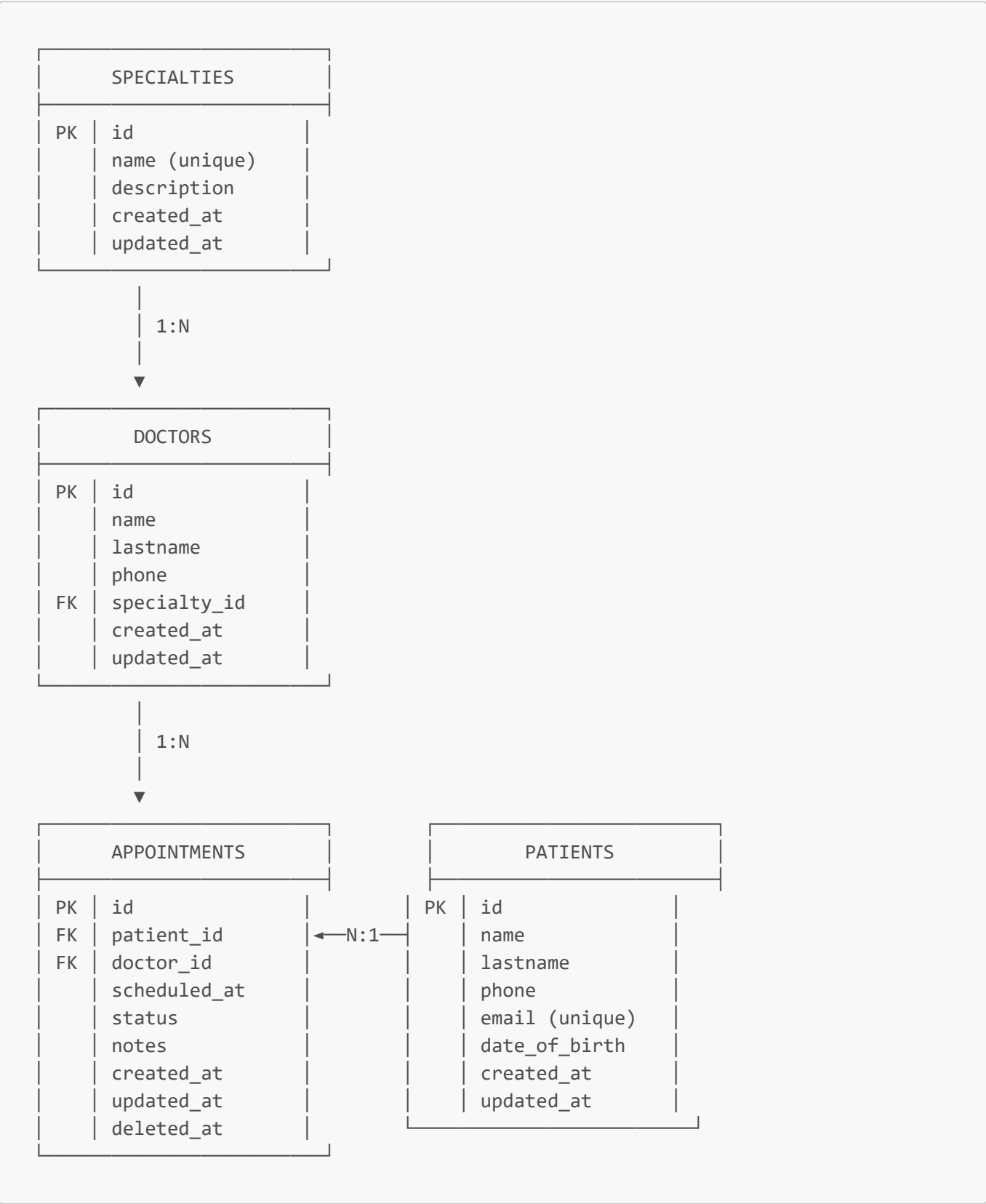


# Diagrama de Base de Datos - Sistema de Citas Médicas

## Estructura de Tablas y Relaciones



## Relaciones

1. Specialties → Doctors (1:N)

- Una especialidad puede tener muchos doctores
- Un doctor pertenece a una especialidad
- **Relación:** `Specialty::hasMany(Doctor::class) ↔ Doctor::belongsTo(Specialty::class)`

## 2. Doctors → Appointments (1:N)

- Un doctor puede tener muchas citas
- Una cita pertenece a un doctor
- **Relación:** `Doctor::hasMany(Appointment::class) ↔ Appointment::belongsTo(Doctor::class)`

## 3. Patients → Appointments (1:N)

- Un paciente puede tener muchas citas
- Una cita pertenece a un paciente
- **Relación:** `Patient::hasMany(Appointment::class) ↔ Appointment::belongsTo(Patient::class)`

# Características Especiales

## Tabla: specialties

- `name`: Único (unique constraint)
- `description`: Nullable

## Tabla: patients

- `email`: Único (unique constraint)
- `date_of_birth`: Tipo date

## Tabla: appointments

- `status`: Enum con valores: 'pending', 'confirmed', 'cancelled', 'completed'
- `notes`: Nullable
- **SoftDeletes:** Implementado (deleted\_at)
- Timestamps automáticos (created\_at, updated\_at)

# Llaves Foráneas y Cascadas

- `doctors.specialty_id` → `specialties.id` (onDelete: CASCADE)
- `appointments.patient_id` → `patients.id` (onDelete: CASCADE)
- `appointments.doctor_id` → `doctors.id` (onDelete: CASCADE)

# Notas de Implementación

1. Todas las tablas usan `id` como clave primaria (auto-incremento)
2. Timestamps automáticos en todas las tablas
3. SoftDeletes solo en la tabla appointments
4. Todas las foreign keys tienen constraint de CASCADE en eliminación