Marathwada Mitramandal's

COLLEGE OF ENGINEERING, PUNE

An Autonomous Institute

Project Based Learning

Title of Project : DEPARTMENT CALENDER

Sr.No	Name of Students	PRN No
1	**sairam pampatwar**	**B24IT1051**
2	**Aditya Patil**	**B24IT1127**
3	**Ranveer Shinde**	**B24IT1053**
4	**Varad Rajboinwad**	**B24IT1059**
5		

Date: **Faculty In-Charge**

1) Research :

Objective: To create a functional Department Calendar that allows event management and provides reminder to users.

Scope:

- Offer an intuitive user interface for organizing departmental events.

- Ensure the solution is responsive and accessible across devices.

Findings:

- **Existing Solutions**: Tools like Google Calendar and Microsoft Outlook are robust but may lack customizability for specific departmental needs.

- **Challenges**:

    - Easy-to-use event management with reminders.

    - Compatibility with various browsers and devices.

2) Analysis :

- Functional:
  - Dynamic calendar views (month).
  - Add, edit, and delete events.
  - Set reminders for specific events.
- Non-Functional:
  - User-friendly interface.
  - Fast and responsive.
  - Data persistence using local storage.

Technological Stack:

- Frontend: HTML, CSS, JavaScript.

3) Ideate :

1. -Core Features:
   - Monthly calendar view with navigation controls.
   - Event form for adding details (name, date, description, reminder).
2. Additional Features:
   - Drag-and-drop functionality for rescheduling.
   - Event search/filter.

- **User Flow:**

- **Home Screen: Displays the calendar with navigation controls.**
- **Add/Edit Event: Opens a form in a modal overlay.**
- **Reminders: Displays a notification at the scheduled time.**

4) Build :

Frontend Development:

- Structure the layout using HTML.
- Style the calendar for responsiveness and accessibility.
- Write JavaScript functions to dynamically generate calendar views and handle user actions.

5) Test:

Testing Process:

- **Unit Testing**:
  - Ensure date calculations are accurate for leap years and varying month lengths.
  - Test event addition and removal functionality.

- **Performance Testing**:
  - Stress-test local storage by adding hundreds of events.

6) Implement:

Deployment:

1. **Local Deployment**: Host the application locally for initial use and further testing.
2. **Live Deployment**:
   - Use GitHub Pages.

     .

Training and Handover:

- Provide documentation with instructions on adding, editing, and managing events.