Investment Banking Deal Pipeline Management System

Complete System Documentation (Frontend + Backend + Database)

---

1. Introduction

Investment banking teams handle multiple high-value deals simultaneously, each progressing through complex stages such as **Prospect → Evaluation → Closure**. Traditionally, many small and mid-sized investment banks rely on spreadsheets, emails, and disconnected tools to manage this process, which leads to inefficiencies, security risks, and lack of visibility.

The **Investment Banking Deal Pipeline Management System** is a secure, centralized, web-based application designed to address these challenges by providing structured deal tracking, role-based access control, and secure collaboration.

---

2. Problem Statement

Existing deal management practices suffer from the following issues:

- Deal information scattered across spreadsheets, emails, and chat tools
- Multiple versions of Excel files edited by different users
- No control over who can view sensitive financial data
- Unstructured communication and notes
- Shared spreadsheets leading to unrestricted access

These limitations increase operational risk and reduce efficiency.

---

3. Solution Overview

The proposed system is a **secure, centralized Deal Pipeline Management Portal** that:

1. Centralizes all deal-related data
2. Enforces role-based access (USER vs ADMIN)
3. Protects sensitive financial information
4. Tracks the complete deal lifecycle
5. Enables structured collaboration through notes
6. Scales with team and deal growth
7. Meets modern security and compliance expectations

---

4. Technology Stack

4.1 Frontend

Layer	Technology
Framework	Angular 17
UI Library	Angular Material
State Management	Angular Services
HTTP	HttpClientModule
Forms	Reactive Forms
Routing	Angular Router
Styling	CSS / Angular Material Themes
Security	JWT Interceptor
Communication	REST API

4.2 Backend

Layer	Technology
Language	Java 17
Framework	Spring Boot
Security	Spring Security + JWT
ORM	Spring Data JPA (Hibernate)
Build Tool	Maven
Utilities	Lombok

4.3 Database

Layer	Technology
Database	MySQL
Type	Relational Database

---

5. Overall System Architecture

The system follows a **layered client-server architecture**.

```
CLIENT (Browser)
   |
   | Angular SPA (localhost:4200)
   |  - Components (Login, Deals, Admin)
   |  - Services & JWT Interceptor
   |
   ▼
```

```
SERVER (Spring Boot – localhost:8080)
  - Spring Security (JWT, RBAC)
  - REST Controllers
  - Service Layer
  - Repository Layer (JPA)
  |
  ▼
DATABASE (MySQL – localhost:3306)
  - Users
  - Deals
  - Notes
```

---

6. Frontend Architecture

6.1 Frontend Modules

- **Auth Module**: Login, JWT handling
- **Deal Module**: Deal listing, deal lifecycle, notes
- **Admin Module**: User and role management
- **Shared Module**: Reusable components and guards

6.2 Frontend Security

- JWT stored securely in browser storage
- HTTP Interceptor appends JWT to API requests
- Route Guards enforce role-based navigation

---

7. Backend Architecture

The backend follows a **layered architecture**:

```
Controller Layer → Service Layer → Repository Layer → Database
```

7.1 Key Backend Components

- **AuthController**: Authentication and JWT generation
- **DealController**: Deal CRUD operations
- **AdminController**: User and role management
- **JWT Filter**: Token validation
- **Service Layer**: Business logic and validation

---

8. Database Design

8.1 Users Table

Column	Description
id	Primary Key
username	Unique username
password	Encrypted password
role	USER / ADMIN

8.2 Deals Table

Column	Description
id	Primary Key
client_name	Client name
stage	Deal stage
value	Deal value

8.3 Notes Table

Column	Description
id	Primary Key
deal_id	Foreign key (Deal)
user_id	Foreign key (User)
note	Collaboration note

---

9. Application Flow

9.1 Login Flow

1. User accesses Angular application
2. Credentials are submitted to backend
3. Backend validates user credentials
4. JWT token is generated and returned
5. Frontend stores JWT
6. User is redirected to dashboard

9.2 Authorized API Access

1. Frontend sends API request with JWT
2. Spring Security validates token

3. Role-based authorization is applied
4. Data is fetched from database
5. Response is returned to frontend

---

10. Security Flow

- Stateless JWT-based authentication
- BCrypt password encryption
- Role-based endpoint access
- Secure REST communication

---

11. Key Features

- Secure login and authentication
- Role-based access control (RBAC)
- Deal lifecycle management
- Structured collaboration via notes
- Centralized data management
- Scalable and maintainable architecture

---

12. Non-Functional Requirements

- High security for sensitive financial data
- Scalability with increasing users and deals
- Maintainability using layered architecture
- High availability and performance

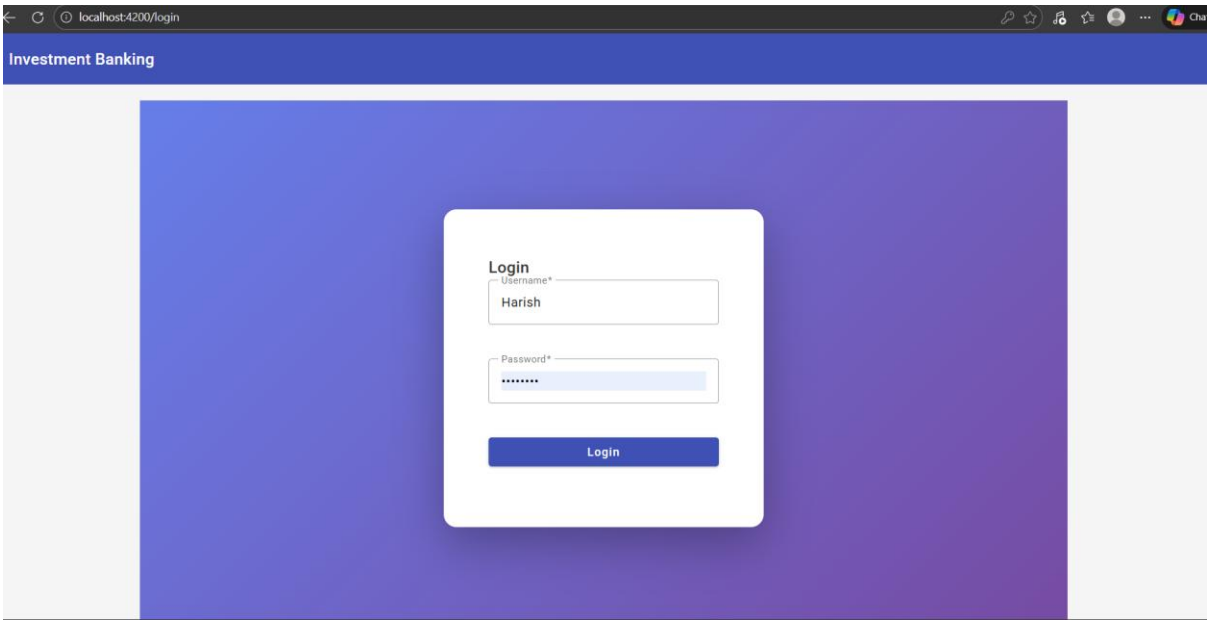
---

13. Screenshots (Application UI)

The following screenshots demonstrate key functionalities of the Investment Banking Deal Pipeline Management System.

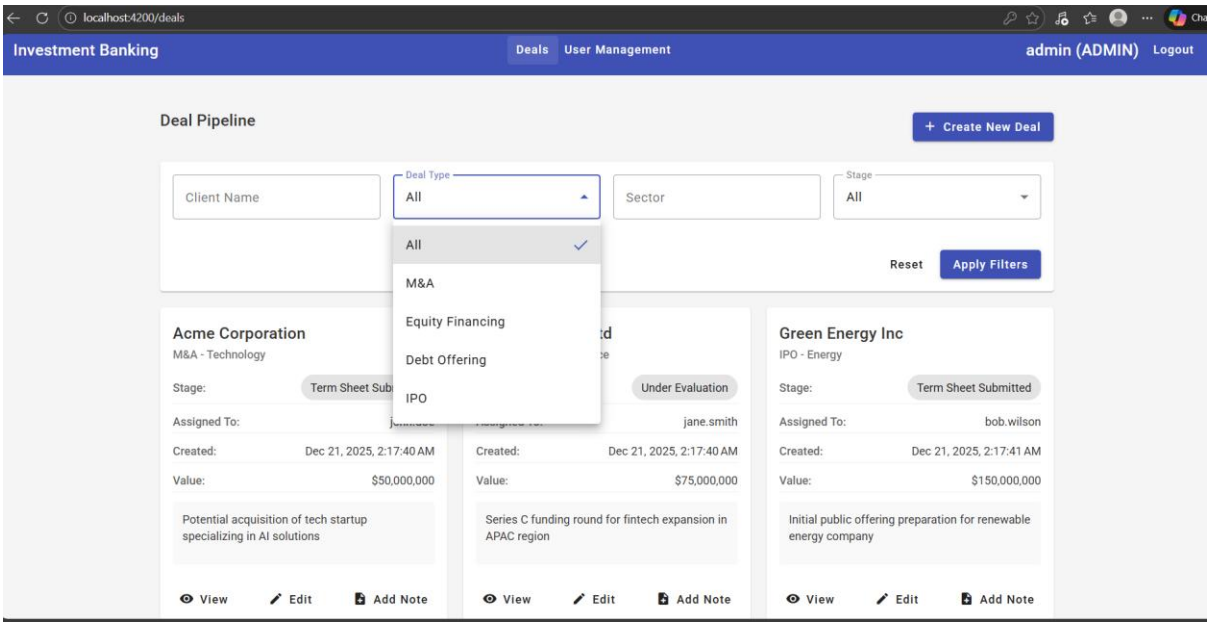
13.1 Login Page

- Secure login using username and password
- JWT-based authentication handled by backend

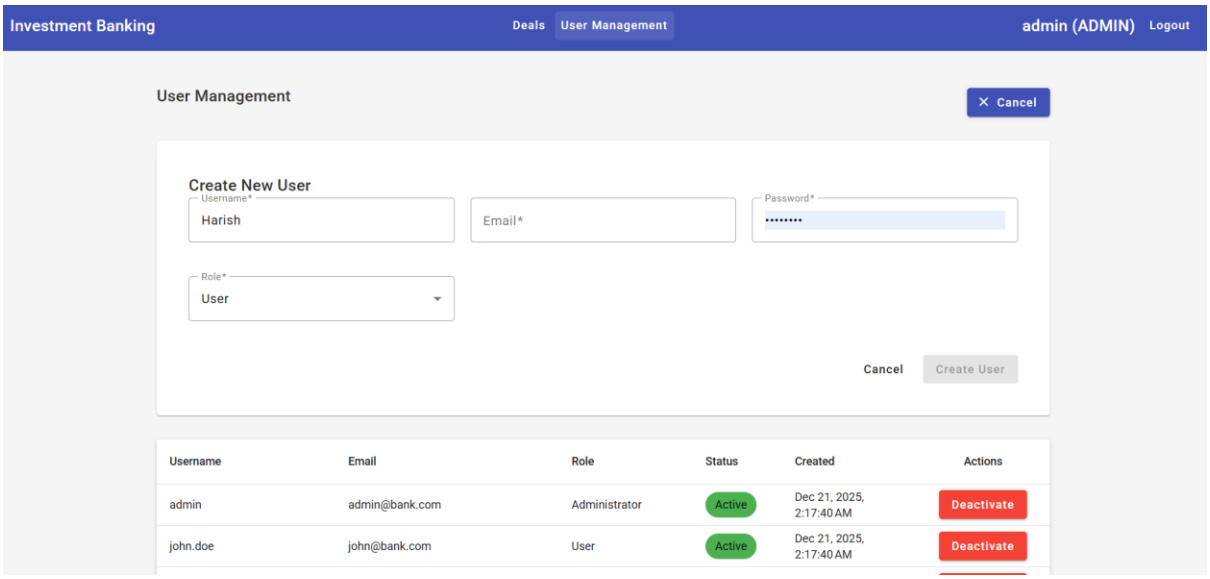


13.2 Deal Pipeline – Admin View

- View all deals across stages
- Filter by Deal Type, Sector, and Stage
- Create, Edit, View deals and add notes
- Full visibility including deal value

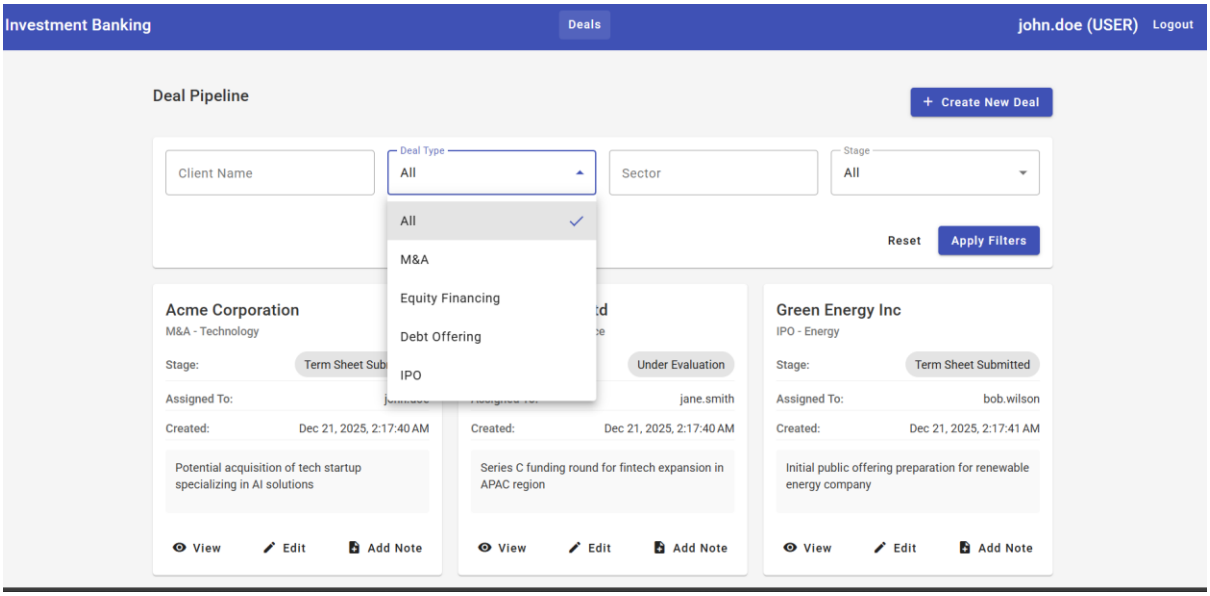


13.3 User Management – Admin Module

- Admin-only access
- Create new users
- Assign roles (ADMIN / USER)
- Activate or deactivate users
- 

13.4 Deal Pipeline – User View

- Restricted navigation (Deals only)
- Role-based access control enforced
- Users can view and update assigned deals
- Sensitive administrative options hidden

- 

---

14. Conclusion

The **Investment Banking Deal Pipeline Management System** provides a modern, secure, and scalable solution for managing investment banking deals. By replacing spreadsheets and emails with a centralized platform, the system significantly improves data security, collaboration, and operational efficiency.

This project demonstrates real-world full-stack development using **Angular, Spring Boot, JWT security, and MySQL**, making it suitable for enterprise use, academic submission, and professional portfolios.

Investment Banking Deal Pipeline Management System

Submitted by:

Your Name : Harish Karanam

Git : Backend: Akhimudiraj/Investment_Banking_backendd

Frontend: Akhimudiraj/Investment_Banking_Frontendd

Mail :karanamharish2003@gmail.com