

Electricity Management System (EMS) - Full Documentation

1. Core Features

MVP Features:

- Customer management (CRUD)
- Meter management (assign, track status)
- Meter readings (monthly tracking)
- Tariff management (slab rates, fixed charges)
- Billing (generate monthly invoice)
- Payments (record and track)
- Reports (history, unpaid invoices)

Advanced Features:

- Late-fee & notifications
- Complaint management
- Multi-user roles (Admin, Clerk, Engineer)
- Export invoices (PDF)
- Payment gateway integration

2. Architecture Options

A. Desktop (JavaFX/Swing + MySQL)

B. Web (Spring Boot REST API + React/Angular)

Recommended: Web architecture for scalability.

3. Database Schema

Tables:

- customer(id, account_number, name, address, phone, email, active)
- meter(id, meter_number, type, installation_date, status, customer_id)
- tariff(id, name, effective_from, effective_to, fixed_charge)
- tariff_slab(id, tariff_id, from_unit, to_unit, rate_per_unit)
- meter_reading(id, meter_id, reading_date, reading_value, recorded_by)
- invoice(id, invoice_number, customer_id, meter_id, period_start, period_end, consumption_units, total_amount, status)
- payment(id, invoice_id, paid_amount, payment_date, method)

4. Domain Model

Entities: Customer, Meter, MeterReading, Tariff, TariffSlab, Invoice, Payment.

All mapped with JPA annotations for persistence.

5. Billing Logic

Steps:

1. Fetch current and previous readings.
2. Calculate consumption (end - start).

3. Fetch tariff for billing date.
4. Apply slab rates and fixed charge.
5. Add tax, generate invoice.

Example: If consumption = 120 units, tariff has:

- 0-50 @ 5.00
- 51-100 @ 6.00
- 101+ @ 8.00

Total = $50 \times 5 + 50 \times 6 + 20 \times 8 = 250 + 300 + 160 = 710$ + fixed charge.

6. REST API Endpoints

- /api/customers (GET, POST, PUT, DELETE)
- /api/meters (GET, POST, PUT, DELETE)
- /api/readings (POST, GET by meter)
- /api/billing/invoice/generate (POST)
- /api/payments (POST, GET by invoice)

7. UI Design

JavaFX (Desktop):

- Dashboard with charts
- Customer & meter management forms
- Invoice generation screen

Web (React/Angular):

- Tables with search & filters
- Invoice preview & download
- Role-based access

8. Non-Functional

- Use Java 17+
- Database: MySQL/PostgreSQL
- Logging with SLF4J
- Backup policies for DB
- Security: HTTPS, authentication
- Testing: unit & integration tests

9. Workflow

1. Record meter readings monthly.
2. Generate invoices automatically or manually.
3. Notify customers of bills.
4. Record payments.
5. Generate reports for unpaid invoices.