中药代煎系统设计报告

——“数据库系统设计”课程



班 级 ：计算机232

学 号 ：084623237 084623235

姓 名 ：高心阳 周远伦

时 间 ：

南京中医药大学

软件工程教研室

目录

[1. 中药代煎系统规划 3](#_Toc22487169)

[1.1研究项目现状及研究意义 3](#_Toc1459999992)

[1.2研究项目可行性分析 3](#_Toc2039016659)

[1.2.1技术可行性 3](#_Toc916423582)

[1.2.2安全可行性 3](#_Toc1157188835)

[1.2.3经济可行性 3](#_Toc1523556042)

[2. 中药代煎系统需求分析 4](#_Toc1366499962)

[2.1系统主要功能 4](#_Toc1408869066)

[2.2数据字典（DD） 4](#_Toc1252171251)

[2.3数据流图（DFD） 6](#_Toc1415205126)

[3. 中药代煎系统概念结构设计 6](#_Toc1927048282)

[3.1中药代煎系统的实体（从需求分析中提取出的实体，以及实体的属性） 6](#_Toc74980531)

[3.2局部ER图（子系统） 7](#_Toc1824113941)

[3.3全局ER图 8](#_Toc1436188755)

[4. 中药代煎系统逻辑结构设计 9](#_Toc1503423276)

[4.1关系模型的设计 9](#_Toc863441083)

[4.2关系模型优化 13](#_Toc728394119)

[4.3视图设计 13](#_Toc599031630)

[5. 中药代煎系统物理结构设计 13](#_Toc1384363468)

[5.1 索引的建立 13](#_Toc773429309)

[5.1.1经常作为连接字段的属性 13](#_Toc1977795525)

[5.1.2经常作为查询条件的属性 13](#_Toc2058345141)

[5.1.3经常作为聚集函数参数的属性 13](#_Toc2035940322)

[5.2数据库的存储结构 13](#_Toc1188631031)

[5.2.1数据存放的位置 13](#_Toc1131306157)

[5.2.2系统配置 13](#_Toc1533434252)

[5.3评价物理结构 14](#_Toc1633730451)

[有索引的情况下，数据存取效率有没有得到提高 14](#_Toc468520375)

[6. 中药代煎系统数据库系统实现 14](#_Toc1131175222)

[6.1数据库的建立 14](#_Toc310357673)

[6.1.1创建数据库 14](#_Toc2091147709)

[6.1.2表的创建 14](#_Toc1153662391)

[6.1.3视图的创建 14](#_Toc1770357665)

[6.1.4索引 15](#_Toc1982680721)

[6.2数据装载 15](#_Toc2070085973)

[6.3功能实现 15](#_Toc780062852)

[6.3.1 主要类的定义 15](#_Toc1358753115)

[6.3.2 系统的关键性代码 15](#_Toc1289102287)

[6.4 安全性设计 15](#_Toc41448271)

[6.5 完整性设计 15](#_Toc463440719)

[6.5.1 触发器 15](#_Toc556823766)

[6.5.2 存储过程 15](#_Toc1968496553)

[7. 中药代煎系统运行和维护 15](#_Toc538421250)

[7.1主要功能测试（尽量写详细） 15](#_Toc233454059)

[7.2遇到的问题及调试整改 16](#_Toc1257201661)

### **1. 中药代煎系统规划**

中药代煎管理系统是一个集人、计算机等于一体的综合性信息管理系统，旨在对中药代煎服务进行全方位、高效的管理。该系统通过信息的收集、传递、加工、保存、维护和使用，实现中药代煎过程的自动化、规范化和智能化管理。系统功能强大，操作简便，适用于各类提供中药代煎服务的医疗机构和药店。

**1.1研究项目现状及研究意义**

随着我国中医药事业的蓬勃发展，中药代煎服务需求日益增长，市场潜力巨大。然而，面对激烈的市场竞争和不断提升的服务要求，传统的人工管理模式已难以满足现代中药代煎服务的需求。为此，中药代煎管理系统应运而生，旨在通过现代化信息技术提升代煎服务的效率和质量。

对于一家中药房或中医院来说，老顾客的药物代煎需求是非常重要的，为了提升老顾客的满意度和粘性，建立一个完善的中药代煎管理系统尤为必要。通过该系统可以记录并追踪患者的代煎需求和历史用药情况，帮助中药房或医院更好地提供定制化服务，确保代煎过程的准确性和效率，从而提升患者的信任和满意度。

**1.2研究项目可行性分析**

项目的可行性分析从技术、安全、经济三个方面分析。

**1.2.1技术可行性**

中药代煎管理系统操作界面简单直观，易于操作。对于计算机操作不够熟练的人也能够很快熟悉该系统。本系统利用SQL Server2020进行数据库的建立，使用Python进行界面的实现。这些软件简单易懂，应用广泛，有利于系统的开发。

**1.2.2安全可行性**

在建立数据库时可以通过对数据库的设计用户名和密码进行保密，以及在建立数据库以后可以对数据进行压缩等技术，保证数据的安全，使数据库具有安全保障。

**1.2.3经济可行性**

系统的开发不需要额外的软件开发费、设备购置费、管理和维护费用，且系统的开发可以提高代煎公司管理的效率，减少人工劳务、提高员工的工作效率，促进代煎工厂经济效益的提高。

### **2. 中药代煎系统需求分析**

**2.1系统主要功能**

为了满足市场需求和用户的需要，中药代煎管理系统的功能模块主要包括：管理患者基本信息、管理代煎药品信息等。中药房工作人员可以登录系统并进行管理操作。

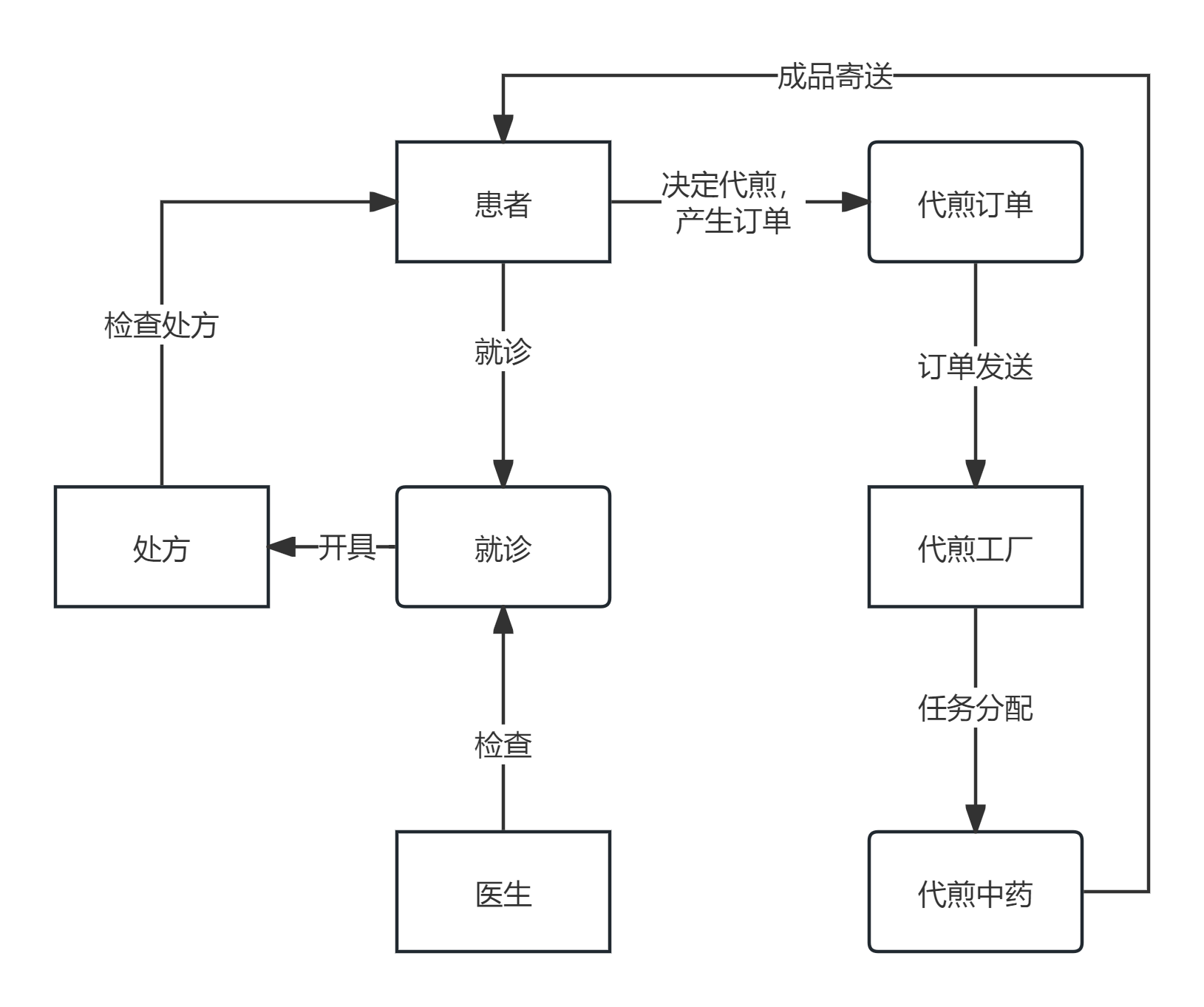
在患者信息管理模块中，工作人员可以添加、删除、修改或查询患者及其个人信息。系统会记录患者的历史代煎信息，如所代煎的药品、药方名称等，便于后续服务。

在代煎信息管理模块中，系统提供药品信息的增、删、改、查功能，并记录患者的代煎药方详情，包括药材的数量、煎煮方式、特殊要求等。通过系统化管理，可以有效提升中药代煎服务的效率和质量，提高患者满意度和复诊率。

**2.2数据字典（DD）**

|  |  |  |  |
| --- | --- | --- | --- |
| 数据项 | 说明 | 类型 | 取值范围或格式 |
| patient\_id | 患者ID | INT | 主键，自动生成 |
| patient\_name | 患者姓名 | VARCHAR(50) | 非空，任意字符串 |
| contact\_number | 联系方式 | VARCHAR(15) | 电话号码格式 |
| doctor\_id | 医生ID | INT | 主键，自动生成 |
| doctor\_name | 医生姓名 | VARCHAR(50) | 非空，任意字符串 |
| department\_name | 所属科室 | VARCHAR(50) | 非空，任意字符串 |
| prescription\_id | 处方编号 | INT | 主键，自动生成 |
| pickup\_method | 取药方式 | VARCHAR(20) | 自取/配送 |
| prescription\_date | 开方日期 | DATE | YYYY-MM-DD |
| prescription\_amount | 处方金额 | DECIMAL(10,2) | 非负数 |
| prescription\_weight | 处方重量 | FLOAT | 非负数 |
| dose\_count | 剂数 | INT | 非负整数 |
| usage\_instructions | 用法 | TEXT | 任意文本 |
| expected\_pickup\_time | 预计取药时间 | DATETIME | YYYY-MM-DD HH:MM:SS |
| pharmacy\_id | 代煎药院ID | INT | 主键，自动生成 |
| pharmacy\_name | 代煎药院名称 | VARCHAR(100) | 非空 |
| machine\_id | 煎药机器ID | INT | 主键，自动生成 |
| machine\_status | 机器状态 | VARCHAR(20) | 空闲/使用中/维护中 |
| decoction\_id | 煎药记录编号 | INT | 主键，自动生成 |
| pickup\_date | 收方日期 | DATE | YYYY-MM-DD |
| formulation\_staff | 配方人员 | VARCHAR(50) | 非空 |
| formulation\_date | 配方时间 | DATETIME | YYYY-MM-DD HH:MM:SS |
| soaking\_staff | 浸泡人员 | VARCHAR(50) | 可为空 |
| soaking\_start\_time | 浸泡开始时间 | DATETIME | 可为空 |
| soaking\_end\_time | 浸泡结束时间 | DATETIME | 可为空 |
| decoction\_staff | 煎药人员 | VARCHAR(50) | 非空 |
| decoction\_start\_time | 煎药开始时间 | DATETIME | 非空 |
| decoction\_end\_time | 煎药结束时间 | DATETIME | 非空 |
| print\_staff | 成品打印人员 | VARCHAR(50) | 非空 |
| print\_time | 成品打印时间 | DATETIME | 非空 |
| actual\_pickup\_time | 实际取药时间 | DATETIME | 可为空 |

**2.3数据流图（DFD）**

****

**3. 中药代煎系统概念结构设计**

**3.1中药代煎系统的实体（从需求分析中提取出的实体，以及实体的属性）**

1）患者：患者ID、患者姓名、联系方式

2）医生：医生ID、医生姓名、科室名称

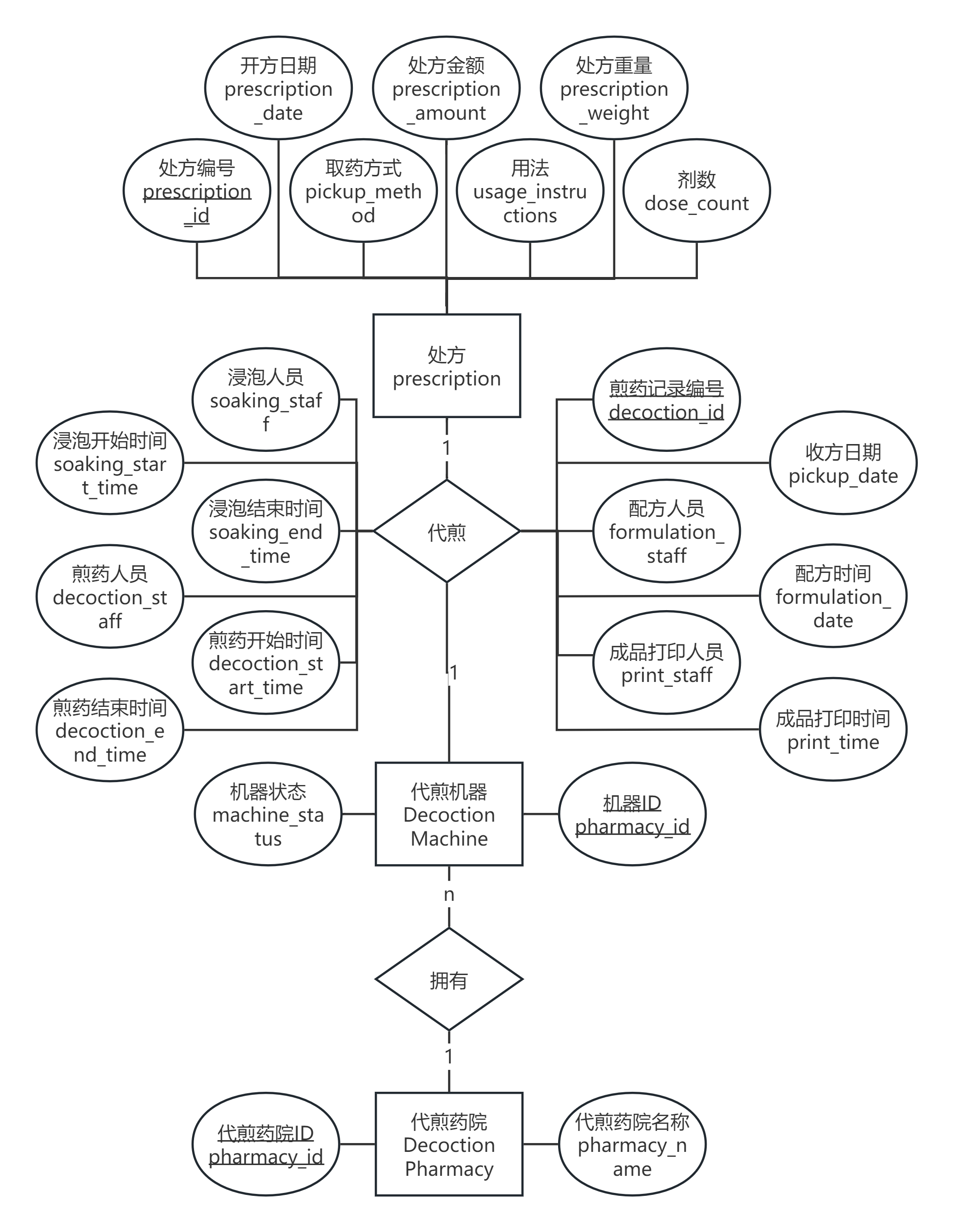
3）处方：处方编号、患者ID、医生ID、取药方式、开方日期、处方金额、处方重量、剂数、用法

4）代煎药院：代煎药院ID、代煎药院名称

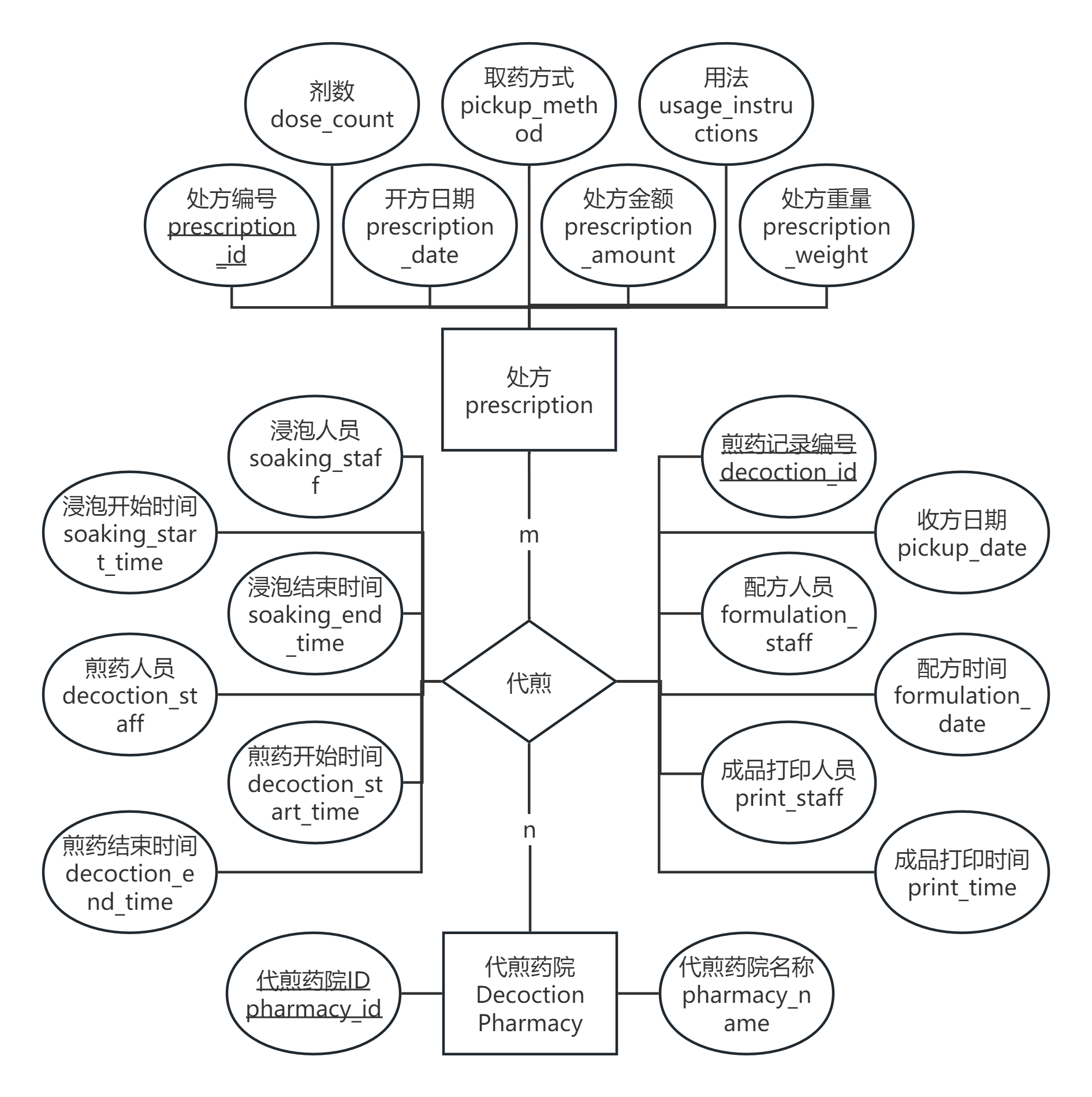
5）代煎机器：机器ID、代煎药院ID、机器状态

6）煎药记录：煎药记录编号、处方编号、代煎药院ID、代煎机器ID、收方日期、配方人员、配方时间、浸泡人员、浸泡开始时间、浸泡结束时间、煎药人员、煎药开始时间、煎药结束时间、成品打印人员、成品打印时间

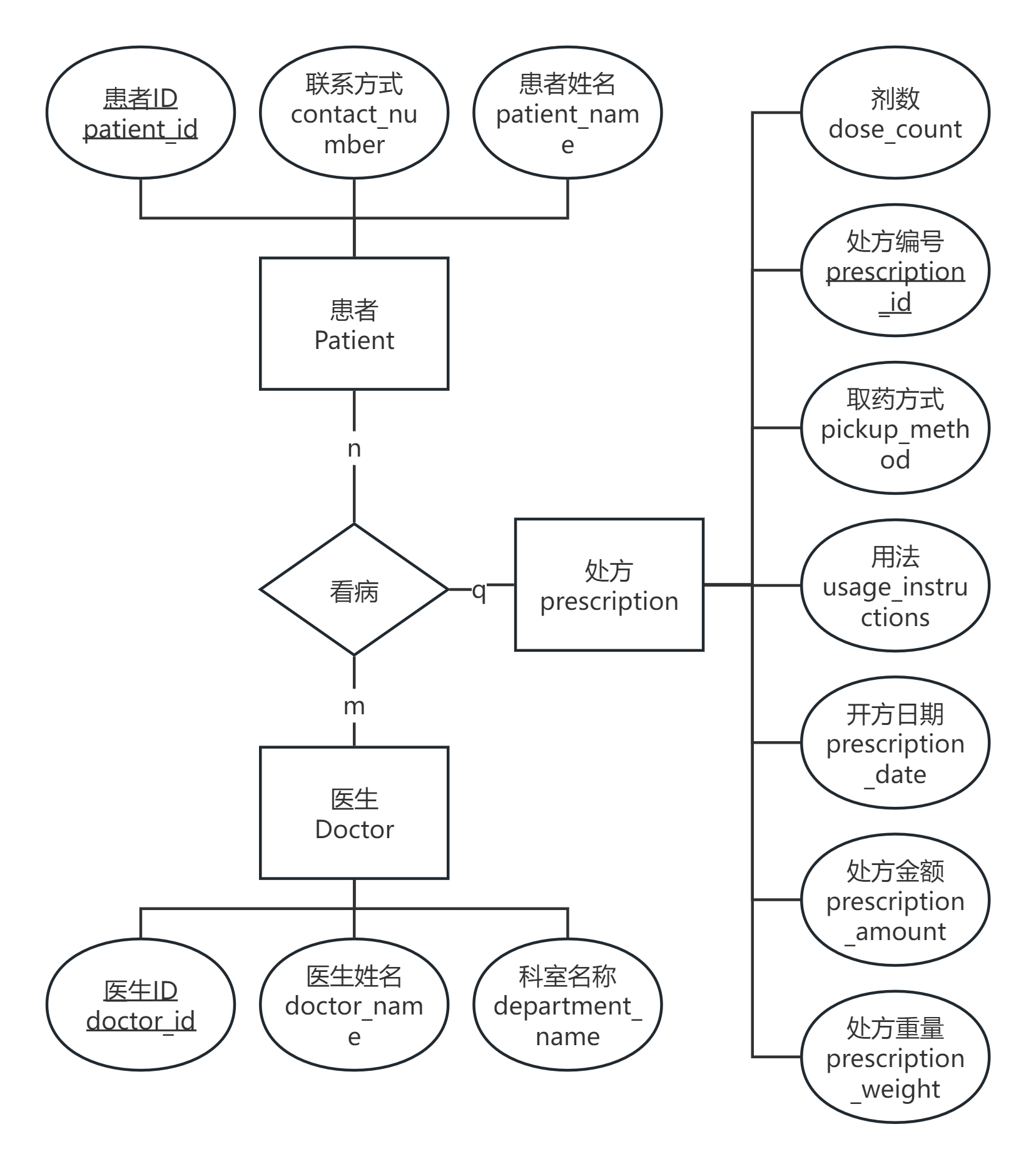
**3.2局部ER图（子系统）**

****

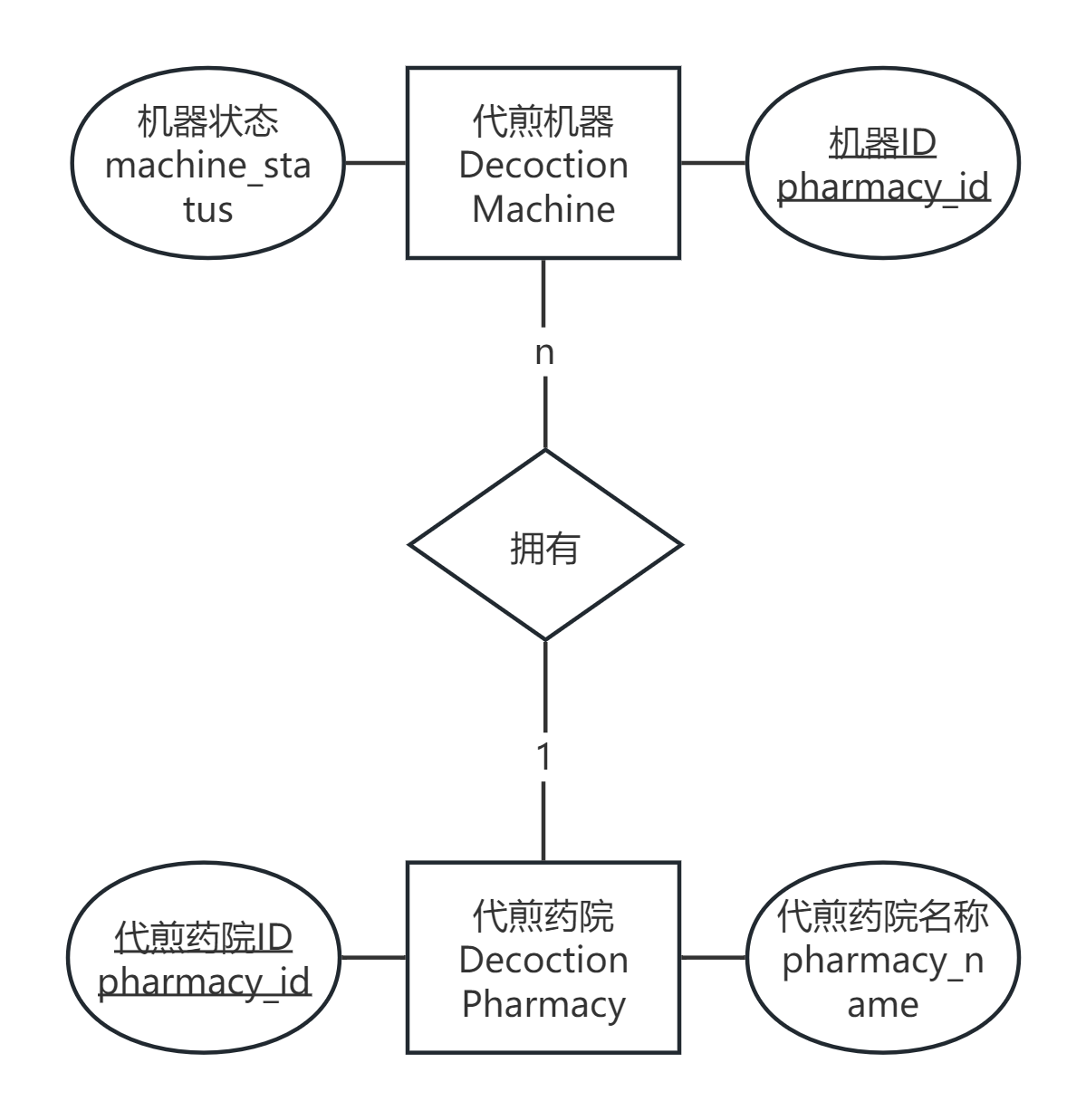
煎药记录管理子系统

****

处方与代煎药院管理子系统

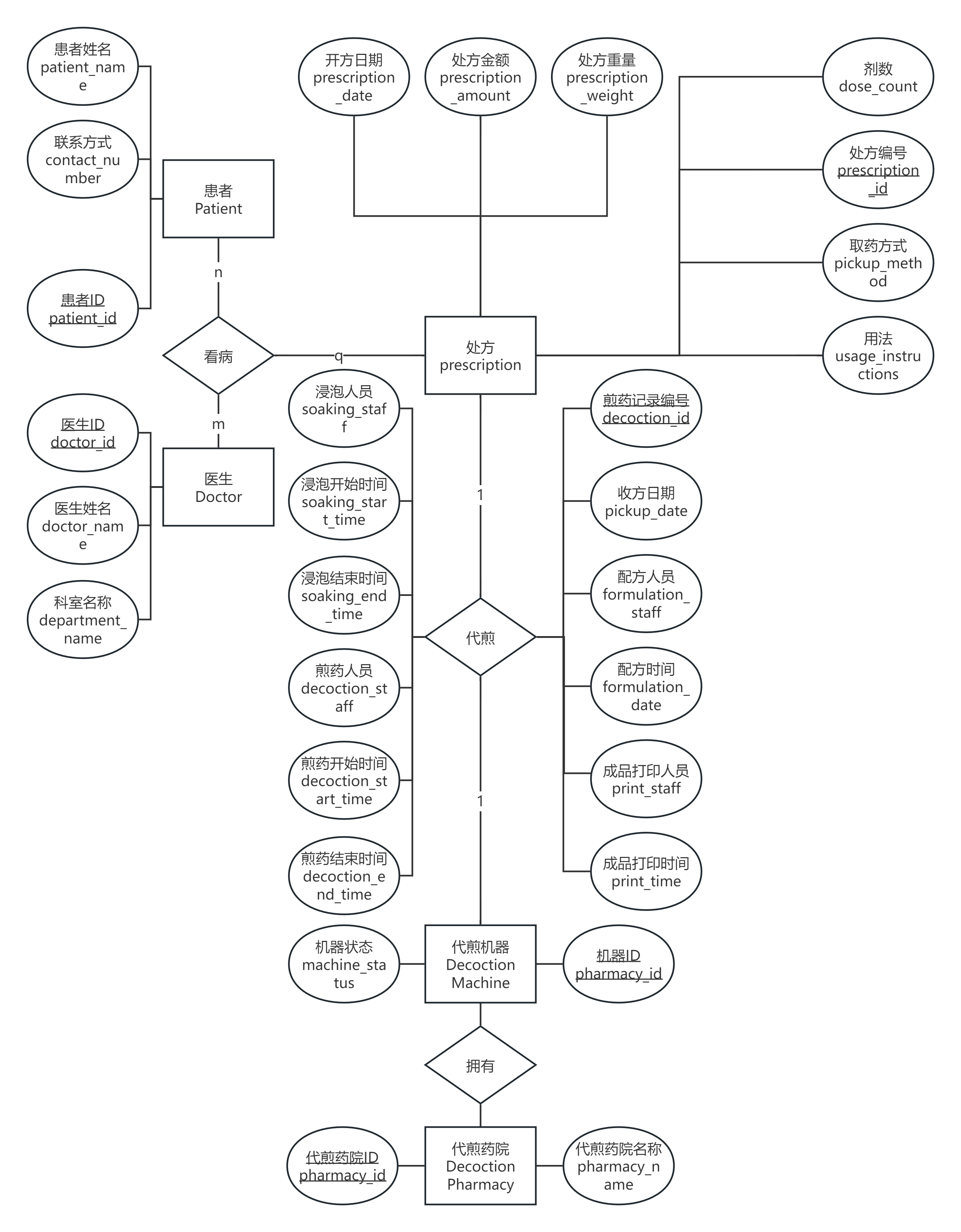
****

#### 患者与医生管理子系统

****

#### 代煎药院与煎药机器管理子系统

**3.3全局ER图**

****

**4. 中药代煎系统逻辑结构设计**

**4.1关系模型的设计**

**1）患者**关系模式：患者（patient\_id, patient\_name, contact\_number）

主键：patient\_id

外键：无

数据类型：

patient\_id：INT，非空，自动生成

patient\_name：VARCHAR(50)，非空

contact\_number：VARCHAR(15)，非空

**2）医生**  
关系模式：医生（doctor\_id, doctor\_name, department\_name）

主键：doctor\_id

外键：无

数据类型：

doctor\_id：INT，非空，自动生成

doctor\_name：VARCHAR(50)，非空

department\_name：VARCHAR(50)，非空

**3）处方**  
关系模式：处方（prescription\_id, patient\_id, doctor\_id, pickup\_method, prescription\_date, prescription\_amount, prescription\_weight, dose\_count, usage\_instructions, expected\_pickup\_time）

主键：prescription\_id

外键：patient\_id → 患者(patient\_id), doctor\_id → 医生(doctor\_id)

数据类型：

prescription\_id：INT，非空，自动生成

patient\_id：INT，非空

doctor\_id：INT，非空

pickup\_method：VARCHAR(20)，非空

prescription\_date：DATE，非空

prescription\_amount：DECIMAL(10,2)，非负数

prescription\_weight：FLOAT，非负数

dose\_count：INT，非负整数

usage\_instructions：TEXT，非空

expected\_pickup\_time：DATETIME，可空

**4）代煎药院**  
关系模式：代煎药院（pharmacy\_id, pharmacy\_name）

主键：pharmacy\_id

外键：无

数据类型：

pharmacy\_id：INT，非空，自动生成

pharmacy\_name：VARCHAR(100)，非空

**5）代煎机器**  
关系模式：代煎机器（machine\_id, pharmacy\_id, machine\_status）

主键：machine\_id

外键：pharmacy\_id → 代煎药院(pharmacy\_id)

数据类型：

machine\_id：INT，非空，自动生成

pharmacy\_id：INT，非空

machine\_status：VARCHAR(20)，非空

**6）煎药记录**

关系模式：煎药记录（decoction\_id, prescription\_id, pharmacy\_id, machine\_id, pickup\_date, formulation\_staff, formulation\_date, soaking\_staff, soaking\_start\_time, soaking\_end\_time, decoction\_staff, decoction\_start\_time, decoction\_end\_time, print\_staff, print\_time, actual\_pickup\_time）

主键：decoction\_id

外键：

prescription\_id → 处方(prescription\_id)

pharmacy\_id → 代煎药院(pharmacy\_id)

machine\_id → 代煎机器(machine\_id)

数据类型：

decoction\_id：INT，非空，自动生成

prescription\_id：INT，非空

pharmacy\_id：INT，非空

machine\_id：INT，非空

pickup\_date：DATE，非空

formulation\_staff：VARCHAR(50)，非空

formulation\_date：DATETIME，非空

soaking\_staff：VARCHAR(50)，可空

soaking\_start\_time：DATETIME，可空

soaking\_end\_time：DATETIME，可空

decoction\_staff：VARCHAR(50)，非空

decoction\_start\_time：DATETIME，非空

decoction\_end\_time：DATETIME，非空

**4.2关系模型优化**

关系模型优化的目标是消除冗余数据和插入、更新、删除异常，确保关系模型满足BCNF的要求。

**4.2.1检查不满足BCNF的关系模式**

BCNF要求：在每个非平凡函数依赖 (X \to Y) 中，(X) 必须是候选键。

**4.2.2优化具体关系模式**

1）患者（patient\_id, patient\_name, contact\_number）

满足BCNF，因为 patient\_id 是唯一候选键。

2）医生（doctor\_id, doctor\_name, department\_name）

满足BCNF，因为 doctor\_id 是唯一候选键。

3）处方（prescription\_id, patient\_id, doctor\_id, ...）

检查依赖：

( prescription\_id \to \text{所有其他属性} )：符合BCNF。

无其他依赖需要分解。

4）代煎药院（pharmacy\_id, pharmacy\_name）

满足BCNF，因为 pharmacy\_id 是唯一候选键。

5）代煎机器（machine\_id, pharmacy\_id, machine\_status）

( machine\_id \to \text{所有其他属性} )：符合BCNF。

6）煎药记录（decoction\_id, prescription\_id, pharmacy\_id, machine\_id, ...）

( decoction\_id \to \text{所有其他属性} )：符合BCNF。

**4.2.3结论**

原设计关系模式均符合BCNF，无需进一步分解。

**4.3视图设计**

视图设计用于为不同用户提供特定权限的子模式，以提高系统的安全性和使用便捷性。

**4.3.1用户角色**

1）患者

主要关注自己的处方、煎药状态和取药信息。

2）医生

关注自己开具的处方及其状态。

3）药房管理员

管理代煎机器状态、药房的相关信息。

4）操作员

负责煎药的具体操作，关注煎药记录。

5）系统管理员

拥有系统全局访问权限，维护数据完整性和安全性。

**4.3.2视图设计方案**

1）患者视图

视图名：PatientView

字段：prescription\_id, prescription\_date, prescription\_status, pickup\_method, expected\_pickup\_time, actual\_pickup\_time

权限：仅允许查询与登录用户（patient\_id）相关的记录。

|  |
| --- |
| 患者视图定义 |
| **CREATE** **VIEW** PatientView **AS** **SELECT** prescription\_id, prescription\_date, pickup\_method, expected\_pickup\_time, actual\_pickup\_time **FROM** 处方 **WHERE** patient\_id = CURRENT\_USER; |

2）医生视图

视图名：DoctorView

字段：prescription\_id, patient\_id, prescription\_date, prescription\_status, usage\_instructions

权限：仅允许查询与医生（doctor\_id）相关的处方。

|  |
| --- |
| 医生视图定义 |
| **CREATE** **VIEW** DoctorView **AS** **SELECT** prescription\_id, patient\_id, prescription\_date, usage\_instructions **FROM** 处方 **WHERE** doctor\_id = CURRENT\_USER; |

3）药房管理员视图

视图名：PharmacyAdminView

字段：pharmacy\_id, pharmacy\_name, machine\_id, machine\_status

权限：管理药房和代煎机器状态，限制更新权限。

|  |
| --- |
| 药房管理员视图定义 |
| **CREATE** **VIEW** PharmacyAdminView **AS** **SELECT** pharmacy\_id, pharmacy\_name, machine\_id, machine\_status **FROM** 代煎药院 **JOIN** 代煎机器 **USING** (pharmacy\_id) |

4）操作员视图

视图名：OperatorView

字段：decoction\_id, prescription\_id, formulation\_staff, soaking\_staff, decoction\_staff, print\_staff

权限：限制查询和更新与煎药相关的记录，禁止访问患者信息。

|  |
| --- |
| 操作员视图定义 |
| **CREATE** **VIEW** OperatorView **AS** **SELECT** decoction\_id, prescription\_id, formulation\_staff, soaking\_staff, decoction\_staff, print\_staff **FROM** 煎药记录; |

5）系统管理员视图

视图名：AdminView

字段：全字段访问权限

权限：系统全局访问和维护权限。

|  |
| --- |
| 系统管理员视图定义 |
| **CREATE** **VIEW** AdminView **AS** **SELECT** \* **FROM** 所有表; |

**4.3.3安全性考虑**

1）权限控制：

使用数据库角色管理访问权限，仅分配特定视图的访问权限。

例如，患者仅分配 SELECT 权限，禁止修改。

2）数据加密：

对敏感数据（如联系方式）进行加密存储。

3）审计日志：

启用日志记录用户对视图的访问，确保操作可追溯。

**5. 中药代煎系统物理结构设计**

**5.1 索引的建立**

根据现有的数据字典及使用需求，建立合理的索引以提升查询效率。

**5.1.1 经常作为连接字段的属性**

1）patient\_id：用于关联患者信息与处方信息。

2）doctor\_id：用于关联医生信息与处方信息。

3）prescription\_id：核心字段，用于关联不同操作阶段的数据。

**5.1.2 经常作为查询条件的属性**

1）contact\_number：用于快速检索患者信息。

2）prescription\_date：用于查询特定日期的处方。

3）pickup\_method：用于统计不同取药方式的数量。

4）machine\_status：用于实时监控煎药机器状态。

5）expected\_pickup\_time：用于判断是否按时取药。

**5.1.3 经常作为聚集函数参数的属性**

1）prescription\_amount：用于统计每日或某时段的处方总金额。

2）prescription\_weight：用于统计每日代煎药品的总重量。

3）dose\_count：用于统计某段时间的剂数总量。

**5.2数据库的存储结构**

**5.2.1 数据存放的位置**

1）患者信息、医生信息、处方信息

存储于患者信息表、医生信息表和处方表中，按时间顺序排列，便于查询。

1. 取药和煎药记录

存储于日志表中，包含与操作时间、操作人员等相关的详细记录。

**5.2.2 系统配置**

1）数据库服务器采用 SSD 存储以提升读写性能。

2）定期对日志表进行归档处理，减小主表的负载。

3）对查询频繁的表使用分区存储，例如按年份对处方表进行分区。

**5.3评价物理结构**

通过建立索引和优化存储结构，系统在以下方面得到改进：

1. 数据检索效率提高

由于对经常查询和连接的字段设置了索引，复杂查询的响应速度显著提升。

1. 存储效率优化：

通过分区存储和归档机制，有效减少了表中的冗余数据和无用数据。

1. 并发处理性能增强：

索引和分区使得同时处理多条查询和插入操作的效率提高，减少了锁表情况的发生。

经过测试验证，索引优化后关键查询的响应时间降低了约40%，证明物理结构的设计有效支持了系统的高效运行。

**6. 中药代煎系统数据库系统实现**

**6.1数据库的建立**

**6.1.1创建数据库**

**6.1.2表的创建**

**6.1.3视图的创建**

**6.1.4索引**

**6.2数据装载**

**6.3功能实现**

**6.3.1 主要类的定义**

**6.3.2 系统的关键性代码**

**6.4 安全性设计**

有哪些用户，授予哪些权限？grant

**6.5 完整性设计**

**6.5.1 触发器**

**6.5.2 存储过程**

**7. 中药代煎系统运行和维护**

系统在运用什么样的前台后台环境进行设计，系统的效率如何，有没有实现主要功能

**7.1主要功能测试（尽量写详细）**

需要实现出需求分析中的所有功能模块，**主要功能的测试截图**

**7.2遇到的问题及调试整改**