

SoftWare Design Document

Topic:Customers Relational Management

1. Introduction

Purpose:

This document describes the design of a CRM system that helps manage customer interactions, employee task assignments, and role-based responsibilities, including email notifications.

Scope:

The CRM will provide customer management, task tracking, and email reminders for users. It will support employees with different roles (Admin, Manager, Sales Agent, etc.) and ensure role-based access.

Audience:

This document is intended for developers, project managers, testers, and stakeholders involved in the system's development.

2. System Overview

- A web-based CRM system.
- Core functionalities: Customer management, task tracking, email alerts.
- Role-based: Admin, Manager, Sales Agent, etc.
- Users will interact with the system via web browsers.

3. Design Considerations

Assumptions:

- System is used in a modern browser (e.g., Google Chrome).
- Internet connection of at least 7 Mbps is required.

Constraints:

- Must provide role-based access.
- System must be responsive and scalable for growing data/users.

4. System Architecture

Architecture Style: 3-tier architecture

- **Frontend:** React.js
- **Backend:** Spring Boot (Java)
- **Database:** MySQL

Layers:

- **Presentation Layer:** Handles UI, user interactions.
- **Application Layer:** Contains business logic, task/customer services.
- **Data Layer:** Manages data access and persistence.

5. Modules & Components

◆ User Module

- Register, login, update profile.
- Role-based access control.

◆ Customer Management Module

- Create, read, update, delete (CRUD) customer data.
- Filter and search customers.

◆ Task Management Module

- Assign tasks to users.
- View pending and completed tasks.
- Notify users about task deadlines.

◆ Email Notification Module

- Send emails to users for:
 - Upcoming tasks
 - Pending tasks
 - System alerts

◆ Admin Module

- Manage users and assign roles.
- Oversee system operations.

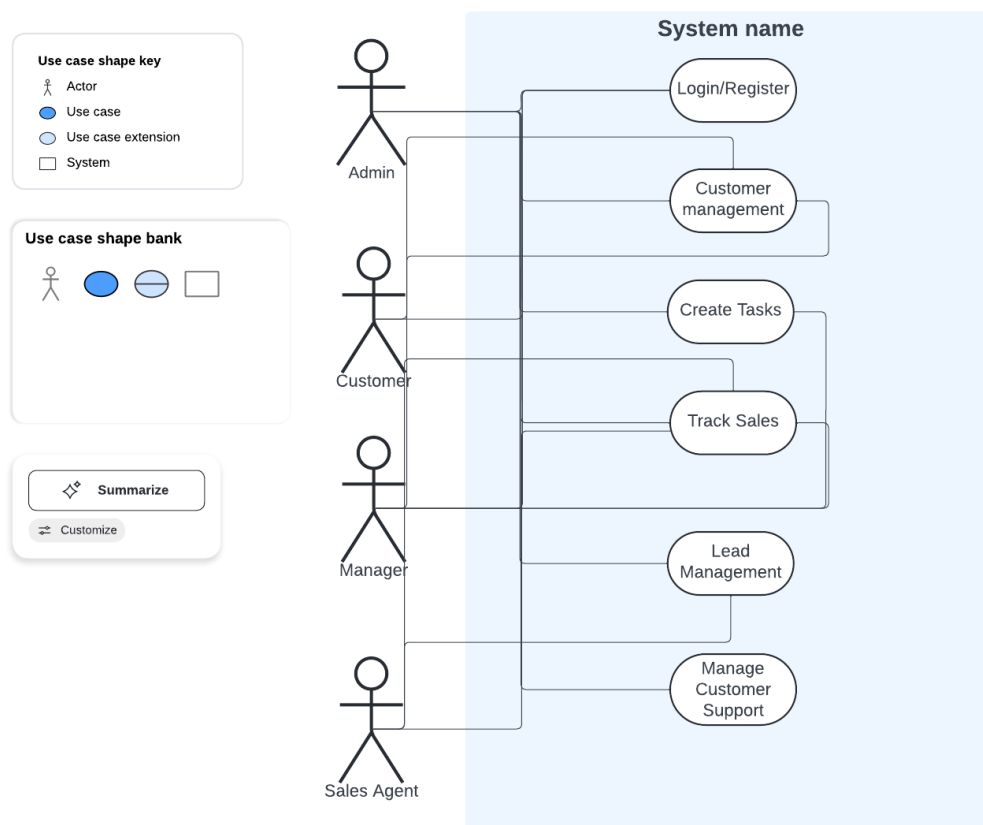
6. Data Design

Database Tables:

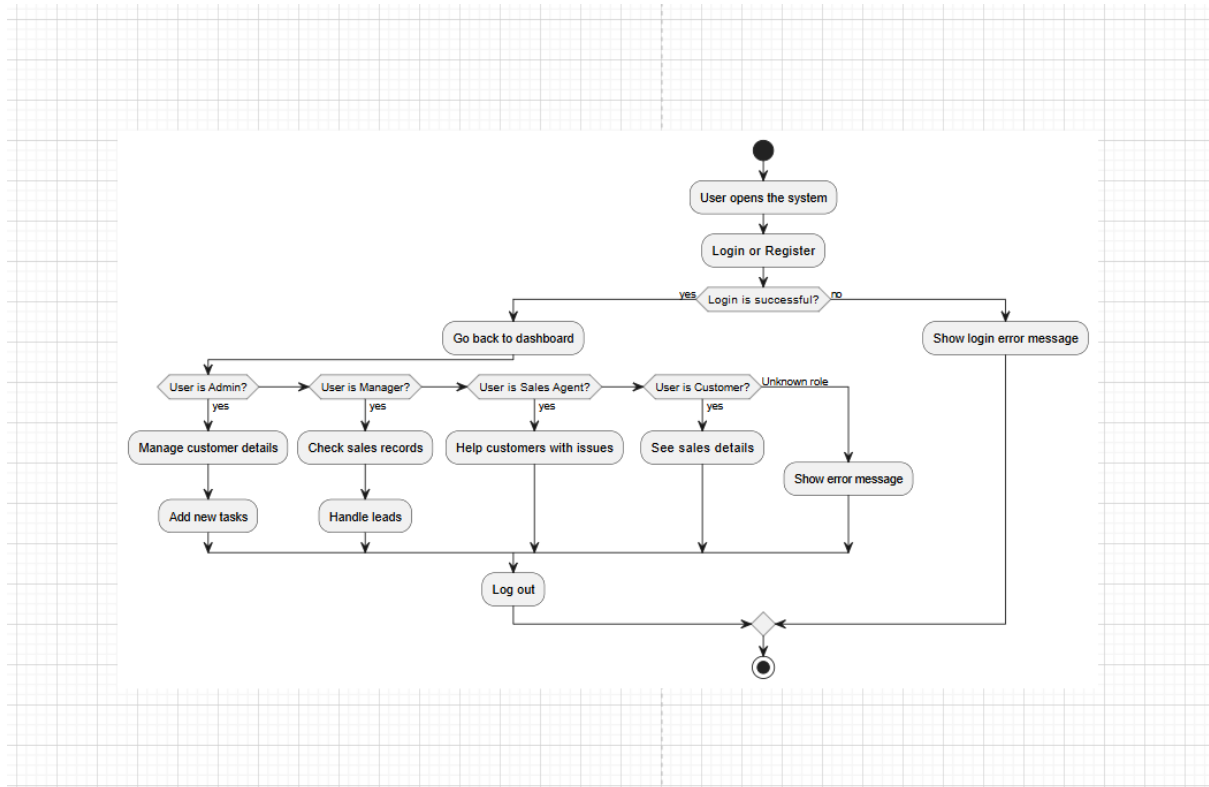
- Users(user_id, name, email, password, role)
 - Customers(customer_id, name, contact_info)
 - Tasks(task_id, title, description, assigned_to, due_date, status)
 - EmailLogs(log_id, user_id, subject, sent_time, status)
-

7. UML Diagrams

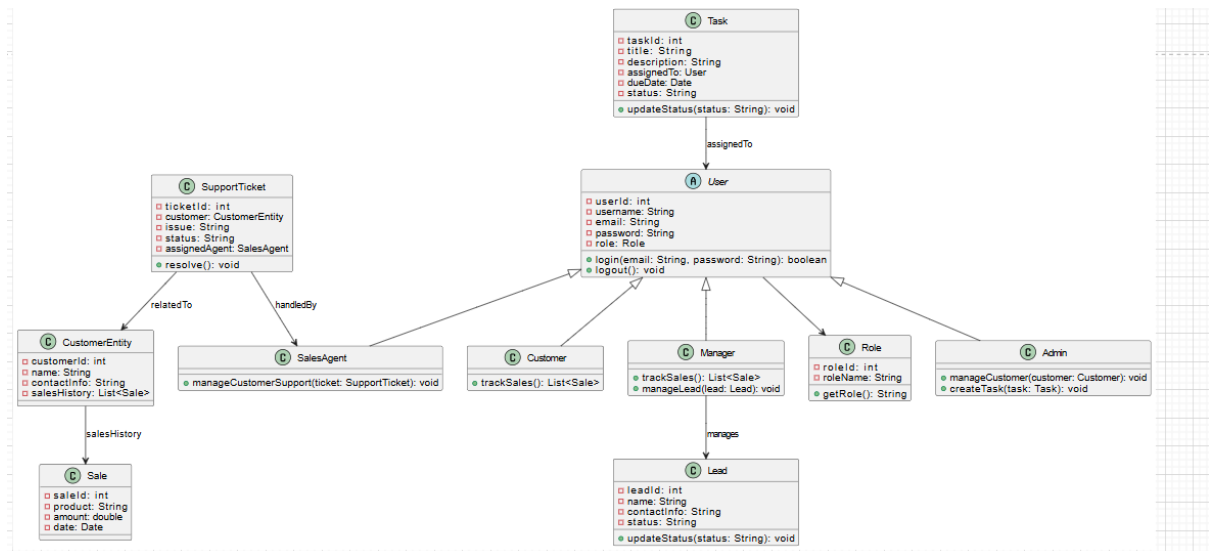
->UseCase Diagram:



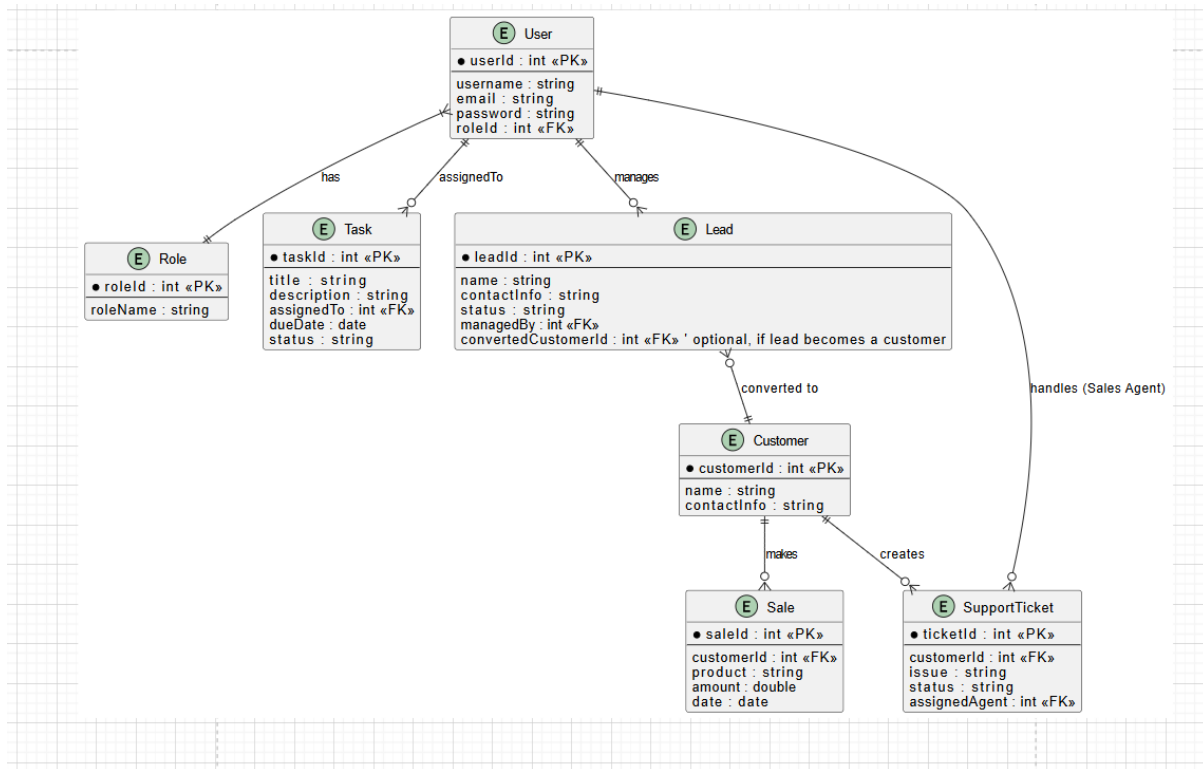
->Activity Diagram:



->Class Diagram:



->ER Diagram:



8. Interface Design

UI Pages:

- Login/Register
- Dashboard (based on role)
- Task list view
- Customer directory

APIs:

- POST /login

- GET /customers
- POST /tasks
- POST /notify