

**COLLEGE CODE:[8223]**

**COLLEGE NAME: [vandayar engineering college ]**

**DEPARTMENT: [Computer science and engineering]**

**STUDENT NM-ID:**

**[532DCAE5AE1924E97FB408085D4F2B4D]**

**ROLL NO:[822323104019]**

**DATE:[26.09.2025]**

**Completed the project named as**

**Phase-3**

**TECHNOLOGY PROJECT NAME: Event Scheduling App**

**SUBMITTED BY,**

**NAME:[parameshwari.M]**

**MOBILE NO:[9043728261]**

## **PROJECT SETUP:**

- ❖ Create repository in GitHub and set up branching strategy.
- ❖ Initialize frontend (React) and backend (Node.js/Express) folders.
- ❖ Configure database connection (PostgreSQL / MongoDB).
- ❖ Install core dependencies and set up environment variables.
- ❖ Establish folder structure for scalability (components, services, tests).
- ❖ Set up Continuous Integration (CI) pipeline for automated builds.

## CORE FEATURES IMPLEMENTATION :

- ❖ User registration and login with secure authentication.
- ❖ Event creation and editing for organizers.
- ❖ Display of upcoming events in calendar or list view.
- ❖ RSVP and attendee management for users.
- ❖ Real-time updates or automatic refresh of event list.
- ❖ Notification or email alerts for event changes.

## DATA STORAGE (LOCAL STATE & DATABASE) :

- ❖ Use Redux/Context API to manage local UI state.

- ❖ **Store event and user data in PostgreSQL or MongoDB collections/tables.**
- ❖ **Implement CRUD operations for events and RSVPs.**
- ❖ **Securely hash and store user credentials.**
- ❖ **Create backup and restore strategies for database.**
- ❖ **Optimize queries to handle large event datasets efficiently.**

## **TESTING CORE FEATURES :**

- ❖ **Write unit tests for critical frontend components and backend services.**
- ❖ **Perform integration testing for API endpoints.**

- ❖ **Conduct manual testing of user flows (login, create event, RSVP).**
- ❖ **Implement automated test scripts to catch regressions.**
- ❖ **Test performance and scalability with sample data.**
- ❖ **Document issues and track fixes in GitHub Issues board.**

## **VERSION CONTROL (GITHUB) :**

- ❖ **Follow a clear branching model (main/develop/feature branches).**
- ❖ **Use pull requests and peer reviews before merging.**
- ❖ **Tag stable releases for milestone tracking.**

❖ Enable GitHub Actions for CI/CD pipelines.

❖ Maintain detailed commit messages and changelogs.

❖ Protect main branch with review and test requirements.

The screenshot shows a code editor interface with the following details:

- Title Bar:** Shows "File Edit Selection View ..." and a search bar with "html nm".
- Left Sidebar:** Includes icons for File, Edit, Selection, View, and a search icon.
- Central Area:** Displays the file "homepage.html" with the following content:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Event Scheduling App - Home</title>
<style>
body {
    font-family: 'Times New Roman', serif;
    margin: 0;
    padding: 0;
    background-color: #f9f9f9;
    text-align: center;
}
header {
    background-color: #4CAF50;
    color: white;
    padding: 20px;
}
nav a {
    color: white;
    margin: 0 15px;
    text-decoration: none;
    font-weight: bold;
}
```
- Right Panel:** Features the "Welcome to Copilot" message, a "Let's get started" button, and three buttons for "Add context (#)", "Build Workspace", and "Show Config". It also includes a note: "Review AI output carefully before use."
- Bottom Bar:** Shows "Ln 11, Col 1" and other file status indicators.

