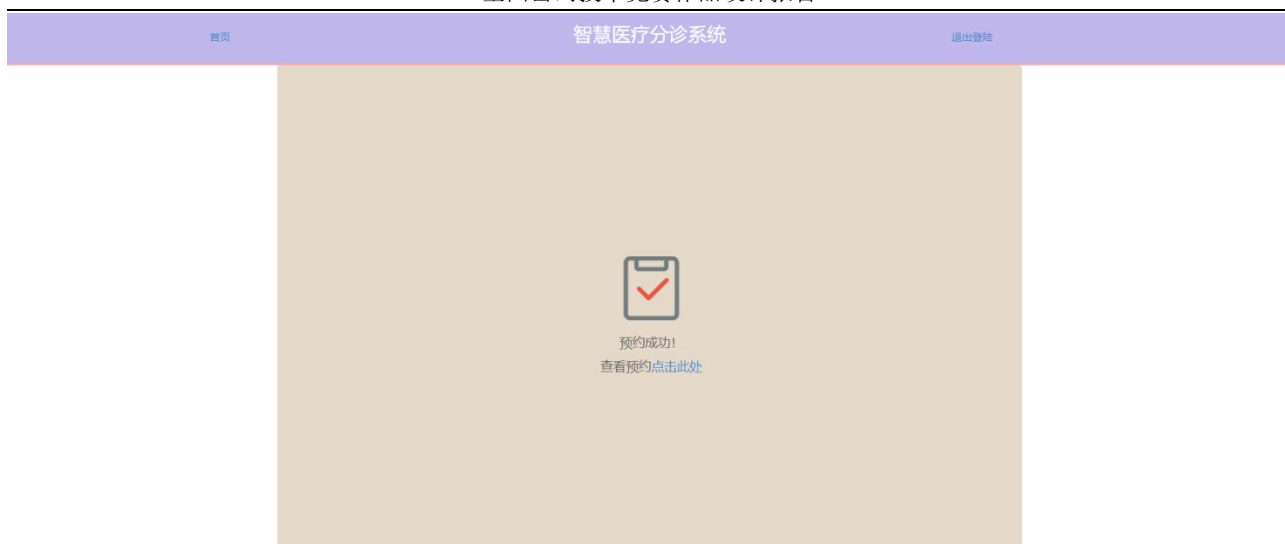


图 2.21 预约成功页面

7.进入个人选择页面，查看已经预约的信息，如图 2.2 所示



图 2.22 预约信息查看

## 2.3 技术指标

本系统的技术指标包括功能和性能两个方面，本系统需实现的功能主要包含以下三个方面：

- (1) 病人账号密钥加密。
- (2) 医生关键信息隐私保护。
- (3) 智慧导诊方便病人快速预约。

本系统的性能主要体现在以下两个方面：

- (1) 能够抵抗攻击者对医疗数据的篡改与伪造。

(2) 能够有效保护医生和病人的隐私。

3.系统测试与结果



图 3.1 系统运行结果图

3.1 测试方案

(1) 测试资源与测试环境

表 3.1 硬件配置

关键项	数量	性能要求	期望到位阶段
测试 PC 机	一台	I7，内存 16G	实现系统所有功能

表 3.2 软件配置

资源名称/类型	配置
操作系统环境	Windows10 64 位
测试方法	手工测试
软件运行工具	Pycharm, node.js,mysql

## 3.2 功能测试

表 3.3 功能测试表

测试范围	验证数据是否正确，数据类型、业务功能等相关方面是 否正确
测试目标	核实所有功能均已正常实现，即是否与需求一致
采用技术	黑盒测试，Debug 调试
工具与方法	手工测试
开始标准	开发阶段对应的功能完成并且测试用例设计完成
完成标准	测试用例通过并且最高级缺陷全部解决

## 3.3 性能测试

表 3.4 性能测试表

测试范围	检索时间与非法攻击
测试目标	检索速度毫秒级以上，可以抵抗惟密文攻击
采用技术	Debug 调试与黑盒测试
工具与方法	手工测试
开始标准	功能测试完成
完成标准	检索数据毫秒级反应以及无安全漏洞
测试重点与优先级	根据实际需求设定
需考虑的特殊事项	根据实际需求设定

## 3.4 测试数据与结果

表 3.5 测试数据与结果

用例编号	病痛部位	操作步骤	预期输出	执行时间
Case01	肺	点击智慧分诊	成功预约	27ms
Case02	心脏	点击智慧分诊	成功预约	23ms

Case03	肺	点击叫号	成功叫号	25ms
Case04	心脏	点击叫号	成功叫号	30ms

## 4.应用前景

2019年3月21日，国家卫生健康委员会就信息化质控与智慧医院建设工作有关情况举行了专场发布会。重点开展以下几方面工作：一是以电子病历为核心推动医疗机构信息化建设。二是落实《国务院办公厅关于促进“互联网+医疗健康”发展的意见》，实施进一步改善医疗服务行动计划，运用信息化手段解决人民群众看病就医过程当中的“难点”、“堵点”问题。三是加强智慧医院建设，出台医院智慧服务分级评估标准体系，推动医院运用智能化、信息化手段，提高医疗质量和效率，提升精细化、信息化管理水平。四是应用信息化手段加强医疗管理。

展望国内外医疗卫生事业的发展前景，信息化、智能化已经成为医疗技术的发展趋势。在国内，医疗卫生事业的信息化建设已经成为新一轮医疗体制改革的重要方面，并且对促进经济转型发挥了积极作用。智慧医疗分诊系统基于B/S架构采用Python语言进行系统开发。借助数字化、可视化模式，实现对病人账户以及医生个人敏感信息的保护，并开发了智慧分诊以及医生叫号的功能，将有限的医疗资源让更多人共享。

## 5.结论

本项目目的在于设计一种基于结合国密算法SM2和SM9的智慧医疗分诊系统，为人们看病就医提供更加方便安全的网上预约挂号。

在一个月的项目开发中，经过了概要设计、详细设计、编码、测试后，本项目已经实现并满足以下功能：

1. 可以对整个系统提供的医疗数据进行加密存储在数据库中。
2. 对医生的关键敏感信息进行加密保护。
3. 防止数据提供者或者网络爬虫检索医疗数据。

通过 SM2 和 SM9 国密算法加密,数据库中的数据可以防止极大程度的唯密文攻击,在如此繁琐的加密和解密的模式下,该系统的功能与时间效率仍然不受影响,检索速度可达毫秒及以上。

通过 SM2 和 SM9 国密算法加密,数据库中的数据可以防止极大程度的唯密文攻击,在如此繁琐的加密和解密的模式下,该系统的功能与时间效率仍然不受影响,检索速度可达毫秒级。