A

SYNOPSIS

of

MINOR PROJECT

on

GITS Innovation Showcase

तमसो मा ज्योतिर्गमय्

Darkness to Light

Submitted by

HUSSAIN, PIYUSH CHOUBISA

(Roll No.-21EGICA014, 21EGICA025)

Project Guide
Mr. Rakshit Kothari

Head of Department
Dr. Mayank Patel

Geetanjali Institute of Technical Studies, Dabok, Udaipur (Raj.)
Department of Computer Science and Engineering
October,2023

Problem Statement:

The lack of an engaging and informative online platform to showcase the GITS Innovation Lab's projects, achievements, and opportunities hinders its ability to attract collaborators, students, and investors.

Brief Description:

Development of a comprehensive website for the GITS Innovation Lab to highlight its creative endeavors, technological advancements, and educational offerings. This website will serve as a central hub for showcasing the lab's work and facilitating communication with stakeholders.

Key Features:

- **Home:** Welcome page with an overview of the lab.
- **About Us:** Description of the lab's mission, vision, and values.
- **Achievements:** Showcase of awards, recognitions, and milestones.
- **Projects:** Detailed information on ongoing and completed projects.
- **Trainings:** Information about available training programs and workshops.
- **Patents:** List of patents filed by the lab.
- **Contact Us:** Contact information and inquiry form.

Objective and Scope:

- **Objective:** To create a dynamic and user-friendly website that effectively communicates the innovation lab's mission, achievements, and offerings.
- **Scope:** The website will serve as an informational and interactive platform for students, collaborators, and potential investors.

Methodology:

❖ **Requirements Gathering:**

- **Stakeholder Consultation:**

  Conduct interviews and meetings with stakeholders (students, faculty, potential investors) to gather detailed requirements.

- **Documentation:**

  Create comprehensive requirement documents outlining functional and non-functional requirements, user stories, and use cases.

- **Feasibility Analysis:**

  Assess the feasibility of the proposed requirements and identify any potential constraints or challenges.

❖ **Design:**

- **System Architecture Design:**

  Develop a detailed architecture plan that defines the overall structure of the website, including the interaction between various components, database schema, and data flow diagrams.

- **UI/UX Design:**

Design user interfaces and user experiences that are intuitive, visually appealing, and aligned with the lab's branding. Create wireframes, mockups, and prototypes to visualize the design.

- **Technical Specifications:**

  Define the technical specifications, including technology stack, development tools, and third-party integrations.

❖ **Development:**

- **Module Development:**

  Develop individual modules for each section of the website (Home, About Us, Achievements, Projects, Trainings, Patents, Contact Us) in parallel sprints.

- **Backend Development:**

  Implement server-side logic, database interactions, and API integrations using selected backend technologies (e.g., Django, Flask).

- **Frontend Development:**

  Create responsive and interactive web pages using HTML5, CSS3, JavaScript, and frontend frameworks like React.js.

- **Integration:**

  Integrate backend and frontend components, ensuring seamless communication between the client and server.

❖ **Testing:**

- **Unit Testing:**

  Write and execute unit tests to verify the functionality of individual components and modules.

- **Integration Testing:**

Test the interaction between different modules and components to ensure they work together as expected.

- **System Testing:**

  Conduct end-to-end testing of the entire system to validate that all requirements are met and the system functions as intended.

- **User Acceptance Testing (UAT):**

  Engage a group of end-users to test the website, gather feedback, and identify any usability issues or bugs.

❖ **Deployment:**

- **Server Setup:**

  Configure and set up a secure web server (e.g., Apache, Nginx) to host the application.

- **Application Deployment:**

  Deploy the website to the server, ensuring all components are correctly configured and operational.

- **User Training:**

  Provide training sessions and documentation to users and administrators on how to use and manage the website.

- **Performance Monitoring:**

  Implement monitoring tools to track the performance, availability, and security of the website.

❖ **Maintenance:**

- **Feedback Collection:**

Continuously gather feedback from users and stakeholders through surveys, feedback forms, and direct communication.

- **Bug Fixes and Updates:**

  Regularly update the website to fix bugs, improve performance, and add new features based on user feedback.

- **Security Audits:**

  Conduct periodic security audits to identify and mitigate potential vulnerabilities.

- **Documentation:**

  Maintain comprehensive documentation for all aspects of the website, including codebase, user guides, and administrative manuals.

- **Continuous Improvement:**

  Implement a continuous improvement process to ensure the website evolves to meet changing user needs and technological advancements.

Hardware and Software Requirements:

Hardware:

- **Processor**: Dual-core processor (e.g., Intel i3 or equivalent)
- **Memory**: Minimum 4 GB RAM
- **Storage**: At least 1 GB free space
- **Webcam**: HD webcam for proctoring
- **Microphone**: Built-in or external microphone
- **Internet**: Stable internet connection with at least 2 Mbps speed
- **Monitor**: Minimum resolution of 1280x720

Software:

- **Operating System**: Linux (Ubuntu, CentOS, or similar) or Windows Server
- **Web Server**: Apache or Nginx
- **Database**: PostgreSQL, MySQL, or MongoDB
- **Programming Language**: Python (for backend logic)
- **Framework**: Django or Flask for Python backend
- **AI/ML Libraries**: OpenCV, TensorFlow, or PyTorch (for proctoring module)
- **Plagiarism Detection API**: Integration with a third-party service like Turnitin or an open-source alternative

Technologies:

- **Frontend:**
  - HTML5, CSS3, JavaScript
  - React.js
  - Bootstrap
- **Backend:**
  - Python, Django, Flask
  - Node.js (optional for specific use cases)
- **Database:**
  - PostgreSQL, MySQL, MongoDB

Database Technologies:

1. **PostgreSQL**: Open-source relational database management system known for its robustness and performance.
2. **MySQL**: Widely used open-source relational database management system.
3. **MongoDB**: NoSQL database for handling large amounts of unstructured data.

Testing Techniques:

• **Unit Testing**: Test individual components and modules for correctness.

• **Integration Testing**: Test the interaction between different modules to ensure they work together seamlessly.

• **System Testing**: Perform end-to-end testing of the entire system to ensure it

meets the requirements.

• **User Acceptance Testing (UAT)**: Conduct testing with a group of end-users to gather feedback and make necessary improvements.

Project Contribution:

- **Requirement Gathering:** Collect input from stakeholders and document functional and non-functional requirements.
- **Sprint Planning:** Break down the project into manageable tasks and sprints, and assign tasks to team members.
- **Development:** Follow Agile methodology for iterative development, conduct regular code reviews, and pair programming sessions.
- **Testing:** Implement automated and manual testing processes, continuously integrate and test new features.
- **Deployment:** Use CI/CD pipelines for smooth and efficient deployment, monitor the live system, and perform regular updates and maintenance.
- **Feedback and Improvement:** Gather feedback from users and stakeholders, and continuously improve the system based on feedback and new requirements.