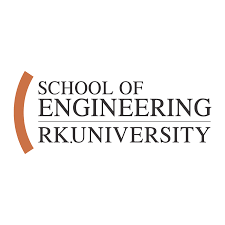
**SOFTWARE REQUIREMENT SPECIFICATION**

**ON**

**INTERN ADMIN COLLABORATION**





**Prepared By : -**

**Name: Enrollment No.:**

**Harendra Chaudhary 21SOECE11607**

**Angad Gupta 21SOECE11614**

**Roni Chandra Sarkar 21SOECE11639**

**SCHOOL OF ENGINEERING**

**RAJKOT – 360020, GUJARAT, INDIA**

**Table of Contents: -**

1. **Introduction**

1.1 Purpose………………………………………………………………………….…3

1.2 Document Conventions…………………………………………………………....3

1.3 Intended Audience and Reading Suggestions……………………………………..3

1.4 Project Scope………………………………………………………………………4

1.5 References…………………………………………………………………………4

1. **Overall Description**

2.1 Product Perspective………………………………………………………………..4

2.2 Product Features………………………………………………………….……..…5

2.3 User Classes and Characteristics…………………………………….……………..5

2.4 Operating Environment…………………………………………………………….5

1. **System Features**

3.1 Dashboard……………………………………………………………….…………7

3.2 Create Admin………………………………………………………………………7

3.3 Add Intern Details……………………………………………………..…………..7

3.4 Intern Leave Details………………………………………………...……………..7

3.5 Add Intern Salary………………………………………………………………….8

3.6 Intern Task Details…………………………………………………………………8

3.7 Intern Certificate Details…………………………………………….…………….9

3.8 Add Department……………………………………………………...……………9

3.9 Add Event Details …………………………………………………………………9

1. **External Interface Requirements**

4.1 User Interfaces……………………………………………...……………………10

4.2 Hardware Interfaces………………………………………...……………………10

4.3 Software Interfaces………………………………………………………………10

4.4 Communications Interfaces………………………………………………...……11

1. **Other Nonfunctional Requirements**

5.1 Performance Requirements………………………………………………………11

5.2 Safety Requirements……………………………………………………..………11

5.3 Security Requirements…………………………………………………...………12

* 1. Software Quality Attributes………………………………………………………12

1. **User Interface Design**

**7.Other Requirements**

Appendix A: Glossary……………………………………………………….……… 19

Appendix B: Analysis Models………………………………...………………… 20-21

Appendix C: Issues List User Interface ……………………………………………..21

Conclusion…………...…………………….………………………………………...22

**1. Introduction**

**1.1 Purpose**

The purpose of the SRS document for the "Intern Admin Collaboration" project is to provide a comprehensive overview of the software system's requirements and functionalities. It serves as a foundational document that outlines the goals and objectives of the project, guiding the development team throughout the software development lifecycle. By detailing the specific requirements and features of the system, the SRS document ensures alignment between stakeholders' expectations and the final product. Ultimately, the purpose of the SRS is to facilitate effective communication, mitigate misunderstandings, and lay the groundwork for successful project implementation.

**1.2 Document Conventions**

Document Conventions in the SRS document establish a standardized approach to formatting, terminology, and notation. These conventions ensure consistency and clarity in the document, making it easier for stakeholders to navigate and understand. Examples of document conventions include using a consistent naming scheme for requirements, employing a specific formatting style for headings and text, and defining terminology to avoid ambiguity. By adhering to document conventions, the SRS document maintains readability and facilitates effective communication among project stakeholders.

**1.3 Intended Audience and Reading Suggestions**

The Intended Audience for the SRS document includes various stakeholders involved in the software development process, such as developers, project managers, testers, and end-users. The document provides guidance on how to interpret its contents, starting with an overview of the project before delving into detailed requirements. Reading suggestions may advise stakeholders to begin with the Introduction section to gain an understanding of the project's purpose and scope before proceeding to specific feature requirements. By clarifying the intended audience and providing reading suggestions, the SRS document ensures that stakeholders can effectively engage with its contents and contribute to the project's success.

**1.4 Project Scope**

The Project Scope of the "Intern Admin Collaboration" project defines the boundaries and limitations of the software system. It outlines the specific functionalities and features that will be included in the system, as well as those that are excluded. In the context of intern management, event coordination, and administrative tasks, the project scope may encompass features such as dashboard management, intern profile creation, event scheduling, and task assignment. By clearly defining the project scope, the SRS document helps manage stakeholders' expectations and ensures that the development team focuses on delivering the essential functionalities required for project success.

**1.5 References**

The References section of the SRS document lists any external documents, standards, or sources that were consulted during the creation of the document. These references may include industry standards, regulatory requirements, previous project documents, or technical specifications relevant to the project. By citing external references, the SRS document provides a foundation of knowledge and ensures that the requirements and functionalities outlined in the document are informed by best practices and industry standards. Additionally, references serve as valuable resources for stakeholders seeking further information or clarification on specific topics addressed in the SRS.

**2. Overall Description**

**2.1 Product Perspective**

The Product Perspective section provides a holistic view of the software system within the larger context of the organization or existing systems. It outlines how the software interacts with other systems, dependencies, and interfaces. This may include diagrams illustrating the system's relationships with external components, data flows between different modules, and integration points with third-party services. By understanding the product perspective, stakeholders gain insight into the system's role in the organization's ecosystem and can better assess its impact on existing workflows and processes.

**2.2 Product Features**

Product Features section offers a high-level overview of the key functionalities and capabilities of the software system. It outlines what users can expect the system to do from their perspective. This includes summarizing features such as user authentication, data management, collaboration tools, reporting capabilities, and any other functionalities relevant to the project. Providing a concise summary of product features helps stakeholders understand the system's value proposition and aligns expectations regarding its capabilities.

**2.3 User Classes and Characteristics**

User Classes and Characteristics section identifies the different types of users who will interact with the system and describes their roles, responsibilities, and characteristics. For the "Intern Admin Collaboration" project, user classes may include administrators responsible for system management, interns accessing and updating their profiles, and possibly other stakeholders such as HR personnel or event coordinators. Understanding the needs and behaviors of each user class is crucial for designing user interfaces, defining access permissions, and tailoring functionalities to meet diverse user requirements.

**2.4 Operating Environment**

The Operating Environment section describes the technical environment in which the software system will operate. It includes hardware, software, and network requirements necessary for deploying and running the system successfully. This may involve specifying server configurations, supported operating systems, database management systems, web servers, and network infrastructure. By clearly defining the operating environment, developers can ensure compatibility and performance optimization, minimizing potential deployment issues and ensuring smooth operation of the system.

**2.5 Design and Implementation Constraints**

Design and Implementation Constraints section highlights any limitations or restrictions that may impact the development process. Constraints could include budgetary constraints, technological limitations, regulatory requirements, or organizational policies. For example, constraints may dictate the use of specific programming languages, compliance with industry standards, or adherence to project timelines and resource allocations. Understanding and documenting these constraints upfront helps the development team make informed decisions and prioritize requirements effectively, mitigating risks and ensuring project success.

**2.6 User Documentation**

User Documentation outlines the documentation requirements for end-users, such as user manuals, guides, or tutorials. It ensures that users have the necessary information to understand and effectively use the system. User documentation may include instructions on system navigation, feature descriptions, troubleshooting tips, and best practices. Providing comprehensive user documentation enhances user experience, reduces support overhead, and empowers users to make the most of the system's capabilities.

**2.7 Assumptions and Dependencies**

Assumptions and Dependencies section lists any assumptions made during the requirements gathering process and identifies external dependencies that may impact the project. Assumptions may include expectations about user behavior, system performance, or data availability. Dependencies may include reliance on third-party APIs, integration with external systems, or availability of specific resources. Documenting assumptions and dependencies helps manage expectations, identify potential risks, and ensures that the project remains on track despite uncertainties or external factors beyond the team's control.

**3. System Features:-**

**3.1 Dashboard**

The Dashboard feature serves as a central hub for administrators to oversee various aspects of the system. It offers a consolidated view of critical information such as recent activities, pending tasks, and important notifications. By providing administrators with a snapshot of the system's status and key metrics, the dashboard enables informed decision-making and efficient navigation to other sections of the system. Its intuitive design and customizable widgets enhance usability, allowing administrators to tailor the dashboard to their specific needs and preferences.

**3.2 Create Admin**

The Create Admin feature empowers authorized users to establish new administrator accounts within the system. Administrators tasked with user management responsibilities can input essential details such as username, password, and permissions for the newly created admin. Robust authentication and authorization mechanisms are essential to safeguard against unauthorized access and ensure that only authorized users possess the privilege to create new admin accounts. Proper validation of input data and adherence to security best practices are critical to maintaining the integrity and security of the system.

**3.3 Add Intern Details**

Add Intern Details feature enables administrators to input and manage information about interns enrolled in the program. It facilitates the collection and storage of crucial details such as personal information, contact details, and educational background within the system. Implementing robust data validation and verification processes ensures the accuracy and completeness of intern records, enhancing the reliability and usefulness of the system for administrators. Additionally, features such as search and filter functionalities streamline the process of accessing and managing intern details efficiently.

**3.4 Intern Approved Details**

Intern Approved Details feature provides administrators with comprehensive information about interns whose applications or requests have been approved. It offers visibility into the list of approved interns along with relevant details such as start date, end date, and assigned mentor. By centralizing approved internship information, this feature facilitates efficient management and coordination of internships within the organization. Advanced filtering and sorting options empower administrators to tailor the view according to specific criteria, enabling targeted decision-making and resource allocation.

**3.5 Intern Leave Details**

Intern Leave Details feature enables administrators to manage intern leave requests and maintain a comprehensive record of leave history within the system. Administrators can review incoming leave requests, approve or deny them based on organizational policies, and track the availability of interns accordingly. Integration with a calendar or scheduling system enhances the functionality of this feature, enabling administrators to visualize leave schedules and plan resource allocation effectively. Clear communication channels and automated notifications ensure timely updates and minimize disruptions caused by intern absences.

**3.6 Add Intern Salary**

Add Intern Salary feature facilitates the input and management of intern salary information within the system. It encompasses functionalities to specify salary rates, payment schedules, and other relevant details pertaining to intern compensation. Robust security measures are imperative to safeguard sensitive salary data and ensure compliance with legal and regulatory requirements. Additionally, features such as salary calculation tools and reporting capabilities enhance administrators' ability to manage intern compensation effectively and ensure transparency and fairness in salary administration processes.

**3.7 Intern Task Details**

Intern Task Details feature enables administrators to assign and track tasks allocated to interns within the organization. Administrators can create tasks, assign them to specific interns, set deadlines, and monitor task progress in real-time. This feature fosters collaboration and productivity by providing a structured framework for task management and accountability. Advanced features such as task prioritization, status tracking, and progress visualization empower administrators to optimize task allocation and resource utilization, driving efficiency and performance across intern teams.

**3.8 Intern Certificate Details**

Intern Certificate Details feature enables administrators to manage certificates awarded to interns upon completion of their internship program. It encompasses functionalities to generate, issue, and track certificates, including details such as certificate type, recipient name, and issuance date. Integration with a document management system streamlines certificate generation and distribution processes, ensuring accuracy and consistency in certificate administration. Advanced reporting capabilities enable administrators to gain insights into certificate issuance trends and track intern achievements effectively, enhancing the value and credibility of the internship program.

**3.9 Contact Us :-**

Integrating the Google Maps API into your ASP.NET project's "Contact Us" section enhances the user experience by providing an interactive and visually appealing way to display your business location. By embedding a Google Map, users can easily visualize your business address, explore nearby areas, and obtain directions. Additionally, the customization options available with the Google Maps API allow you to tailor the map's appearance and functionality to suit your specific needs, such as adding custom markers to highlight key locations or enabling geolocation services for seamless navigation. Overall, incorporating the Google Maps API enriches the "Contact Us" section of your ASP.NET project, offering users a convenient and intuitive way to connect with your business.

**3.10 Add Event Details**

Add Event Details feature allows administrators to create and manage event details within the system. Administrators can input essential event information such as title, description, date, time, location, and attendee list. Integration with calendar applications or event management tools enhances the functionality of this feature, enabling administrators to synchronize event details across multiple platforms seamlessly. Advanced features such as event reminders, notifications, and attendee communication tools streamline event coordination and ensure a seamless experience for participants. By centralizing event management processes within the system, administrators can optimize resource allocation and maximize the impact of events on intern engagement and development within the organization.

**4.1 User Interfaces**

User Interfaces (UI) are the graphical interfaces that allow users to interact with the software system. These interfaces should be intuitive, user-friendly, and responsive to ensure a positive user experience. For the "Intern Admin Collaboration" project, the user interfaces should be designed to accommodate administrators, interns, and other stakeholders with varying levels of technical expertise. The UI design should focus on providing easy access to system functionalities, clear navigation paths, and visually appealing layouts. Usability testing and feedback from end-users should inform the design process to ensure that the UI meets the needs and preferences of the target audience.

**4.2 Hardware Interfaces**

Hardware Interfaces describe the physical devices or components required to operate the software system. This may include servers, computers, networking equipment, and peripherals necessary for system deployment and operation. For the "Intern Admin Collaboration" project, hardware requirements may vary depending on factors such as the scale of deployment, user load, and performance expectations. The system should be designed to be compatible with a range of hardware configurations to accommodate different deployment scenarios and ensure scalability and reliability.

**4.3 Software Interfaces**

Software Interfaces specify how the software system interacts with other software components or external systems. This may include Application Programming Interfaces (APIs), libraries, databases, or third-party services integrated into the system architecture. For the "Intern Admin Collaboration" project, software interfaces may include integration with email services for notifications, authentication services for user login, or database management systems for data storage. Clear documentation and well-defined interfaces are essential to facilitate seamless integration with external systems and ensure interoperability and data consistency.

**4.4 Communications Interfaces**

Communications Interfaces define the protocols and mechanisms used for communication between different system components or external systems. This may include network protocols, data formats, and security mechanisms to ensure reliable and secure communication. For the "Intern Admin Collaboration" project, communication interfaces may include HTTP protocols for web-based interactions, RESTful APIs for data exchange, or encryption protocols for data security. Compliance with industry standards and best practices is critical to ensure interoperability, data integrity, and protection against security threats.

**5. Other Nonfunctional Requirements**

**5.1 Performance Requirements**

Performance Requirements specify criteria such as response time, throughput, and resource utilization that the system must meet to satisfy user expectations. For the "Intern Admin Collaboration" project, performance requirements may include ensuring fast loading times for dashboards, efficient processing of large datasets, and scalability to accommodate a growing number of users and data. Performance testing and optimization are essential to ensure that the system performs reliably under expected user loads and usage scenarios.

**5.2 Safety Requirements**

Safety Requirements outline measures to ensure the safety of users and data within the system. This may include implementing backup and recovery mechanisms to prevent data loss, ensuring data integrity and confidentiality, and complying with relevant safety standards and regulations. For the "Intern Admin Collaboration" project, safety requirements may include data encryption, access controls, and disaster recovery procedures to protect sensitive information and maintain system availability in case of unforeseen events.

**5.3 Security Requirements**

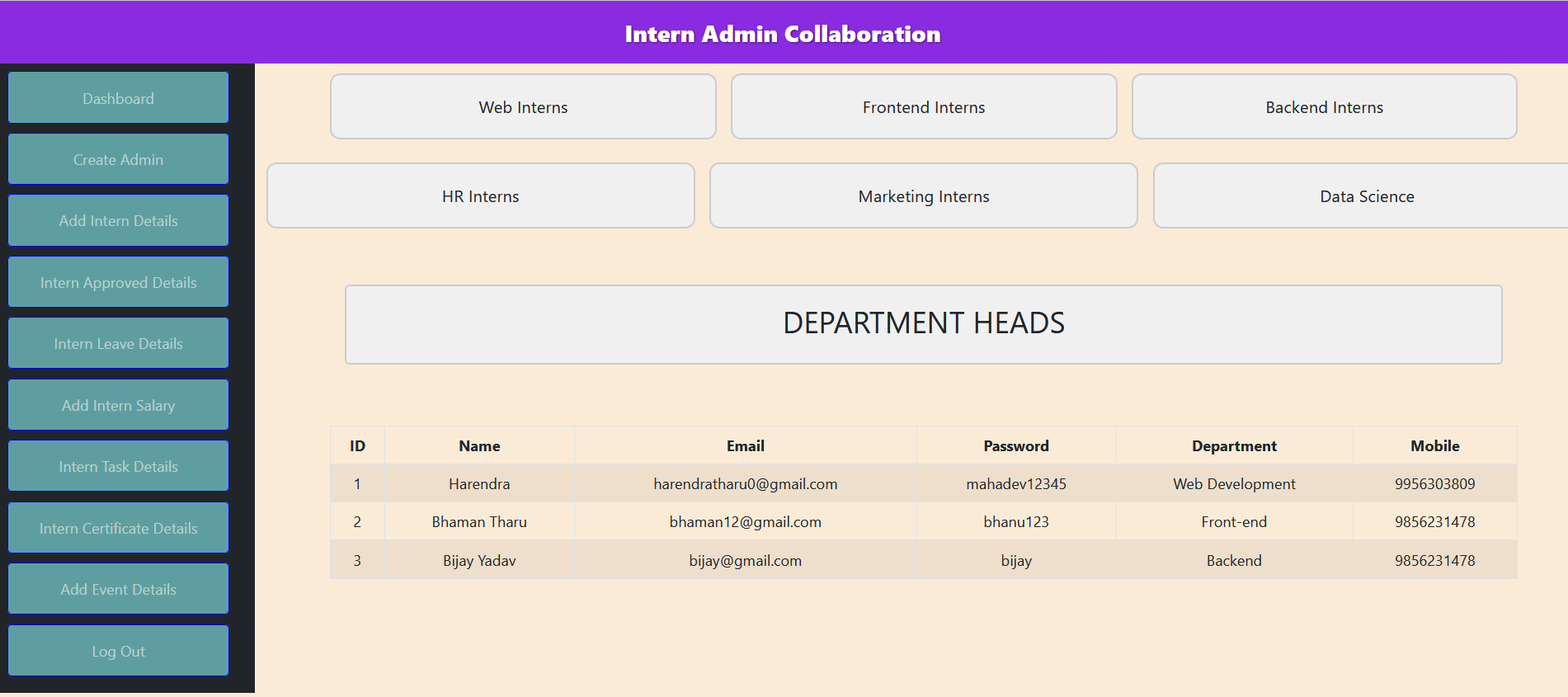
Security Requirements define the security measures implemented to protect the system from unauthorized access, data breaches, and other security threats. This may include user authentication and authorization mechanisms, encryption of sensitive data, audit trails for tracking user actions, and adherence to security best practices and compliance standards. For the "Intern Admin Collaboration" project, security requirements are critical to safeguarding confidential intern information, preventing unauthorized access to administrative functionalities, and ensuring the integrity and availability of the system.

**5.4 Software Quality Attributes**

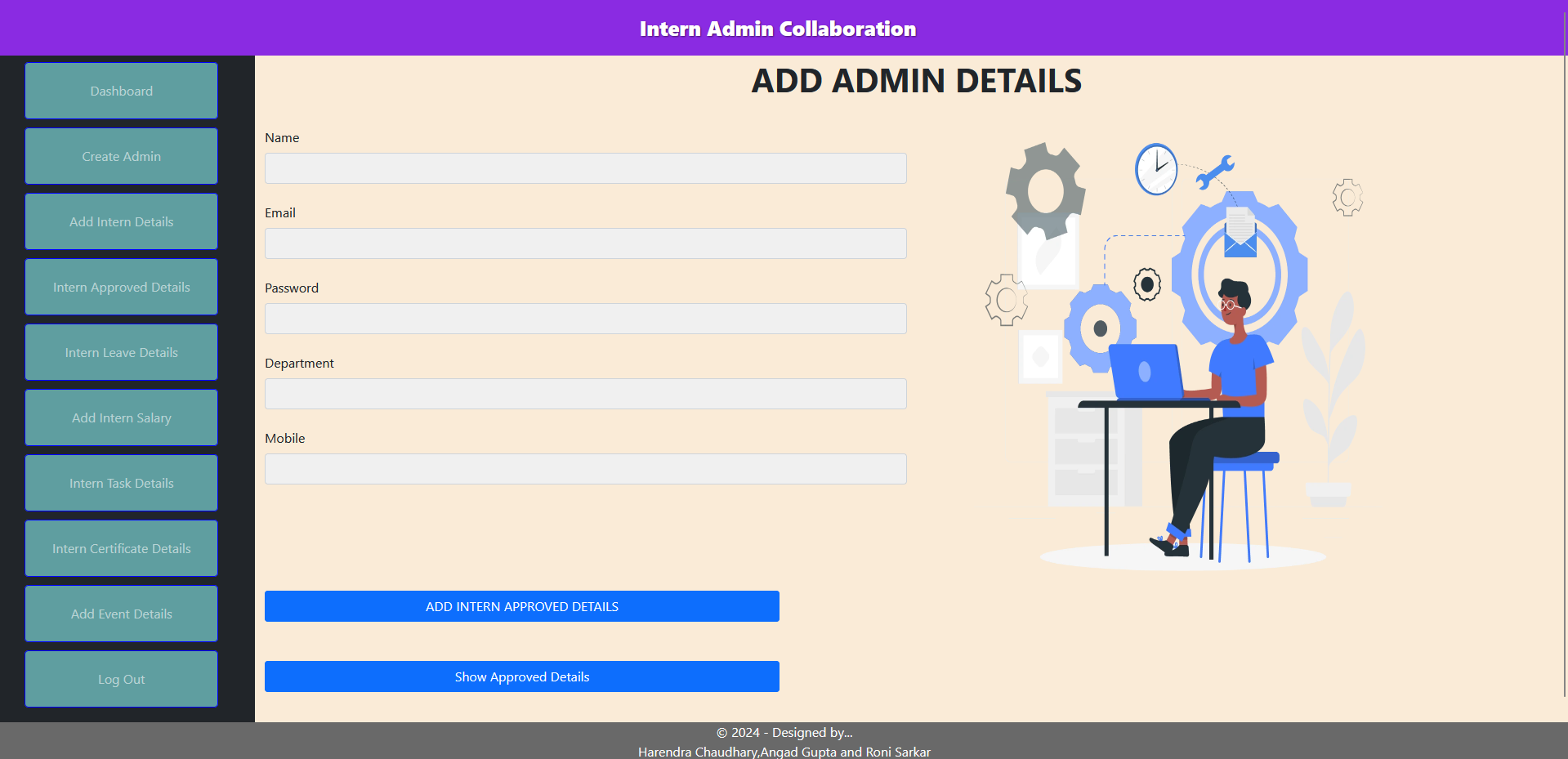
Software Quality Attributes specify characteristics such as reliability, maintainability, usability, and portability that contribute to the overall quality and usability of the software system. For the "Intern Admin Collaboration" project, software quality attributes may include providing clear and intuitive user interfaces, ensuring robustness and stability of the system, facilitating ease of maintenance and updates, and supporting deployment across different environments. Focusing on software quality attributes during the development process helps mitigate risks, improve user satisfaction, and enhance the long-term viability of the system.

**7.User Interface Design :-**

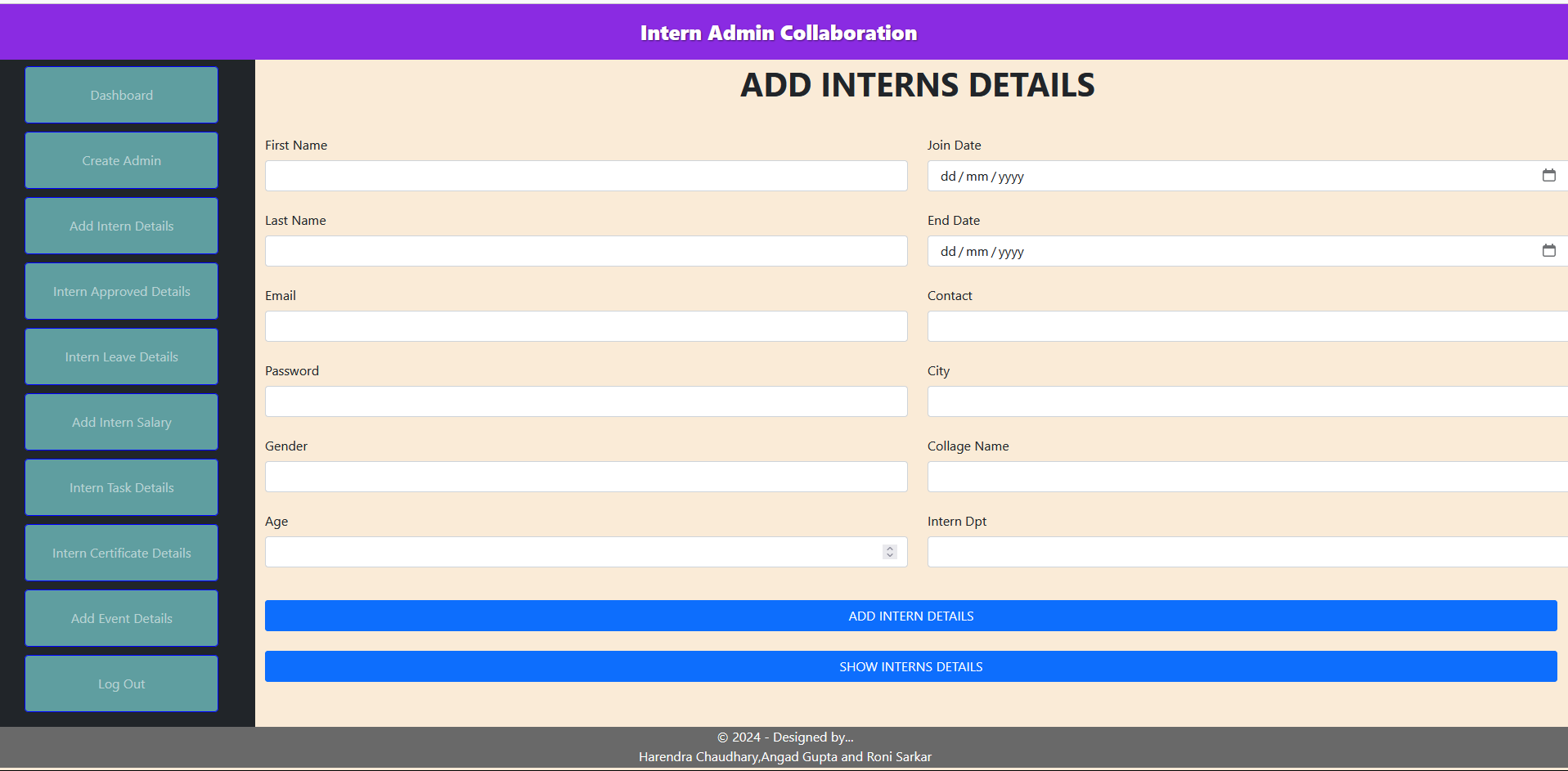
**7.1 Dashboard**

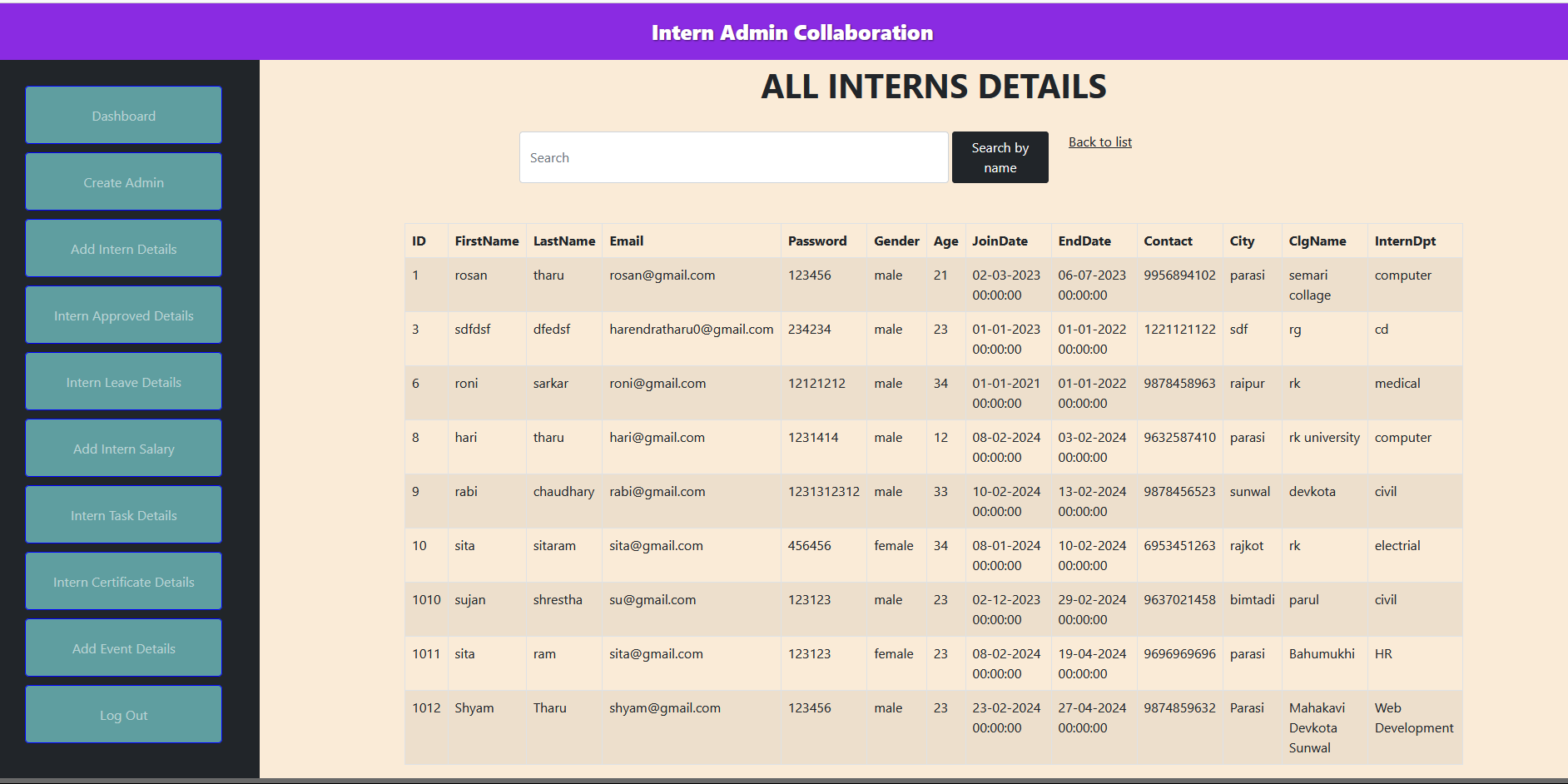
****

**7.2 Create Admin :-**

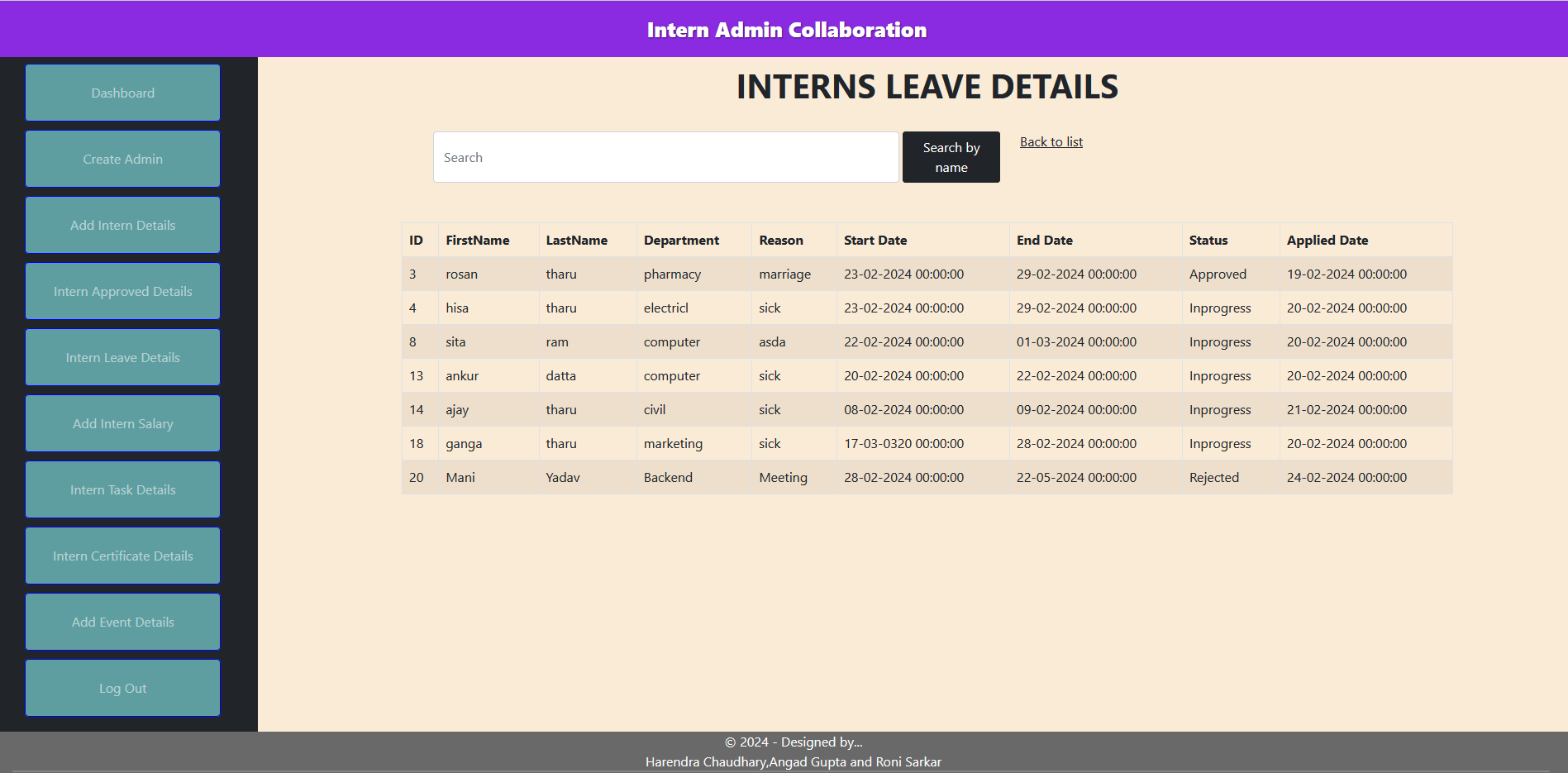
****

**7.3 Add Intern Details :-**

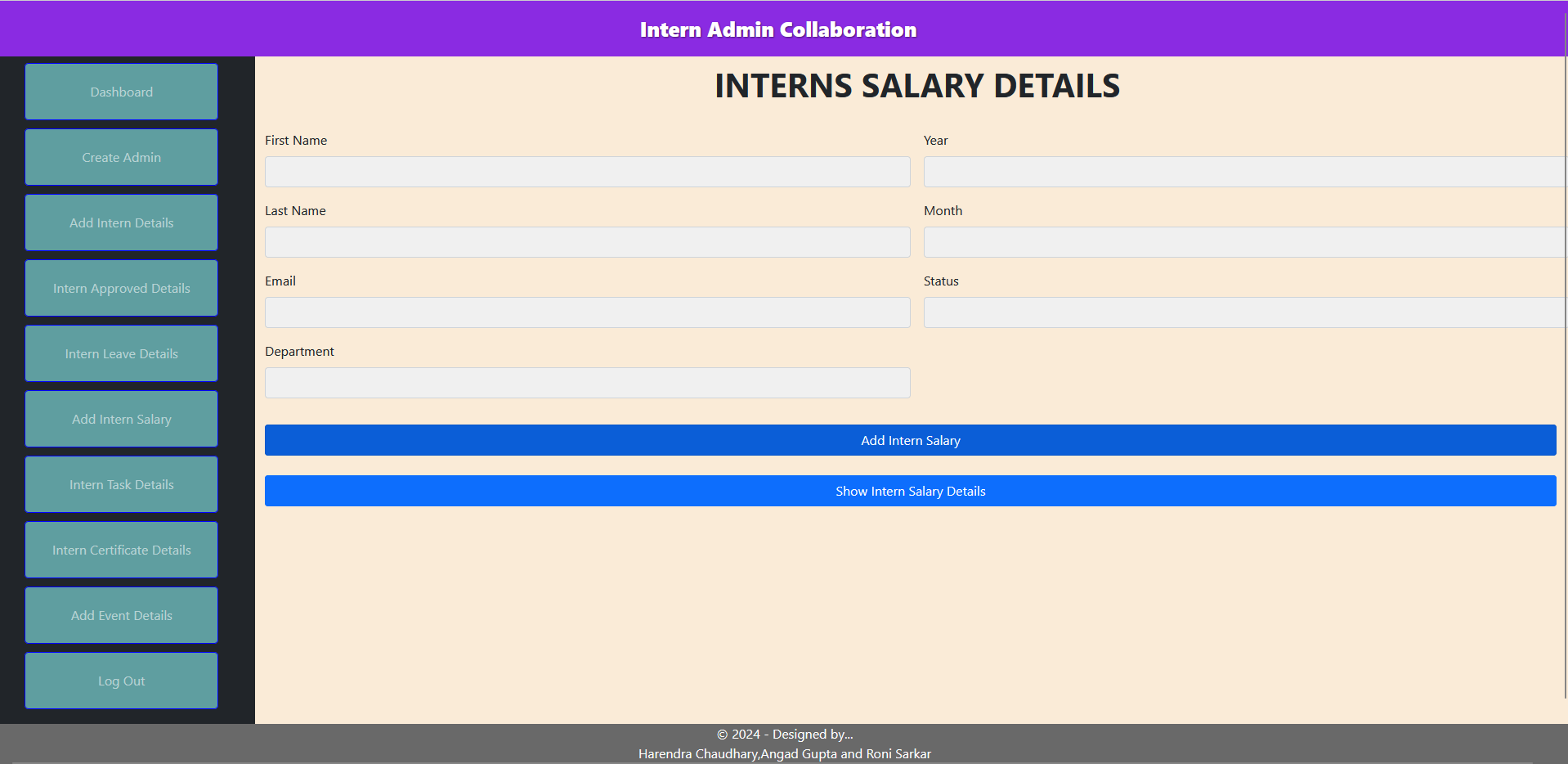
****

****

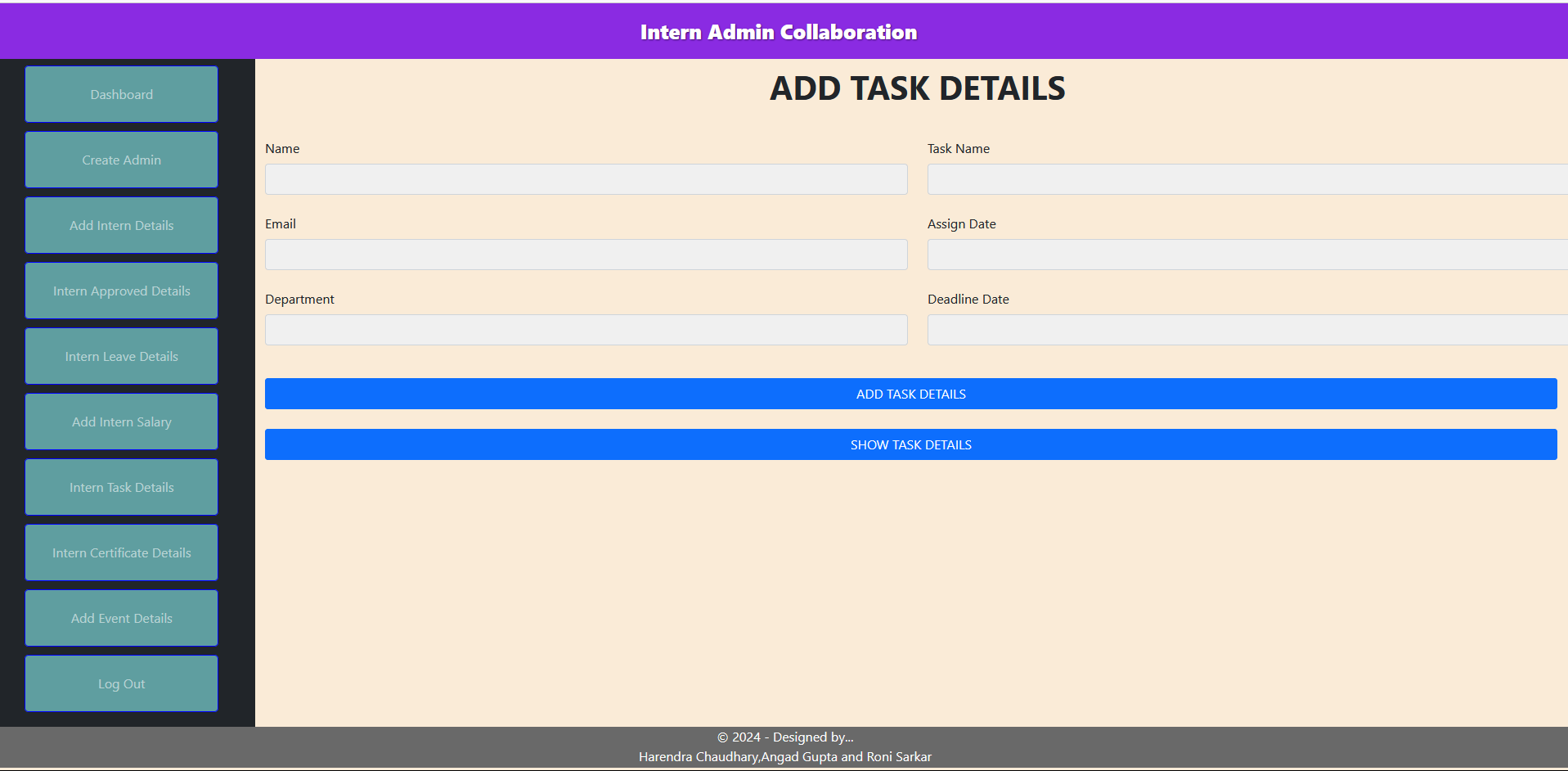
**7.4 Intern Leave Details :-**

****

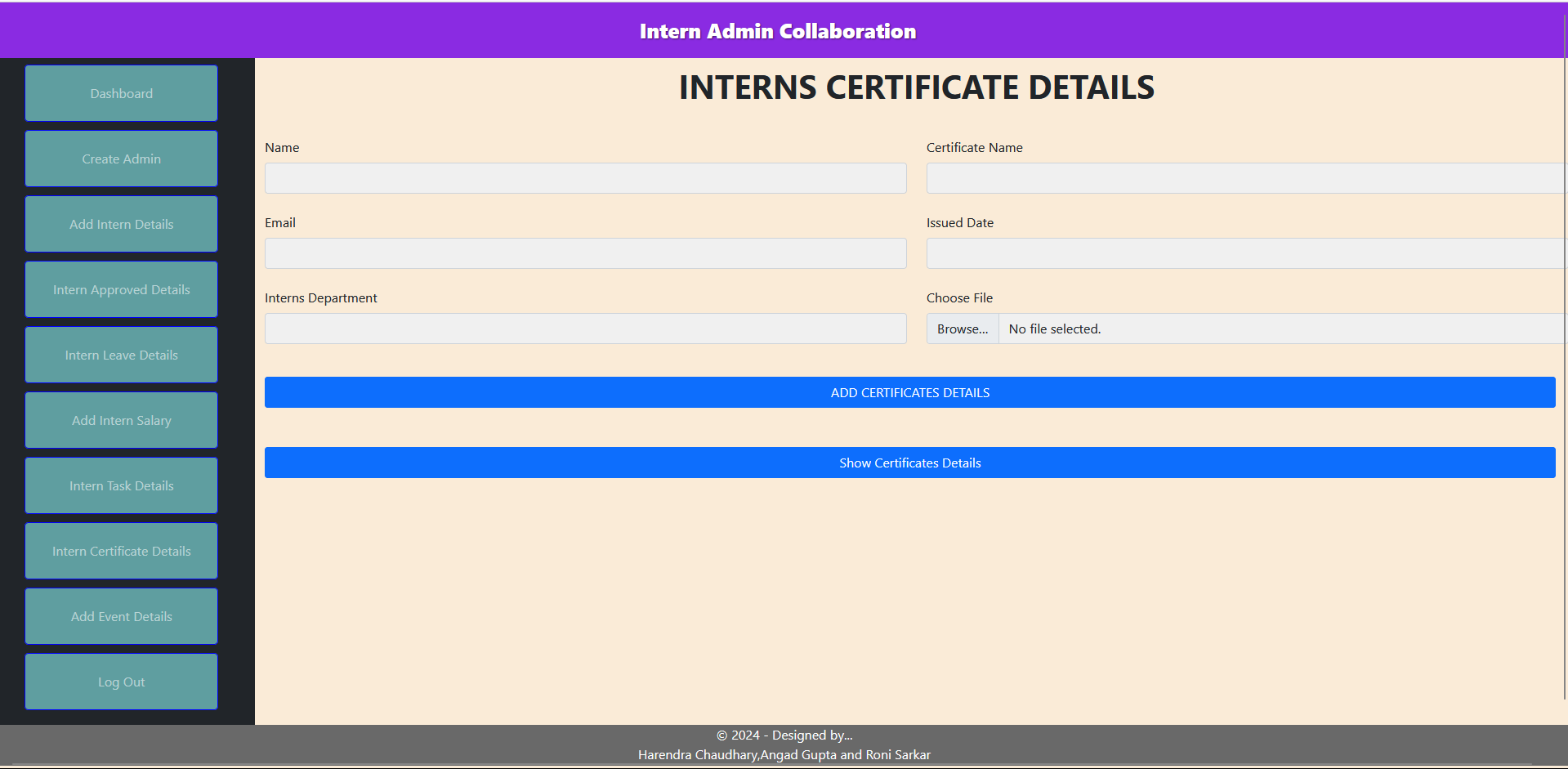
**7.5 Add Intern Salary :-**



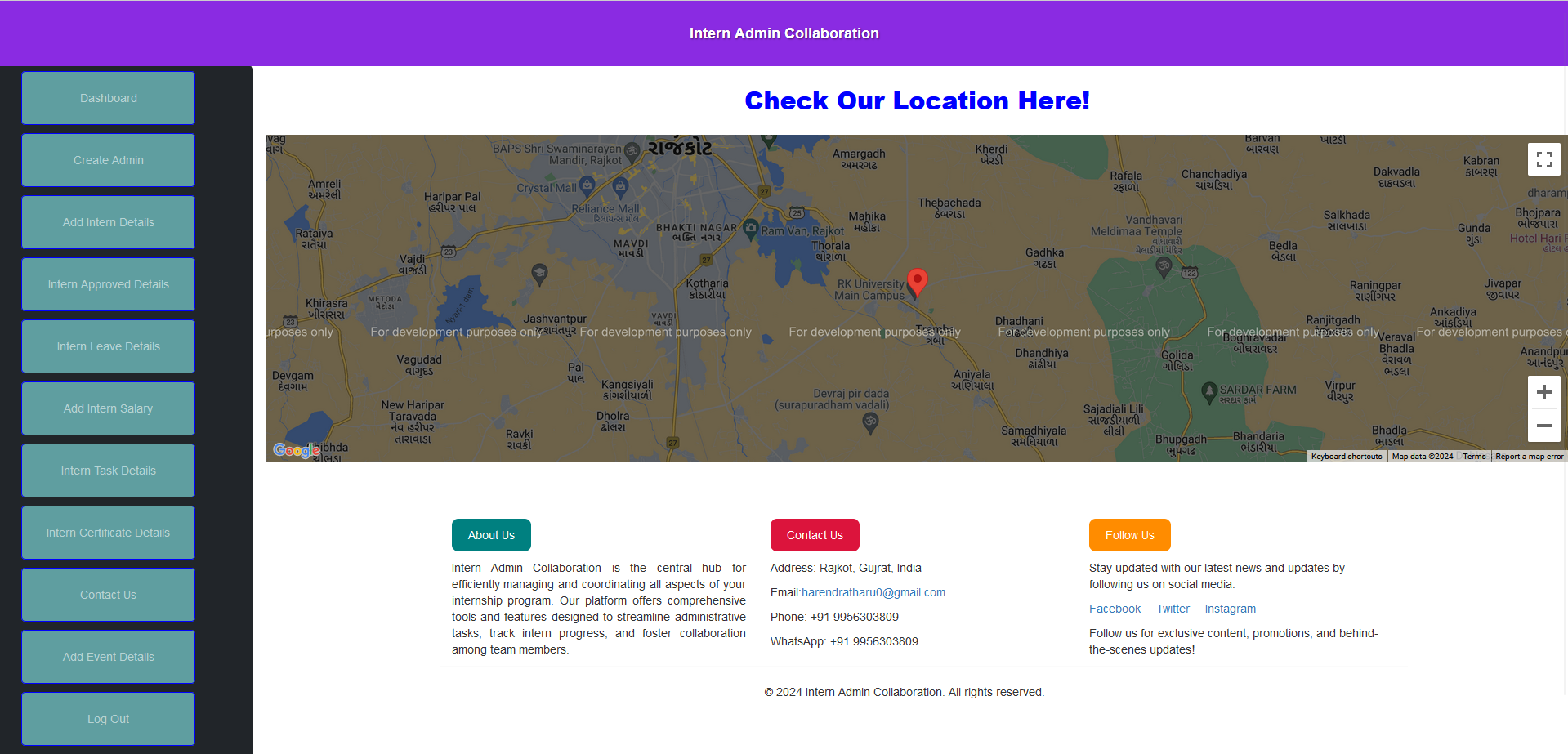
**7.6 Intern Task Details :-**

****

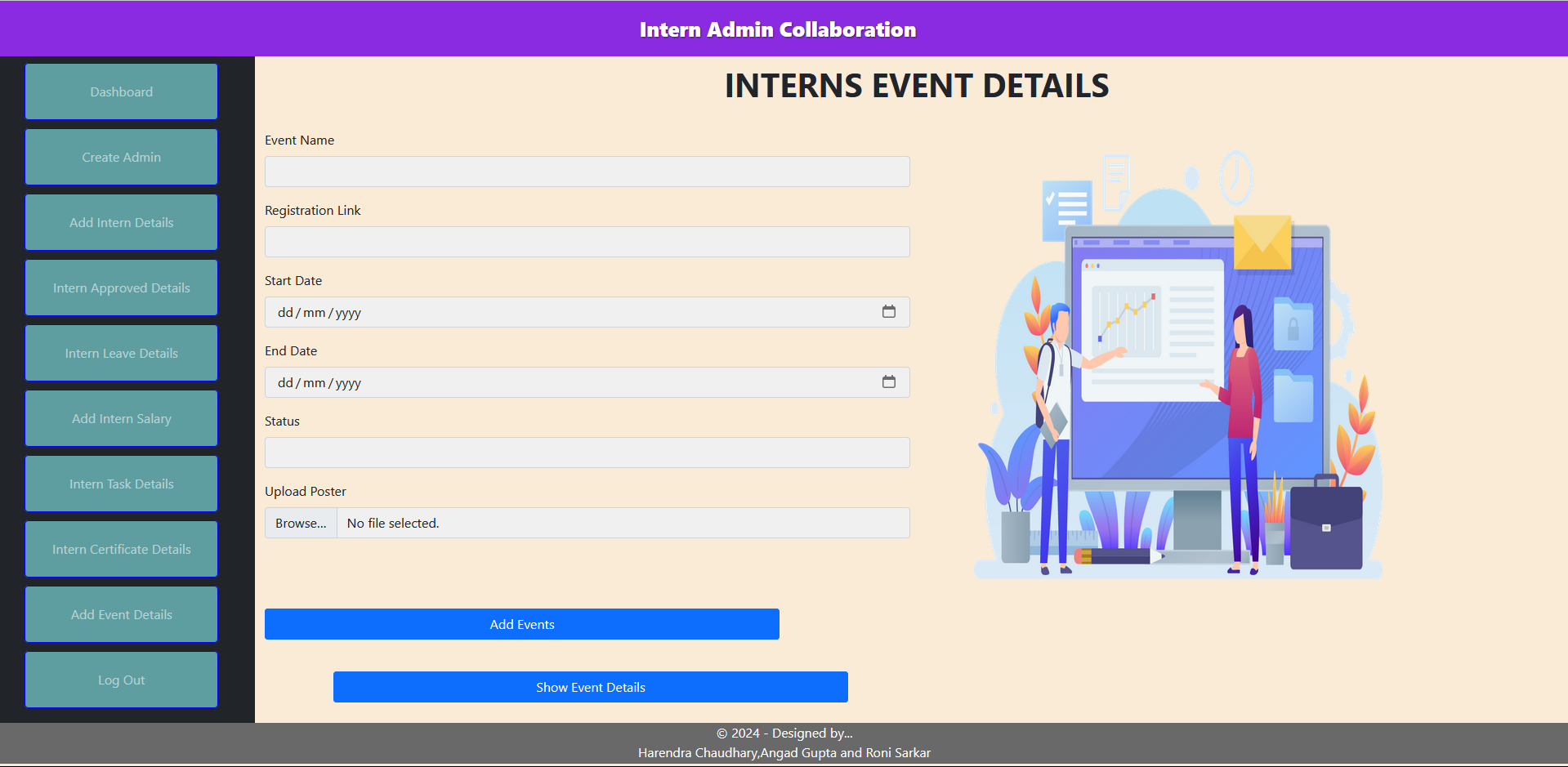
**7.7 Intern Certificate Details :-**

****

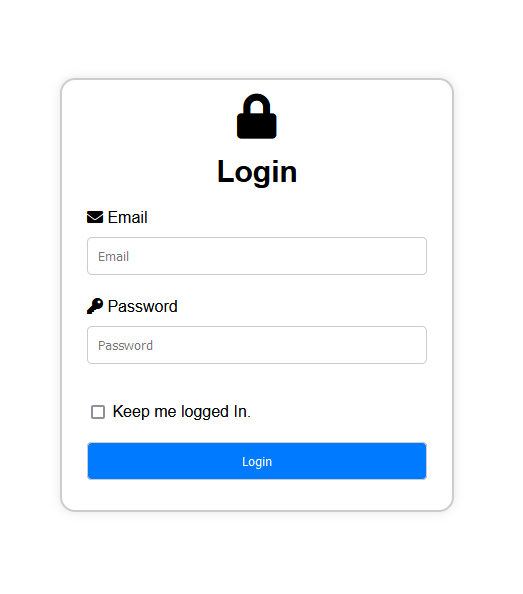
**7.8 Contact Us :-**

****

**7.8 Add Event Details :-**

****

**7.9 UI of Login Page :-**

****

**Other Requirements**

This section may include any additional requirements or considerations that do not fit into the previous categories. It could cover aspects such as legal and regulatory compliance, localization and internationalization requirements, integration with existing systems, or any other specific needs identified during the requirements analysis phase. Documenting these requirements ensures that all relevant considerations are addressed and incorporated into the software development process to meet the needs and expectations of stakeholders.

**Appendices**

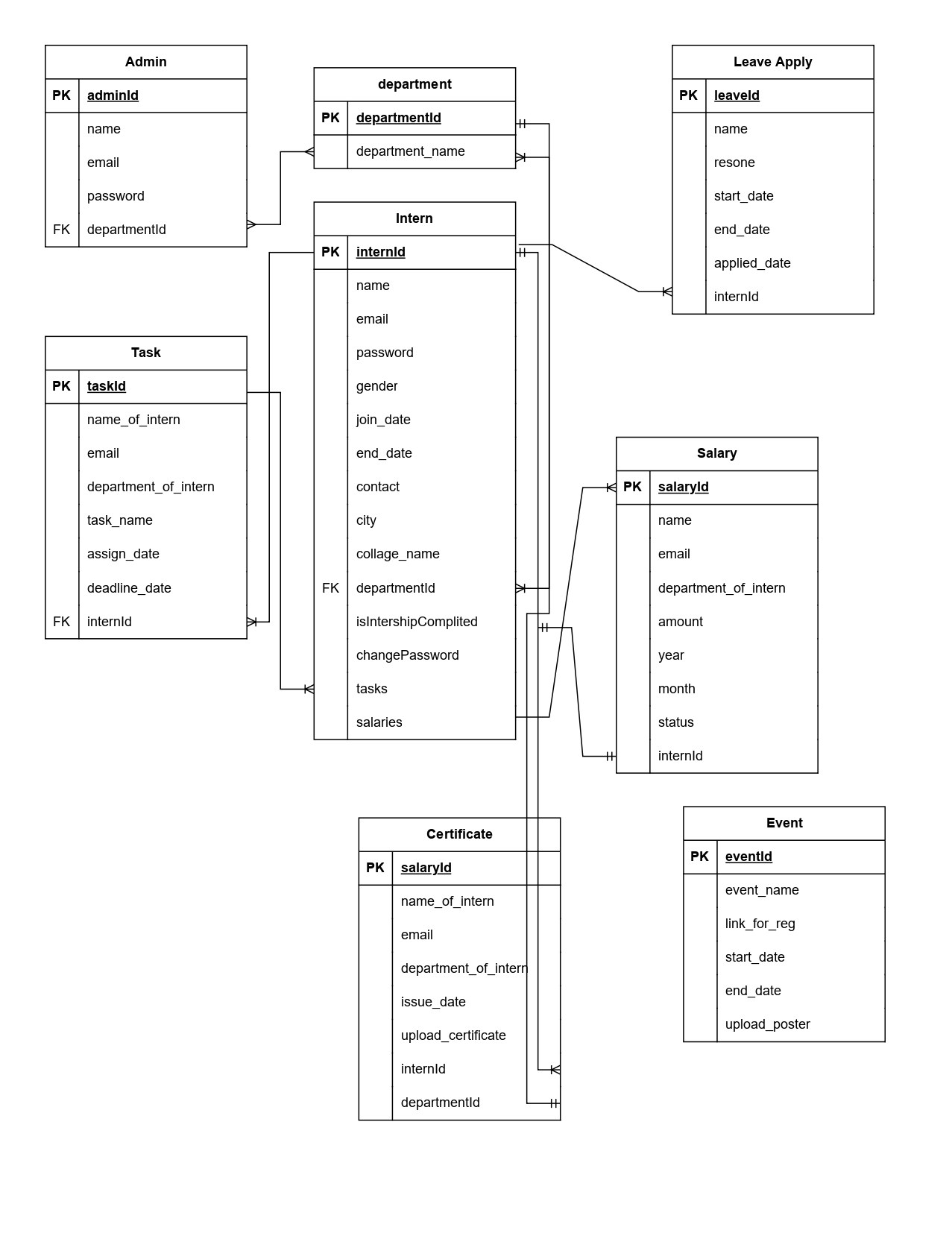
**Appendix A: Glossary**

The Glossary provides definitions of key terms and concepts used throughout the SRS document. It ensures clarity and consistency in communication among project stakeholders by establishing a common understanding of terminology.

**Appendix B: Analysis Models**

Analysis Models include diagrams, charts, or other visual representations used to analyze and illustrate various aspects of the system, such as use case diagrams, sequence diagrams, and data flow diagrams. These models help stakeholders better understand the system requirements and design.

**ER Diagram Of Interns Admin Collaboration :-**

****

**Use-Case Diagram of Intern Admin Collaboration :-**

****

**Appendix C: Issues List**

The Issues List documents any unresolved issues, questions, or concerns identified during the requirements gathering process. It serves as a reference for tracking and addressing issues throughout the project lifecycle, ensuring that all stakeholders' concerns are adequately addressed.

**Conclusion :-**

In conclusion, the "Intern Admin Collaboration" project aims to develop a comprehensive software system to facilitate seamless collaboration between administrators and interns within the organization. The SRS document outlined the project's objectives, scope, and requirements in detail, providing a clear roadmap for the development team.

Throughout the document, we defined various system features and functionalities, including the Dashboard, User Management, Intern Details Management, Event Coordination, and Administrative Tasks. These features are designed to streamline intern management processes, enhance communication, and improve overall administrative efficiency.

Additionally, the SRS document outlined external interface requirements, nonfunctional requirements, and other considerations such as performance, safety, security, and software quality attributes. These requirements ensure that the system meets the organization's needs while adhering to industry standards and best practices.

The inclusion of analysis models, such as Data Flow Diagrams (DFD), Use Case diagrams, and flowcharts, provides stakeholders with visual representations of the system's architecture, functionality, and interactions.

In conclusion, the "Intern Admin Collaboration" project represents a significant opportunity to enhance organizational efficiency, improve collaboration, and provide a better experience for administrators and interns alike. By adhering to the requirements outlined in this document, we aim to deliver a robust and user-friendly software solution that meets the organization's needs and exceeds stakeholder expectations.