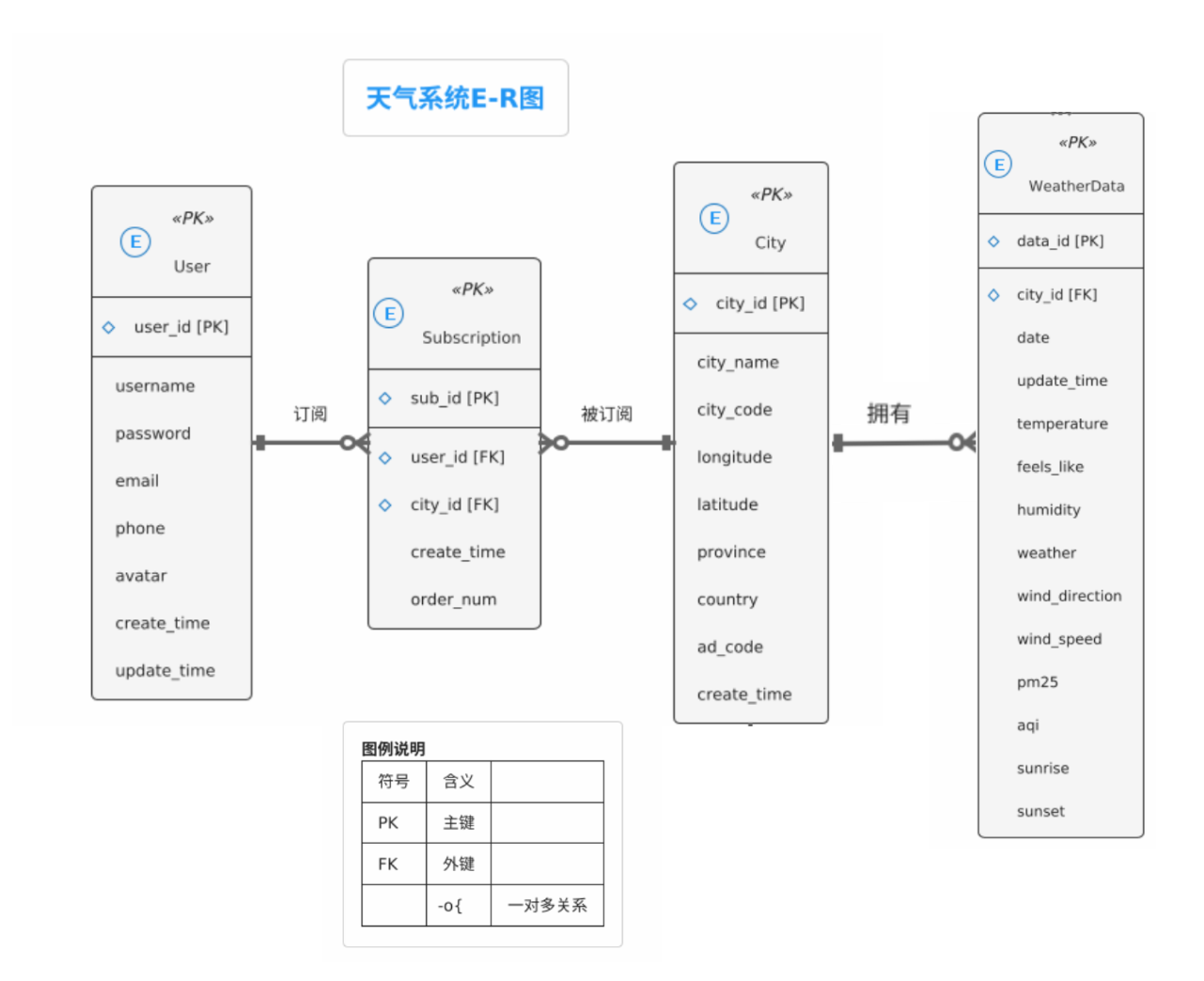
# **7 数据库结构设计**

## **7.1 E-R图设计**



## 

## **7.2 数据库表结构设计**

### **7.2.1 用户表(user)**

**表7.1 用户表结构**

| 字段名 | 数据类型 | 长度 | 允许空 | 主键 | 默认值 | 说明 |
| --- | --- | --- | --- | --- | --- | --- |
| user\_id | bigint | 20 | 否 | 是 | 自增 | 用户ID |
| username | varchar | 50 | 否 | 否 | 无 | 用户名 |
| password | varchar | 100 | 否 | 否 | 无 | 密码(加密存储) |
| email | varchar | 100 | 否 | 否 | 无 | 邮箱 |
| phone | varchar | 20 | 是 | 否 | NULL | 手机号 |
| avatar | varchar | 255 | 是 | 否 | NULL | 头像URL |
| create\_time | datetime | - | 否 | 否 | CURRENT\_TIMESTAMP | 创建时间 |
| update\_time | datetime | - | 否 | 否 | CURRENT\_TIMESTAMP | 更新时间 |

### **7.2.2 城市表(city)**

**表7.2 城市表结构**

| 字段名 | 数据类型 | 长度 | 允许空 | 主键 | 默认值 | 说明 |
| --- | --- | --- | --- | --- | --- | --- |
| city\_id | bigint | 20 | 否 | 是 | 自增 | 城市ID |
| city\_name | varchar | 50 | 否 | 否 | 无 | 城市名称 |
| city\_code | varchar | 20 | 否 | 否 | 无 | 城市代码 |
| longitude | decimal | (10,6) | 否 | 否 | 无 | 经度 |
| latitude | decimal | (10,6) | 否 | 否 | 无 | 纬度 |
| province | varchar | 50 | 否 | 否 | 无 | 所属省份 |
| country | varchar | 50 | 否 | 否 | ‘中国’ | 所属国家 |
| ad\_code | varchar | 20 | 是 | 否 | NULL | 行政区划代码 |
| create\_time | datetime | - | 否 | 否 | CURRENT\_TIMESTAMP | 创建时间 |

### **7.2.3 天气数据表(weather\_data)**

**表7.3 天气数据表结构**

| 字段名 | 数据类型 | 长度 | 允许空 | 主键 | 默认值 | 说明 |
| --- | --- | --- | --- | --- | --- | --- |
| data\_id | bigint | 20 | 否 | 是 | 自增 | 数据ID |
| city\_id | bigint | 20 | 否 | 否 | 无 | 城市ID(外键) |
| date | date | - | 否 | 否 | 无 | 日期 |
| update\_time | datetime | - | 否 | 否 | 无 | 数据更新时间 |
| temperature | decimal | (3,1) | 否 | 否 | 无 | 温度(℃) |
| feels\_like | decimal | (3,1) | 是 | 否 | NULL | 体感温度(℃) |
| humidity | int | 11 | 是 | 否 | NULL | 湿度(%) |
| weather | varchar | 50 | 否 | 否 | 无 | 天气现象 |
| wind\_direction | varchar | 20 | 是 | 否 | NULL | 风向 |
| wind\_speed | decimal | (5,2) | 是 | 否 | NULL | 风速(km/h) |
| pm25 | int | 11 | 是 | 否 | NULL | PM2.5值 |
| aqi | int | 11 | 是 | 否 | NULL | 空气质量指数 |
| sunrise | time | - | 是 | 否 | NULL | 日出时间 |
| sunset | time | - | 是 | 否 | NULL | 日落时间 |

### **7.2.4 用户关注表(subscription)**

**表7.4 用户关注表结构**

| 字段名 | 数据类型 | 长度 | 允许空 | 主键 | 默认值 | 说明 |
| --- | --- | --- | --- | --- | --- | --- |
| sub\_id | bigint | 20 | 否 | 是 | 自增 | 关注ID |
| user\_id | bigint | 20 | 否 | 否 | 无 | 用户ID(外键) |
| city\_id | bigint | 20 | 否 | 否 | 无 | 城市ID(外键) |
| create\_time | datetime | - | 否 | 否 | CURRENT\_TIMESTAMP | 创建时间 |
| order\_num | int | 11 | 是 | 否 | 0 | 排序序号 |

补充:

**数据表与系统功能的对应关系**

1. 用户表(user)

■ 用户注册与登录功能：存储用户账号、密码等基本信息，支撑系统登录验证

■ 个人中心管理功能：维护用户名、头像、邮箱、手机号等个人信息

■ 账号安全功能：通过邮箱和手机号实现密码找回与身份验证

1. 城市表(city)

■ 城市查询功能：提供城市名称、代码等基础信息检索

■ 地图定位功能：通过经纬度数据实现城市地理位置展示

■ 区域分类功能：按省份、国家维度组织城市数据

■ 数据关联功能：作为外键关联天气数据表的核心实体

1. 天气数据表(weather\_data)

■ 实时天气展示功能：提供温度、体感温度、湿度等实时数据

■ 预报预警功能：存储历史与未来日期的天气预测数据

■ 环境监测功能：记录PM2.5、AQI等空气质量指标

■ 生活服务功能：提供日出日落时间等参考信息

■ 数据分析功能：支持生成天气变化趋势图表

1. 用户关注表(subscription)

■ 城市收藏功能：保存用户关注的常用城市列表

■ 个性化展示功能：按用户定制顺序排列城市天气信息

■ 消息推送功能：基于关注城市触发天气异常提醒

**12 关键技术**

**12.1 开发环境**

**·JDK 17：**项目基于Java 17开发，支持现代语法特性（如Records、Switch表达式等）

**·Apache Maven 3.8+：**依赖管理与项目构建工具，管理第三方库（如Spring Boot、MyBatis等）

**·IntelliJ IDEA：**主开发IDE，支持Spring生态链高效开发

**12.2 编译和部署**

**·Spring Boot 3.1+：**内嵌Tomcat服务器，支持快速打包（mvn package生成可执行JAR）

**·Docker：**容器化部署MySQL、Redis等服务，通过docker-compose.yml一键启动依赖环境

**12.3 天气API**

**· 中国气象局数据接口：**通过HttpClient调用权威天气数据（需申请API Key）

**·JSON解析：**Jackson库处理API返回的JSON数据，转换为Java实体类

**12.4 地图和定位服务**

**·高德地图API：**集成高德WebService SDK实现地理编码（地址转坐标）

**·IP定位：**通过第三方服务（如ip-api.com）或Nginx GeoIP模块获取用户初始城市

**12.5 数据库技术**

**·MySQL 8.0：**主数据库，使用MyBatis-Plus进行ORM操作，UTF8MB4字符集支持

**·Redis 7.0：**缓存热点数据（如城市天气），Redisson客户端实现分布式锁

**12.5 源代码目录**

├── pom.xml （项目依赖管理）

├── src （源代码）

│ ├── main （主程序包）

│ │ ├── java

│ │ │ └── com

│ │ │ └── wtqjsz

│ │ │ └── mywea （应用模块）

│ │ │ ├── myweaApplication.java （启动类）

│ │ │ ├── controller （请求处理）

│ │ │ │ ├── user （MYWEA-1.1, MYWEA-2.x）

│ │ │ │ ├── weather （MYWEA-1.2, MYWEA-3.x）

│ │ │ │ └── extension （MYWEA-1.3, MYWEA-4.x）

│ │ │ ├── entity （实体类）

│ │ │ │ ├── user （用户相关实体）

│ │ │ │ └── weather （天气数据实体）

│ │ │ ├── mapper （MyBatis映射）

│ │ │ │ ├── user （MYWEA-2.x 用户操作）

│ │ │ │ ├── weather （MYWEA-3.x 天气工具）

│ │ │ │ └── extension （MYWEA-4.x 应用扩展）

│ │ │ ├── service （服务层）

│ │ │ │ ├── impl （实现类）

│ │ │ │ │ ├── UserServiceImpl （MYWEA-2.x）

│ │ │ │ │ ├── WeatherServiceImpl （MYWEA-3.x）

│ │ │ │ │ └── ExtensionServiceImpl （MYWEA-4.x）

│ │ │ │ ├── UserService （MYWEA-2.x 接口）

│ │ │ │ ├── WeatherService （MYWEA-3.x 接口）

│ │ │ │ └── ExtensionService （MYWEA-4.x 接口）

│ │ │ └── utils （工具类）

│ │ │ └── data （MYWEA-5 数据处理包）

│ │ └── resources （项目资源）

│ │ ├── application.yml （SpringBoot配置）

│ │ ├── mapper （XML映射文件）

│ │ │ ├── user

│ │ │ ├── weather

│ │ │ └── extension

│ │ ├── static （静态资源）

│ │ │ ├── js （前端脚本）

│ │ │ ├── css （样式文件）

│ │ │ └── images （天气图标/分享图片）

│ │ └── templates （HTML模板）

│ │ ├── user （用户管理界面模板）

│ │ ├── weather （天气信息界面模板）

│ │ └── extension （功能扩展界面模板）

│ └── test （单元测试）

│ └── java

│ └── com

│ └── wtqjsz

│ └── mywea

│ ├── controller

│ └── service

└── target （maven 项目编译后的文件夹）

**模块对应关系：**

· **MYWEA-1.x（界面包）** → controller 层

· **MYWEA-2.x（用户操作包）** → service 层（用户相关）

· **MYWEA-3.x（天气工具包）** → service + mapper 层（天气数据）

· **MYWEA-4.x（应用扩展包）** → service + controller 层（分享/安装）

· **MYWEA-5（数据包）** → mapper + utils.data（数据库读写）