Smart Habit Tracker - Project Report

# 1. Project Information

Group Number: GR00070

Project Title: Smart Habit Tracker

Course: Advanced Object-Oriented Programming

# 2. Team Members

• Nand Kapatel

• Srushti Chavan

# 3. Project Idea

The Smart Habit Tracker is a Java-based web application developed with Spring Boot and Thymeleaf. It enables users to build and manage personal habits such as 'Drink Water' or 'Go for a Walk'. The app tracks habit progress, allows users to mark daily completions, and sends notifications for reminders.

# 4. Technologies Used

• Java 21

• Spring Boot 3.2.5

• Spring Security

• Spring Data JPA

• Thymeleaf

• MySQL

• Maven

• Lombok

# 5. Object-Oriented Programming Concepts

• Inheritance - Entities may share common behavior via base classes.

• Polymorphism - Overridden methods and service implementations.

• Interfaces - Used in Repositories and Service Layer.

• Encapsulation - All fields are private with getters/setters.

• Generics - Used in repository interfaces like JpaRepository.

# 6. Application Design

The application follows the MVC (Model-View-Controller) architecture and uses layered design. Each component such as controllers, services, repositories, and entities are separated for maintainability and scalability.

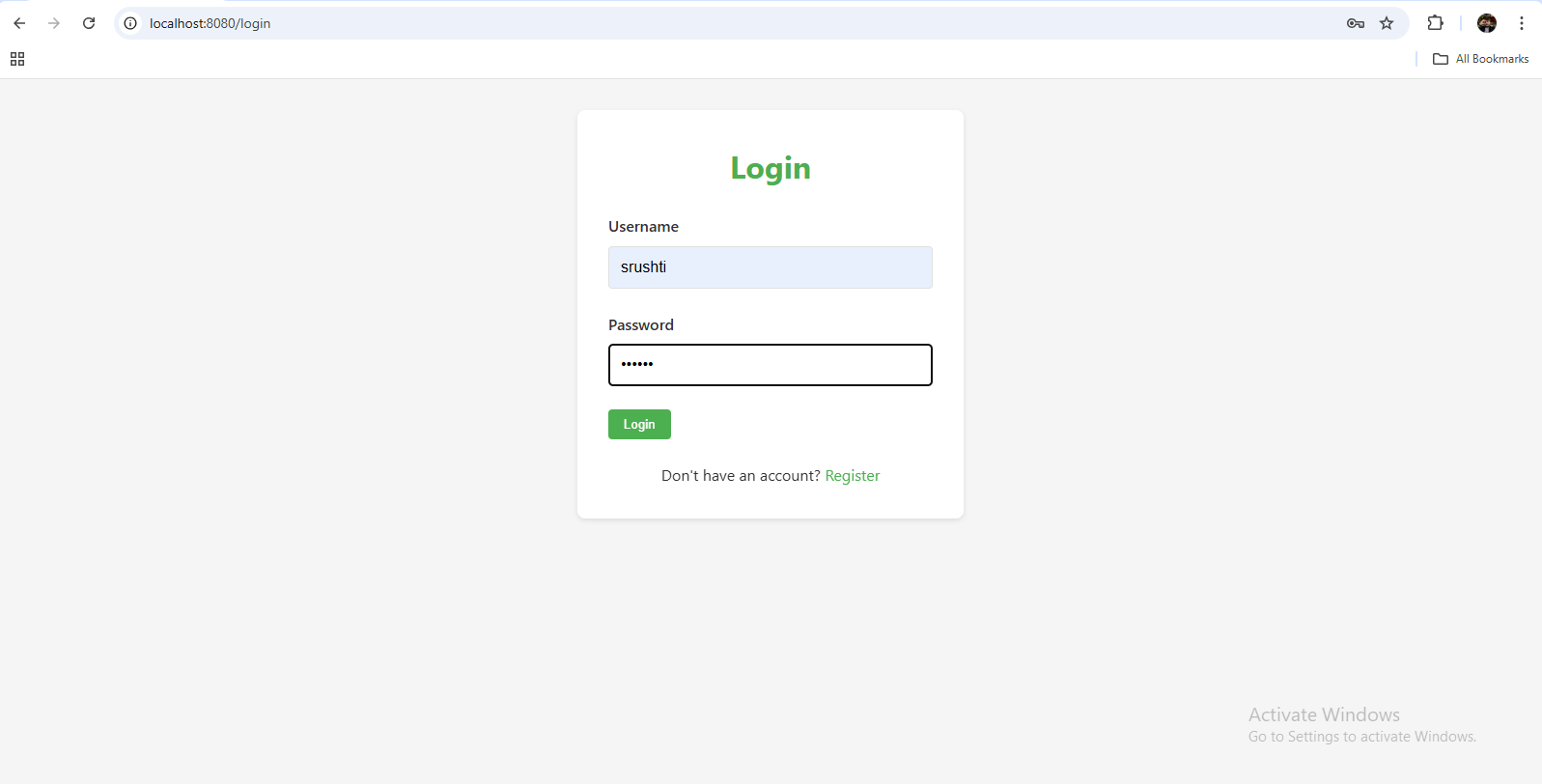
# 7. Inputs and Outputs

• Inputs: Habit name, frequency, user login data, registration form

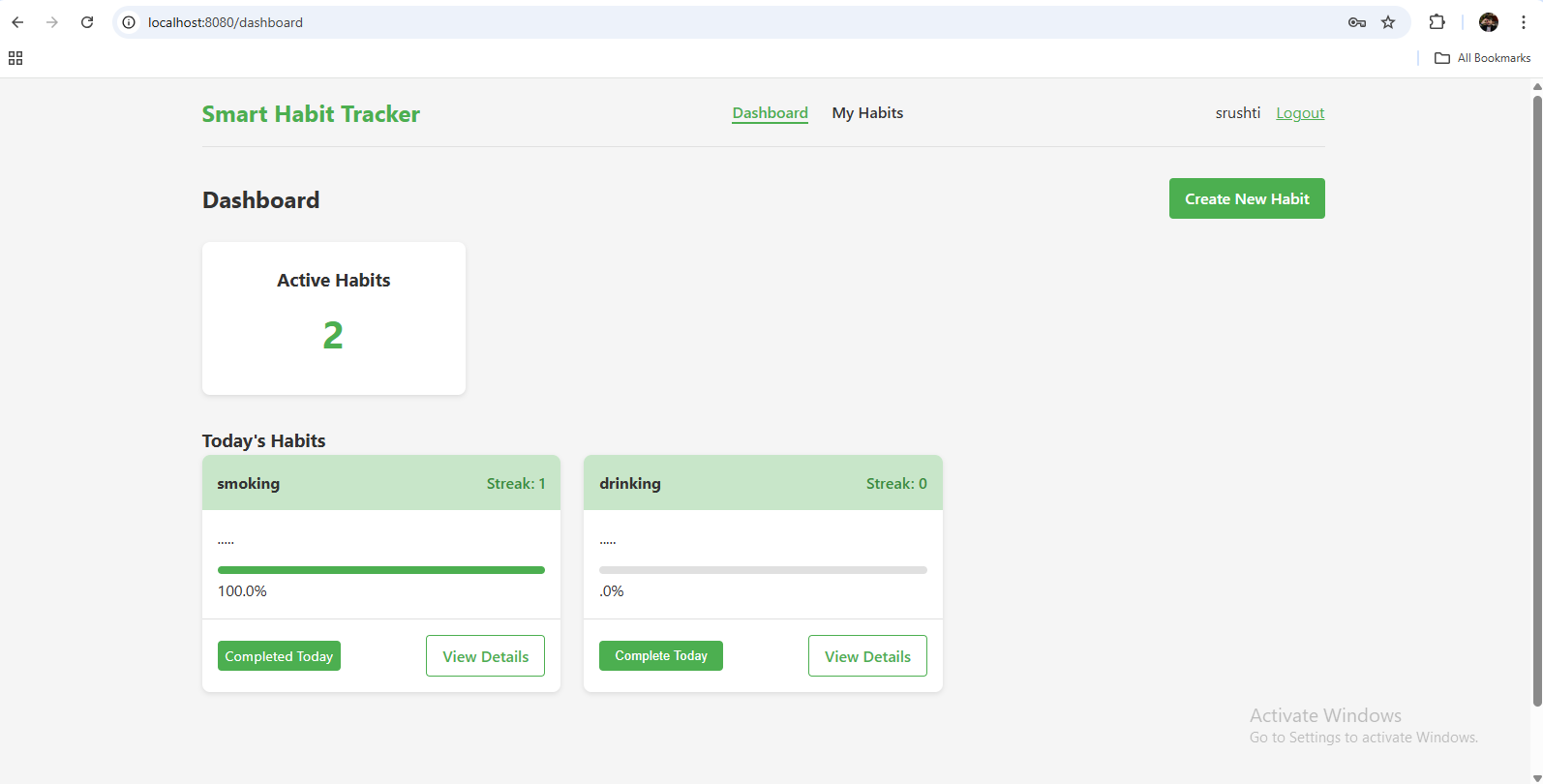
• Outputs: Dashboard displaying habits, notifications, and user progress

# 8. Screenshots

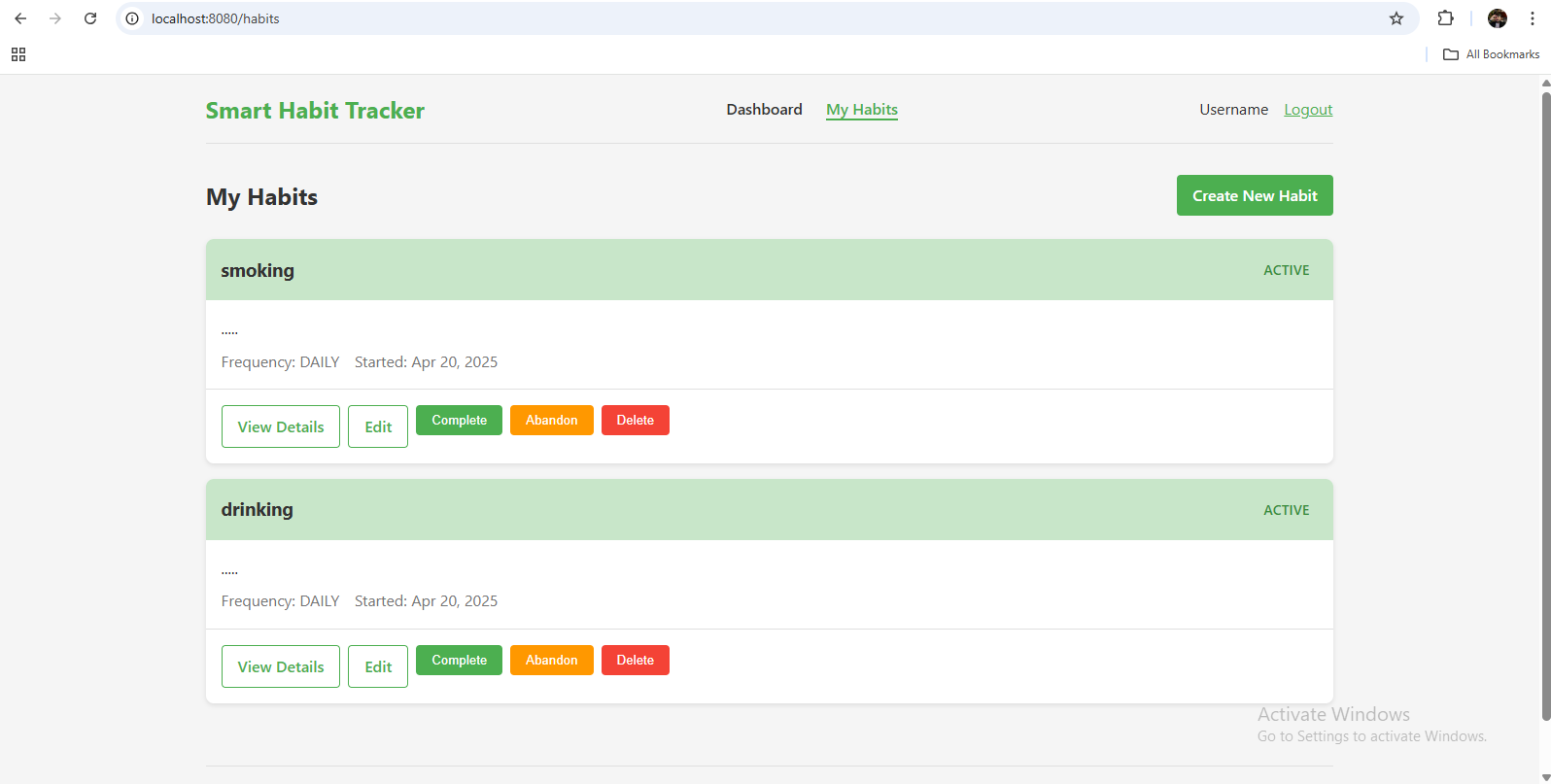
Login Page



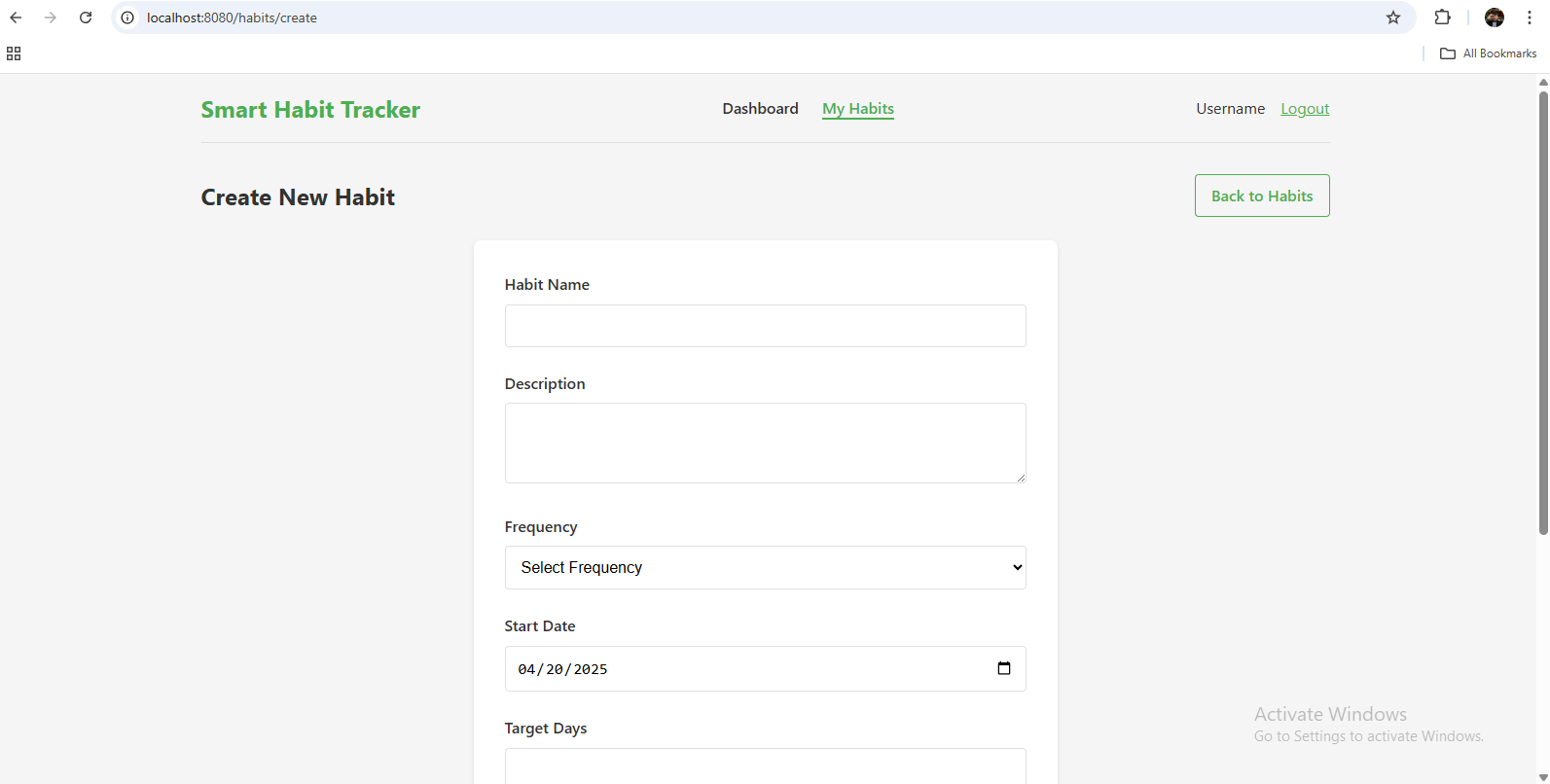
Dashboard



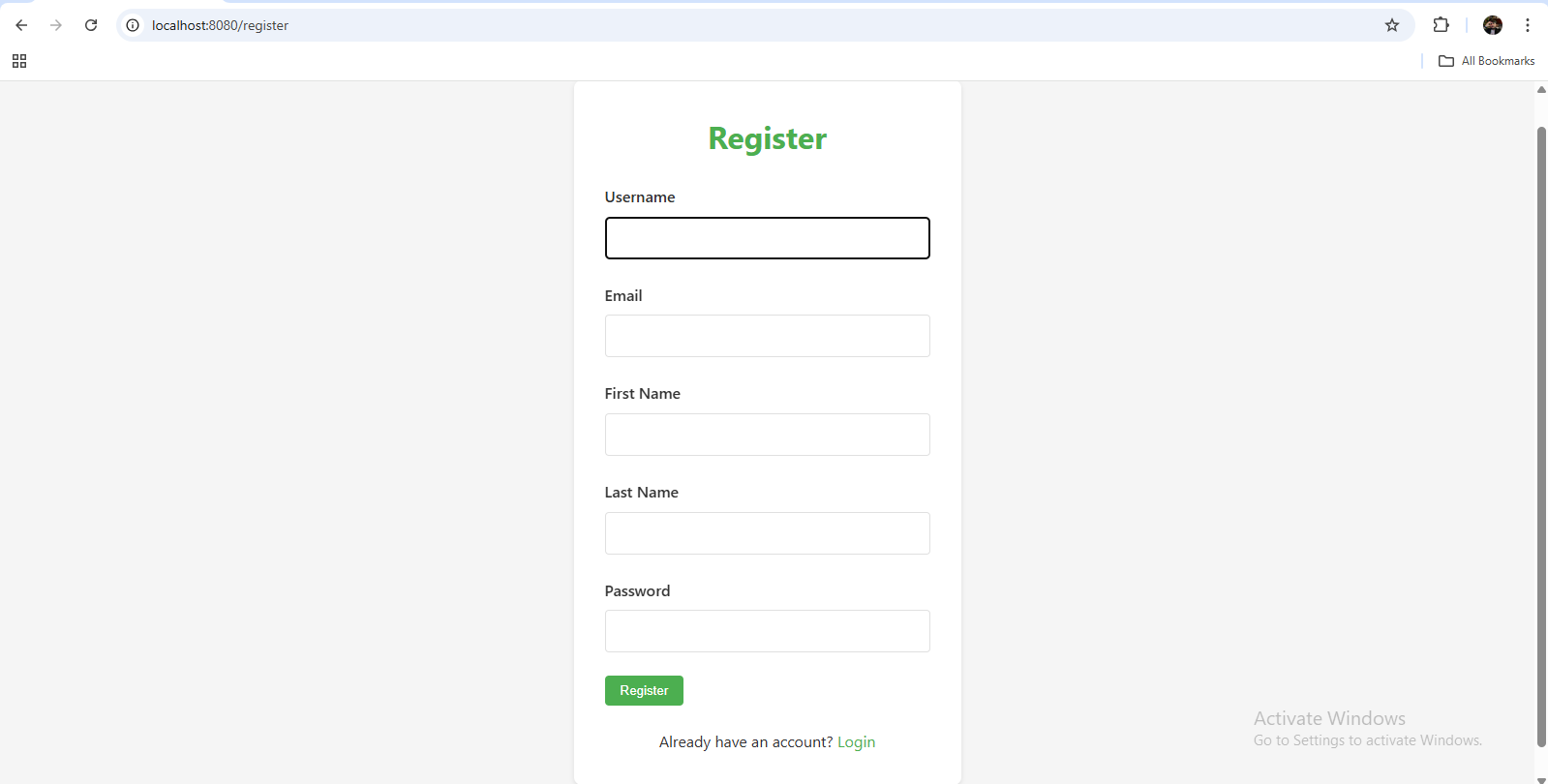
My Habits



Create Habit



Register Page

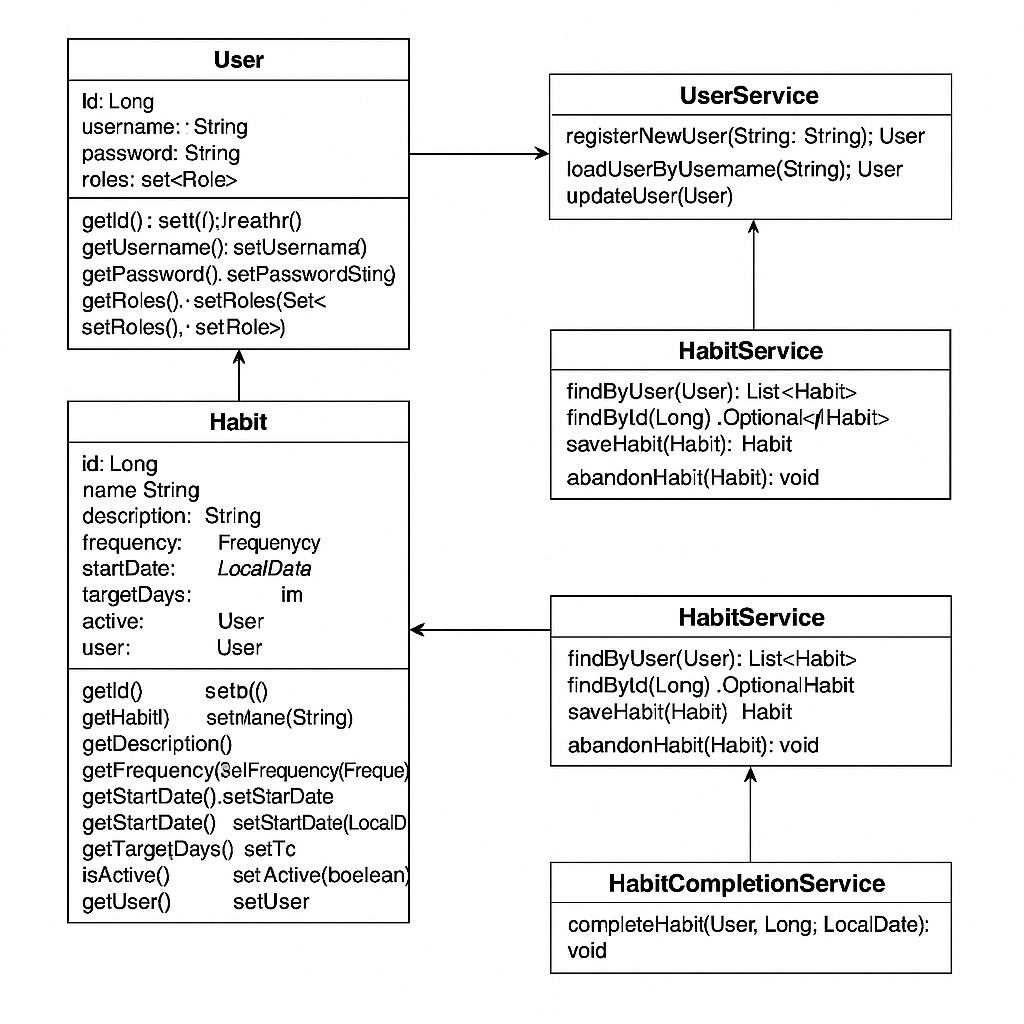


# 9. Team Contributions

• Nand Kapatel: Backend development, database design, security configuration

• Srushti Chavan: Frontend design (HTML/CSS), testing and styling, login configuration

# 10. UML



# 11. Challenges Faced

Some of the key challenges faced during development were:  
- Configuring Spring Security for login/logout handling  
- Setting up the database and resolving schema generation errors  
- Debugging Thymeleaf templates to connect frontend with backend  
- Implementing real-time notifications and scheduling reminders

# 12. Conclusion

This project helped us learn the implementation of Object-Oriented Programming concepts in a real-world Java Spring Boot application. It improved our understanding of MVC, database integration, authentication systems, and full-stack web development practices.