**Customer Loan Management System for a Banking Client**

**Overview:**

Develop a web application that allows bank customers to manage loan applications, track the status of existing loans, and communicate with bank representatives. It includes features like loan application submission, document uploads, status tracking, payment schedules, and communication tools for customer support.

**Core Features:**

1. **User Authentication & Authorization:**
   * Use **Spring Security** with OAuth 2.0 for customer login and role-based access control (e.g., Admin, Customer, Bank Representative).
2. **Loan Application Module:**
   * Implement loan application forms using **Angular** (Version 15) with a dynamic form builder.
   * Use **Spring Boot** and **Spring Data JPA** to manage backend data related to loan applications.
   * Store application data in **MySQL** or **MongoDB**, depending on the requirement for structured or flexible data storage.
3. **Document Uploads:**
   * Allow customers to upload necessary documents (e.g., ID, income proof) using **AWS S3** for storage.
   * Use a microservice to handle document validation and storage with **Spring Boot** and **AWS SDK**.
4. **Loan Status Dashboard:**
   * Provide a dashboard for customers to check the current status of their loan applications.
   * Use **Angular Material** components for a clean and modern UI, including data tables, charts, and progress bars.
   * Implement asynchronous updates with **RxJS** in Angular.
5. **Loan Payment Scheduling:**
   * Create a scheduling service using **Spring Batch** to manage payment schedules and reminders.
   * Use **Java 17** with Spring Core features to handle concurrent processing of payments and updates.
6. **Customer Support Chat:**
   * Integrate a chat feature using **WebSockets** with **Spring MVC** for real-time communication between bank representatives and customers.
   * Store chat history in **NoSQL MongoDB** for fast retrieval and scalability.
7. **APIs & Integration:**
   * Develop **REST APIs** for managing loan applications, payment processing, and document handling using **Spring Boot**.
   * Utilize **SOAP** for secure transactions and communication with other internal banking systems if needed.
   * Expose APIs for integration with other bank systems or third-party analytics tools.

**Technology Stack:**

* **Frontend:** HTML5, CSS3, Bootstrap 3, JavaScript, TypeScript, Angular 15, Angular Material.
* **Backend:** Java (JDK 17), Spring Boot, Spring Security, Spring Data JPA, Spring Batch.
* **Database:** MySQL or MongoDB (depending on the data type), Oracle (for core banking data).
* **Web Services:** REST APIs, SOAP-based services using JAX-WS if internal communication needs it.
* **Cloud:** AWS (S3 for document storage, EC2 for deployment).
* **Development Tools:** Visual Studio Code, Eclipse, STS, Maven, log4j for logging.
* **Project Management:** Jira.
* **Version Control:** Git.
* **Operating Systems:** Linux, Windows.

This project would allow you to utilize a wide range of your skills while building a practical, scalable solution for the banking sector.

4o