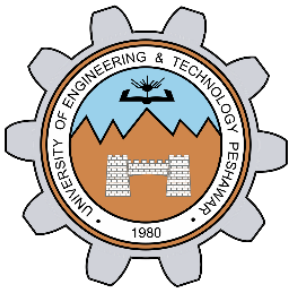
DBMS PROJECT

PHARMACY MANAGEMENT SYSTEM



Spring 2025

Submitted by: **Muhammad Musab, Ahmad Sadiq**

Registration No.: **22PWCSE2201, 22PWCSE2187**

Class Section: **A**

Student Signature: _____

Submitted to:

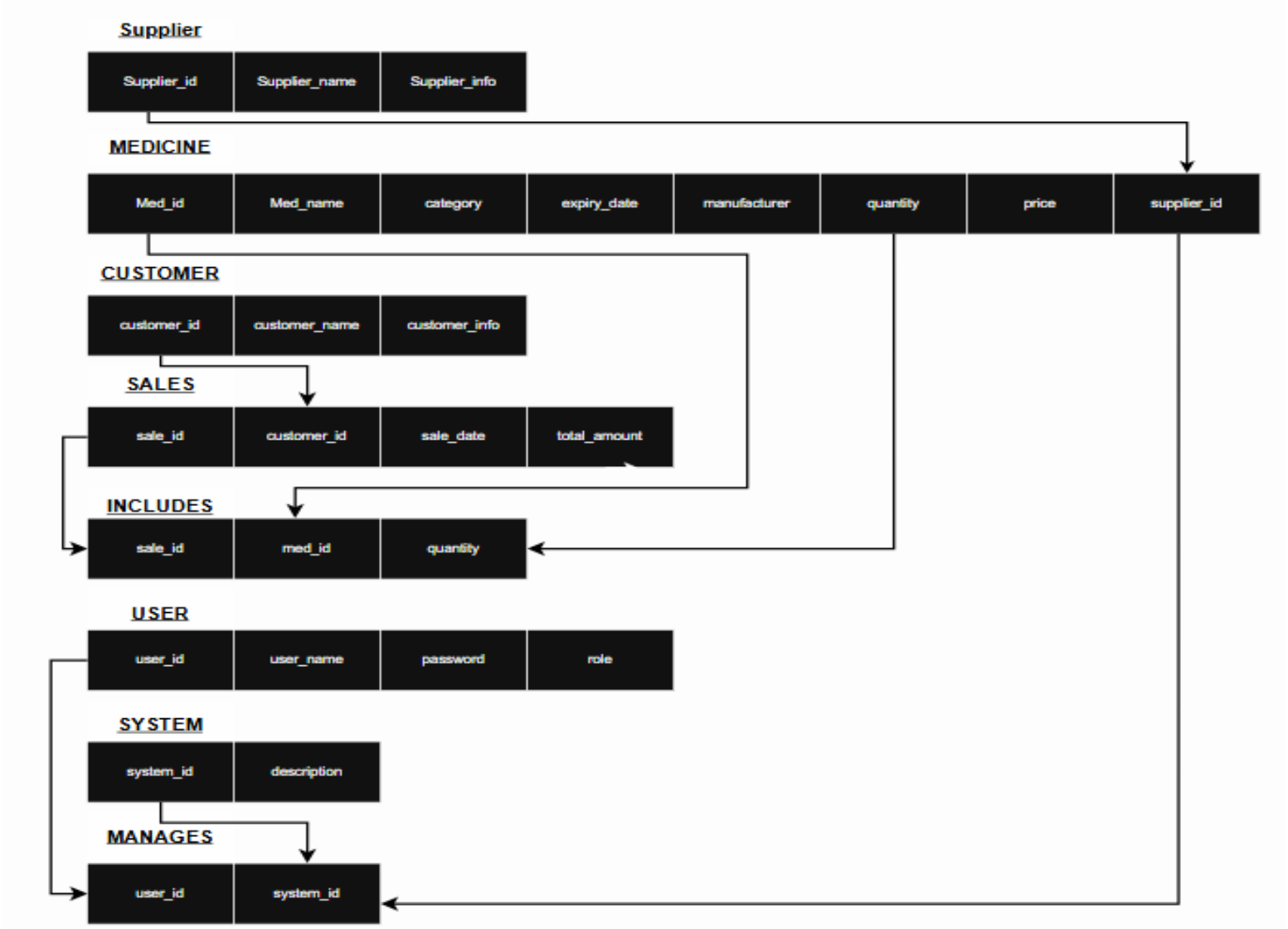
Engr. Sumayya Salahudin

Department of Computer Systems Engineering

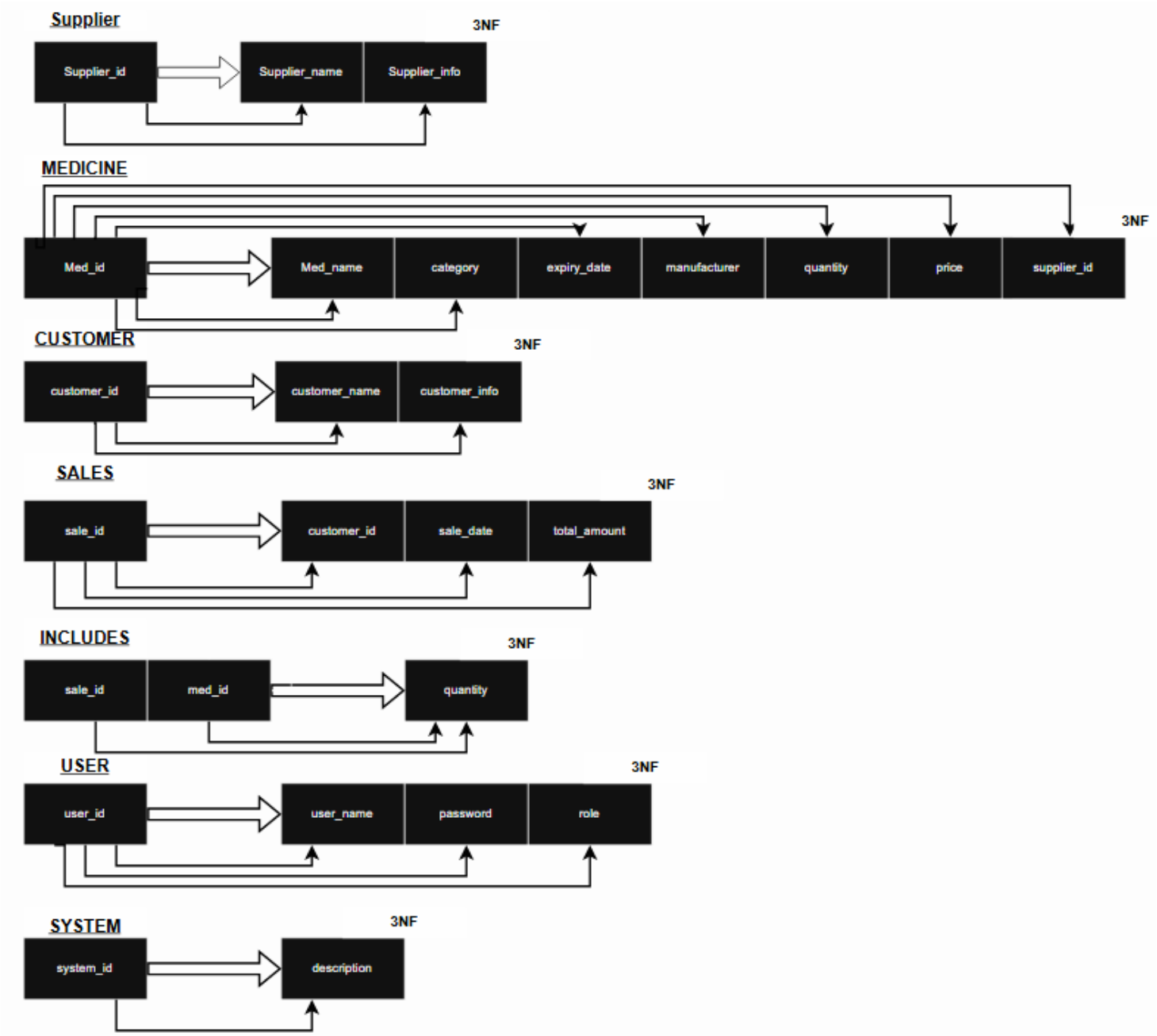
University of Engineering and Technology, Peshawar

Pharmacy Management System Project

Finalized Conceptual Schema



Finalized Normalized Relations:



Comprehensive Implementation Details (Laravel)

Laravel organizes the application into the following main directories:

- **app/Http/Controllers** – contains controllers that handle business logic.
- **app/Models** – holds Eloquent models corresponding to database tables.
- **database/migrations** – contains migration files for table structure.
- **resources/views** – contains Blade templates for UI.
- **routes/web.php** – defines all web routes.
- public/ – publicly accessible files (e.g., CSS, JS, assets).

Models and Relationships:

1. **Supplier:**

- Has many medicines.
- Fields: id, name, contact_info, created_at, updated_at

2. **Medicin**e:

- Belongs to a supplier.
- Has many sale items and prescription medicines.
- Fields: id, name, brand, quantity, price_per_unit, expiry_date, supplier_id

3. **Customer:**

- Has many sales.
- Fields: id, name, email, phone

4. **Sale:**

- Belongs to a customer.
- Has many sale items.
- Fields: id, customer_id, total_amount, created_at

Migrations:

Each model has a corresponding migration file that defines its table structure, foreign key constraints, and indexes.

Controllers:

Each entity has a resource controller:

- MedicineController, SupplierController, SaleController, etc.
- Functions: index, create, store, show, edit, update, destroy

Additional function:

- SaleController::report() – generates a sales report.

Views:

Blade templates are used for:

- Index, Create, Edit, and Show pages
- Shared layout via layouts/app.blade.php
- Custom homepage: resources/views/home.blade.php

Routing:

In routes/web.php:

- Resource routes for CRUD: Route::resource(...)
- Custom route: Route::get('/sales-report', ...) for reports
- Homepage route: Route::get('/', ...)

Features Implemented:

- CRUD operations for all major entities
- Supplier dropdown in medicine form
- Sales entry form with dynamic medicine selection
- Sales report page with summary
- Bootstrap for responsive design
- Homepage with block-based navigation

Testing:

- Feature test for Sales Report page using Laravel's built-in test tools:
    - Visit route /sales-report
    - Check for HTTP 200 response
    - Assert page contains 'Sales Report'

Configuration:

- .env configured for MySQL connection
- SQLite used for testing environment
- Storage linked using php artisan storage:link (if needed)

Security:

- CSRF tokens in forms
- Mass assignment protection via $fillable in models

SQL Database Tables and Queries

Main Tables:

- suppliers
  - id (PK)
  - name
  - contact_info
  - timestamps
- medicines
  - id (PK)
  - name
  - brand
  - quantity
  - price_per_unit
  - expiry_date
  - supplier_id (FK)
  - timestamps
- customers
  - id (PK)
  - name
  - email
  - phone
  - timestamps
- sales
  - id (PK)
  - customer_id (FK)
  - total_amount
  - timestamps

SQL Queries

1. Insert a new medicine

```
INSERT INTO medicines (name, brand, quantity, price_per_unit, expiry_date, supplier_id,
created_at, updated_at)
VALUES ('Paracetamol', 'GSK', 100, 0.25, '2025-12-31', 1, NOW(), NOW());
```

---

2. Get all medicines with supplier names

```
SELECT m.id, m.name, m.brand, m.price_per_unit, s.name AS supplier_name
FROM medicines m
JOIN suppliers s ON m.supplier_id = s.id;
```

---

3. Insert a new sale

```
INSERT INTO sales (customer_id, total_amount, created_at, updated_at)
VALUES (1, 450.00, NOW(), NOW());
```

---

4. Get sales report (total sales per day)

```
SELECT DATE(created_at) AS sale_date, SUM(total_amount) AS total_sales
FROM sales
GROUP BY DATE(created_at)
ORDER BY sale_date DESC;
```

---

5. Get all prescriptions for a customer

```
SELECT p.id, p.date, p.notes
FROM prescriptions p
WHERE p.customer_id = 1;
```

---

6. Insert prescription medicine link

```
INSERT INTO prescription_medicine (prescription_id, medicine_id, dosage)
VALUES (2, 5, '1 tablet twice daily');
```
