PROJECT DOCUMENTATION: STUDYSPHERE WEBSITE

**Table of Contents**

1. Introduction
2. Objectives
3. System Requirements
   * Hardware Requirements
   * Software Requirements
4. Features of the Website
5. Architecthure
6. User Roles and Functionalities
7. Implementation Details
   * Technologies Used
8. Testing
9. Deployment
10. Future Enhancements
11. Conclusion

**1. Introduction**

StudySphere is an interactive website designed to provide a centralized platform for study resources. It offers features such as categorized browsing, and collaborative tools for students and educators. Built using Java, HTML, and Maven, the platform ensures scalability, modularity, and ease of maintenance.

**2. Objectives**

* Provide a seamless platform for sharing and accessing study materials.
* Ensure secure data handling and resource management.
* Facilitate collaboration between students and educators.

**3. System Requirements**

**Hardware Requirements:**

* Processor: Intel i3 or higher
* RAM: 4GB minimum
* Storage: 10GB minimum

**Software Requirements:**

* Operating System: Windows/Linux/MacOS
* JDK: Java Development Kit 11 or higher
* Maven: Build tool
* Database: MySQL 8.0 or higher
* Web Browser: Chrome/Firefox/Edge
* IDE: IntelliJ IDEA/Eclipse

**4. Features of the Website**

* **Resource Management:** Upload, categorize, and manage study materials.
* **Search and Filters:** Advanced search with filters by subject, author, and type.
* **User Collaboration:** Discussion forums and comments on resources.
* **Personalized Dashboard:** User-specific recommendations and updates.
* **Download Management:** Secure and trackable resource downloads.

**5. Architecture**

StudySphere follows a three-tier architecture:

* **Presentation Layer:** HTML, CSS, and JavaScript for the front-end.
* **Business Logic Layer:** Java-based RESTful services.
* **Database Layer:** MySQL for data storage.

**6. User Roles and Functionalities**

**Administrator:**

* Manage users and resources.
* Moderate discussions and comments.

**Educator:**

* Upload and categorize resources.
* Engage in discussions.

**Student:**

* Browse, download, and comment on resources.
* Participate in discussions.

**7. Implementation Details**

**Technologies Used:**

* Programming Language: Java (Backend)
* Frontend: HTML, CSS, JavaScript
* Framework: Maven (Project Management)
* REST APIs: Implemented using Spring Boot

**8. Testing**

* **Unit Testing:** Validate individual modules and APIs.
* **Integration Testing:** Ensure seamless data flow between the front-end, back-end, and database.
* **User Acceptance Testing (UAT):** Gather feedback from a pilot group of students and educators.

**9. Deployment**

* Package the project as a WAR file using Maven.
* Deploy the application on a Tomcat server.
* Host the database on a cloud-based MySQL server.
* Configure DNS and SSL for the website.

**10. Future Enhancements**

* Integrate AI-based recommendations for resources.
* Add video tutorials and live sessions.
* Implement a mobile app for StudySphere.

**11. Conclusion**

StudySphere serves as a comprehensive platform for sharing and managing study resources. Its robust architecture and user-friendly design ensure accessibility and scalability, making it a valuable tool for learners and educators.