

# “守沪者”——基于公众参与的城市治安协同治理系统

数据库系统原理课程设计报告



学号 \_\_\_\_\_

姓名 \_\_\_\_\_

专业      计算机科学与技术

授课老师      李文根

## 目录

一. 概述 .....	4
1. 1 课题背景 .....	4
1. 2 编写目的 .....	4
1. 2. 1 公众与警务端的实时互联 .....	4
1. 2. 2 治安案件全流程监测与查询 .....	5
1. 2. 3 警力协同调度与精准响应 .....	5
1. 3 用户需求 .....	5
1. 3. 1 管理员用户 .....	5
1. 3. 2 普通用户 .....	6
二. 需求分析 .....	7
2. 1 功能需求 .....	7
2. 1. 1 案件管理模块 .....	7
2. 1. 2 账户管理模块 .....	7
2. 1. 3 个人信息管理模块 .....	8
2. 1. 4 警员管理模块（管理员端） .....	8
2. 2 数据字典 .....	9
2. 2. 1 数据项定义 .....	9
2. 2. 2 数据结构定义 .....	10
2. 2. 3 数据存储定义 .....	11
2. 3 数据流图 .....	12
三. 可行性分析 .....	12
3. 1 技术可行性 .....	12
3. 2 经济可行性 .....	12
3. 3 运营可行性 .....	13
四. 概念设计 .....	13
4. 1 实体 .....	13
4. 2 实体属性局部 E-R 图 .....	14
4. 2. 1 登录账户实体 .....	14
4. 2. 2 案件实体 .....	14
4. 2. 3 警员实体 .....	15
4. 2. 4 涉案人员实体 .....	15
4. 2. 5 地理位置实体 .....	16
4. 2. 6 涉案记录实体 (Case_Persons) .....	17
4. 2. 7 办案记录实体 (CaseOfficers) .....	17
4. 2. 8 用户角色关联实体 (User_Roles) .....	18
4. 3 全局 E-R 图 .....	18
五. 逻辑设计 .....	18
5. 1 E-R 图向关系模型的转变 .....	18
5. 1. 1 地理位置信息表 (Locations) .....	19
5. 1. 2 人员基础信息表 (Persons) .....	20
5. 1. 3 警员信息表 (Officers) .....	20

5.1.4 案件主表 (Cases) .....	21
5.1.5 案件与涉案人员关联表 (Case_Persons) .....	22
5.1.6 案件与警员关联表 (Case_Officers) .....	22
5.1.7 系统登录账户表 (user_accounts) .....	22
5.1.8 角色表 (Roles) .....	23
5.1.9 权限表 (Permissions) .....	23
5.1.10 用户与角色关联表 (User_Roles) .....	23
5.1.11 角色与权限关联表 (Role_Permissions) .....	23
5.2 数据模型的优化及规范化设计 .....	24
5.2.1 规范化理论基础 .....	24
5.2.2 关系模式的规范化分析 .....	24
5.2.3 数据模型的优化策略 .....	25
六. 项目管理 .....	25
6.1 框架选择 .....	25
6.1.1 前端框架 .....	26
6.1.2 后端框架 .....	26
6.2 开发平台 .....	27
七. 系统实现 .....	28
7.1 系统架构搭建 .....	28
7.2 系统逻辑设计 .....	29
7.3 具体功能编写 .....	29
7.3.1 登录注册模块 .....	29
7.3.2 用户信息管理模块 .....	31
7.3.3 案件上报与追踪模块 (市民端) .....	33
7.3.4 案件处置与调度模块 (警务端) .....	35
7.4 功能测试 .....	38
7.4.1 案件创建 .....	39
7.4.2 案件信息更新 .....	39
7.4.3 案件删除 .....	40
7.4.4 用户登录验证 .....	41
7.4.5 用户注册 .....	41
7.4.6 用户信息修改 .....	42
7.4.7 文件上传 (案件附件) .....	43
7.4.8 权限控制测试 .....	43
7.4.9 案件搜索 .....	44
八. 总结 .....	45

## 摘要

随着城市化进程的加速，上海作为国际化大都市，其城市治安治理面临着日益复杂的挑战。传统的治安管理模式往往依赖于警务人员的单向巡查与处置，存在信息更新滞后、警力资源分配不均、公众参与度低等问题，难以满足现代超大城市精细化治理的需求。为了解决上述痛点，响应“人民城市人民建”的号召，本课程设计基于数据库系统原理，设计并研发了“守沪者”——基于公众参与的城市治安协同治理系统。该系统旨在构建一个警民联动的数字化平台，通过移动互联网技术连接普通市民与警务部门。系统核心功能涵盖了案件的多维上报（支持地理定位、多媒体上传）、案件的全生命周期管理（创建、分配、更新、结案）、警力资源的数字化管理以及个人信息安全维护。在技术实现上，本项目采用前后端分离架构，后端基于 Java Spring Boot 框架，前端采用 Vue.js 框架，数据库选用 MySQL 进行持久化存储。本报告详细阐述了从需求分析、概念结构设计、逻辑结构设计到物理结构设计及系统实现的完整流程。重点探讨了实体关系模型（E-R 图）的构建、数据库规范化理论在表结构优化中的应用，以及索引设计对查询性能的提升。通过本系统的实施，不仅有效提升了治安事件的响应速度和处置透明度，也为构建共建、共治、共享的社会治理格局提供了有力的技术支撑。

**关键词：**数据库系统设计；城市治安；协同治理

## 一. 概述

### 1.1 课题背景

近年来，随着“智慧城市”和“平安中国”战略的深入推进，利用大数据、云计算等新一代信息技术提升警务效能已成为行业共识。在当今信息化高度发达的社会，犯罪手段日益高科技化、隐蔽化，这就要求公安机关必须具备快速反应、精准打击和全流程信息管控的能力。

当前，基层警务工作普遍面临警力资源紧张与案件数量激增的矛盾。一线民警在接处警、案件录入、侦查协作等环节消耗了大量精力，若缺乏高效的信息系统支撑，极易出现信息录入滞后、案件线索遗漏、协同作战受阻等情况。此外，公众对于公共安全服务的期待也在不断提升，普通市民希望拥有更便捷的报案渠道和更透明的案件处理进度查询机制。上海作为特大型城市，人口流动性大、治安要素复杂。然而，目前的治安管理系统中，公众参与的渠道相对有限，且数据流转主要集中在公安内部网，缺乏一个面向社会公众、能够实时交互的轻量化协同平台。

在此背景下，开发一套集案件上报、警力调度、涉案人员管理、数据分析于一体的“守沪者”治安协同治理系统具有重要的现实意义。该系统旨在通过现代数据库技术和 Web 开发技术，构建一个统一、规范、高效的治安协同治理系统，实现案件全生命周期的数字化管理。它不仅仅是提升警务效能的技术手段，更是连接警民的纽带，通过数据的实时流动，实现治安隐患的快速发现与精准处置。

### 1.2 编写目的

#### 1.2.1 公众与警务端的实时互联

公众端与警务管理端的实时连接是“守沪者”治安协同治理系统的核心功能之一。通过建立这一连接，系统能够打破传统治安管理中信息传递滞后的壁垒，实时获取来自普通市民的第一手现场数据。市民通过移动端与系统的通信，可以上传案发地点的精确定位、现场照片及视频等多媒体证据，从而使警务部门能够准确把握治安事件的实时动态。这一连接不仅

令警务部门具备了更全面、透彻的“千里眼”观测能力，也为后续的案件定性、警力分配提供了详实、实时的底层数据支持。

### 1.2.2 治安案件全流程监测与查询

治安案件的全流程监测与查询功能一方面使得警务部门能够即时发布案件的处理进度与结果，另一方面同时允许报案市民灵活地查询与追踪案件状态。警务人员更新的案件状态（如已受理、侦查中、已结案）通过数据库实时同步，并向相关用户推送反馈信息。这一功能保障了市民能够在第一时间获取到自己所报案件的最新进展，有效提高了治安治理的透明度与公信力。用户可以通过系统进行灵活的历史记录查询，满足不同用户对个人安全信息的管理需求，为公众提供及时、闭环的治安信息服务。

### 1.2.3 警力协同调度与精准响应

警力协同调度与精准响应功能通过数据库基础的“区域-部门-警员”关联关系，实现了高效的治安资源配置。当系统接收到新的报案请求时，将根据案发地点的行政区划与辖区警员的关联关系，将案件信息精准推送至对应的负责部门或警员终端。每个案件存储了案发地的地理坐标，而城区与警务部门的多对多或一对多关系确保了派警指令能够准确、迅速地传递给辖区内的执法力量。这一功能能够有效覆盖城市各个网格区域，从而提高接处警的响应速度和实效性。通过精准的信息分发，系统为城市安全管理提供了强有力的技术支撑，使潜在的治安隐患得到及时处置，从而最大程度降低对社会秩序的影响。

## 1.3 用户需求

### 1.3.1 管理员用户

管理员用户主要由警务部门的指挥调度人员或系统后台运维人员担任，他们拥有对系统数据的最高管理权限，负责案件的流转控制与警务资源的配置。针对管理员用户，系统需提供以下核心功能：

- **案件接报与处置：** 管理员需具备对案件进行立案审查与全流程跟踪的能力。其中包括依据市民上报的信息进行“案件创建”，确立案件档案；随着调查深入，执行“案件更

新”操作，实时修改案件状态（如从“待受理”流转至“侦查中”），并对案件基础信息进行修正。此外，系统需支持管理员将案件与具体的“关联警员”及“关联涉案人员”进行绑定，实现“案人关联”，确保案件侦办过程的责任落实与轨迹清晰。

- 警务人员管理：为保障执法力量的数字化管理，系统需提供警员信息的维护功能。管理员可通过“新增警员”模块，录入新入职警员的详细档案，包括警员编号、身份证号、所属部门及职位等关键履历信息。同时，支持对现有警员信息的更新与维护，确保警力数据库的鲜活准确，为突发事件下的警力快速调度提供数据支撑。

### 1.3.2 普通用户

普通用户是“守沪者”治安协同治理系统的广泛参与群体，即广大的上海市民。他们是城市治安的“眼睛”，需要一个便捷、安全的渠道来上报身边的治安隐患，并期望获得警方的及时响应。对于广大市民群众，系统需提供以下功能：

- 多维度的案件上报：市民用户可在平台上发起案件上报请求，实现治安线索的数字化提交。用户需要能够灵活选择案件类型，填写详细的情况描述，并利用系统的定位服务自动或手动确认案发地点和行政区划。为了提高案情描述的准确性，平台还需支持多媒体上传功能，允许用户上传现场照片或视频证据。这一功能保障了用户能够在第一时间将现场情况以富媒体形式传送给警务部门，有效降低了沟通成本，为警方快速研判案情提供了详实的数据基础。
- 个人信息管理与状态追踪：平台不仅要作为报案工具，还需成为用户与警方互动的窗口。用户需要具备修改个人信息（如联系方式、居住地址）的权限，以确保警方在必要时能与其取得联系。同时，基于协同治理的理念，用户期望获得案件处理进度的反馈。虽然 UML 图中侧重于上报，但在实际需求中，用户提交信息后，系统应能反馈案件是否已被受理或处理完毕，这种双向的信息交互能增强市民参与社会治理的积极性与信任感。

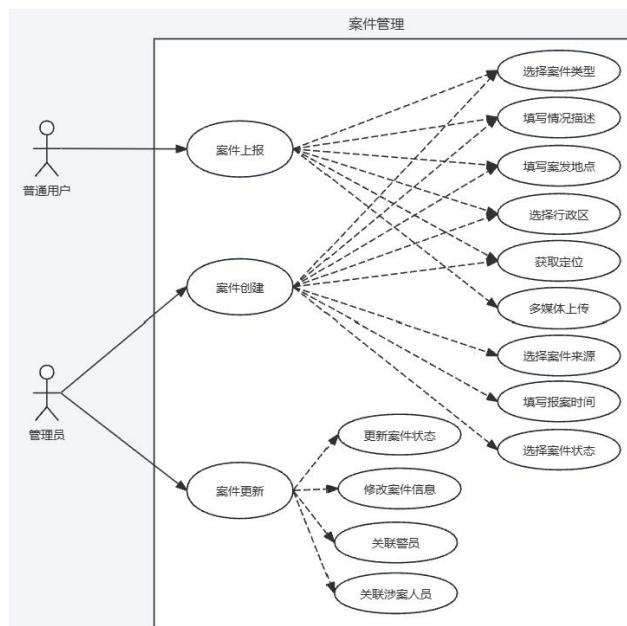
## 二. 需求分析

### 2.1 功能需求

根据前期的调研与 UML 用例图分析，系统主要包含以下四大功能模块：

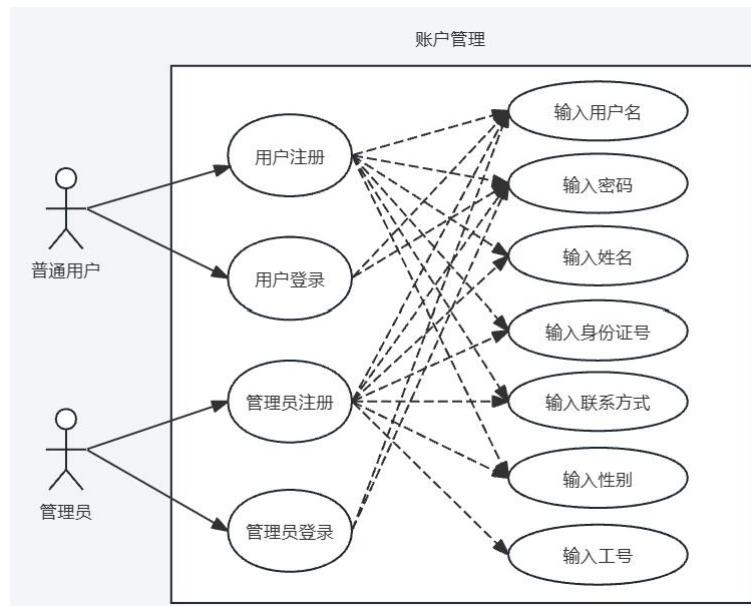
#### 2.1.1 案件管理模块

- 案件上报（普通用户）： 支持选择案件类型（如盗窃、纠纷、诈骗等）、填写详细情况描述、自动/手动填写案发地点及行政区、上传多媒体附件、记录报案时间。
- 案件创建与分配（管理员）： 审核上报信息，正式立案，并根据辖区分配警员。
- 案件更新（管理员/警员）： 更新案件状态（待处理、侦查中、已结案）、修改案件补充信息、关联具体涉案警员及涉案人员。



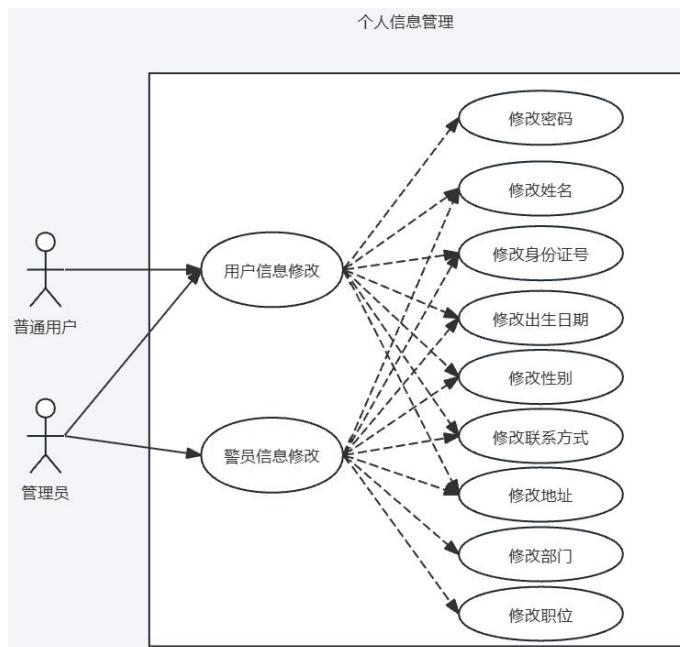
#### 2.1.2 账户管理模块

- 支持普通用户和管理员的注册与登录。
- 登录时需校验用户名（或工号）、密码及身份类型。
- 注册时需录入包括身份证号、联系方式、性别等关键身份信息，确保实名制。



### 2.1.3 个人信息管理模块

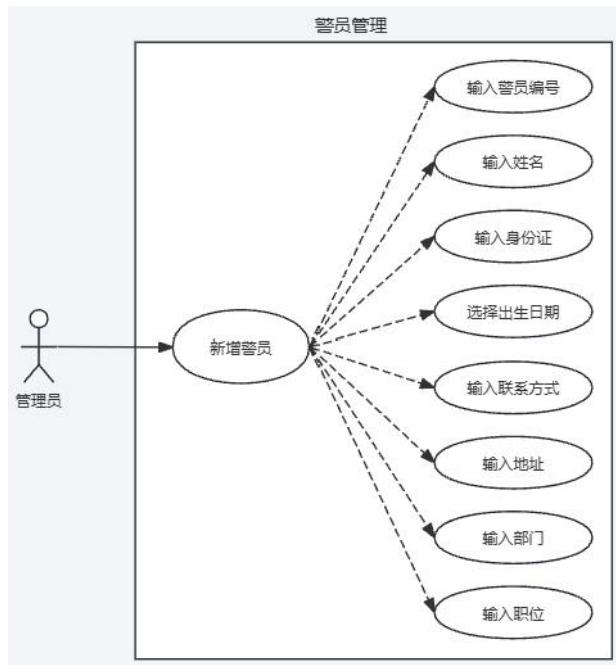
- 用户与警员均可修改各自的非敏感信息（如联系方式、地址）。
- 关键信息修改（如身份证号、部门职位）需经过特定的权限验证或由管理员操作。



### 2.1.4 警员管理模块（管理员端）

- 新增警员： 管理员可录入新警员的编号、姓名、身份证件、出生日期、联系方式、所属

部门及职位信息。



## 2.2 数据字典

### 2.2.1 数据项定义

编号	数据项名	别名	类型	长度/范围	简要说明
DI-01	用户 ID	UserID	数值型	系统自增	系统内部唯一标识用户的流水号
DI-02	登录名	Username	字符型	4-50 位	用户登录系统的唯一账号标识
DI-03	身份证号	IDCard	字符型	18 位	符合国家标准的居民身份证号码，用于实名认证
DI-04	警员编号	BadgeNum	字符型	6-10 位	公安机关颁发的唯一警号
DI-05	案件状态	Status	枚举型	-	取值：已接报、立案侦查、已告破、已归档

DI-06	案件类型	CaseType	字符型	-	如：盗窃、诈骗等
DI-07	报案时间	ReportTime	日期时间型	YYYY-MM-DD HH:MM:SS	用户提交案件的时间
DI-08	经纬度	Lat/Lon	浮点数值型	精度 8 位	案件发生的地理坐标，用于 GIS 定位
DI-09	多媒体文件	Media	文本型	-	存储图片或视频在服务器上的相对路径 (JSON 格式)
DI-10	涉案角色	Role	枚举型	-	取值：嫌疑人、受害人、证人

## 2. 2. 2 数据结构定义

### 1. 用户账户信息 (User\_Account\_Info)

- 说明： 描述系统所有登录用户的基本账户信息。
- 组成： 用户 ID + 登录名 + 密码哈希 + 用户类型（普通用户/管理员） + 激活状态 + 创建时间 + (关联警号)

### 2. 案件详细信息 (Case\_Info)

- 说明： 描述一起治安案件的完整属性。
- 组成： 案件 ID + 案件标题 + 案件类型 + 案件描述 + 发生地点信息 + 报案时间 + 当前状态 + 案件来源 + 报案人 ID + 多媒体附件

### 3. 警员档案信息 (Officer\_Profile)

- 说明： 描述警务人员的职能档案。
- 组成： 警员编号 + 姓名 + 性别 + 身份证号 + 所属部门 + 职务 + 联系电话

#### 4. 涉案人员记录 (Involved\_Person\_Record)

- 说明：描述案件中涉及的具体自然人及其在案件中的身份。
- 组成：案件 ID + 人员身份证号 + 涉案角色 + 人员姓名 + 联系方式

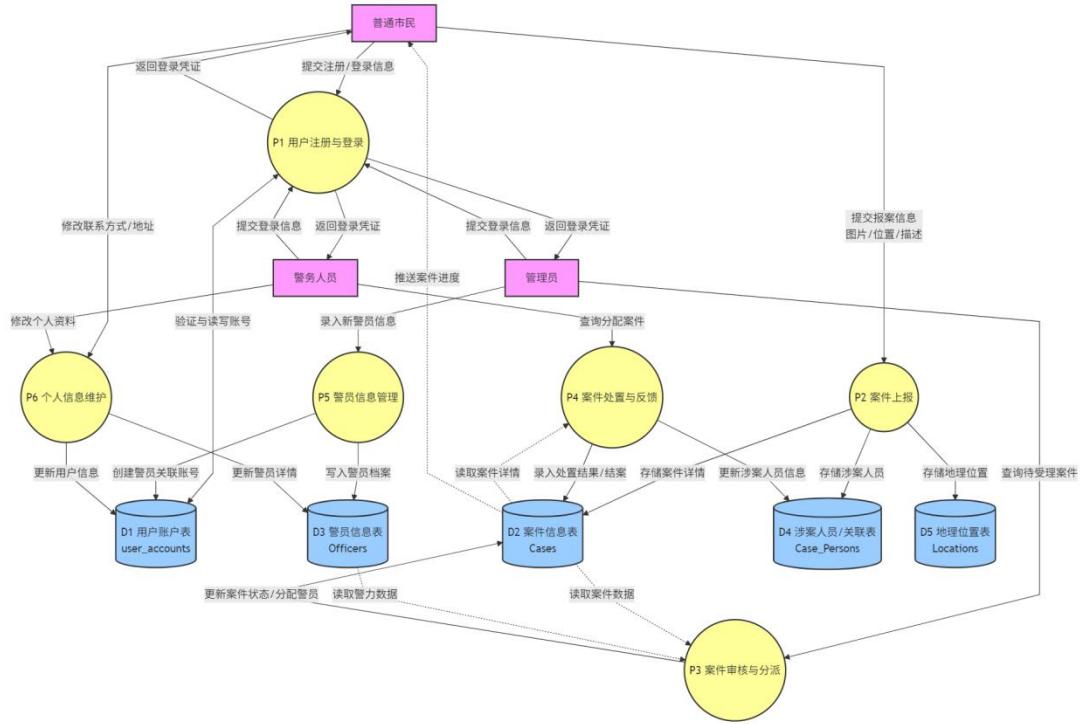
#### 5. 地理位置信息 (Location\_Info)

- 说明：描述案件发生的空间属性，用于地图服务。
- 组成：位置 ID + 行政区 + 详细街道地址 + 经度 + 纬度

### 2.2.3 数据存储定义

编号	数据存储名	说明	输入数据流	输出数据流
DS-01	用户信息库	存储所有注册用户的认证信息及实名档案	用户注册信息	用户登录验证结果、个人信息查询
DS-02	案件信息库	存储所有上报的案件及其处理进度的动态记录	报案提交信息、案件状态更新	案件列表查询、案件详情展示
DS-03	警力资源库	存储警员的档案及部门归属信息	新警员录入	警员信息验证、派警查询
DS-04	关联关系库	存储案件与人员、案件与警员的对应关系	关联操作提交	涉案人员查询、办案警员查询

## 2.3 数据流图



## 三. 可行性分析

### 3.1 技术可行性

在“守沪者”治安协同治理系统的技术方案中，整体设计清晰、逻辑简练。从治安监测与可视化角度，利用成熟的 GIS 地理信息技术（如高德地图 API）进行案发地点定位与热力图展示已经具备广泛的应用基础；从数据库系统与全栈开发角度，该工作难度适中，只需具备扎实的 Java 与前端 Vue 编程基础，结合前后端分离的开发能力便能很好地完成系统建设。通过深入学习 Spring Boot 后端框架与 MySQL 数据库优化知识，预计能够高质量实现该系统，技术上的可行性得到有效保障。

### 3.2 经济可行性

本系统的服务对象涵盖警部门与广大上海市民，数字化管理流程的引入有效提升了警

情处置的决策效率。同时，这一系统通过公众线上报案，显著降低了传统电话报警与线下笔录的人力沟通成本，为社会创造了显著的隐性经济效益。通过公众参与的“众包”模式获取治安线索使得数据采集成本大幅降低，而通过配置标准化的云服务器或本地服务器，能够以相对较低的经济成本维持系统的长期运行。因此，从经济角度看，本系统的建设具有高度可行性。

### 3.3 运营可行性

“守沪者”系统的实施不仅极大地提升了超大城市治安管理的信息化水平，更显著地提高了警务人员的办案效率和民众的参与便捷度。系统交互模块严格根据“警员”与“市民”的不同角色进行差异化智能设计，使得无论是报案群众还是基层民警都能快速上手，操作简便易行。整体系统采用轻量级架构，部署灵活且易于维护，为城市治安协同治理注入了新的技术活力，使得该数据库系统具备了长期运营和迭代维护的可行性，为上海的城市安全管理奠定了坚实的数字基础。

## 四. 概念设计

### 4.1 实体

根据前期的需求分析与业务流程梳理，本系统主要包含以下核心实体：

#### 1. 基础实体：

- 登录账户 (User Account)
- 警员 (Officer)
- 涉案人员 (Person)
- 案件 (Case)
- 地理位置 (Location)
- 角色 (Role) & 权限 (Permission)

#### 2. 关联实体（由关系转化）：

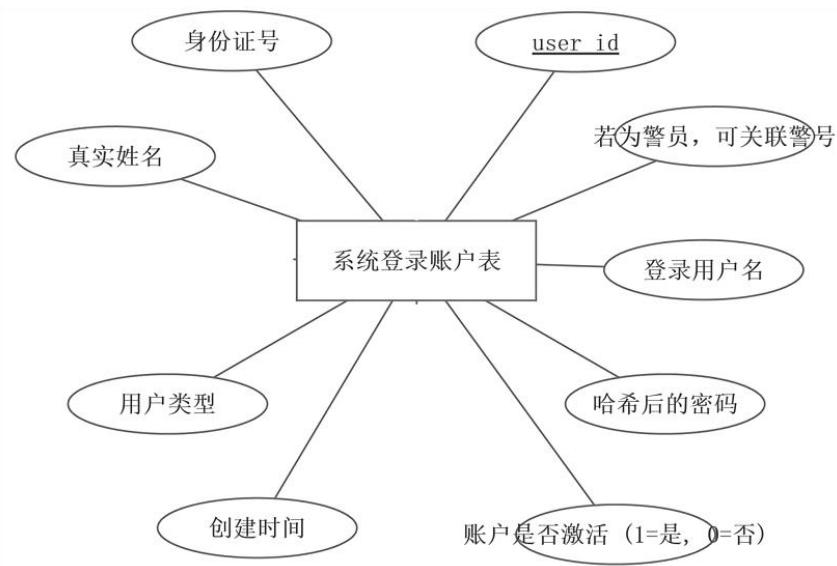
- 涉案记录 (Case-Person Link): 记录特定人员在特定案件中的角色。

- 办案记录 (Case-Officer Link): 记录警员参与的具体案件。
- 授权记录 (Role-Permission Link): 记录角色拥有的具体权限。

## 4.2 实体属性局部 E-R 图

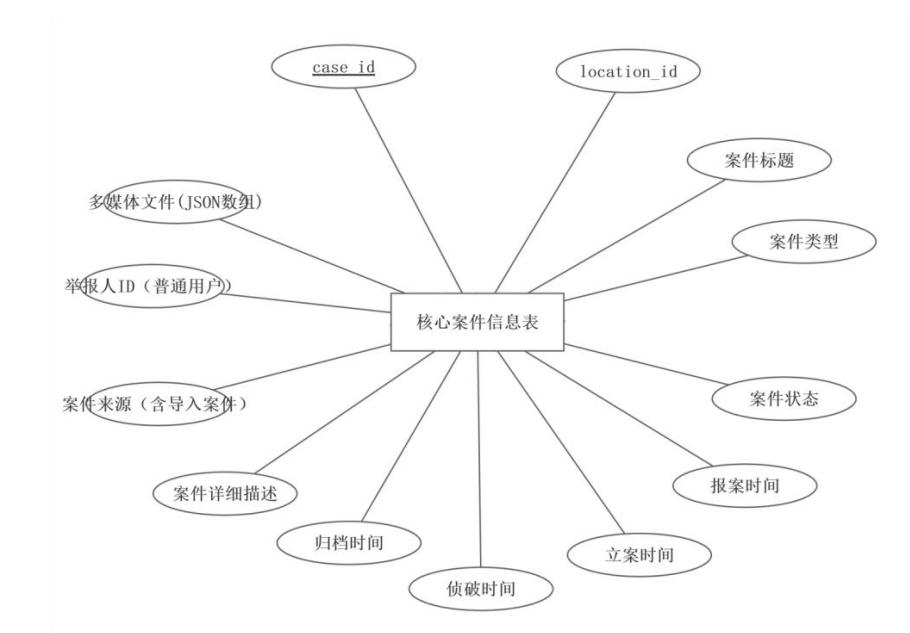
### 4.2.1 登录账户实体

登录账户实体用于管理系统的访问权限。其中 `user_id` 为唯一标识，`username` 为登录凭证。为了支持警务人员登录，该实体通过 `officer_id` 与警员实体建立关联。



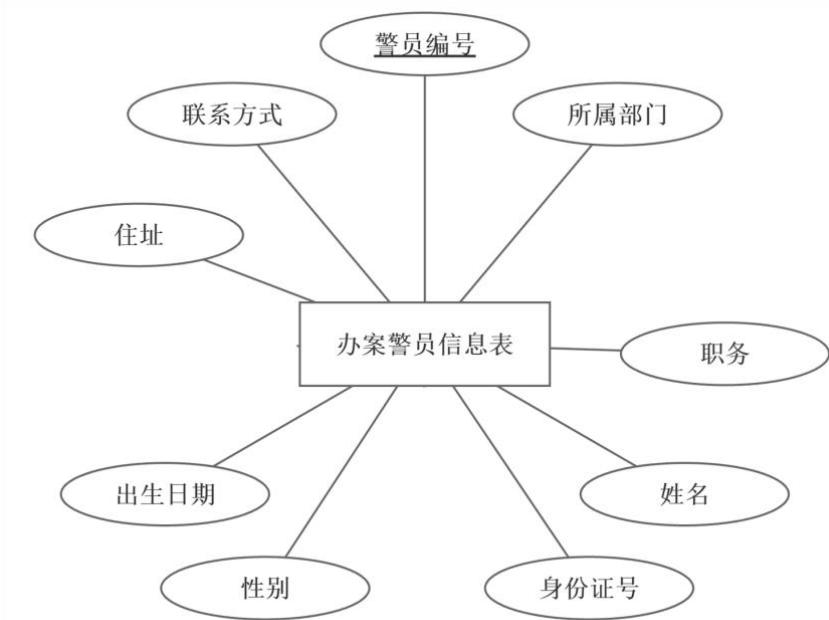
### 4.2.2 案件实体

案件实体是整个协同治理系统的核心，记录了从报案到结案的全过程信息。`case_id` 是案件的唯一流水号。为了支持数字化证据，实体中包含了 `media_files` 属性以存储多媒体路径。



#### 4.2.3 警员实体

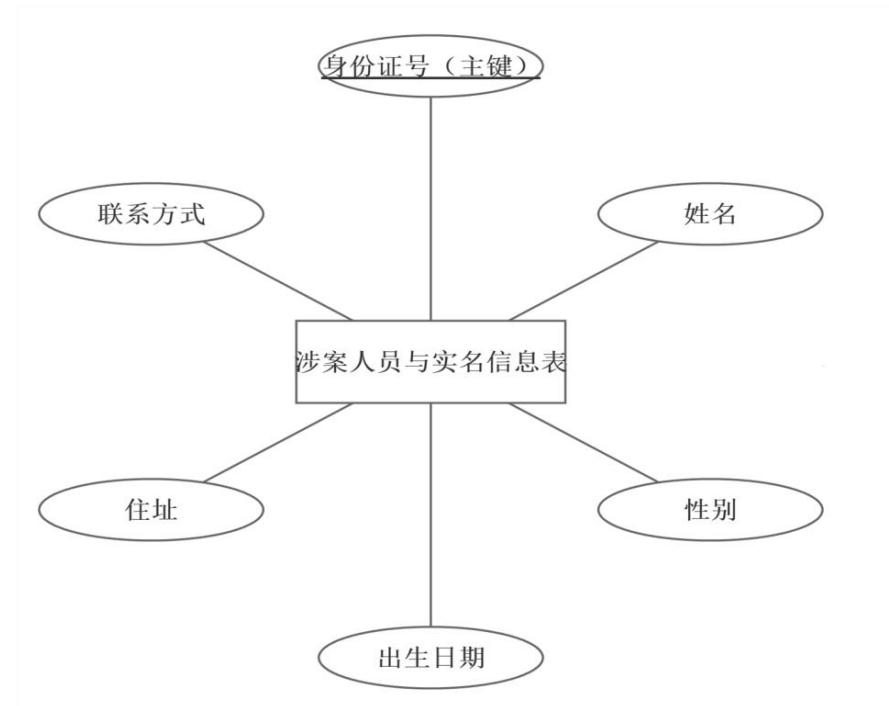
警员实体记录了执法人员的详细档案。officer\_id (警号) 作为主键，用于唯一区分每一位警员。



#### 4.2.4 涉案人员实体

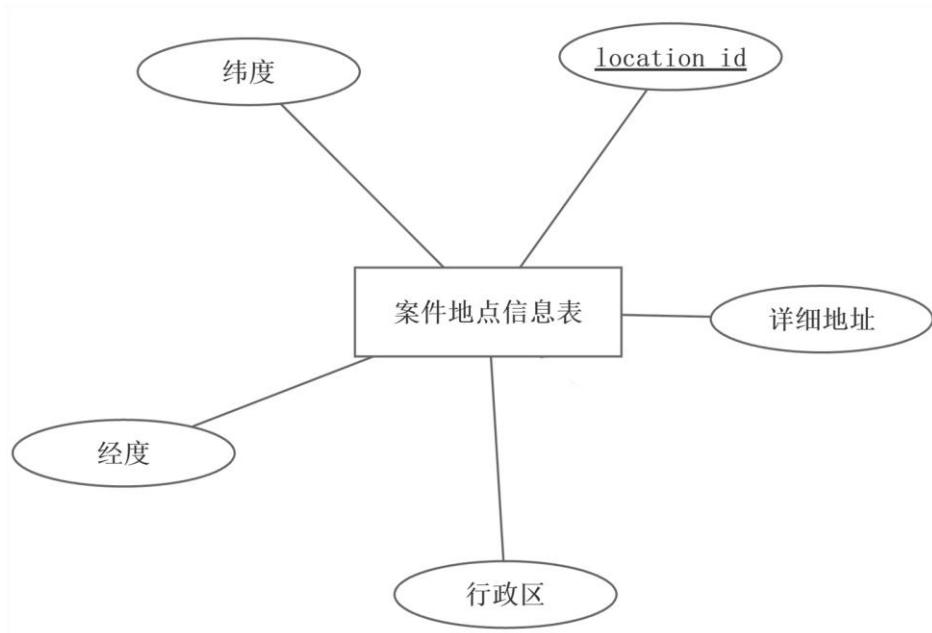
该实体通过 id\_number (身份证号) 作为主键，实现了对实名制信息的严格管理。它用

于存储案件中涉及的各类人员的基础身份信息。



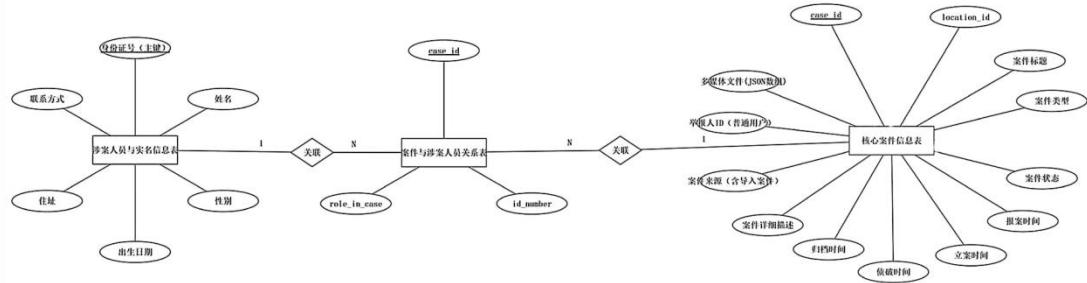
#### 4.2.5 地理位置实体

为了支持基于 GIS 的可视化管理，位置实体不仅包含文本地址，还独立存储了经纬度坐标。



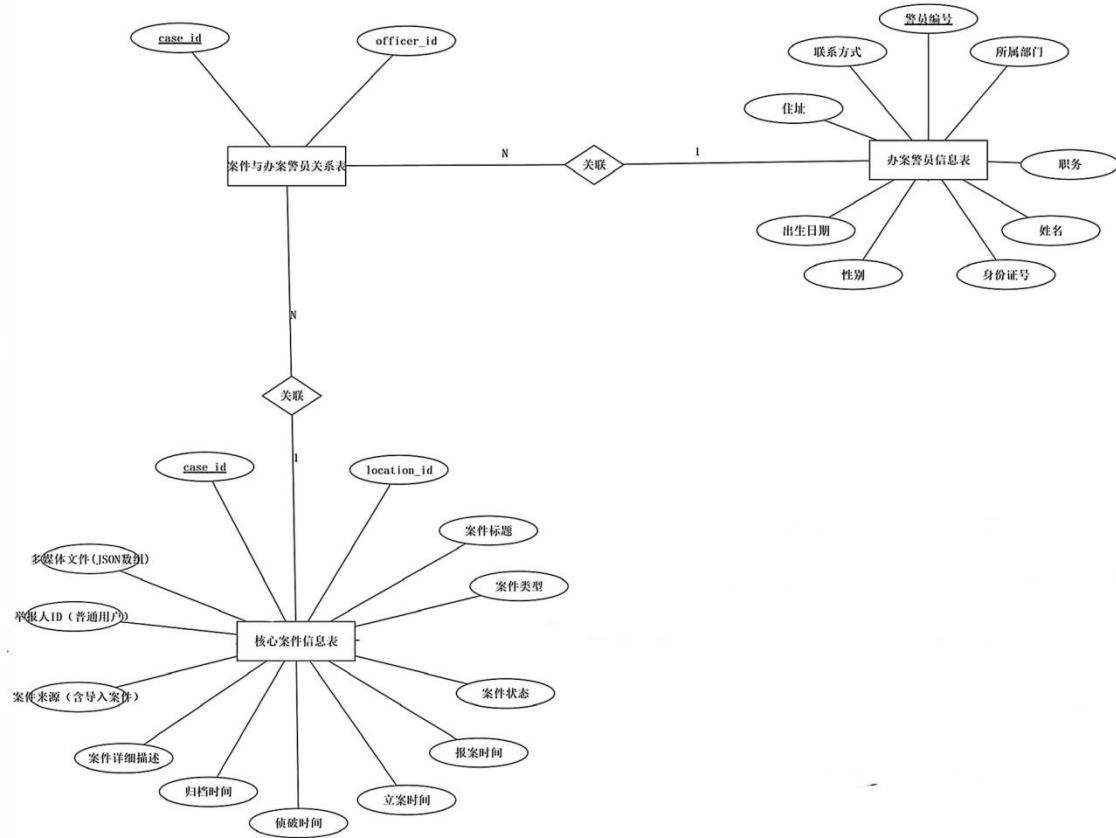
#### 4. 2. 6 涉案记录实体 (Case\_Persons)

解决“一个案件涉及多人，一人涉及多案”的 M:N 关系。该实体不仅连接了案件与人员，还包含了一个核心属性——“涉案角色”



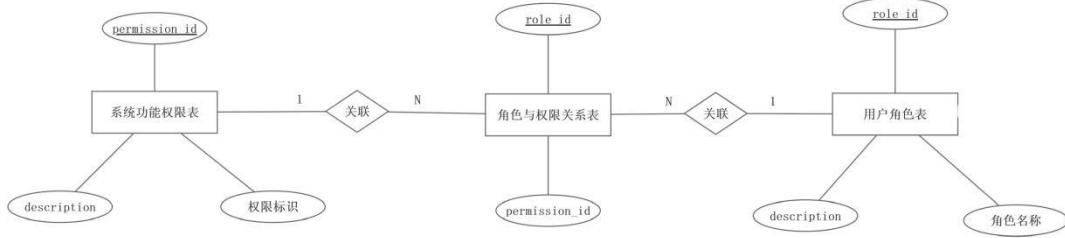
#### 4. 2. 7 办案记录实体 (CaseOfficers)

解决“联合办案”场景下的 M:N 关系。记录了哪些警员负责处理哪些案件。

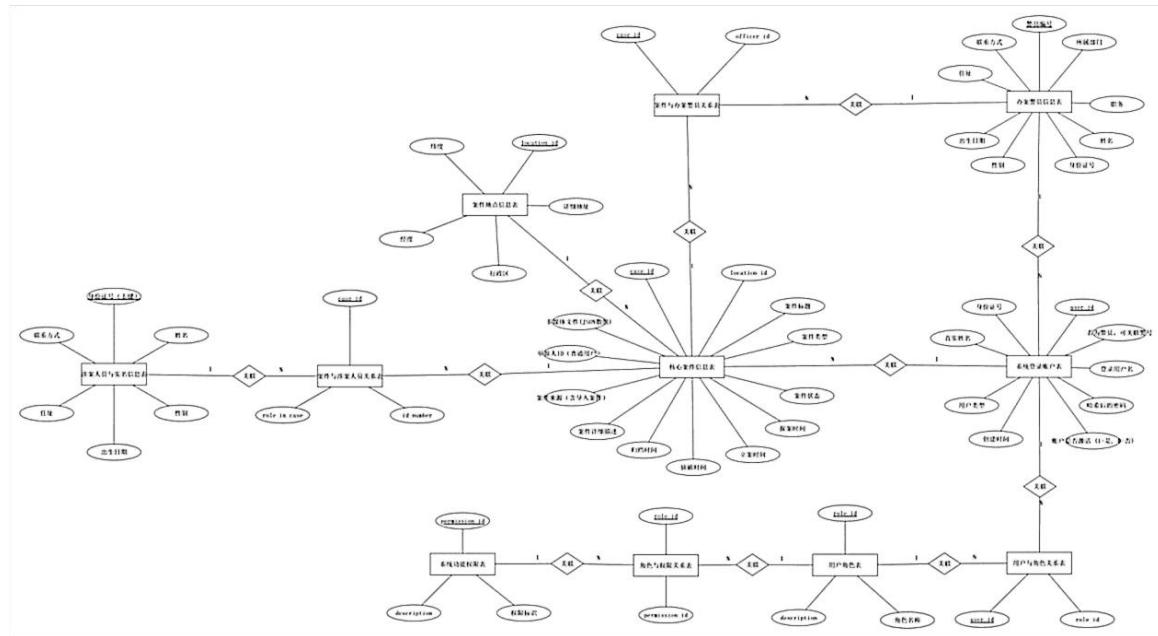


#### 4.2.8 用户角色关联实体 (User\_Roles)

实现 RBAC 模型，允许一个用户拥有多个角色



### 4.3 全局 E-R 图



五. 逻辑设计

## 5.1 E-R 图向关系模型的转变

进行如下的实体间联系转换：

- 对于弱实体集：将标识其的强实体集的主键加入到弱实体集中的表中，与分辨符共同作

为其主键；

- 对于 1 对 1 联系（本次数据库设计中体现为用户账户与警员档案的关联）：在任意一方中加入另一方的主码并设为其外码，并加入联系本身的属性；
- 对于上述联系中的 1 对 n 联系：将 1 方的主码加入 n 方作为外码，同时将联系的属性加入 n 方（例如：用户与案件的上报关系）；
- 对于上述联系中的 n 对 n 联系：将联系本身转换为一个关系模式，将联系双方的主码加入其中设为码，并将联系的属性也加入其中（例如：案件与涉案人员、案件与警员的关联）。

根据上述规则，设计出的关系 E-R 图可以转换为如下关系模型：

登录账户实体 User\_Account: User\_Account(user\_id, username, password\_hash, user\_type, real\_name, id\_number, officer\_id, is\_active, created\_at)

警员实体 Officer: Officer(officer\_id, name, department, position, id\_number, gender, birth\_date, address, contact\_info)

案件实体 Case: Case(case\_id, case\_title, case\_type, status, report\_time, filing\_time, solve\_time, archive\_time, description, source, reporter\_id, location\_id, media\_files)

涉案人员实体 Person: Person(id\_number, name, gender, birth\_date, address, contact\_info)

地理位置实体 Location: Location(location\_id, address, district, longitude, latitude)

角色实体 Role: Role(role\_id, role\_name, description)

权限实体 Permission: Permission(permission\_id, permission\_name, description)

案件与涉案人员的联系关系表 Case\_Person: Case\_Person(case\_id, id\_number, role\_in\_case)

案件与警员的联系关系表 Case\_Officer: Case\_Officer(case\_id, officer\_id)

用户与角色的联系关系表 User\_Role: User\_Role(user\_id, role\_id)

### 5.1.1 地理位置信息表 (Locations)

字段名	说明	数据类型	长度/精度	约束	备注
location_id	位置 ID	INT	-	PK, Auto Inc	唯一标识

address	详细地址	VARCHAR	255	NOT NULL	案件发生的具体街道门牌号
district	行政区	VARCHAR	50	-	如：杨浦区、嘉定区
longitude	经度	DECIMAL	11, 8	-	用于 GIS 定位
latitude	纬度	DECIMAL	10, 8	-	用于 GIS 定位

### 5.1.2 人员基础信息表 (Persons)

字段名	说明	数据类型	长度	约束	备注
id_number	身份证号	VARCHAR	18	PK	唯一标识，自然主键
name	姓名	VARCHAR	50	NOT NULL	-
gender	性别	ENUM	-	NOT NULL	取值：'男', '女', '未知'
birth_date	出生日期	DATE	-	-	-
address	住址	VARCHAR	255	-	户籍地址或常住地址
contact_info	联系方式	VARCHAR	100	-	电话或电子邮箱

### 5.1.3 警员信息表 (Officers)

字段名	说明	数据类型	长度	约束	备注
officer_id	警员编号	VARCHAR	20	PK	警号，唯一标识
name	姓名	VARCHAR	50	NOT NULL	-
department	所属部门	VARCHAR	100	-	如：刑侦支队、派出所
position	职务	VARCHAR	50	-	如：队长、警员
id_number	身份证号	VARCHAR	18	NOT NULL	-

gender	性别	ENUM	-	NOT NULL	取值: '男', '女', '未知'
birth_date	出生日期	DATE	-	-	-
address	住址	VARCHAR	255	-	-
contact_info	联系方式	VARCHAR	100	-	工作联系电话

#### 5.1.4 案件主表 (Cases)

字段名	说明	数据类型	长度	约束	备注
case_id	案件 ID	INT	-	PK, Auto Inc	系统生成的唯一流水号
case_title	案件标题	VARCHAR	100	NOT NULL	简要描述案情
case_type	案件类型	VARCHAR	50	NOT NULL	如: 盗窃、诈骗、纠纷
status	案件状态	ENUM	-	NOT NULL	默认为'已接报'。取值: '已接报', '立案侦查', '已告破', '已归档'
report_time	报案时间	DATETIME	-	NOT NULL	-
filng_time	立案时间	DATETIME	-	-	状态变更为立案时更新
solve_time	侦破时间	DATETIME	-	-	状态变更为已告破时更新
archive_time	归档时间	DATETIME	-	-	案件结束时更新
description	案件描述	TEXT	-	-	详细案情经过
source	案件来源	VARCHAR	50	NOT NULL	默认为'在线举报'
reporter_id	举报人 ID	INT	-	FK	关联 user_accounts.user_id
location_id	地点 ID	INT	-	FK	关联 Locations.location_id
media_files	多媒体文件	TEXT	-	-	存储图片/视频路径的 JSON 数组

### 5.1.5 案件与涉案人员关联表 (Case\_Persons)

字段名	说明	数据类型	长度	约束	备注
case_id	案件 ID	INT	-	PK, FK	联合主键, 关联 Cases
id_number	身份证号	VARCHAR	18	PK, FK	联合主键, 关联 Persons
role_in_case	涉案角色	ENUM	-	PK, NOT NULL	取值:'嫌疑人','受害人','证人'

### 5.1.6 案件与警员关联表 (CaseOfficers)

字段名	说明	数据类型	长度	约束	备注
case_id	案件 ID	INT	-	PK, FK	联合主键, 关联 Cases
officer_id	警员编号	VARCHAR	20	PK, FK	联合主键, 关联 Officers

### 5.1.7 系统登录账户表 (user\_accounts)

字段名	说明	数据类型	长度	约束	备注
user_id	用户 ID	INT	-	PK, Auto Inc	内部唯一标识
username	用户名	VARCHAR	50	Unique, Not Null	登录账号
password_hash	密码	VARCHAR	255	NOT NULL	加密存储的密码
user_type	用户类型	ENUM	-	NOT NULL	取值: '普通用户','管理员'
officer_id	警员编号	VARCHAR	20	FK	若为管理员/警员, 关联 Officers 表
real_name	真实姓名	VARCHAR	50	-	普通用户的实名信息
id_number	身份证号	VARCHAR	18	-	普通用户的实名信息
is_active	激活状态	TINYINT	1	Default 1	1=启用, 0=禁用

created_at	创建时间	TIMESTAMP	-	Default Current	-
------------	------	-----------	---	-----------------	---

### 5.1.8 角色表 (Roles)

字段名	说明	数据类型	长度	约束	备注
role_id	角色 ID	INT	-	PK, Auto Inc	-
role_name	角色名称	VARCHAR	50	Unique, Not Null	如：系统管理员、警务人员
description	描述	TEXT	-	-	-

### 5.1.9 权限表 (Permissions)

字段名	说明	数据类型	长度	约束	备注
permission_id	权限 ID	INT	-	PK, Auto Inc	-
permission_name	权限标识	VARCHAR	100	Unique, Not Null	如： case:create
description	描述	TEXT	-	-	-

### 5.1.10 用户与角色关联表 (User\_Roles)

字段名	说明	数据类型	长度	约束	备注
user_id	用户 ID	INT	-	PK, FK	关联 user_accounts
role_id	角色 ID	INT	-	PK, FK	关联 Roles

### 5.1.11 角色与权限关联表 (Role\_Permissions)

字段名	说明	数据类型	长度	约束	备注
role_id	角色 ID	INT	-	PK, FK	关联 Roles
permission_id	权限 ID	INT	-	PK, FK	关联 Permissions

## 5.2 数据模型的优化及规范化设计

### 5.2.1 规范化理论基础

规范化设计是数据库逻辑设计的核心环节，其目的是通过分析关系模式中属性间的函数依赖关系，消除数据冗余和更新异常。本系统设计严格遵循第三范式 (3NF) 的要求。

3NF 范式要求关系模式满足以下条件：

- 第一范式 (1NF): 强调属性的原子性，即关系中每一列都是不可再分的最小数据单位。
- 第二范式 (2NF): 在 1NF 的基础上，消除非主属性对码的部分依赖。即对于拥有复合主键的表，非主属性必须依赖于整个主键，而不能只依赖于主键的一部分。
- 第三范式 (3NF): 在 2NF 的基础上，消除非主属性对码的传递依赖。即任何非主属性都必须直接依赖于主键，而不能通过其他非主属性间接依赖于主键。

### 5.2.2 关系模式的规范化分析

对核心实体表和关联关系表进行范式检测与分析：

#### 1. 基础实体表的 3NF 分析（以案件实体为例，其主键为 case\_id）

- 表中包含的属性如 case\_title、report\_time、status 等，均直接描述该特定案件的特征，完全依赖于 case\_id。
- 消除传递依赖的处理：在早期的设计构思中，案件表可能包含“发生地点”的详细描述。如果直接将 district 和 longitude 放在案件表中，可能会出现数据冗余。本系统通过分离出独立的地理位置表，在 Cases 表中仅保留外键 location\_id。这样，address 和 district 等属性直接依赖于 Locations 表的主键 location\_id，而不是间接依赖于 case\_id，从而完美符合 3NF 要求，消除了传递依赖。

同样，在警员实体中，警员的姓名、性别、职位、联系方式等属性都直接依赖于主键 officer\_id。系统中未存储冗余的“部门详细地址”等可能导致传递依赖的信息，而是通过部门名称或部门 ID 进行关联，符合 3NF 标准。

#### 2. 关联关系表的 3NF 分析（本系统存在多组多对多关系，例如案件与涉案人员关联表）

- 主键分析：该表的主键是由 case\_id、id\_number 和 role\_in\_case 组成的联合主键。

- 依赖分析：该表不存在其他非主属性，因此不存在非主属性对主键的部分依赖或传递依赖。即使未来增加“备注”字段，该备注也是特指“某人在某案件中的具体情况”，依赖于完整的联合主键。因此该关联表结构稳健，符合 3NF。

### 5.2.3 数据模型的优化策略

在满足 3NF 的基础上，结合“城市治安协同治理”的高并发写和多维查的业务特性，本设计还进行了以下优化：

- 空间数据独立存储：将地理信息单独抽取为 Locations 表。这不仅符合规范化要求，更重要的是为 GIS（地理信息系统）的扩展提供了便利。当需要进行“热力图分析”或“周边警力调度”时，只需对 Locations 表的经纬度字段建立空间索引，而无需全表扫描庞大的案件详情表，极大地提升了空间查询效率。
- 适当的字段冗余：虽然 3NF 要求消除冗余，但在 Cases 表中保留了 case\_type 和 source 作为直接字段，而不是将其拆分为单独的字典表。这是考虑到这两个字段属于高频查询过滤条件，且枚举值相对固定。将其保留在主表中可以避免在查询案件列表时进行过多的 JOIN 操作，属于以空间换时间的合理优化。
- 索引覆盖优化：针对系统高频的查询场景，如“查询辖区内的未结案案件”，在物理设计阶段规划了针对 status 和 report\_time 的复合索引，确保逻辑模型在落地为物理模型时具备高性能。

综上所述，本系统的数据库模型设计在严格遵守第三范式的前提下，针对特定的业务场景进行了合理的结构优化，确保了数据的一致性、完整性以及系统运行的高效性。

## 六. 项目管理

### 6.1 框架选择

本系统采用当前业界流行的前后端分离架构，确保了系统的可扩展性与维护性。

### 6.1.1 前端框架

- **Vue 3 JavaScript 框架**

本项目前端采用 Vue 3 作为主要的开发框架。Vue 3 提供了基于响应式数据流的组件化开发方式，性能优于前一版本，并引入 Composition API，使业务逻辑组织更加灵活与可维护。其核心库支持 tree-shaking，在构建阶段仅保留实际使用的模块，从而有效减少最终打包体积，提升前端加载效率。此外，Vue 3 与 Vite 生态深度整合，使整个开发过程具有更快的热更新速度和更简洁的工程配置。

- **Element Plus 组件库**

项目使用 Element Plus 作为主要的前端 UI 框架。Element Plus 基于 Vue 3 开发，提供了一套风格统一、易使用且可扩展的组件体系，包括表格、表单、对话框、分页、消息提示等。利用 Element Plus，开发者可以快速构建稳定、美观、高可用性的前端界面，从而显著提升开发效率并保证用户界面的一致性。

- **vue-amap 高德地图封装库**

为了满足项目中地理信息展示与案件位置标注等需求，前端集成了 vue-amap。该库基于 Vue 封装了高德地图 API，支持声明式地图配置、地图交互事件、路径绘制、点标记与地图插件调用等功能，使地理信息功能的开发更加简便高效。通过 vue-amap，项目可以轻松实现案件地理分布展示、定位查询等相关功能。

- **ECharts 可视化图表库**

项目使用 ECharts 实现数据可视化功能。ECharts 支持柱状图、折线图、饼图、散点图、热力图等多种图表类型，适合展示案件数据统计结果、趋势分析等内容。其高度可定制化的图表配置和对动态数据更新的强支持，使系统可以构建灵活、直观的可视化分析模块，增强用户对数据的理解能力。

### 6.1.2 后端框架

- **Spring Boot 后端开发框架**

后端采用 Spring Boot 作为主要的 Web 应用开发框架。Spring Boot 通过约定优于配置的设计理念，提供了快速的项目初始化方式与自动化的依赖管理机制。其内嵌 Tomcat 服务器使项目能够独立运行，部署方式简单高效。此外，Spring Boot 拥有完善的生态系统，包

括安全认证、数据访问、异常处理等扩展能力，能够满足本系统的复杂业务需求。

- Spring Security 权限与认证框架

为了实现用户登录、角色权限控制等功能，本项目集成了 Spring Security。该框架提供了完善的认证授权机制，可实现基于角色的访问控制（RBAC）。通过自定义过滤器、加密策略和用户详情服务，系统能够有效管理不同类型用户的访问权限，确保系统数据的安全性。

- Spring Data JPA 数据访问框架

Spring Data JPA 用于简化数据库访问操作。它基于 JPA 规范，在底层使用 Hibernate 作为 ORM 引擎，使开发者能够以面向对象的方式完成数据库操作，而无需编写大量 SQL 语句。通过 Repository 接口即可完成增删改查、分页查询、条件查询等功能，显著提升开发效率，同时减少人为编写 SQL 带来的错误风险。

- MySQL 数据库

本项目使用 MySQL 作为后端数据库。MySQL 具备高稳定性、性能优秀、广泛支持等特点，适用于项目中的用户管理、案件信息管理、文件信息记录等结构化数据存储需求。通过 SQL 初始化脚本（`crime_platform_init.sql`），系统可一次性完成数据库结构的创建与初始化，便于部署与迁移。

## 6.2 开发平台

- 操作系统: Windows 11 64 位

- 前端开发环境: Node.js + Vite + Vue CLI 工具链

通过 VS Code 进行开发，使用 `npm/pnpm` 管理依赖，利用 Vite 实现快速构建和热更新。

- 后端开发环境: JDK 17 + Maven + IntelliJ IDEA

后端基于 JDK 17 构建，使用 Maven 管理依赖与构建流程，IDEA 作为主要开发工具。

- 数据库环境: MySQL

使用 MySQL 作为数据存储系统，并通过统一的 SQL 初始化脚本进行数据库结构部署。

## 七. 系统实现

### 7.1 系统架构搭建

└─ crime-platform-api/	# 后端代码目录 (Spring Boot)
└─ src/	
└─ main/	
└─ java/	
└─ com/	
└─ crimeplatform/	
└─ controller/	# REST 接口定义, 处理前端请求
└─ service/	# 业务逻辑层, 封装核心操作
└─ repository/	# JPA 数据访问层, 与数据库交互
└─ entity/	# 数据表对应的实体类
└─ security/	# 登录、权限、角色认证配置
└─ util/	# 工具类或辅助函数
└─ resources/	
└─ application.yml	# Spring Boot 全局配置文件
└─ static/	# 静态资源目录
└─ uploads/	# 文件上传保存目录, 如图片、附件等
└─ pom.xml	# Maven 构建配置文件
└─ README.md	# 后端说明文档
└─ crime-platform-ui/	# 前端代码目录 (Vue 3 + Vite)
└─ index.html	# 项目入口 HTML 文件
└─ package.json	# 前端依赖声明与脚本
└─ vite.config.js	# Vite 构建与开发服务器配置
└─ pnpm-lock.yaml	# pnpm 锁文件, 记录依赖版本
└─ public/	
└─ logo.png	# 静态资源, 如项目 logo
└─ src/	
└─ App.vue	# Vue 根组件
└─ main.js	# 应用入口文件, 用于初始化 Vue
└─ assets/	# 图片、图标、样式等静态资源
└─ axios/	
└─ axios.js	# Axios 配置文件, 封装请求对象
└─ router/	
└─ index.js	# Vue Router 入口, 定义路由系统
└─ routes.js	# 路由表, 配置页面路径和组件映射
└─ views/	# 视图组件文件夹, 对应各页面 UI
└─ README.md	# 前端说明文档
└─ crime_platform_init.sql	# 一次性数据库初始化脚本
└─ uploads/	# 包含建库、建表、初始数据、权限配置等
└─ README.md	# 根目录文件上传示例 (可能用于调试)
	# 实际部署中会放置多媒体材料文件
	# 项目总说明文档, 介绍功能与部署流程

## 7.2 系统逻辑设计

基于系统需求分析与架构设计，将“守沪者”城市治安协同治理系统划分为如下四个核心业务模块：

- 登录注册模块
  - ✓ 用户注册
  - ✓ 普通市民登录
  - ✓ 管理员登录
- 用户信息管理模块
  - ✓ 市民个人信息更新
  - ✓ 警员个人信息维护
  - ✓ 警力资源管理（管理员）
- 案件上报与追踪模块（市民端）
  - ✓ 在线案件上报
  - ✓ GIS 辅助定位
  - ✓ 历史报案查询
  - ✓ 案件进度追踪
- 案件处置与调度模块（警务端）
  - ✓ 新案接报与审核
  - ✓ 警力协同分派
  - ✓ 涉案人员关联
  - ✓ 案件状态更新
  - ✓ 案件可视化查询

## 7.3 具体功能编写

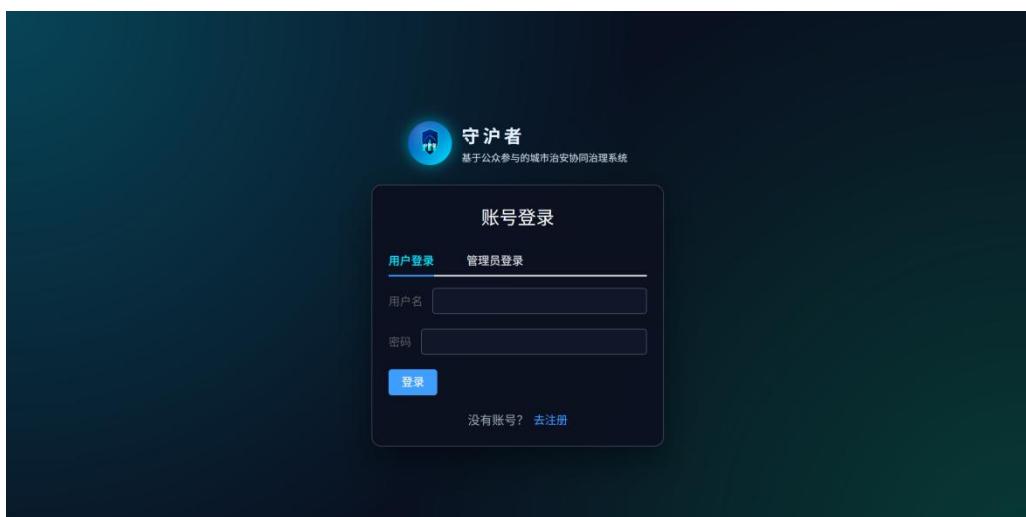
### 7.3.1 登录注册模块

在普通市民注册功能的实现中，前端 Register.vue 组件首先负责收集用户输入的用户名、密码、身份证号及手机号，并利用正则表达式对身份证格式进行严格的预校验。一旦表单提

交，系统即调用后端 /api/auth/register 接口。后端 Service 层会执行核心业务逻辑：首先查询 user\_accounts 表以确保用户名未被占用，同时比对 Persons 表检查身份证号是否已存在，若不存在则同步创建基础人员档案。随后将数据写入数据库，并将默认的用户类型设定为“普通用户”，最终向前端返回注册成功的状态码以触发页面跳转。

The screenshot shows a registration form titled '守沪者 · 创建账号' (Guardian · Create Account). It includes a note: '完成实名认证后即可在线举报、查询案件进度，与警方共同守护城市安全。' (After completing real-name authentication, you can file online reports and query case progress, working together with the police to protect the city's safety.). The form has two tabs: '创建用户账号' (Create User Account) and '找回密码' (Reset Password). Under '创建用户账号', there are fields for: '用户名' (Username) with placeholder '请输入用户名', '密码' (Password) with placeholder '请输入密码', '确认密码' (Confirm Password) with placeholder '请再次输入密码'. Below these are sections for '个人信息' (Personal Information) with fields: '姓名' (Name) with placeholder '请输入真实姓名', '身份证号' (ID Number) with placeholder '请输入18位身份证号', '性别' (Gender) with a dropdown menu, '出生日期' (Date of Birth) with a date picker, '地址' (Address) with placeholder '请输入地址', and '联系方式' (Contact Information) with placeholder '请输入手机号或邮箱'. At the bottom are '创建' (Create) and '返回登录' (Return to Login) buttons.

对于系统的登录功能，无论是市民还是警务人员，均通过统一的身份认证流程。用户在前端输入凭证后，后端 /api/auth/login 接口接收请求并查询数据库。系统会自动比对数据库中存储的哈希密码与用户输入密码，验证通过后，集成的 JWT 工具类会生成包含用户 ID、角色权限及过期时间的 Token 令牌。前端接收到令牌后，将其持久化存储于 localStorage 及 Pinia 状态管理库中。随后，路由守卫（Router Guard）会根据令牌解析出的 user\_type 字段进行判断，将普通市民重定向至服务首页，而将管理员或警员重定向至后台管理驾驶舱。



管理员与警务人员的注册流程则更为严谨，采取了身份绑定机制。在注册界面，警员必须输入唯一“警号”作为身份凭证。后端在接收到注册请求后，会优先检索 Officers 警员信息表，确认该警号真实有效且尚未绑定登录账户。验证通过后，系统在创建账户的同时，会将 user\_accounts 表中的 officer\_id 外键与该警号进行关联，并赋予其“管理员”的高级权限，从而实现物理身份与数字身份的绑定。

The screenshot shows a registration form titled '守沪者 · 创建账号' (Shanghai Protector · Create Account). It is a two-page form. The first page is for account creation, featuring fields for user type (selected as 'Administrator'), username, password, confirmation, and police number. The police number field is highlighted in red, indicating it is a required field. A note below the police number field states: '管理员注册必须提供有效的警号' (Administrators must provide a valid police number) and '管理员注册必须提供警号' (Administrators must provide a police number). The second page is for personal information entry, including fields for name, ID card number, gender, birthdate, address, and contact method. Both pages have 'Create' and 'Return to Login' buttons at the bottom.

### 7.3.2 用户信息管理模块

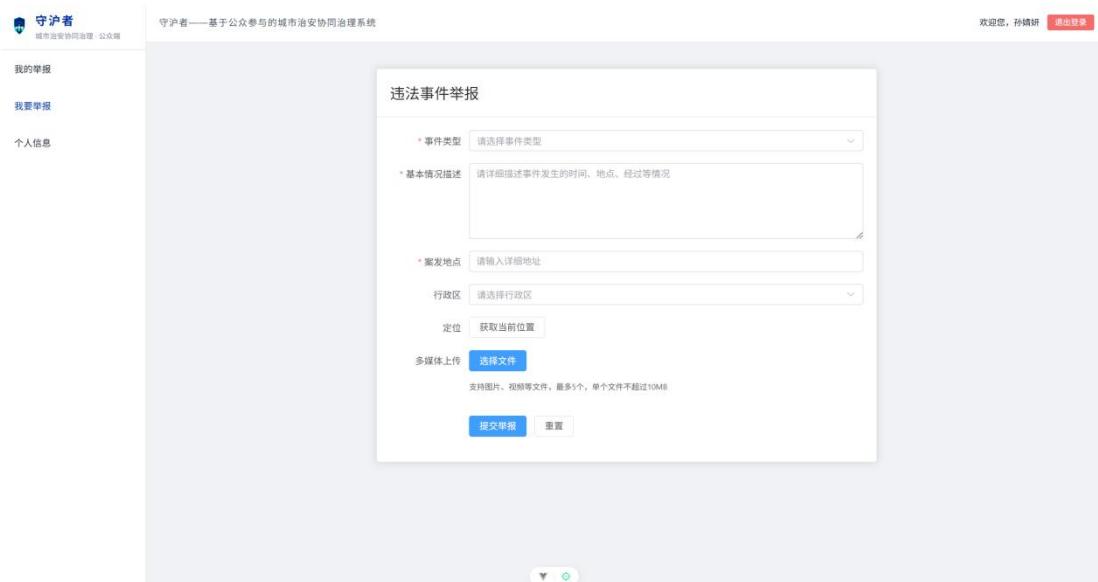
市民用户的个人信息维护主要聚焦于联系方式与居住地址的变更。用户在个人中心修改信息并保存后，前端会封装一个 PUT 请求发送至后端 /api/user/profile 接口。后端通过解析请求头中的 Token 获取当前操作用户的 ID，进而更新 user\_accounts 表中的非敏感字段。考虑到数据的真实性，对于姓名、身份证号等关键实名信息，系统在逻辑上设定为不可直接修改，若需变更则提示用户前往线下柜台进行人工核验，以此确保系统实名制数据的严肃性。

警务侧的信息管理则分为个人维护与资源管理两个维度。拥有权限的管理员可通过“警员管理”模块对辖区警力进行统筹。当管理员执行“新增警员”操作时，后端 /api/officer/add 接口会开启事务，先在 Officers 表中建立警员档案，随即自动在账户表中生成对应的初始登录账号。

警员编号	姓名	身份证号	性别	出生日期	部门	职位	联 操作
001	张小红	001	女	1975-12-01	嘉定派出所	所长	11 [查看] [编辑] [删除]
002	王大壮	222	男	1989-11-17	嘉定派出所	民警	12 [查看] [编辑] [删除]

### 7.3.3 案件上报与追踪模块（市民端）

在线案件上报是连接警民的数字化桥梁。市民在填写报案单时，除了输入案件标题与描述外，系统特别集成了多媒体取证功能。通过 Element Plus 的上传组件，用户可上传案发现场的图片或视频，后端 FileController 接收文件流后将其存储至服务器静态资源目录，并将生成的相对路径 URL 返回给前端。整个表单数据最终被打包发送至 /api/case/report 接口，后端在 Cases 表中插入新记录的同时，会将案件状态初始化为“已接报”，并自动记录当前的系统时间作为报案时间。



The screenshot displays the 'Violation Report' (违法事件举报) form within the 'Guardian' (守沪者) platform. The form is titled 'Violation Report' (违法事件举报). It contains several input fields: 'Event Type' (事件类型), 'Description' (基本情况描述), 'Location' (案发地点), 'Administrative Area' (行政区), 'Location Selection' (定位), and a 'Media Upload' (多媒体上传) section. Below the form is a map interface for selecting the location. The top navigation bar includes links for 'My Reports' (我的举报), 'Report Now' (我要举报), and 'Personal Information' (个人信息). The top right corner shows a welcome message 'Hello, Sun Jingyan' (欢迎您, 孙婧妍) and a 'Logout' (退出登录) button.

为了提高案发地点定位的精准度，上报页面深度集成了基于 Vue-AMap 的 GIS 地图组件。用户既可以拖动地图指针进行选点，也可以通过关键词搜索具体街道。组件会实时返回选定位置的经纬度坐标及结构化地址，这些数据会被自动填充至表单的隐藏域中。在后端处理环节，系统会采用事务机制，先在 Locations 表中插入独立的地理位置记录，再将生成的主键 location\_id 回填至案件表中，从而实现了案件信息与空间地理信息的解耦存储。

### 违法事件举报

\* 事件类型: 盗窃

\* 基本情况描述: 2025年12月10日19时左右我的电动车在同济大学嘉定校区图书馆门口被偷了

\* 案发地点: 上海市嘉定区曹安公路4800号同济大学图书馆门口

行政区: 嘉定区

定位: 获取当前位置 经度: 121.2073726, 纬度: 31.2884798

多媒体上传:   
支持图片、视频等文件，最多5个，单个文件不超过10MB

案件提交后，市民可通过历史报案查询功能追踪处理进度。前端调用接口获取当前用户的历史案件列表，后端通过 MyBatis 的级联查询功能，一次性提取案件基础信息及其关联的地理位置信息。在案件详情页，系统利用步骤条组件直观展示了案件从“已接报”到“立案侦查”再到“已结案”的状态流转。若案件已被分配给具体警员，后端还会查询 CaseOfficers 关联表，将负责警员的联系方式展示给报案人，极大地方便了警民之间的后续沟通。

### 案件详情

案件ID	15	案件标题	用户举报：盗窃
案件类型	盗窃	状态	已接报
报案来源	在线举报	行政区	嘉定区
报案时间	2025/12/11 03:24:56	立案时间	—
侦破时间	—	归档时间	—
案发地点	上海市嘉定区曹安公路4800号同济大学图书馆门口		
案件描述	2025年12月10日晚上19点左右，我的电动车在学校图书馆门口被偷了		

### 案件状态时间轴

- 案件报案  
案件已接报，开始处理流程  
2025/12/11 03:24:56
- 案件立案  
案件尚未立案  
未到达
- 案件侦破  
案件尚未侦破  
未到达

### 7.3.4 案件处置与调度模块（警务端）

在警务工作台，管理员的首要任务是对新进案件进行接报审核。系统界面实时展示状态为“已接报”的待处理案件列表，管理员点击审核后，可查阅市民上传的现场证据。若判定案情属实，管理员点击“立案”按钮，后端随即更新案件状态为“立案侦查”并写入立案时间；若属于无效或重复报警，则可执行驳回操作。审核通过的案件随后进入警力协同分派流程，管理员在弹出的辖区警力列表中勾选指派对象，后端通过在 CaseOfficers 关联表中批量插入记录，完成任务的下发。

案件标题	案件类型	状态	案件来源	行政区	操作
校园笔记本电脑失窃案	盗窃	已归档	短信报警	徐汇区	<button>查看</button> <button>更新状态</button> <button>删除</button>
电诈案件	诈骗	立案侦查	导入案件	浦东新区	<button>查看</button> <button>更新状态</button> <button>删除</button>
手机被盗	盗窃	已告破	线下报警	黄浦区	<button>查看</button> <button>更新状态</button> <button>删除</button>
入室抢劫	抢劫	立案侦查	电话报警	杨浦区	<button>查看</button> <button>更新状态</button> <button>删除</button>

1	校园笔记本电脑失窃案	盗窃	已归档	短信报警	徐汇区	19 00:00:00	济大学四平路校区	<button>查看</button> <button>更新状态</button> <button>删除</button>
5	电诈案件	诈骗	立案侦查	导入案件	浦东新区	2025-10-18 20:24:57	张江路xxx号	<button>查看</button> <button>更新状态</button> <button>删除</button>
7	手机被盗	盗窃	已告破	线下报警	黄浦区	2025-10-20 02:41:17	南京东路xxx号	<button>查看</button> <button>更新状态</button> <button>删除</button>
9	入室抢劫	抢劫	立案侦查	电话报警	杨浦区	2025-10-21 02:41:17	政教路xxx号	<button>查看</button> <button>更新状态</button> <button>删除</button>
10	用户举报：交通违法	交通违法	已接报	在线举报	杨浦区	2025-11-17 11:34:25	同济大学四平路校区	<button>查看</button> <button>更新状态</button> <button>删除</button>
11	绑架案	伤害	已归档	线下报警	闵行区	2025-11-17 00:00:00	闵行区A小区	<button>查看</button> <button>更新状态</button> <button>删除</button>
14	用户举报：诈骗	诈骗	已接报	在线举报	嘉定区	2025-11-17 17:02:22	上海市嘉定区同济大学	<button>查看</button> <button>更新状态</button> <button>删除</button>
15	用户举报：盗窃	盗窃	已接报	在线举报	嘉定区	2025-12-11 03:24:56	上海市嘉定区鲁安公路4800号同济大学图书馆门口	<button>查看</button> <button>更新状态</button> <button>删除</button>

案件侦办过程中，涉案人员信息的关联是核心环节。警员在办理具体案件时，可通过输入身份证号在 Persons 表中检索人员信息，若库中无此人则即时新建档案。随后，警员选择该人员在当前案件中的角色（如嫌疑人、受害人或证人），后端将这一关系写入 Case\_Persons 表，从而构建起“案-人”关联系统。当案件调查终结，警员在系统内提交结案陈词，后端将案件状态更新为“已归档”，并记录侦破与归档时间，系统同时触发消息通知机制，向报案市民反馈最终处理结果。

案件状态时间轴

- 案件报案  
案件已接报，开始处理流程  
2025/12/11 03:24:56
- 案件立案  
案件尚未立案  
未到达
- 案件侦破  
案件尚未侦破  
未到达
- 案件归档  
案件尚未归档  
未到达

办案警员

警员编号	姓名	部门	职位	操作
002	王大壮	嘉定派出所	民警	<button>移除</button>

涉案人员

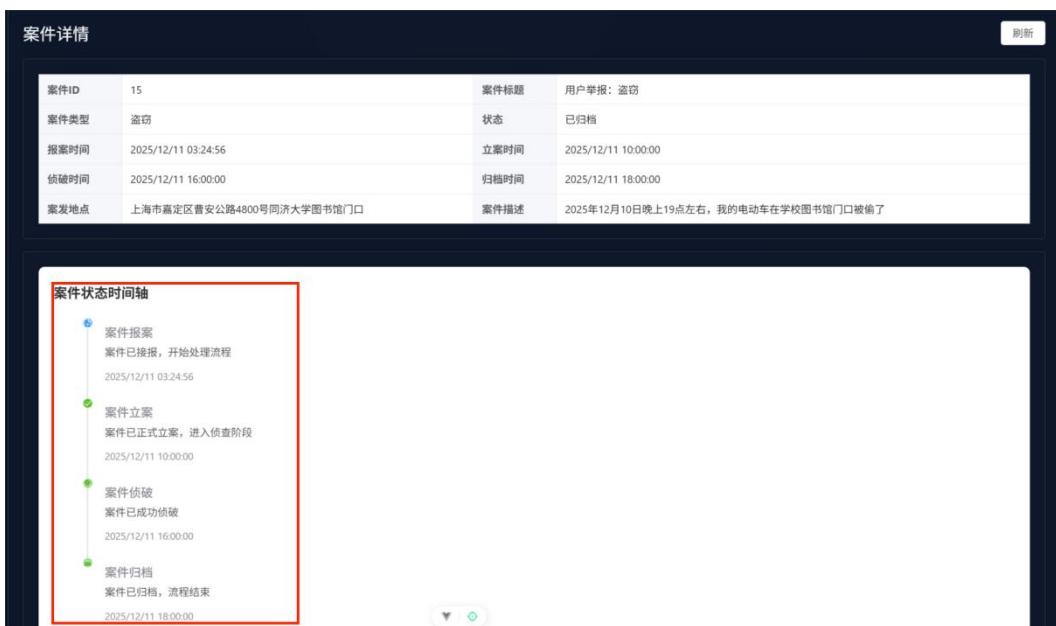
姓名	身份证号	角色	操作
孙婧妍	650203200505141827	受害人	<button>移除</button>

关联成功

更新案件状态

* 案件标题	用户举报：盗窃	* 案件类型	盗窃
* 案件状态	已归档	* 报案时间	2025-12-11 03:24:56
立案时间	2025-12-11 10:00:00	侦破时间	2025-12-11 16:00:00
归档时间	2025-12-11 18:00:00	* 行政区	嘉定区
* 详细地址	上海市嘉定区曹安公路4800号同济大学图书馆门口		
案件描述	2025年12月10日晚上19点左右，我的电动车在学校图书馆门口被偷了		

保存 取消



为了辅助宏观决策，系统还提供了可视化的辖区治安态势查询功能。后端 /api/stats/map 接口聚合了辖区内所有案件的经纬度与类型数据，前端利用 ECharts 与地图组件的结合，将这些数据渲染为动态的热力图或散点图。不同颜色的标记点代表不同类型的案件，警务人员可以直观地看到治安热点区域的分布情况，并支持按时间段筛选数据，从而为警力部署的优化调整提供科学的数据支撑。





#### 7.4 功能测试

序号	功能名称	功能描述	输入	输出
1	案件创建	在系统中新增案件记录	案件名称、案件类型、发生地点、时间、描述	创建成功，案件信息存储于数据库
2	案件信息更新	修改已存在案件的基本信息	案件 ID、新的案件属性字段	更新成功，案件信息已刷新
3	案件删除	删除指定案件记录	案件 ID	删除成功，数据库中对应记录被移除
4	用户登录验证	验证用户登录凭证	用户名、密码	登录成功或失败的提示
5	用户注册	新用户创建系统账户	用户名、密码、城市、电话等信息	注册成功，新账号写入数据库
6	用户信息修改	用户更新自己的个人资料	新用户名、电话、城区等	更新成功，个人信息已更改
7	文件上传（案件附件）	为案件上传图片或文档作为补充资料	案件 ID、上传的文件	上传成功，文件信息与路径记录于数据库
8	权限控制测试	验证普通用户是否无法操作管理员功能	普通用户尝试访问管理员接口	系统返回权限不足提示

9	案件搜索	根据案件名称或类型查询案件	搜索关键词	展示匹配的案件列表
---	------	---------------	-------	-----------

以下图片展示测试过程中前端反馈:

#### 7.4.1 案件创建

违法事件举报

\* 事件类型 交通违法

\* 基本情况描述 2025年12月10日曹安公路4800号十字路口发生交通事故

\* 案发地点 上海市嘉定区曹安公路4800号同济大学南门十字路口

行政区 嘉定区

定位 获取当前位置 经度: 121.2073723, 纬度: 31.2884765

多媒体上传 选择文件

支持图片、视频等文件，最多5个，单个文件不超过10MB

同济嘉定.png

提交举报 重置

公众参与的城市治安协同治理系统 欢迎您

我的举报

案件ID	案件标题	事件类型	状态	行政区	举报时间	案发地点	操作
15	用户举报：盗窃	盗窃	已归档	嘉定区	2025-12-11 03:24:56	上海市嘉定区曹安公路 4800号同济大学图书馆 门口	查看详情
16	用户举报：交通违法	交通违法	已接报	嘉定区	2025-12-11 03:51:08	上海市嘉定区曹安公路 4800号同济大学南门十 字路口	查看详情

#### 7.4.2 案件信息更新

- 管理员端更新

更新案件状态

案件标题	用户举报：交通违法	案件类型	交通违法
案件状态	已接报	报案时间	2025-12-11 03:51:08
行政区	已接报	详细地址	上海市嘉定区曹安公路4800号同济大学南门十字路口
案件来源	立案侦查		
案件描述	已告破 已归档		

**保存 取消**

案件详情

案件ID	16	案件标题	用户举报：交通违法
案件类型	交通违法	状态	立案侦查
报案时间	2025/12/11 03:51:08	立案时间	2025/12/11 10:00:00
侦破时间	—	归档时间	—
案发地点	上海市嘉定区曹安公路4800号同济大学南门十字路口	案件描述	2025年12月10日曹安公路4800号十字路口发生交通事故

案件状态时间轴

- 案件报案  
案件已接报，开始处理流程  
2025/12/11 03:51:08
- 案件立案**  
案件已正式立案，进入侦查阶段  
**2025/12/11 10:00:00**
- 案件侦破  
案件尚未侦破  
未到达
- 案件归档  
案件尚未归档

### ● 用户端同步更新相应案件状态

我的举报

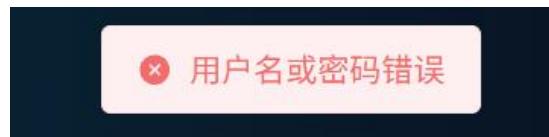
案件ID	案件标题	事件类型	状态	行政区	举报时间	案发地点	操作
15	用户举报：盗窃	盗窃	已归档	嘉定区	2025-12-11 03:24:56	上海市嘉定区曹安公路4800号同济大学图书馆门口	<b>查看详情</b>
16	用户举报：交通违法	交通违法	<b>立案侦查</b>	嘉定区	2025-12-11 03:51:08	上海市嘉定区曹安公路4800号同济大学南门十字路口	<b>查看详情</b>

### 7.4.3 案件删除



#### 7.4.4 用户登录验证

- 用户名/密码错误时



#### 7.4.5 用户注册

- 重复注册时

守护者 · 创建账号

完成实名认证后即可在线举报、查询案件进度，与警方共同守护城市安全。

创建用户账号

\* 用户类型  普通用户  管理员

\* 用户名 巴巴利波

\* 密码 \*\*\*

\* 确认密码 \*\*\*

(\*) 用户名已存在

守护者 · 创建账号

完成实名认证后即可在线举报、查询案件进度，与警方共同守护城市安全。

创建用户账号

\* 用户类型  普通用户  管理员

\* 用户名 巴巴拉拉

\* 密码 \*\*\*

\* 确认密码 \*\*\*

(\*) 该身份证号已被注册

创建用户账号

\* 用户类型  普通用户  管理员

\* 用户名 巴巴拉拉

\* 密码 \*\*\*

\* 确认密码 \*\*\*

个人信息

\* 姓名 孙婧妍

\* 身份证号 650203200505141827

- 身份证格式不匹配时

个人信息

* 姓名	张三
* 身份证号	111111111111 身份证号格式或校验码不正确
* 性别	男

- 创建管理员使用警号不存在时

守护者 · 创建账号

✖ 警号不存在，请确认警号是否正确

完成实名认证后即可在线举报、查询案件进度，与警方共同守护城市安全。

创建用户账号

\* 用户类型  普通用户  管理员

\* 用户名 管理员1

\* 密码 \*\*\*

\* 确认密码 \*\*\*

\* 警号 003

管理员注册必须提供有效的警号

#### 7.4.6 用户信息修改

- 用户端修改

个人信息

个人信息已更新

已完善

用户名	孙婧妍
用户类型	普通用户
* 真实姓名	孙婧妍 已验证的姓名不可修改
身份证号	650203200505141827 已验证的身份证号不可修改
* 性别	女
出生日期	2005-05-14
联系地址	上海市嘉定区曹安公路4800号同济大学
* 联系方式	111111111111

保存修改 重置

- 管理员端同步查看



#### 7.4.7 文件上传（案件附件）



#### 7.4.8 权限控制测试



### 7.4.9 案件搜索

#### ● 按案件类型

**案件管理**

新建案件

案件ID	案件标题	案件类型	状态	案件来源	行政区	报案时间	案发地点	操作
1	校回笔记本电脑失窃案	盗窃	已归档	短信报警	徐汇区	2025-10-19 00:00:00	上海市同济大学四平路校区	<button>查看</button> <button>更新状态</button> <button>删除</button>
7	手机被盗	盗窃	已告破	线下报警	黄浦区	2025-10-20 02:41:17	南京东路xxx号	<button>查看</button> <button>更新状态</button> <button>删除</button>
15	用户举报：盗窃	盗窃	已归档	在线举报	嘉定区	2025-12-11 03:24:56	上海市嘉定区曹安公路4800号同济大学图书馆门口	<button>查看</button> <button>更新状态</button> <button>删除</button>

Total 3

#### ● 按案件状态

**案件管理**

新建案件

案件ID	案件标题	案件类型	状态	案件来源	行政区	报案时间	案发地点	操作
5	电诈案件	诈骗	立案侦查	导入案件	浦东新区	2025-10-18 20:24:57	张江路xxx号	<button>查看</button> <button>更新状态</button> <button>删除</button>
9	入室抢劫	抢劫	立案侦查	电话报警	杨浦区	2025-10-21 02:41:17	政益路xxx号	<button>查看</button> <button>更新状态</button> <button>删除</button>

Total 2

#### ● 按案件来源

**案件管理**

新建案件

案件ID	案件标题	案件类型	状态	案件来源	行政区	报案时间	案发地点	操作
7	手机被盗	盗窃	已告破	线下报警	黄浦区	2025-10-20 02:41:17	南京东路xxx号	<button>查看</button> <button>更新状态</button> <button>删除</button>
11	绑架案	伤害	已归档	线下报警	闵行区	2025-11-17 00:00:00	闵行区A小区	<button>查看</button> <button>更新状态</button> <button>删除</button>

Total 2

### ● 按报案时间

案件管理									<button>新建案件</button>
类型	输入类型	状态	全部	案件来源	全部	报案时间	2025-11-01 00:00:00	2025-12-01 00:00:00	重置
案件ID	案件标题	案件类型	状态	案件来源	行政区	报案时间	案发地点	操作	
10	用户举报：交通违法	交通违法	已接报	在线举报	杨浦区	2025-11-17 11:34:25	同济大学四平路校区	<button>查看</button> <button>更新状态</button> <button>删除</button>	
11	绑架案	伤害	已归档	线下报警	闵行区	2025-11-17 00:00:00	闵行区A小区	<button>查看</button> <button>更新状态</button> <button>删除</button>	
14	用户举报：诈骗	诈骗	已接报	在线举报	嘉定区	2025-11-17 17:02:22	上海市嘉定区同济大学	<button>查看</button> <button>更新状态</button> <button>删除</button>	

## 八. 总结

通过本次数据库系统设计大作业，我对城市治安协同治理系统的构建进行了系统性的分析与实践。项目从需求调研、数据库建模、接口设计到前后端实现，全流程的开发经历不仅强化了我对数据库系统的理论理解，也提升了我将技术应用于真实城市治理场景的能力。

在需求分析阶段，我深入研究了城市治安协同治理的业务流程，区分了普通市民、公安机关以及平台管理员三类核心用户，并明确了事件上报、警情派发、事件处置、信息反馈和综合统计等关键功能。通过对公众参与机制和治安信息流转的梳理，我认识到平台需要具备高实时性的数据处理能力与完备的权限控制体系，为后续系统设计提供了方向。

在概念设计阶段，我构建了系统的全局 E-R 图，确定了用户、事件、警力资源、处理记录、位置数据等实体，并对其关联关系进行了结构化建模。该设计为逻辑模型的构建提供了清晰依据。逻辑设计中，我严格遵循数据库规范化原则，对关系模式进行了第三范式处理，减少冗余，提高数据一致性。在物理设计阶段，我重点对事件表、用户表等高频访问的数据结构设计了合适的索引策略，尤其针对事件状态筛选、时间区间查询和地理位置关联等场景进行了定向优化，使系统在动态数据处理上具备更高的执行效率。

在系统实现过程中，我采用了 Vue3、Element Plus 与 ECharts 构建前端界面，实现了事件地图展示、警情流转状态可视化、用户信息管理等功能；后端使用 Flask 与 Flask-SQLAlchemy 实现 RESTful 服务，结合权限校验与数据验证机制，使系统具备较好

的安全性与可扩展性。在此过程中，我加深了对前后端分离架构、接口设计规范、数据库事务管理等内容的理解，也意识到高质量数据库设计对系统稳定性关键作用。

虽然本次项目的规模有限，但它让我对治安协同治理场景下的数据组织方式与业务流程有了更深刻的掌握。未来，我希望进一步研究更复杂的地理空间数据库、实时事件流分析以及人工智能在城市安全预警中的应用，例如利用深度学习模型对治安事件进行分类预测，提高平台的智能化水平。我也希望将本次项目的经验迁移到城市管理、公共安全、应急响应等更广泛的场景中，探索技术在社会治理中的更多价值。

通过本次大作业，我更加认识到数据库系统是信息化平台的核心基础，而扎实的数据建模与实现能力是软件工程师必备的关键技能。在未来的学习与工作中，我将继续深化数据库、分布式系统与智能分析技术的学习，期望能够在更复杂的工程场景中发挥自身能力，创造更高的社会价值。