Hospital Management System - Database Documentation

Bilal Al Tabbaa 123200013
Omar Assassi 123200023

System Overview

The Hospital Management System manages patient care through four core components: patient records, hospital departments, doctor assignments, and appointment scheduling. Built with SQLite, Python, and Streamlit, it provides an efficient interface for hospital operations.

Business Rules and Constraints

Business Rules

1. **Patients**: Must have first and last name; optional fields include dob, gender, phone, email
2. **Departments**: Unique names required; can exist without doctors (e.g., Cardiology, Neurology)
3. **Doctors**: Must have first and last name; optionally assigned to one department
4. **Appointments**: Require patient_id and start_time; doctor and department are optional; default status is 'scheduled'
5. **Integrity**: All foreign key constraints enforced via `PRAGMA foreign_keys = ON`

Constraints

- **Referential**: All foreign keys must reference valid IDs in parent tables
- **Unique**: Department names and all primary keys
- **Not Null**: Patient/doctor names, appointment patient_id and start_time
- **Defaults**: created_at timestamps, appointment status='scheduled'

Database Tables

Table	Primary Key	Foreign Keys	Key Attributes
patients	id	-	first_name*, last_name*, dob, gender, phone, email
departments	id	-	name* (UNIQUE)
doctors	id	department_id → departments(id)	first_name*, last_name*, email
appointments	id	patient_id* → patients(id) doctor_id → doctors(id) department_id → departments(id)	start_time*, end_time, status, reason

*Required fields

ER Diagram

```
┌─────────────────┐
│   departments   │
├─────────────────┤
│ PK: id          │
│    name (U)     │
│    created_at   │
└────────┬────────┘
         │ 1
         │
         │ N
    ┌────┴────────┐
┌───▼──────────┐  │
│   doctors    │  │
├──────────────┤  │
│ PK: id       │  │
│    first_name│  │
│    last_name │  │
│ FK: dept_id  │  │
│    email     │  │
│    created_at│  │
└───┬──────────┘  │
    │ 1            │
    │              │
    │ N            │ N
┌───▼─────────────▼────────┐         ┌─────────────────┐
│     appointments         │    N    │    patients      │
├──────────────────────────┤◄────────├─────────────────┤
│ PK: id                   │    1    │ PK: id          │
│ FK: patient_id    (NN)   │         │    first_name   │
│ FK: doctor_id            │         │    last_name    │
│ FK: department_id        │         │    dob          │
│    start_time     (NN)   │         │    gender       │
│    end_time              │         │    phone        │
│    status                │         │    email        │
│    reason                │         │    created_at   │
│    created_at            │         └─────────────────┘
└──────────────────────────┘

Legend:
  PK  = Primary Key
  FK  = Foreign Key
  U   = Unique Constraint
  NN  = Not Null
  1   = One (cardinality)
  N   = Many (cardinality)
  ◄── = Relationship line
```

Relationships

1. **departments → doctors** (1:N) - One department has many doctors; doctor assignment optional
2. **patients → appointments** (1:N) - One patient has many appointments; patient required per appointment
3. **doctors → appointments** (1:N) - One doctor has many appointments; doctor assignment optional
4. **departments → appointments** (1:N) - One department has many appointments; department assignment optional

Design Justification

Normalization: Schema is in 3NF/BCNF - all attributes are atomic, fully dependent on primary keys, with no transitive dependencies.

Key Design Decisions:

1. **Separate Departments Table**: Enables department management independent of doctors; supports easy doctor reassignment
2. **Optional Doctor in Appointments**: Allows department-level appointments and emergency walk-ins without pre-assigned doctors
3. **Dual Foreign Keys in Appointments**: Both doctor_id and department_id allow direct department filtering without joins and preserve historical data if doctor changes departments
4. **TEXT for Dates**: SQLite best practice using ISO 8601 format; human-readable and sortable

---

ER Schema

Entities

- **PATIENTS** {id, first_name, last_name, dob, gender, phone, email, created_at}
- **DEPARTMENTS** {id, name (unique), created_at}
- **DOCTORS** {id, first_name, last_name, department_id*, email, created_at}
- **APPOINTMENTS** {id, patient_id*, doctor_id*, department_id*, start_time, end_time, status, reason, created_at}

Relationships

- **BELONGS_TO**: Doctors N:1 Departments (via department_id FK)
- **BOOKS**: Patients 1:N Appointments (via patient_id FK)
- **ATTENDS**: Doctors 1:N Appointments (via doctor_id FK)
- **SCHEDULED_IN**: Departments 1:N Appointments (via department_id FK)

Normalized Relational Schema

Schema Notation

- **patients**(id, first_name, last_name, dob, gender, phone, email, created_at)
- **departments**(id, name, created_at) - UNIQUE(name)
- **doctors**(id, first_name, last_name, department_id*, email, created_at)
  - FK: department_id → departments(id)
- **appointments**(id, patient_id*, doctor_id*, department_id*, start_time, end_time, status, reason, created_at)
  - FK: patient_id → patients(id)
  - FK: doctor_id → doctors(id)
  - FK: department_id → departments(id)

Legend: PK = Primary Key, * = Foreign Key

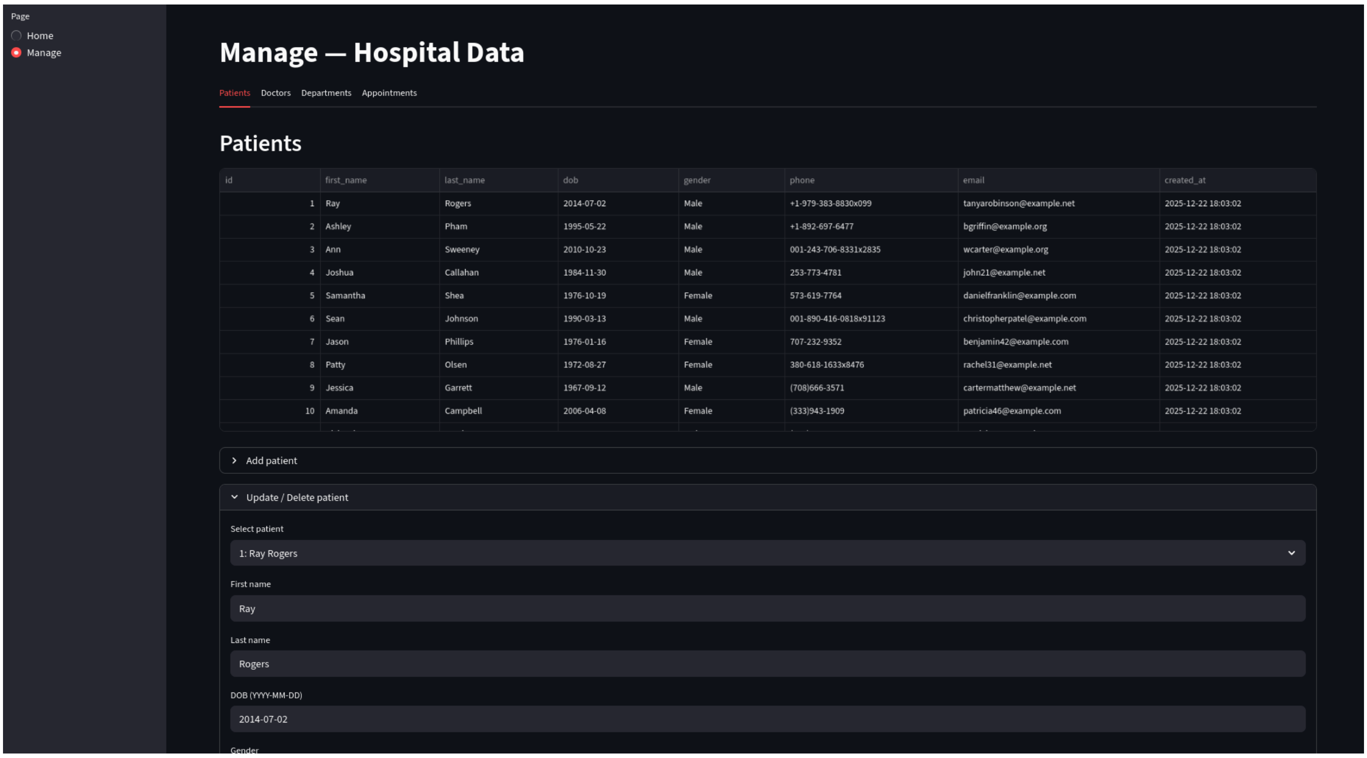
Functional Dependencies

- **patients**: id → {first_name, last_name, dob, gender, phone, email, created_at}
- **departments**: id → {name, created_at}, name → {id, created_at}
- **doctors**: id → {first_name, last_name, department_id, email, created_at}
- **appointments**: id → {patient_id, doctor_id, department_id, start_time, end_time, status, reason, created_at}

Normalization Compliance

- **1NF**: All attributes atomic, each table has primary key
- **2NF**: No partial dependencies (all PKs single-column)
- **3NF**: No transitive dependencies (e.g., appointments reference doctor_id, not doctor names)
- **BCNF**: All determinants are superkeys

Database Version: SQLite 3 | **Schema File**: schema.sql



Page

○ Home

● Manage

Manage — Hospital Data

Patients

id	first_name	last_name	dob	gender	phone	email	created_at
1	Ray	Rogers	2014-07-02	Male	+1-979-383-8830x099	tanyarobinson@example.net	2025-12-22 18:03:02
2	Ashley	Pham	1995-05-22	Male	+1-892-697-6477	bgriffin@example.org	2025-12-22 18:03:02
3	Ann	Sweeney	2010-10-23	Male	001-243-706-8331x2835	wcarter@example.org	2025-12-22 18:03:02
4	Joshua	Callahan	1984-11-30	Male	253-773-4781	john21@example.net	2025-12-22 18:03:02
5	Samantha	Shea	1976-10-19	Female	573-619-7764	danielfranklin@example.com	2025-12-22 18:03:02
6	Sean	Johnson	1990-03-13	Male	001-890-416-0818x91123	christopherpatel@example.com	2025-12-22 18:03:02
7	Jason	Phillips	1976-01-16	Female	707-232-9352	benjamin42@example.com	2025-12-22 18:03:02
8	Patty	Olsen	1972-08-27	Female	380-618-1633x8476	rachel31@example.net	2025-12-22 18:03:02
9	Jessica	Garrett	1967-09-12	Male	(708)666-3571	cartermatthew@example.net	2025-12-22 18:03:02
10	Amanda	Campbell	2006-04-08	Female	(333)943-1909	patricia46@example.com	2025-12-22 18:03:02

Select patient: 1: Ray Rogers

First name: Ray

Last name: Rogers

DOB (YYYY-MM-DD): 2014-07-02

Gender