

**STUDENT NAME : AMRITHA V**

**REGISTER NUMBER : 7376221CD102**

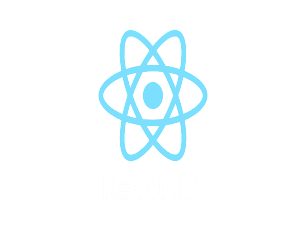
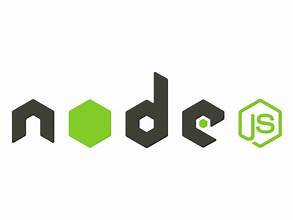
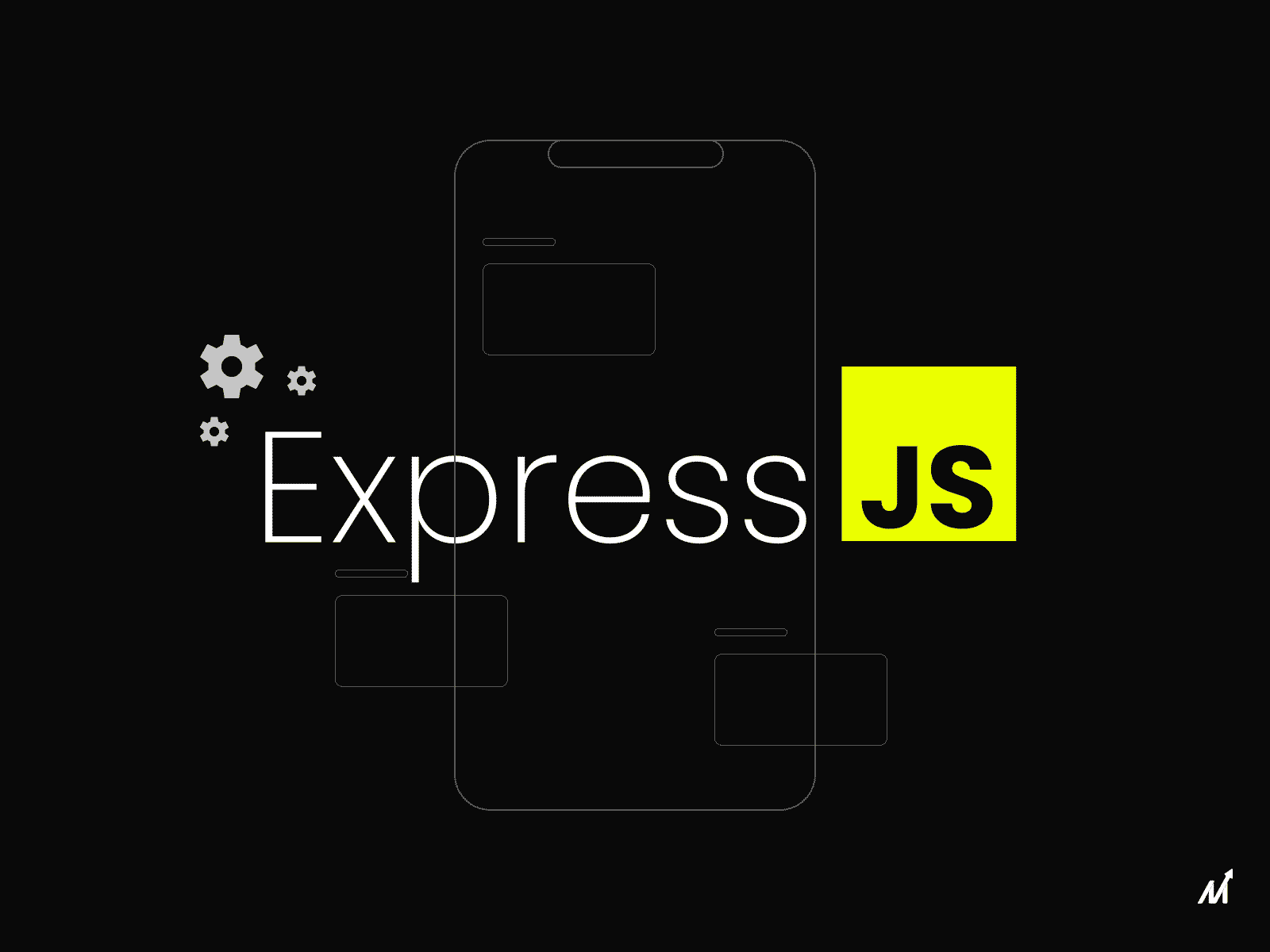
**SEAT NUMBER : 160**

**PROJECT ID : 29**

**PROJECT TITLE : COURSE FILE AUTOMATION**

**TECHNOLOGY STACK ALLOCATED : MERN STACK**

**TECHNICAL COMPONENTS :**

**   **

**React.js Node.js Express.js MongoDB**

|  |  |
| --- | --- |
| **Front -End** | **React.js** |
| **Back - End** | **Node.js , Express.js** |
| **Database** | **MongoDB** |
| **API** | **Express.js** |

**IMPLEMENTATION TIMELINE :**

|  |  |  |
| --- | --- | --- |
| **PHASE** | **STATUS** | **NOTES** |
| **STAGE 1** | In progress | **Planning and Requirement Gathering** |
| **STAGE 2** | Not started | **Design and Prototyping** |
| **STAGE 3** | Not started | **DB Designing** |
| **STAGE 4** | Not started | **Back-End Implementation** |
| **STAGE 5** | Not started | **Testing and Implementation** |

**Table showing where each component of the MERN stack comes into play for the faculty dashboard application:**

|  |  |  |
| --- | --- | --- |
| **COMPONENT** | **DESCRIPTION** | **SPECIFIC ROLE** |
| **MongoDB** | **NoSQL database** | **Stores:**  **- Faculty credentials and profiles**  **- Course information and details**  **- Criterion definitions and data** |
| **Express.js** | **Node.js web framework** | **Handles:**  **- API requests for faculty login and authentication**  **- API requests for data entry and PDF report generation** |
| **React.js** | **Frontend JavaScript library** | **Faculty dashboard UI components |**  **- Data entry forms and PDF report** |
| **Node.js** | **JavaScript runtime environment** | **- Express.js server for API requests**  **- Connects to MongoDB for data storage and retrieval** |

**PROBLEM STATEMENT :**

**The Current Process of Tracking and Documenting Academic Course Data is Inefficient :**

The existing process of tracking and documenting academic course data is manual, time-consuming, and prone to errors. Faculty members spend a significant amount of time and effort collecting, recording, and updating course data, leading to inefficiencies and inaccuracies. This manual process lacks automation, resulting in:

- Inconsistent data entry and formatting

- Errors in data transcription and recording

- Difficulty in tracking and updating course changes

- Inefficient use of faculty time and resources

- Limited accessibility and sharing of course data

**The Need for a Streamlined and Automated System**

To address these challenges, faculty members require a streamlined and automated system to input and track data related to academic courses. This system should ensure accuracy, efficiency, and ease of use, enabling faculty to focus on teaching and research rather than administrative tasks. The ideal system should:

- Automate data entry and tracking

- Ensure consistency and accuracy in data recording

- Provide easy access and sharing of course data

- Save faculty time and resources

- Enhance the overall management of academic course data

**Benefits of an Automated System :**

An automated system for tracking and documenting academic course data would bring numerous benefits, including:

- Improved accuracy and consistency in course data

- Enhanced efficiency in data management

- Increased productivity for faculty members

- Better accessibility and sharing of course data

- Informed decision-making through data analysis

**PROJECT WORKFLOW :**

**Step 1: Faculty Login**

Faculty members log in to the application using their unique credentials, ensuring secure access to their respective dashboards.

**Step 2: Enter Course Information**

Faculty enter essential course details, including course code, title, and faculty name, to identify the course and faculty associated with the data entry.

**Step 3: Select Criterion**

Faculty select one of the 21 predefined criteria to enter data for, ensuring that all necessary information is captured.

**Step 4: Enter Data for Selected Criterion**

Faculty enter data for the selected criterion in the provided format, ensuring accuracy and consistency.

**Step 5: Save and Submit Data**

Faculty save and submit their entered data, which is then stored in the application database.

**Step 6: Generate PDF Report**

The application generates a PDF report based on the entered data, providing a comprehensive summary of the course information.

**Step 7: HOD Review and Approval**

The Head of Department (HOD) reviews and approves the generated PDF report, ensuring that the data is accurate and meets the required standards.  
  
**FUNCTIONAL REQUIREMENTS :**

**1. User Authentication**

The application shall provide a secure login mechanism for faculty members to access their respective dashboards using unique credentials.

**2. Course Information**

The application shall allow faculty members to enter essential course details, including course code, title, and faculty name, to identify the course and faculty associated with the data entry.

**3. Criterion Selection**

The application shall provide a dropdown list of 21 predefined criteria for faculty members to select from, ensuring that all necessary information is captured.

**4. Data Entry**

The application shall provide a user-friendly interface for faculty members to enter data for the selected criterion in the provided format, ensuring accuracy and consistency.

**5. PDF Report Generation**

The application shall automatically generate a PDF report based on the entered data, providing a comprehensive summary of the course information.

**6. HOD Review and Approval**

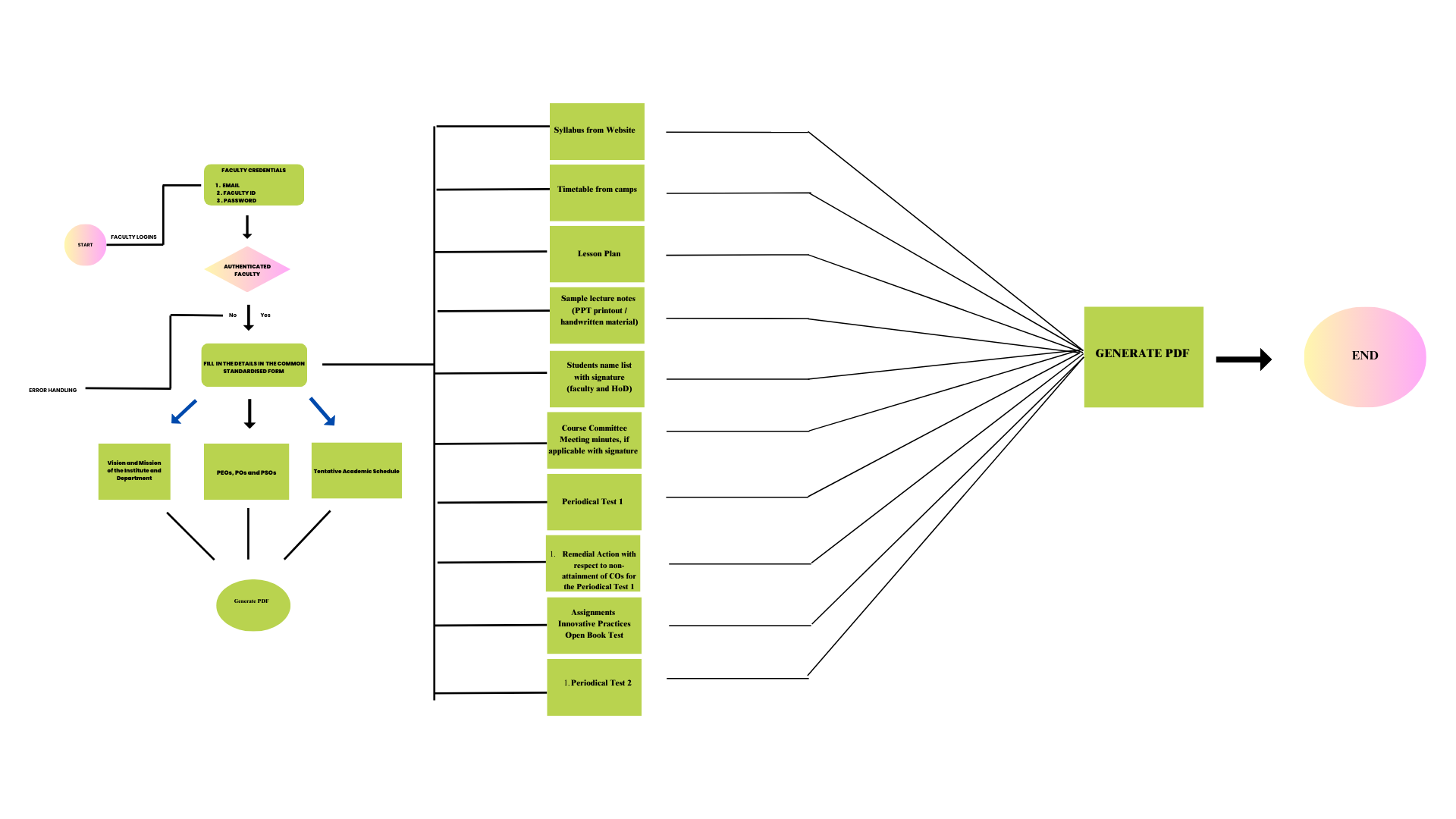
The application shall allow the Head of Department (HOD) to review and approve the generated PDF report, ensuring that the data is accurate and meets the required standards.

**7. Data Security**

The application shall implement measures to ensure data security, integrity, and confidentiality, including encryption and access controls.

**8. User Support**

The application shall provide faculty members with training and support to effectively use the application, including user guides and helpdesk support.

****