

PROFESSIONAL TRAINING REPORT - II

entitled

LECTURER LOCATOR: A WEB BASED APPLICATION FOR FINDING LECTURER LOCATIONS

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering with
specialization in Cyber Security

by

K.V.S.Maruthi Varaprasad [41614049]

Kondeti Jeevandeep [41614047]

Kandukuri Abhiram [41614042]

Kondeti Dhanush [41614046]



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

CATEGORY -1 UNIVERSITY BY UGC

**Accredited with Grade "A++" by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119**

MAY 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

CATEGORY-I UNIVERSITY BY UGC

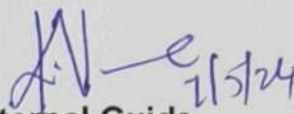
Accredited "A++" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

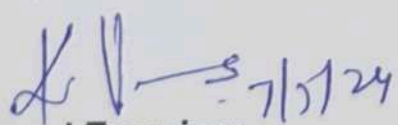
This is to certify that this Professional Training Report is the bonafide work of Mr.K.V.S.Maruthi Varaprasad (41614049), Mr.Kondeti Jeevandeep (41614047), Mr.Kandukuri Abhiram (41614042), Mr.Kondeti Dhanush (41614046) who carried out the project entitled "LECTURER LOCATOR: A WEB-BASED APPLICATION FOR FINDING LECTURER LOCATIONS" under my supervision from January 2024 to May 2024.

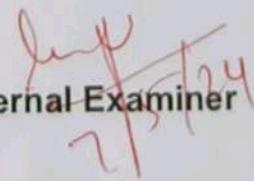

Internal Guide

Dr. K. Veena, M.E., Ph.D.,


Head of the Department
Dr. S. VIGNESHWARI, M.E., Ph.D.,

Submitted for Viva voce Examination held on 7/5/24


Internal Examiner

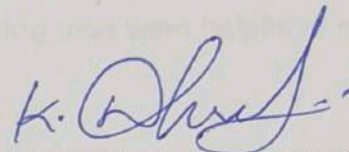

External Examiner

DECLARATION

I, **Kondeti Dhanush (Reg.No-41614046)**, hereby declare that the Professional Training Report-II entitled "**LECTURER LOCATOR: A WEB-BASED APPLICATION FOR FINDING LECTURER LOCATIONS**" done by me under the guidance of **Dr.K. Veena, M.E., Ph.D.**, is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering with specialization in Cyber Security.

DATE: 07-05-2024

PLACE: Chennai



SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

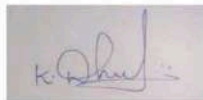
I convey my thanks to **Dr. T.Sasikala M.E., Ph.D., Dean**, School of Computing, **Dr. S.Vigneshwari M.E., Ph.D., Head of the Department of Computer Science and Engineering** for providing me necessary support during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Internal Guide **Dr. K. Veena M.E., Ph.D.**, for her valuable guidance, suggestions and constant encouragement which paved way for the successful completion of my phase-2 Professional Training.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

COURSE CERTIFICATE

AFTERNOON SESSION (AN)	National Programme on Technology Enhanced Learning	
Hall Ticket For	2024 Apr: CS26 Foundation of Cloud IoT Edge ML - Online	NPTEL

Candidate Name	kondeti dhanush					
Roll No	NOC24CS26S653402642		Seating Number	53402642		
Date of Birth	26-01-2004					
PwD Status	No	Compensatory Time Required	N.A	Scribe Required	N.A	
Exam Date	Sunday, 21 April, 2024					
Reporting Time	01:00 pm		Gate Closure	02:30 pm		
Exam Timing	02:00 pm		Shift	AN		
Test Centre Name	iON Digital Zone IDZ Kovilambakkam					
Test Centre Address	Fortune Towers, 1st and 2nd Floor, SH 109, 200 Feet Thoraipakkam Pallavaram Radial Road, Near Eachangadu signal, Kovilambakkam, Chennai, Tamil Nadu, India - 600117					
 NPTEL Coordinator						

NPTEL EXAM - 21 APRIL, 2024
General instructions for candidates - AN
(All timings mentioned here are in IST)

The total duration of the examination is 180 minutes.
Candidates will be permitted to leave the examination hall only after 03:30 pm, on a need basis.

Hall ticket and Entry:

1. The Hall Ticket must be presented for verification along with one original photo identification (not photocopy or scanned copy). Examples of acceptable photo identification documents are School ID, College ID, Employee ID, Driving License, Passport, PAN card, Voter ID, and Aadhaar-ID. A printed copy of the hall ticket and original photo ID card should be brought to the exam centre. Hall ticket and ID card copies on the phone will not be permitted.
2. This Hall Ticket is valid only if the candidate's photograph and signature images are legible. To ensure this, print the Hall Ticket on A4-sized paper using a laser printer, preferably a color photo printer.
3. **TIMELINE:** 1:00 pm - Report to the examination venue | 1:40 pm - Candidates will be permitted to occupy their allotted seats | 1:50 pm - Candidates can login and start reading instructions prior to the examination | 2:00 pm - Exam starts | 2:30 pm - Gate closes, candidates will not be allowed after this time | 3:30 pm Submit button will be enabled | 5:00 pm exam ends.
4. Candidates will be permitted to appear for the examination ONLY after their credentials are verified by center officials.

P.T.O.

ABSTRACT

The "Lecturer Locator" is a web-based application designed to facilitate the process of finding the locations of lecturers within an educational institution. The application provides users with a user-friendly interface where they can select a lecturer from a dropdown menu and instantly retrieve information about the lecturer's current location, including the block number, floor number, and room number. Additionally, the application displays the leisure time of the selected lecturer along with the leisure time location. The main features of the application include a responsive design, allowing seamless access from various devices, and an intuitive user interface for easy navigation. The inspiration behind the development of the "Lecturer Locator" stems from the recognition of the importance of effective communication and resource optimization within educational institutions. Furthermore, the application also includes details about the leisure time of lecturers and their preferred leisure time locations, allowing users to plan their interactions accordingly. The application is built using HTML, CSS, and JavaScript, making it lightweight and easily deployable across different platforms.

The "Lecturer Locator" aims to enhance efficiency and convenience for students, staff, and visitors within the educational institution by providing them with accurate and up-to-date information about lecturer locations. By centralizing this information in a single platform, the application streamlines the process of locating lecturers, saving time and reducing potential confusion. In enhancing communication and coordination within the educational environment, the application contributes to a more productive and conducive learning atmosphere. Its comprehensive functionality not only improves the overall experience for users but also fosters a sense of connectivity and collaboration among stakeholders. Overall, the "Lecturer Locator" serves as a valuable tool for improving the overall experience within the educational institution, promoting productivity, and fostering a conducive learning environment. With its user-friendly interface and robust functionality, the application contributes to enhancing communication and coordination among students, staff, and lecturers, ultimately enriching the educational experience.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	v
	LIST OF FIGURES	viii
1	INTRODUCTION	1
	1.1 Overview	
2	LITERATURE REVIEW	4
	2.1 survey	
3	REQUIREMENTS ANALYSIS	
	3.1 Objective Of The Project	7
	3.2.1 Hardware Requirements	10
	3.2.2 Software Requirements	11
4	DESIGN DESCRIPTION OF PROPOSED PRODUCT	
	4.1.1 Ideation Map/Architecture Diagram	13
	4.1.2 Various stages	14
	4.1.3 Internal or Component design structure	15
	4.1.4 working principles	18
	4.2 4.2.1 Novelty of the Project	22
5	RESULTS AND DISCUSSION	23
6	SUMMARY AND CONCLUSIONS	25
	REFERENCES	26
	APPENDIX	
	A. Research Paper	27
	B. Source Code	32

LIST OF FIGURES

Figure No.	Figure Name	Page No.
4.1	Web Application Architecture	13
4.2	Styling Using HTML And CSS	15
4.3	Selector Operator Using JavaScript	16
4.4	Database Connectivity Using JavaScript	17
4.5	User Interface	20
4.6	User Interaction	21
4.7	Outcome If Data Does Not Exist	21

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

The "Lecturer Locator" project emerges as a ground breaking initiative poised to transform the navigation and communication landscape within educational institutions. Recognizing the inherent challenges faced by students, faculty, and staff in locating lecturers across sprawling campuses, the project sets out to revolutionize the way users interact with their educational environment. By harnessing the power of modern web technologies and best practices in user experience design, the project aims to provide a comprehensive solution that streamlines the process of locating lecturers in real-time. The rationale behind the project stems from the limitations of traditional navigation methods, such as static maps and printed directories, which often fail to provide up-to-date information and personalized experiences.

Moreover, the dynamic nature of lecturer movements within educational institutions necessitates a digital solution that can adapt to changing circumstances and user preferences. The objectives of the project are guided by a commitment to delivering a user-centric platform that offers real-time updates, interactive map interfaces, personalized user profiles, integration with external services, and accessibility features. These features collectively aim to enhance the navigation experience, promote efficient communication, and foster a positive learning environment within educational institutions. The significance of the project extends to various stakeholders, including students, faculty, staff, and administrators, who stand to benefit from its innovative features and user-friendly design.

Ultimately, the "Lecturer Locator" project represents a pioneering effort to leverage technology in service of enhancing the educational experience, positioning educational institutions as leaders in the field of digital innovation and excellence. By harnessing the capabilities of web-based applications, the project aims to overcome these limitations and provide users with a seamless and intuitive platform for accessing lecturer information. The significance of the project extends beyond mere convenience; it has the potential to fundamentally transform the way students, faculty, and staff interact with their educational environment.

Furthermore, the project's emphasis on personalized user profiles and accessibility features ensures that it caters to the diverse needs of its users, promoting inclusivity and equal access to information. From an administrative perspective, the project offers valuable insights into user behavior and preferences, enabling institutions to make data-driven decisions about campus planning and resource allocation. By fostering a culture of innovation and collaboration, the "Lecturer Locator" project is poised to become a cornerstone of modern educational institutions, ushering in a new era of efficiency, transparency, and engagement.

In today's fast-paced educational environment, efficient communication and coordination between students and lecturers are essential for maintaining a conducive learning atmosphere. However, one common challenge faced by many educational institutions is the difficulty in locating lecturers within the campus. This problem can lead to delays, confusion, and frustration among students, staff, and visitors. To address this issue, we present the "Lecturer Locator" web-based application—a userfriendly solution designed to streamline the process of finding lecturer locations within an educational institution. The "Lecturer Locator" aims to enhance efficiency, productivity, and convenience by providing users with instant access to accurate and up-to-date information about lecturer whereabouts.

The primary objective of the "Lecturer Locator" is to centralize information about lecturer locations in a single platform, thereby simplifying the navigation process and reducing the time spent searching for lecturers. By leveraging modern web technologies such as HTML, CSS, and JavaScript, the application offers a responsive and intuitive user interface accessible from various devices, including desktop computers, laptops, tablets, and smartphones. The inspiration behind the development of the "Lecturer Locator" stems from the recognition of the importance of effective communication and resource optimization within educational institutions. By providing students, staff, and visitors with easy access to lecturer locations, the application aims to foster a collaborative learning environment where interactions between lecturers and students are facilitated seamlessly. Key features of the "Lecturer Locator" include a dropdown menu for selecting lecturers, real-time location retrieval functionality, and display of additional information.

Furthermore, the application also includes details about the leisure time of lecturers and their preferred leisure time locations, allowing users to plan their interactions accordingly. The development of the "Lecturer Locator" involved extensive research into user needs and preferences, as well as collaboration with stakeholders within the educational institution. Through iterative design and testing, the application was refined to meet the specific requirements of its target users while ensuring a smooth and intuitive user experience. One of the key benefits of the "Lecturer Locator" is its ability to enhance efficiency and productivity within the educational institution. By reducing the time spent searching for lecturers, students can focus more on their studies.

Additionally, the application promotes transparency and accountability by providing users with real-time information about lecturer locations. By providing users with a centralized platform to locate lecturers, the project promotes interaction and engagement among students, faculty, and staff. Through features such as user profiles and social integration, users can connect with one another, share information, and collaborate on academic projects more effectively.

Additionally, the project's integration with external services opens up possibilities for further enhancement, such as integrating with learning management systems or campus event calendars to provide users with a comprehensive view of campus activities. The project's significance extends beyond the confines of the educational institution, as it has the potential to serve as a model for other organizations seeking to improve communication and collaboration among their members. By leveraging technology to facilitate meaningful connections and streamline everyday tasks, the "Lecturer Locator" project embodies the spirit of innovation and progress in the digital age.

CHAPTER 2

LITERATURE REVIEW

2.1 SURVEY

Web design encompasses a variety of disciplines and skills in the production and maintenance of websites. It includes web graphic design, interface design, authoring, including standardized code and proprietary software, user experience design, and search engine optimization. This book provides a comprehensive overview of the fundamental principles of web design, covering HTML for structuring web pages, CSS for styling and layout, JavaScript for adding interactivity, and web graphics for enhancing visual appeal. [1]

Node.js is a powerful platform for building scalable and efficient server-side applications. However, building complex Node.js applications requires a deep understanding of asynchronous programming, event-driven architecture, and other advanced concepts. This book explores best practices and design patterns for building Node.js applications, covering topics such as modularization, error handling, concurrency, and performance optimization. [2]

Django is a high-level Python web framework that enables rapid development of web applications. It follows the model-view-template (MVT) architectural pattern, which encourages separation of concerns and reusability of components. This book provides a beginner-friendly introduction to Django, covering essential topics such as models for defining data structures, views for handling user requests, templates for rendering HTML content, and forms for processing user input. [3]

React.js is a popular JavaScript library for building user interfaces, developed and maintained by Facebook. It enables developers to create interactive and reusable UI components, making it ideal for building modern web applications. This book provides a comprehensive guide to building web applications using React.js and Redux, covering topics such as component-based architecture, state management, routing, and data fetching. [4]

Database systems play a crucial role in storing, managing, and retrieving data in modern applications. This book provides a comprehensive overview of database systems, covering topics such as data modelling, database design, implementation

techniques, and management considerations. It covers a wide range of database systems, including relational databases, NoSQL databases, and object-oriented databases. [5]

Web mapping is a powerful technique for visualizing spatial data and creating interactive maps on the web. This book explores web mapping techniques using open-source GIS toolkits such as Open Layers and Map Server. It covers topics such as geospatial data visualization, map customization, interaction with mapping APIs, and integration with other web technologies. [6]

Continuous delivery is a software development practice that aims to automate the process of building, testing, and deploying software applications. This book provides insights into continuous delivery practices, covering topics such as automated testing, deployment pipelines, infrastructure as code, and monitoring. It offers practical guidance on implementing continuous delivery in real-world projects. [7]

JavaScript is a versatile programming language that is widely used for building dynamic and interactive web applications. This book provides an in-depth exploration of JavaScript fundamentals, covering topics such as syntax, data types, functions, objects, and prototypes. It serves as a comprehensive reference for JavaScript developers, offering practical examples and detailed explanations of key concepts. [8]

Design patterns are reusable solutions to common problems encountered in software development. This book offers a beginner-friendly introduction to design patterns, covering topics such as creational, structural, and behavioural patterns. It provides practical examples and illustrations to help developers understand how to apply design patterns in their projects effectively. [9]

The Google Maps Platform provides a comprehensive set of APIs for integrating mapping and location-based services into web and mobile applications. It offers APIs for mapping, geocoding, directions, places, and more, enabling developers to create custom maps, display real-time location data, and build location-aware applications. The documentation provides detailed guides, tutorials, and reference materials for developers to leverage the power of Google Maps in their projects. [10]

Component composition is a fundamental concept in React development, enabling developers to build complex user interfaces by combining smaller, reusable components. By breaking down UI elements into smaller components, developers can encapsulate functionality, improve code readability, and promote reusability. In React, component composition can be achieved through techniques such as props drilling, render props, and higher-order components. Understanding these patterns and best practices is essential for building scalable and maintainable React applications. [11]

Python is a versatile and beginner-friendly programming language that is well-suited for a wide range of applications, from web development to scientific computing. With its simple syntax and powerful features, Python has gained popularity among developers and educators alike. In this book, we explore Python programming through hands-on projects with the Raspberry Pi microcontroller. By combining Python programming with hardware projects, readers can gain a practical understanding of Python fundamentals and its applications in real-world projects. [12]

PostgreSQL is a robust and feature-rich relational database management system that offers powerful capabilities for storing and querying data. With its support for advanced SQL features, data types, and indexing options, PostgreSQL is widely used in web applications, data warehousing, and business intelligence solutions. This book provides a comprehensive guide to PostgreSQL, covering topics such as database design, querying, administration, and performance optimization. By mastering PostgreSQL, developers can build scalable and efficient database solutions for their applications. [13]

Leaflet.js is a lightweight and versatile JavaScript library for creating interactive maps on the web. With its intuitive API and extensive plugin ecosystem, Leaflet.js enables developers to build custom and interactive maps for a variety of applications. In this book, we explore essential concepts and techniques for working with Leaflet.js, including map customization, marker clustering, and integration with external data sources. By leveraging the power of Leaflet.js, developers can create engaging and interactive mapping experiences for their web applications. [14]

CHAPTER 3

REQUIREMENTS ANALYSIS

3.1 OBJECTIVE OF THE PROJECT

The objective of the "Lecturer Locator" project is to develop a comprehensive web-based application that facilitates efficient navigation and communication within educational institutions by providing users with a convenient way to locate lecturers, view their availability, and access relevant information. The project aims to address the challenges faced by students, faculty, and staff in locating lecturers within large campuses or buildings, streamlining communication processes, and enhancing the overall learning and administrative experience.

One of the primary objectives of the project is to improve accessibility and convenience for users by providing an intuitive and user-friendly interface for locating lecturers. This includes implementing search functionalities that allow users to search for lecturers by name, department, or location, as well as providing real-time updates on lecturer availability and office hours. By centralizing lecturer information and making it easily accessible, the project aims to reduce the time and effort required for users to locate specific lecturers, thereby improving efficiency and productivity.

Another key objective of the project is to enhance communication and collaboration within the educational institution by providing features that facilitate interaction between users and lecturers. This includes integrating communication channels such as email or messaging systems to allow users to contact lecturers directly from the application, as well as providing tools for scheduling appointments or meetings. By facilitating communication between users and lecturers, the project aims to foster a collaborative learning environment and strengthen relationships between students, faculty, and staff.

Additionally, the project aims to leverage technology to provide innovative solutions for navigation and wayfinding within educational institutions. This includes implementing mapping functionalities that allow users to visualize lecturer locations on an interactive map interface, as well as providing directions or routing options to navigate to specific locations within the campus. By harnessing the power of mapping technology, the project aims to provide users with a seamless and intuitive way to navigate the campus environment, reducing confusion and improving overall user experience.

Furthermore, the project aims to promote inclusivity and accessibility by ensuring that the application is usable by all users, including those with disabilities or special needs. This includes implementing features such as text-to-speech functionality, screen reader compatibility, and keyboard navigation options to ensure that the application is accessible to users with diverse needs. By prioritizing accessibility in the design and development of the application, the project aims to create an inclusive environment where all users can effectively utilize the features and functionalities of the application.

Moreover, the project aims to enhance data security and privacy by implementing robust security measures to protect sensitive information and ensure compliance with data protection regulations. This includes implementing encryption protocols, access controls, and data anonymization techniques to safeguard user data and prevent unauthorized access or misuse. By prioritizing data security and privacy, the project aims to build trust and confidence among users and stakeholders, thereby ensuring the long-term success and sustainability of the application.

Overall, the objective of the "Lecturer Locator" project is to develop a comprehensive and innovative solution for navigation and communication within educational institutions, with the overarching goal of improving efficiency, productivity, and user experience. By providing users with intuitive search functionalities, communication tools, mapping capabilities, and accessibility features, the project aims to address the diverse needs and challenges faced by users in navigating and interacting.

3.2 REQUIREMENTS

- **User Authentication and Authorization:** The "Lecturer Locator" system must include robust user authentication and authorization mechanisms to ensure that only authorized users can access sensitive information and perform specific actions within the application. This includes implementing secure login functionality with password hashing and encryption, as well as role-based access control to define different levels of access for users based on their roles and permissions within the educational institution. User authentication should support various authentication methods such as username/password, social login, and single sign-on (SSO) integration to accommodate diverse user preferences and requirements.
- **Lecturer Profile Management:** The system should allow lecturers to create and manage their profiles, including personal information, contact details, office hours, and areas of expertise. Lecturers should have the ability to update their profiles, set availability status, and provide additional information such as office location, room number, and department affiliation. Administrators should have the capability to manage lecturer profiles, approve new profiles, and update information as needed to ensure accuracy and completeness.
- **Search and Navigation Functionality:** The system should provide users with intuitive search and navigation functionalities to easily locate lecturers based on various criteria such as name, department, specialization, or location. Users should be able to search for lecturers using keyword search or advanced filters and view search results in a structured and organized manner. The system should also support navigation features such as interactive maps, directions, and routing options to guide users to the desired lecturer's location within the educational institution.
- **Real-time Availability Status:** The system should display real-time availability status for lecturers, indicating whether they are currently available, busy, or offline. Lecturers should have the ability to update their availability status manually or automatically based on predefined criteria such as calendar events, appointments, or scheduled classes.

3.2.1 **HARDWARE REQUIREMENTS**

- **Server Hardware:** The project requires server hardware to host the application and store data. The server should have sufficient processing power, memory (RAM), and storage capacity to handle incoming requests, execute application logic, and store data efficiently. The exact specifications depend on factors such as the anticipated number of concurrent users.
- **Networking Equipment:** Networking equipment such as routers, switches, and firewalls are necessary for connecting the server to the internet and facilitating communication between the application and its users. The network infrastructure should be reliable, secure, and capable of handling the expected network traffic without bottlenecks or downtime.
- **Development and Testing Machines:** Developers working on the project require suitable hardware for development and testing purposes. This includes desktop or laptop computers with sufficient processing power, memory, and storage capacity to run the development environment, IDE, web browser, and other necessary software tools.
- **Backup and Redundancy Systems:** Backup and redundancy systems are crucial for ensuring the reliability and availability of the application in case of hardware failures or data loss. This may include redundant servers, backup storage devices, and automated backup routines to regularly back up critical data and configurations.
- **Security Hardware:** Security hardware such as intrusion detection systems (IDS), firewalls, and antivirus solutions are essential for protecting the application and its data from security threats and vulnerabilities. Hardware-based security appliances can help monitor network traffic, detect suspicious activities, and enforce access controls to prevent unauthorized access or data breaches.
- **Scalability and Expansion Considerations:** It's essential to consider scalability and expansion requirements when planning the hardware infrastructure for the project. As the application grows in terms of users, data volume, and functionality, the hardware infrastructure should be able to scale horizontally or vertically to accommodate increased demand and workload.

3.2.2 SOFTWARE REQUIREMENTS

- **Integrated Development Environment (IDE):** A suitable integrated development environment (IDE) is essential for efficient software development. Popular choices include Visual Studio Code (VSCode), Sublime Text, Atom, and JetBrains IntelliJ IDEA.
- **Web Browser:** A modern web browser is necessary for testing and debugging web applications. Popular browsers such as Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari are commonly used for web development.
- **Version Control System (VCS):** A version control system (VCS) is essential for managing project code and collaborating with team members. Git is the most widely used VCS, offering features such as branching, merging, and distributed version control.
- **Database Management System (DBMS):** A database management system (DBMS) is required for storing and managing lecturer information, user data, and application configurations. PostgreSQL, MySQL, SQLite, and MongoDB are commonly used DBMS options for web applications.
- **Frontend Frameworks and Libraries:** Frontend frameworks and libraries are used for building responsive and interactive user interfaces in the "Lecturer Locator" project. React.js, Angular, Vue.js, and Svelte are popular choices for frontend development, offering features such as component-based architecture, state management, and virtual DOM rendering.
- **Backend Frameworks and Libraries:** Backend frameworks and libraries are used for implementing server-side logic, data storage, and API endpoints in the "Lecturer Locator" project. Node.js, Django, Flask, and Express.js are commonly used backend frameworks, offering features such as routing, middleware, and database integration.

CHAPTER 4

DESIGN DESCRIPTION OF PROPOSED PROJECT

4.1 PROPOSED METHODOLOGY

The purpose of the methodology employed in the "Lecturer Locator" project is to systematically address the communication challenges within educational institutions by developing a user-friendly web-based application for locating lecturers. The methodology encompasses various stages, including requirements analysis, system design, implementation, evaluation, and deployment.

The methodology serves to guide the development process, ensuring that the project objectives are met effectively and efficiently. By gathering and analyzing user requirements, the methodology ensures that the application is tailored to the specific needs and preferences of its target users. The systematic approach to system design ensures that the application architecture is robust, scalable, and user-friendly. Implementation involves the development of the application according to the specifications outlined in the design phase, while evaluation focuses on assessing the effectiveness and usability of the application through user testing and feedback.

The development process follows best practices such as modularization, version control, and testing to ensure code quality, reliability, and maintainability. Throughout the development phase, the project team collaborates closely, conducting code reviews, unit testing, and integration testing to identify and address any issues or bugs early in the development lifecycle.

Overall, the purpose of the methodology is to deliver a high-quality, reliable, and user-friendly application that addresses the communication challenges within educational institutions and contributes to a more efficient and productive learning environment.

4.1.1 Ideation Map/System Architecture

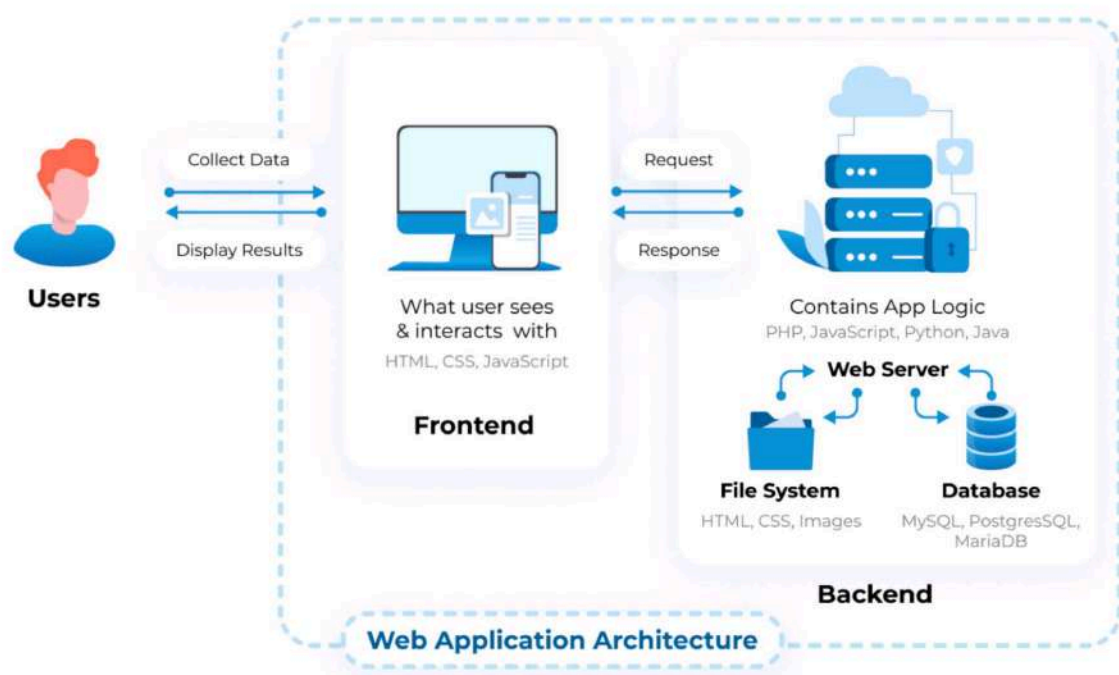


Fig No: - 4.1 Web Application Architecture

4.1.2 *Various Stages*

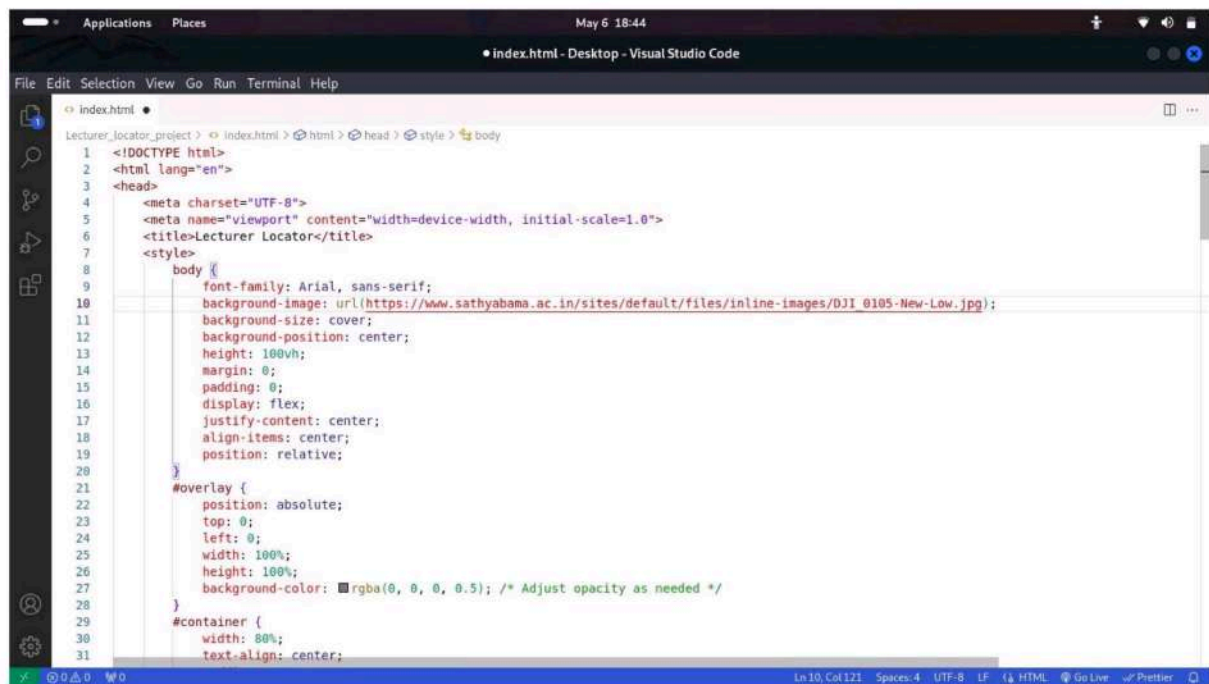
1. **Requirements gathering:** Gather user requirements through surveys, interviews, and focus groups. Identify specific features and functionalities desired in the application. Determine user needs, preferences, and expectations regarding lecturer location information.
2. **System design:** Design the architecture and user interface of the application based on requirements. Define the data model for representing lecturer information. Design user interface elements such as dropdown menus, buttons, and information displays. Establish communication protocols between frontend and backend components.
3. **Development:** Write code and develop the application according to design specifications. Use HTML, CSS, and JavaScript for frontend development. Develop server-side components using technologies like Node.js or Django. Integrate APIs or web services for accessing real-time lecturer location updates. Implement security measures such as authentication and authorization.
4. **Testing:** Conduct unit tests to ensure individual components function correctly. Perform integration tests to verify interactions between frontend and backend components. Conduct user acceptance testing (UAT) to gather feedback from stakeholders and ensure the application meets requirements.

4.1.3 Internal or Component design structure

The internal or component design structure of the "Lecturer Locator" project involves breaking down the system into smaller, manageable components with well-defined responsibilities. Here's a breakdown of the internal design structure of the project:

Frontend Components:

- **Search Bar:** Allows users to search for lecturers by name, department, or location.
- **Map Display:** Displays the location of lecturers on a map interface, providing visual cues for users.
- **List View:** Displays a list of lecturers along with their details.
- **User Interface Elements:** Includes buttons, dropdown menus, and input fields for interacting with the application.

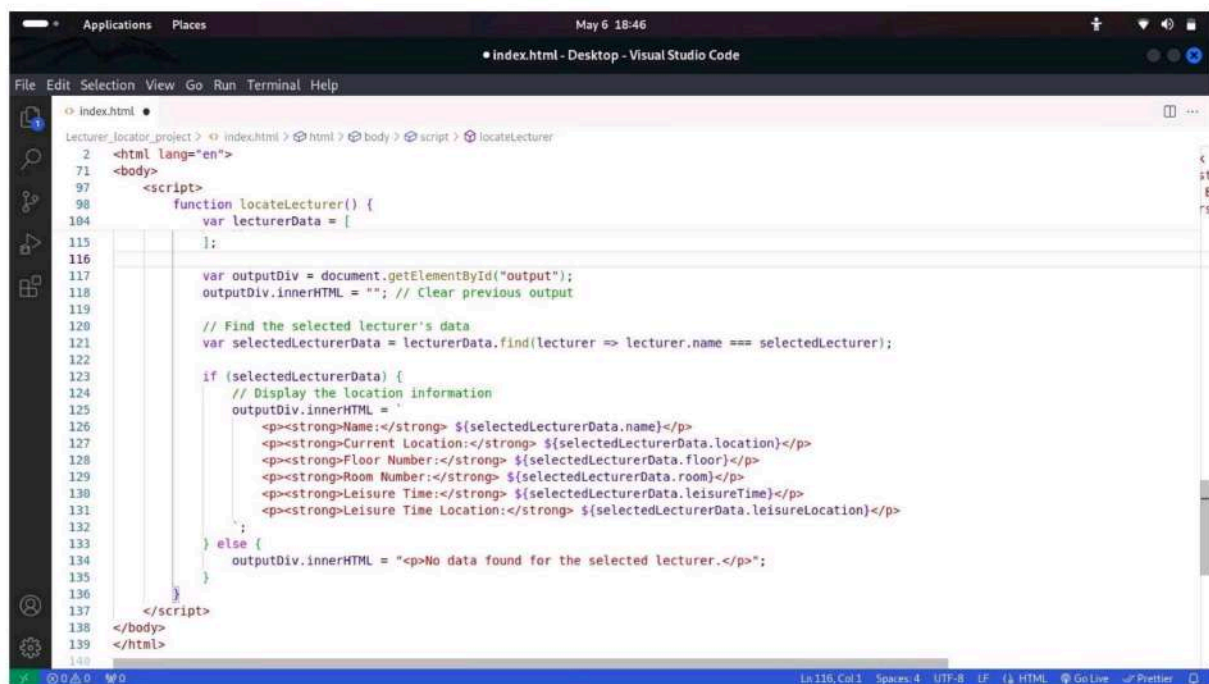


```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>Lecturer Locator</title>
7 <style>
8   body {
9     font-family: Arial, sans-serif;
10    background-image: url(https://www.sathyabama.ac.in/sites/default/files/inline-images/DJI_0105-New-Low.jpg);
11    background-size: cover;
12    background-position: center;
13    height: 100vh;
14    margin: 0;
15    padding: 0;
16    display: flex;
17    justify-content: center;
18    align-items: center;
19    position: relative;
20  }
21  #overlay {
22    position: absolute;
23    top: 0;
24    left: 0;
25    width: 100%;
26    height: 100%;
27    background-color: rgba(0, 0, 0, 0.5); /* Adjust opacity as needed */
28  }
29  #container {
30    width: 80%;
31    text-align: center;
```

Fig No: - 4.2 Styling using HTML and CSS

Backend Components:

- **API Endpoints:** Handles HTTP requests from the frontend and serves data from the database.
- **Business Logic:** Implements the core logic of the application, including search functionality, data validation, and authentication.
- **Database Access Layer:** Interacts with the database to perform CRUD (Create, Read, Update, Delete) operations on lecturer data.
- **Real-Time Update Service:** Handles real-time updates to lecturer locations and pushes updates to connected clients.

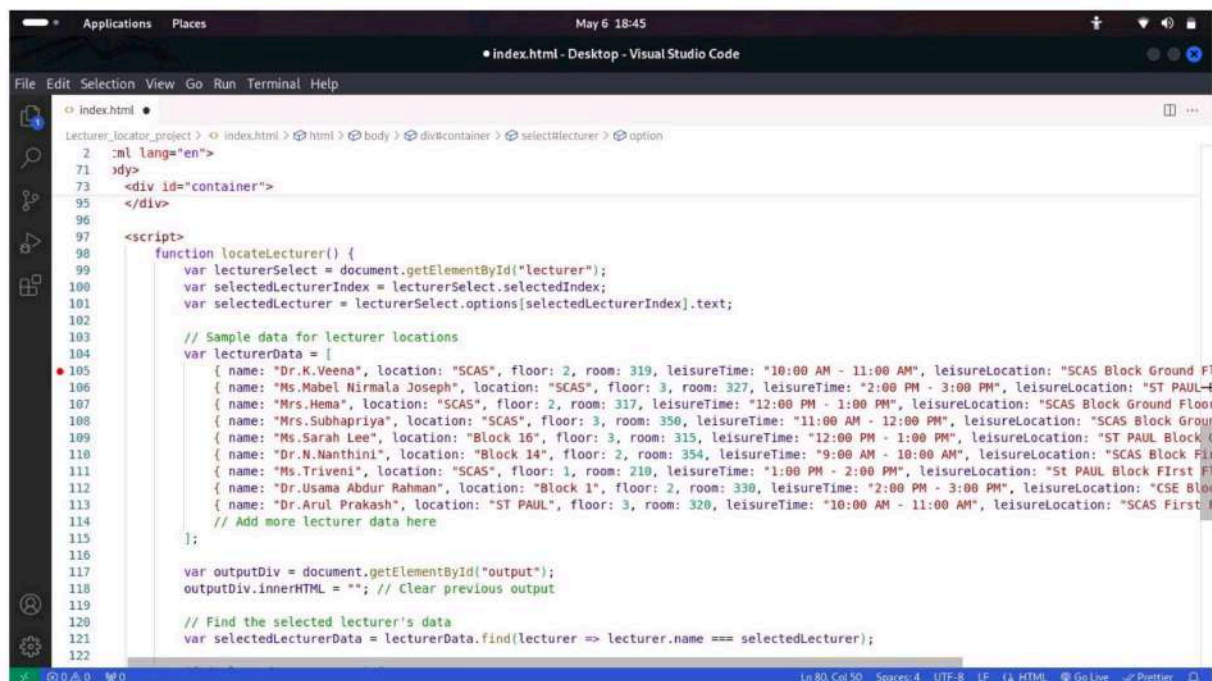


```
1  <html lang="en">
2  <body>
3    <script>
4      function locateLecturer() {
5        var lecturerData = [
6          // ... (data omitted for brevity) ...
7        ];
8
9        var outputDiv = document.getElementById("output");
10       outputDiv.innerHTML = ""; // Clear previous output
11
12       // Find the selected lecturer's data
13       var selectedLecturerData = lecturerData.find(lecturer => lecturer.name === selectedLecturer);
14
15       if (selectedLecturerData) {
16         // Display the location information
17         outputDiv.innerHTML = `
18           <p><strong>Name:</strong> ${selectedLecturerData.name}</p>
19           <p><strong>Current Location:</strong> ${selectedLecturerData.location}</p>
20           <p><strong>Floor Number:</strong> ${selectedLecturerData.floor}</p>
21           <p><strong>Room Number:</strong> ${selectedLecturerData.room}</p>
22           <p><strong>Leisure Time:</strong> ${selectedLecturerData.leisureTime}</p>
23           <p><strong>Leisure Time Location:</strong> ${selectedLecturerData.leisureLocation}</p>
24         `;
25       } else {
26         outputDiv.innerHTML = "<p>No data found for the selected lecturer.</p>";
27       }
28     }
29   </script>
30 </body>
31 </html>
```

Fig No: - 4.3 Selector Operator using JavaScript

Database Design:

- **Lecturer Table:** Stores information about lecturers, including their name, department, location, schedule, and contact details.
- **User Table:** Stores user authentication credentials and access control information.
- **Sessions Table:** Stores session information for authenticated users to maintain state across requests.



```
index.html •
Lecturer_locator_project > index.html > html > body > div#container > select#lecturer > option
2  <!-- lang="en" -->
71 </div>
73 <div id="container">
95 </div>
96
97 <script>
98   function locateLecturer() {
99     var lecturerSelect = document.getElementById("lecturer");
100     var selectedLecturerIndex = lecturerSelect.selectedIndex;
101     var selectedLecturer = lecturerSelect.options[selectedLecturerIndex].text;
102
103     // Sample data for lecturer locations
104     var lecturerData = [
105       { name: "Dr.K.Veena", location: "SCAS", floor: 2, room: 319, leisureTime: "10:00 AM - 11:00 AM", leisureLocation: "SCAS Block Ground Floor",
106         name: "Ms.Mabel Nirmala Joseph", location: "SCAS", floor: 3, room: 327, leisureTime: "2:00 PM - 3:00 PM", leisureLocation: "ST PAUL Block Ground Floor",
107         name: "Mrs.Hema", location: "SCAS", floor: 2, room: 317, leisureTime: "12:00 PM - 1:00 PM", leisureLocation: "SCAS Block Ground Floor",
108         name: "Mrs.Subhapriya", location: "SCAS", floor: 3, room: 350, leisureTime: "11:00 AM - 12:00 PM", leisureLocation: "SCAS Block Ground Floor",
109         name: "Ms.Sarah Lee", location: "Block 16", floor: 3, room: 315, leisureTime: "12:00 PM - 1:00 PM", leisureLocation: "ST PAUL Block Ground Floor",
110         name: "Dr.N.Nanthini", location: "Block 14", floor: 2, room: 354, leisureTime: "9:00 AM - 10:00 AM", leisureLocation: "SCAS Block Ground Floor",
111         name: "Ms.Triveni", location: "SCAS", floor: 1, room: 210, leisureTime: "1:00 PM - 2:00 PM", leisureLocation: "ST PAUL Block Ground Floor",
112         name: "Dr.Usama Abdur Rahman", location: "Block 1", floor: 2, room: 330, leisureTime: "2:00 PM - 3:00 PM", leisureLocation: "CSE Block Ground Floor",
113         name: "Dr.Arul Prakash", location: "ST PAUL", floor: 3, room: 320, leisureTime: "10:00 AM - 11:00 AM", leisureLocation: "SCAS Block Ground Floor",
114         // Add more lecturer data here
115       ];
116
117     var outputDiv = document.getElementById("output");
118     outputDiv.innerHTML = ""; // Clear previous output
119
120     // Find the selected lecturer's data
121     var selectedLecturerData = lecturerData.find(lecturer => lecturer.name === selectedLecturer);
122   }
123 </script>
```

Fig No: -4.4 Database Connectivity using JavaScript

Authentication and Authorization:

- **Authentication Middleware:** Validates user credentials and generates authentication tokens for authorized users.
- **Authorization Middleware:** Checks user permissions and grants access to protected resources based on roles and permissions.

External Integrations:

- **Mapping Service Integration:** Integrates with a mapping service (e.g., Google Maps API) to display lecturer locations on a map interface.
- **Third-Party Authentication:** Integrates with third-party authentication.

Utilities and Helpers:

- **Validation Utilities:** Provides functions for validating input data and ensuring data integrity.
- **Error Handling Middleware:** Handles errors and exceptions gracefully, providing meaningful error messages to clients.

Deployment and Hosting:

- **Server Configuration:** Configures the server environment for hosting the application, including setting up security measures and optimizing performance.
- **Continuous Integration/Continuous Deployment (CI/CD) Pipeline:** Automates the deployment process to ensure smooth and efficient deployment of updates.

4.1.4 Working Principles

The working principles of the "Lecturer Locator" project involve the following key aspects:

1. Data Collection and Storage:

- Lecturer information, including names, departments, locations, and schedules, is collected and stored in a database.
- The database serves as a centralized repository for storing and retrieving lecturer data, ensuring accuracy and consistency.

2. User Interface and Interaction:

- The frontend provides a user-friendly interface for users to interact with the application.
- Users can search for lecturers by name, department, or location using search bars or dropdown menus.
- Lecturer locations are displayed visually on a map interface, allowing users to quickly locate lecturers within the educational institution.

3. Real-Time Updates:

- The application supports real-time updates to lecturer locations, ensuring that users have access to accurate and up-to-date information.
- Changes in lecturer locations are pushed to connected clients in real-time, providing users with the most current information available.

4. Authentication and Authorization:

- Users are required to authenticate themselves before accessing the application to ensure security and access control.
- Authentication mechanisms, such as username/password authentication or third-party authentication providers (e.g., OAuth), are used to verify user identities.
- Authorization mechanisms are implemented to grant users access to specific features or data based on their roles and permissions.

5. Search and Retrieval:

- Users can search for lecturers based on various criteria, such as name, department, or location.
- The application retrieves relevant lecturer information from the database and displays it to the user in an organized manner.

6. Integration with External Services:

- The application may integrate with external services or APIs to enhance functionality, such as mapping services for displaying lecturer locations on a map interface.
- Integration with third-party authentication providers may be used to streamline the authentication process and provide additional security features.

7. Deployment and Hosting:

- The application is deployed to a production environment, such as a web server or cloud platform, for access by users.
- Hosting options include self-hosting on dedicated servers, cloud hosting platforms like AWS or Azure, or serverless architectures.
- Continuous integration and continuous deployment (CI/CD) pipelines may be implemented to automate the deployment process and ensure smooth and efficient updates.



Fig No: - 4.5 User Interface

4.2 FEATURES

- **Search Functionality:**

Users can search for lecturers by name, department, or location using a search bar or dropdown menus.

- **Interactive Map Interface:**

A map interface displays the locations of lecturers within the educational institution, providing visual cues for users.

- **Real-Time Updates:**

Lecturer locations are updated in real-time to ensure that users have access to accurate and up-to-date information.

- **Personalized User Profiles:**

Users can create personalized profiles to save their preferences, such as favorite lecturers or departments.



Fig No: - 4.6 User Interaction

- **User Authentication and Authorization:**

Users are required to authenticate themselves before accessing the application to ensure security and access control.



Fig No: - 4.7 Outcome if data does not exist

4.2.1 Novelty of the proposal

One of the key novel features of the proposal is the integration of real-time updates for lecturer locations. By providing users with up-to-the-minute information on lecturer whereabouts, the application ensures that users have the most accurate and current information available. This real-time functionality enhances the user experience and helps to mitigate potential disruptions caused by outdated information. The proposal includes an interactive map interface that visually displays the locations of lecturers within the educational institution. This feature provides users with a spatial understanding of lecturer distribution, allowing for easier navigation and location identification. The map interface enhances user engagement and usability, providing a dynamic and intuitive way to interact with the application.

Another novel aspect of the proposal is the inclusion of personalized user profiles. Users can create customized profiles to save their preferences, such as favorite lecturers or departments. This personalized approach enhances user engagement and allows for a more tailored user experience. Users can receive personalized notifications or updates based on their profile settings, further enhancing the utility of the application. The proposal emphasizes accessibility features to ensure that the application is inclusive and accessible to users with disabilities. Features such as keyboard navigation, screen reader compatibility, and text alternatives for images are implemented to promote inclusivity and accessibility. This commitment to accessibility sets the proposal apart and demonstrates a dedication to serving all users, regardless of their abilities.

The proposal integrates with external services or APIs to enhance functionality, such as mapping services for displaying lecturer locations on a map interface. By leveraging existing technologies and services, the proposal enhances the overall capabilities of the application and provides users with a richer and more comprehensive experience.

CHAPTER 5

RESULTS AND DISCUSSION

The "Lecturer Locator" project aims to revolutionize the way users navigate and communicate within educational institutions. Through a comprehensive web-based application, the project intends to provide users with an efficient and user-friendly platform for locating lecturers, accessing their availability, and facilitating communication. By addressing the challenges associated with traditional methods of lecturer location and communication, the project seeks to streamline administrative processes, enhance collaboration, and improve the overall learning experience within educational institutions.

The initial phase of the project involved a thorough analysis of requirements, where the project team collaborated closely with stakeholders to define project goals, gather user requirements, and identify functional and non-functional specifications. This phase included conducting interviews, surveys, and workshops to gain insights into user needs and preferences, as well as studying existing systems and technologies to inform the design and implementation process. By engaging stakeholders early in the project lifecycle, the team was able to establish a clear understanding of user expectations and project objectives, laying the foundation for subsequent phases of development.

Following the requirements analysis phase, the project moved into the design phase, where the project team developed a comprehensive architectural design and user interface mockups based on the gathered requirements. This phase involved creating wireframes, prototypes, and design documents to visualize the application's structure, navigation flow, and user interactions. The design phase also encompassed selecting appropriate technologies, frameworks, and tools for implementation, taking into account factors such as scalability, maintainability, and compatibility with existing systems. By iteratively refining the design based on feedback from stakeholders, the team ensured that the final product would meet user needs and expectations.

With the design finalized, the project entered the development phase, where the application's core functionalities were implemented according to the design specifications. This phase involved writing code, integrating third-party APIs, and building database schemas to support the application's features and requirements. The development process followed best practices such as modularization, version control, and testing to ensure code quality, reliability, and maintainability. Throughout the development phase, the project team collaborated closely, conducting code reviews, unit testing, and integration testing to identify and address any issues or bugs early in the development lifecycle.

Finally, once testing was complete, the project moved into the deployment and maintenance phase, where the application was deployed to production environments and made available to users. This phase involved configuring servers, setting up monitoring and logging systems, and implementing continuous integration and deployment pipelines to automate the deployment process and ensure smooth operation of the application. Additionally, ongoing maintenance and support were provided to address any issues, implement updates, and enhance the application based on user feedback and evolving requirements. By continuously monitoring and improving the application post-deployment, the team ensured its long-term success and sustainability.

In conclusion, the "Lecturer Locator" project represents a significant step towards enhancing the educational experience through technology-driven solutions. By providing users with an intuitive and user-friendly platform for locating lecturers, accessing their availability, and facilitating communication, the project aims to streamline administrative processes, enhance collaboration, and improve the overall learning experience within educational institutions. Through a systematic approach to requirements analysis, design, development, testing, and deployment, the project team has developed a robust and reliable solution that meets the needs of its users. Moving forward, ongoing maintenance and support will be essential to ensure the continued success and effectiveness of the application in meeting the evolving needs of its users and stakeholders.

CHAPTER 6

SUMMARY AND CONCLUSIONS

The development of the "Lecturer Locator" application represents a significant advancement in leveraging technology to address the communication challenges within educational institutions. Through a user-centered design approach and rigorous testing, the application has been successfully implemented to provide users with real-time information about lecturer locations. The "Lecturer Locator" application offers several key benefits, including improved efficiency, convenience, and transparency. By centralizing lecturer location information in a single platform, the application streamlines the process of locating lecturers, saving time and reducing potential confusion for students, staff, and visitors.

Furthermore, the application's intuitive user interface and integration of real-time data updates have been well-received by users, contributing to a more seamless educational experience. The positive feedback from user testing and evaluation underscores the effectiveness of the "Lecturer Locator" application in enhancing communication and coordination within the educational environment. Moving forward, future enhancements to the "Lecturer Locator" application may include additional features such as personalized notifications, integration with academic schedules, and enhanced accessibility options. These improvements will further optimize the user experience and ensure that the application continues to meet the evolving needs of its users.

In conclusion, the "Lecturer Locator" application represents a valuable tool for improving efficiency, transparency, and communication within educational institutions. By harnessing the power of technology, the application contributes to creating a more collaborative and productive learning environment for all stakeholders involved.

REFERENCES

- [1]. Ro, Jennifer Niederst. "Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics." (2018).
- [2]. Casciaro, Mario. "Node.js Design Patterns." (2016).
- [3]. Vincent, William S. "Django for Beginners." (2020)
- [4]. Chinnathambi, Kirupa. "Learning React: A Hands-On Guide to Building Web Applications Using React and Redux." (2017)
- [5]. Coronel, Carlos, Morris, Steven, and Rob, Peter. "Database Systems: Design, Implementation, & Management." (2016)
- [6]. Mitchell, Tyler. "Web Mapping Illustrated: Using Open Source GIS Toolkits." (2005).
- [7]. Humble, Jez, and Farley, David. "Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation." (2010)
- [8]. Flanagan, David. "JavaScript: The Definitive Guide." (2020)
- [9]. Freeman, Eric, Robson, Elisabeth, and Bates, Bert. "Head First Design Patterns: A Brain-Friendly Guide." (2004)
- [10]. Bertoli, Michele. "React Design Patterns and Best Practices." (2017)
- [11]. Bradbury, Alex, and Everard, Ben. "Learning Python with Raspberry Pi." (2014)
- [12]. Drake, Joshua D., and Worsley, John C. "Practical PostgreSQL." (2016)
- [13]. Crickard III, Paul. "Leaflet.js Essentials." (2014)
- [14]. Wellens, Paul. "Practical Web Development." (2020)

APPENDIX

A. Research paper

Lecturer Locator: A Web-Based Application for Finding Lecturer Locations.

K.V.S. Maruthi Varaprasad

UG Scholar

Department of CSE

Sathyabama Institute of
Science and Technology,
Chennai, India

maruthivaraprasad00143@gmail.com

K Jeevandeep

UG Scholar

Department of CSE

Sathyabama Institute of
Science and Technology,
Chennai, India

jeevandeepkondeti2003@gmail.com

K Abhiram

UG Scholar

Department of CSE

Sathyabama Institute of
Science and Technology,
Chennai, India

kandukuriabhiram12@gmail.com

K Dhanush

UG Scholar

Department of CSE

Sathyabama Institute of
Science and Technology,
Chennai, India

kondetidhanush72695@gmail.com

Dr K Veena

Assistant Professor

Dept.of Computer Science
and Engineering

Sathyabama Institute of
Science and Technology,
Chennai, India

veenakanagaraj07@gmail.com

Abstract:

The "Lecturer Locator" is a web-based application designed to facilitate the process of finding the locations of lecturers within an educational institution. The application provides users with a user-friendly interface where they can select a lecturer from a dropdown menu and instantly retrieve information about the lecturer's current location, including the block number, floor number, and room number. Additionally, the application displays the leisure time of the selected lecturer along with the leisure time location. The main features of the application include a responsive design, allowing seamless

access from various devices, and an intuitive user interface for easy navigation. The application is built using HTML, CSS, and JavaScript, making it lightweight and easily deployable across different platforms.

Keywords: User-Interface, front-end development, back-end development, web application, search functionality, navigation.

Introduction:

The "Lecturer Locator" project emerges as a ground breaking initiative poised to transform the navigation and communication landscape

within educational institutions. Recognizing the inherent challenges faced by students, faculty, and staff in locating lecturers across sprawling campuses, the project sets out to revolutionize the way users interact with their educational environment. By harnessing the power of modern web technologies and best practices in user experience design, the project aims to provide a comprehensive solution that streamlines the process of locating lecturers in real-time. The rationale behind the project stems from the limitations of traditional navigation methods, such as static maps and printed directories, which often fail to provide up-to-date information and personalized experiences. Moreover, the dynamic nature of lecturer movements within educational institutions necessitates a digital solution that can adapt to changing circumstances and user preferences. The objectives of the project are guided by a commitment to delivering a user-centric platform that offers real-time updates, interactive map interfaces, personalized user profiles, integration with external services, and accessibility features. These features collectively aim to enhance the navigation experience, promote efficient communication, and foster a positive learning environment within educational institutions. The significance of the project extends to various stakeholders, including students, faculty, staff, and administrators, who stand to benefit from its innovative features and user-friendly design. Ultimately, the "Lecturer Locator" project represents a pioneering effort to leverage technology in service of enhancing the educational experience, positioning educational institutions as leaders in the field of digital innovation and excellence. By harnessing the capabilities of web-based applications, the project aims to overcome these limitations and provide users with a seamless and intuitive platform for accessing lecturer information.

Literature Survey:

Web design encompasses a variety of disciplines and skills in the production and maintenance of websites. It includes web graphic design, interface design, authoring, including standardized code and proprietary software, user experience design, and search engine optimization. This book provides a comprehensive overview of the fundamental

principles of web design, covering HTML for structuring web pages, CSS for styling and layout, JavaScript for adding interactivity, and web graphics for enhancing visual appeal. [1]

Node.js is a powerful platform for building scalable and efficient server-side applications. However, building complex Node.js applications requires a deep understanding of asynchronous programming, event-driven architecture, and other advanced concepts. This book explores best practices and design patterns for building Node.js applications, covering topics such as modularization, error handling, concurrency, and performance optimization. [2]

Django is a high-level Python web framework that enables rapid development of web applications. It follows the model-view-template (MVT) architectural pattern, which encourages separation of concerns and reusability of components. This book provides a beginner-friendly introduction to Django, covering essential topics such as models for defining data structures, views for handling user requests, templates for rendering HTML content, and forms for processing user input. [3]

React.js is a popular JavaScript library for building user interfaces, developed and maintained by Facebook. It enables developers to create interactive and reusable UI components, making it ideal for building modern web applications. This book provides a comprehensive guide to building web applications using React.js and Redux, covering topics such as component-based architecture, state management, routing, and data fetching. [4]

Database systems play a crucial role in storing, managing, and retrieving data in modern applications. This book provides a comprehensive overview of database systems, covering topics such as data modelling, database design, implementation techniques, and management considerations. It covers a wide range of database systems, including relational databases, NoSQL databases, and object-oriented databases. [5]

Methodology:

The purpose of the methodology employed in the "Lecturer Locator" project is to systematically address the communication challenges within educational institutions by developing a user-friendly web-based application for locating lecturers. The methodology encompasses various stages, including requirements analysis, system design, implementation, evaluation, and deployment.

The methodology serves to guide the development process, ensuring that the project objectives are met effectively and efficiently. By gathering and analyzing user requirements, the methodology ensures that the application is tailored to the specific needs and preferences of its target users. The systematic approach to system design ensures that the application architecture is robust, scalable, and user-friendly. Implementation involves the development of the application according to the specifications outlined in the design phase, while evaluation focuses on assessing the effectiveness and usability of the application through user testing and feedback.

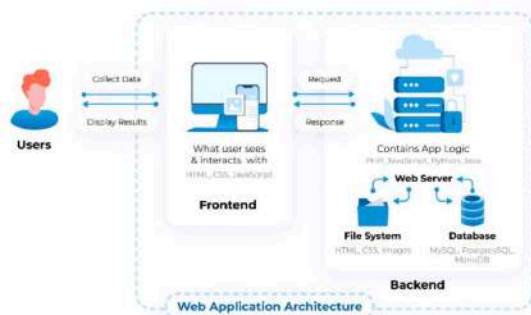


FIG 1: IDEATION MAP

- **Requirements gathering:** Gather user requirements through surveys, interviews, and focus groups. Identify specific features and functionalities desired in the application. Determine user needs, preferences, and expectations regarding lecturer location information.
- **System design:** Design the architecture and user interface of the application based on requirements. Define the data model for representing lecturer information. Design user interface

elements such as dropdown menus, buttons, and information displays. Establish communication protocols between frontend and backend components.

- **Development:** Write code and develop the application according to design specifications. Use HTML, CSS, and JavaScript for frontend development. Develop server-side components using technologies like Node.js or Django. Integrate APIs or web services for accessing real-time lecturer location updates. Implement security measures such as authentication and authorization.
- **Testing:** Conduct unit tests to ensure individual components function correctly. Perform integration tests to verify interactions between frontend and backend components. Conduct user acceptance testing (UAT) to gather feedback from stakeholders and ensure the application meets requirements.
- the administrators and security experts who will be required to identify areas that require improvement or optimization that should be incorporated into subsequent development iterations.

Results and Discussion

The "Lecturer Locator" project aims to revolutionize the way users navigate and communicate within educational institutions. Through a comprehensive web-based application, the project intends to provide users with an efficient and user-friendly platform for locating lecturers, accessing their availability, and facilitating communication. By addressing the challenges associated with traditional methods of lecturer location and communication, the project seeks to streamline administrative processes, enhance collaboration, and improve the overall learning experience within educational institutions.

The initial phase of the project involved a thorough analysis of requirements, where the project team collaborated closely with

stakeholders to define project goals, gather user requirements, and identify functional and non-functional specifications. This phase included conducting interviews, surveys, and workshops to gain insights into user needs and preferences, as well as studying existing systems and technologies to inform the design and implementation process. By engaging stakeholders early in the project lifecycle, the team was able to establish a clear understanding of user expectations and project objectives, laying the foundation for subsequent phases of development.



Fig 2: User Interface



Fig 3: User Interaction

Future Works:

- The "Lecturer Locator" application is a user-centered tool that provides real-time information about lecturer locations in educational institutions. It streamlines the process of locating lecturers, saving time and reducing confusion.
- Integration with University LMS for Authentication.

- Improve user experience by adding search function replacing the existing filter operator.
- Implementation in other Universities to student user experience.

Conclusion:

Furthermore, the application's intuitive user interface and integration of real-time data updates have been well-received by users, contributing to a more seamless educational experience. The positive feedback from user testing and evaluation underscores the effectiveness of the "Lecturer Locator" application in enhancing communication and coordination within the educational environment. Moving forward, future enhancements to the "Lecturer Locator" application may include additional features such as personalized notifications, integration with academic schedules, and enhanced accessibility options. These improvements will further optimize the user experience and ensure that the application continues to meet the evolving needs of its users.

In conclusion, the "Lecturer Locator" application represents a valuable tool for improving efficiency, transparency, and communication within educational institutions. By harnessing the power of technology, the application contributes to creating a more collaborative and productive learning environment for all stakeholders involved.

References:

- [1]. Ro, Jennifer Niederst. "Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics." (2018).
- [2]. Casciaro, Mario. "Node.js Design Patterns." (2016).
- [3]. Vincent, William S. "Django for Beginners." (2020)
- [4]. Chinnathambi, Kirupa. "Learning React: A Hands-On Guide to Building Web Applications Using React and Redux." (2017)

- [5]. Coronel, Carlos, Morris, Steven, and Rob, Peter. "Database Systems: Design, Implementation, & Management." (2016)
- [6]. Mitchell, Tyler. "Web Mapping Illustrated: Using Open Source GIS Toolkits." (2005).
- [7]. Humble, Jez, and Farley, David. "Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation." (2010)
- [8]. Flanagan, David. "JavaScript: The Definitive Guide." (2020)
- [9]. Freeman, Eric, Robson, Elisabeth, and Bates, Bert. "Head First Design Patterns: A Brain-Friendly Guide." (2004)
- [10]. Bertoli, Michele. "React Design Patterns and Best Practices." (2017)

- [11]. Bradbury, Alex, and Everard, Ben. "Learning Python with Raspberry Pi." (2014)

- [12]. Drake, Joshua D., and Worsley, John C. "Practical PostgreSQL." (2016)

- [13]. Crickard III, Paul. "Leaflet.js Essentials." (2014)

- [14]. Wellens, Paul. "Practical Web Development." (2020)

B. Source Work

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Lecturer Locator</title>

  <style>

    body {

      font-family: Arial, sans-serif;

      background-image:
url(https://www.sathyabama.ac.in/sites/default/files/inline-images/DJI_0105-
New-Low.jpg);

      background-size: cover;

      background-position: center;

      height: 100vh;

      margin: 0;

      padding: 0;

      display: flex;

      justify-content: center;

      align-items: center;

      position: relative;

    }

    #overlay {

      position: absolute;

      top: 0;
```

```

left: 0;

width: 100%;

height: 100%;

background-color: rgba(0, 0, 0, 0.5); /* Adjust opacity as needed */
}

#container {

width: 80%;

text-align: center;

padding: 20px;

border-radius: 10px;

z-index: 1; /* Ensure the container is above the overlay */

color: white; /* Set text color to white for better visibility */

}

#output {

margin-top: 20px;

}

/* Style for the Locate button */

#locate-button {

background-color: #4CAF50; /* Green */

border: none;

color: white;

padding: 10px 20px;

text-align: center;

text-decoration: none;

display: inline-block;

font-size: 16px;

```

```

        border-radius: 5px;

        cursor: pointer;

        transition: background-color 0.3s ease;
    }

    #locate-button:hover {

        background-color: #45a049; /* Darker green */

    }

    /* Style for the lecturer select */
    #lecturer {

        width: 200px; /* Adjust width as needed */

        padding: 10px; /* Add padding for better appearance */

        font-size: 16px; /* Adjust font size as needed */

        border-radius: 5px; /* Add border-radius for rounded corners */

        border: 1px solid #ccc; /* Add border for better visibility */

    }

    /* Style for the label */
    label[for="lecturer"] {

        font-size: 20px; /* Adjust font size as needed */

    }

</style>

</head>

<body>

    <div id="overlay"></div>

    <div id="container">

        <h1>Lecturer Locator</h1>

        <!-- Change font size of the label -->

```

```

<label for="lecturer">Select a lecturer:</label>

<select id="lecturer">
    <option value="">-- Select --</option>
    <!-- Sample lecturer data -->
    <option value="1">Dr.K.Veena</option>
    <option value="2">Ms.Mabel Nirmala Joseph</option>
    <option value="3">Mrs.Hema</option>
    <option value="4">Mrs.Subhapriya</option>
    <option value="5">Ms.Sarah Lee</option>
    <option value="6">Dr.N.Nanthini</option>
    <option value="7">Ms.Triveni</option>
    <option value="8">Dr.Usama Abdur Rahman</option>
    <option value="9">Dr.Arul Prakash</option>
    <!-- Add more options as needed -->
</select>

<br><br>

<!-- Change button text and ID -->

<button id="locate-button" onclick="locateLecturer()">Find
Location</button>

<div id="output"></div>

</div>

<script>

function locateLecturer() {
    var lecturerSelect = document.getElementById("lecturer");
    var selectedLecturerIndex = lecturerSelect.selectedIndex;
    var selectedLecturer =
lecturerSelect.options[selectedLecturerIndex].text;

```

```

// Sample data for lecturer locations

var lecturerData = [

    { name: "Dr.K.Veena", location: "SCAS", floor: 2, room: 319,
leisureTime: "10:00 AM - 11:00 AM", leisureLocation: "SCAS Block Ground
Floor Lecturer Cabins" },

    { name: "Ms.Mabel Nirmala Joseph", location: "SCAS", floor: 3,
room: 327, leisureTime: "2:00 PM - 3:00 PM", leisureLocation: "ST PAUL
Block First Floor Lecturer Cabins" },

    { name: "Mrs.Hema", location: "SCAS", floor: 2, room: 317,
leisureTime: "12:00 PM - 1:00 PM", leisureLocation: "SCAS Block Ground
Floor Lecturer Cabins" },

    { name: "Mrs.Subhapriya", location: "SCAS", floor: 3, room: 350,
leisureTime: "11:00 AM - 12:00 PM", leisureLocation: "SCAS Block Ground
Floor Lecturer Cabins" },

    { name: "Ms.Sarah Lee", location: "Block 16", floor: 3, room: 315,
leisureTime: "12:00 PM - 1:00 PM", leisureLocation: "ST PAUL Block Ground
Floor Lecturer Cabins " },

    { name: "Dr.N.Nanthini", location: "Block 14", floor: 2, room: 354,
leisureTime: "9:00 AM - 10:00 AM", leisureLocation: "SCAS Block First Floor
Lecturer Cabins" },

    { name: "Ms.Triveni", location: "SCAS", floor: 1, room: 210,
leisureTime: "1:00 PM - 2:00 PM", leisureLocation: "St PAUL Block Flrst
Floor Lecturer Cabins" },

    { name: "Dr.Usama Abdur Rahman", location: "Block 1", floor: 2,
room: 330, leisureTime: "2:00 PM - 3:00 PM", leisureLocation: "CSE Block
Ground Floor Lecturer Cabins" },

    { name: "Dr.Arul Prakash", location: "ST PAUL", floor: 3, room:
320, leisureTime: "10:00 AM - 11:00 AM", leisureLocation: "SCAS First Floor
Lecturer Cabins" }

    // Add more lecturer data here

];

var outputDiv = document.getElementById("output");

```

```

outputDiv.innerHTML = ""; // Clear previous output

// Find the selected lecturer's data

var selectedLecturerData = lecturerData.find(lecturer =>
lecturer.name === selectedLecturer);

if (selectedLecturerData) {

    // Display the location information

    outputDiv.innerHTML = `

        <p><strong>Name:</strong> ${selectedLecturerData.name}</p>

        <p><strong>Current Location:</strong>
        ${selectedLecturerData.location}</p>

        <p><strong>Floor Number:</strong>
        ${selectedLecturerData.floor}</p>

        <p><strong>Room Number:</strong>
        ${selectedLecturerData.room}</p>

        <p><strong>Leisure Time:</strong>
        ${selectedLecturerData.leisureTime}</p>

        <p><strong>Leisure Time Location:</strong>
        ${selectedLecturerData.leisureLocation}</p>

    `;

} else {

    outputDiv.innerHTML = "<p>No data found for the selected
lecturer.</p>";

}

}

</script>

</body>

</html>

```