**BUSINESS REQUIREMENT DOCUMENT (BRD)**

Sujith Sam Saju (EmpId:- 285577) , Sanjeet Tumkur(EmpId:- 285485)

September 2024

**Project Name:** **E- Learning management System**

1. **INTRODUCTION**
   1. **Purpose**

This Business Requirements Document (BRD) outlines the development of an E-Learning Portal designed to facilitate students' shift from traditional classrooms to online learning. Key challenges include the lack of one-on-one interaction, which can lead to isolation and reduced motivation, potentially increasing dropout rates. Many students struggle with self-directed learning and personalized instruction, especially without AI support to adapt to individual learning styles. Additionally, managing deadlines and tracking submissions can cause stress, and the isolation of online learning may heighten anxiety. The E-Learning Portal aims to address these issues by offering a more interactive, supportive, and adaptable digital learning experience to enhance student engagement and success.

**1.2Scope**

The E- Learning Management System will include features like:

* + - Result Metrics: Students can view their overall scores of the courses they’ve enrolled in and track their performances.
    - Resource Support: Teachers can provide essential study materials directly to students, including YouTube URLs and PDFs. Additionally, professors have access to a resource section for searching books and materials related to their topics.

**1.3 Definitions**

* + - **Result Metrics:** Interactive features that facilitate communication and collaboration among students and between students and teachers. This may include discussion forums, group projects, peer feedback systems, and other tools designed to foster a supportive learning environment.

**1.4 Stakeholders**

* + - **End Users:** Students and teachers using the platform for educational purposes.
    - **Development Team:** Engineers and developers responsible for building and maintaining the application.

**Academic Institutions:** Schools, colleges, and universities that will implement and utilize the platform.

1. **BUSINESS OBJECTIVES**
   1. **Goals**

.

* + - **Provide Comprehensive Resource Support:** Ensure students have easy access to essential study materials, including digital content and educational resources.
  1. **Benefits**
     + **Centralized Performance Insights:** Provide students with a clear and comprehensive view of their performance metrics, enabling better tracking of progress and areas for improvement.
     + **Easy Access to Resources:** Offer students convenient access to lecture notes, reading lists, and other vital materials to support their learning.

1. **FUNCTIONAL REQUIREMENTS**

**Result Metrics**

**Details:**

* + - * **Visualize Scores:** Display scores in a bar graph format for each course.
      * **Performance History:** Provide a historical view of scores and trends over time to help students track their progress.
  1. **Resource Support**
     + **Requirement:** Teachers must be able to upload and organize study materials for students.
     + **Details:**
       - **Upload Resources:** Teachers can upload lecture notes, reading lists, practice exercises, and even YouTube links etc.
       - **Organize Content:** Resources are organized by course and topic for easy access.

1. **NON-FUNCTIONAL REQUIREMENTS**
   1. **Usability**
      * **Requirement:** The platform must be user-friendly and intuitive for both students and teachers.
      * **Details:**
        + **User Interface:** The UI should be clear, well-organized, and easy to navigate, with a consistent design.
        + **Cross-Device Compatibility:** The platform must be accessible on both desktop and mobile devices, ensuring a seamless experience across different screen sizes.
   2. **Performance**
      * **Requirement:** The platform should perform efficiently even under high user loads.
      * **Details:**
        + **Fast Loading Times:** Pages and resources should load quickly to minimize delays, particularly during peak usage times.
        + **Responsive Interactions:** The platform should remain responsive even with multiple active users.
        + **Efficient Data Handling:** The system should process and retrieve data efficiently, minimizing latency and optimizing overall performance.
   3. **Reliability**
      * **Requirement:** The platform must be reliable and available at all times, especially during critical academic periods.
      * **Details:**
        + **High Availability:** The platform should have minimal downtime and a robust disaster recovery plan in place.
        + **Robust Infrastructure:** Ensure a stable and reliable backend to support all users with regular maintenance and upgrades
        + **Error Handling:** Implement comprehensive error tracking and handling mechanisms to address issues promptly and minimize disruptions
   4. **Scalability**
      * **Requirement:** The platform must be able to scale according to increasing user numbers and features.
      * **Details:**
        + **Horizontal and Vertical Scaling:** Support both horizontal scaling (adding more servers) and vertical scaling (upgrading existing servers) to handle growing demands.
        + **Adaptability:** The platform should be flexible and easily adaptable to accommodate future growth and evolving requirements.
        + **Load Balancing**: Implement load balancing techniques to distribute traffic evenly across servers and ensure optimal performance.
   5. **Interoperability**
      * **Requirement:** The platform should be compatible with various systems, devices, and educational tools.
      * **Details:**
        + **Cross-Platform Compatibility:** Support for different operating systems and browsers to ensure broad accessibility.
        + **Support for Common File Formats:** Ensure the platform can handle common file formats such as CSV, PDF, and DOCX for easy data import and export.
        + **Data Exchange Standards:** Adhere to standard data exchange protocols and formats to ensure seamless interoperability with external systems.
2. **COMPONENTS AND SERVICES**

**Frontend (Angular, HTML, CSS, JavaScript, TypeScript):**

• **Components:**

* **UserComponent:** For handling CRUD operations related to the User database.
* **ProfessorComponent:** For handling CRUD operations related to the professor database.
* CourseComponent: For handling CRUD operations related to the course database

**Backend (Java, Spring Boot):**

• **Services:**

* **UserService:** Manages the database interactions for users.
* **ProfessorService:** Manages the database interactions for professors.
* **CourseService:** Manages the database interactions for course.

1. **INTERACTION**
   1. **Frontend and Backend Communication**
      * **HTTP Requests:** Angular frontend communicates with Spring Boot backend via HTTP requests for all operations.
      * **Data Binding:** Data models are exchanged between frontend and backend to keep the application synchronized.
   2. **Real-Time Updates**
      * **Angular Components:** Reactively update the UI based on data changes from the backend.
      * **Spring Boot Services:** Process requests and manage data, ensuring consistency and reliability.