**A Midterm Progress Report**

**on**

**SMART WORKSPHERE**

**Submitted in partial fulfillment of the requirements for the award of**

**the degree of**

**BACHELOR OF TECHNOLOGY**

COMPUTER SCIENCE AND ENGINEERING

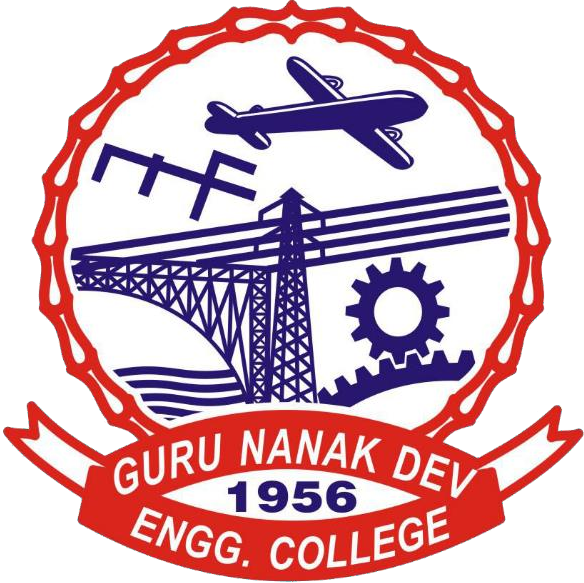
SUBMITTED BY-

Aditya Yadav(2104059)

Ajay Kumar(2104061)

Aryan Garg(2104077)

Under the guidance of Dr. Vivek Thapar



**Department of Computer Science and Engineering**

**GURU NANAK DEV ENGINEERING COLLEGE, LUDHIANA**

**TABLE OF CONTENTS**

**Contents Page No.**

INTRODUCTION 1

SYSTEM REQUIREMENTS 4

SOFTWARE REQUIREMENT ANALYSIS 7

SOFTWARE DESIGN 11

TESTING MODULE 17

PERFORMANCE OF THE PROJECT DEVELOPED 20

OUTPUT SCREENS 22

REFERENCES 29

**INTRODUCTION**

**Smart Worksphere** is a comprehensive Human Resource Management System (HRMS) designed to streamline and automate the core processes of an organization. The project aims to provide a centralized platform that facilitates employee management, attendance tracking, payroll generation, project management, and client interactions. By integrating essential HR functions into a single system, Smart Worksphere reduces manual work, minimizes errors, and enhances organizational efficiency.

Smart Worksphere is an all-in-one Human Resource Management System (HRMS) designed to simplify and streamline various HR-related tasks within an organization. Developed using a robust technology stack that includes **PHP (Laravel)**, **React.js**, **Node.js**, **MySQL**, **HTML**, **CSS**, **Bootstrap**, and **JavaScript**, the system ensures seamless operations across different departments. It serves as a centralized platform, connecting employees, administrators, and management for efficient communication and task management.

The platform offers a comprehensive set of features, including **employee management**, **attendance tracking**, **payroll generation**, **project management**, and **client handling**. Additionally, it supports real-time communication through a **One-to-One Chat App**, ensuring quick collaboration and decision-making. The system also provides a **ticketing system** for employees to raise concerns regarding assets or technical issues, which are then addressed by the administrative team.

To address the challenges faced by traditional HR management systems, **Smart Worksphere** incorporates automated payroll generation, attendance management, and employee record maintenance. The inclusion of **role-based access control (RBAC)** ensures data security by restricting unauthorized access to sensitive information. Each user is assigned specific roles and permissions, providing a personalized experience based on their responsibilities.

The **Project Management Module** further enhances collaboration by allowing managers to assign tasks, monitor progress using a taskboard, and set deadlines. Employees can view their assigned tasks, update their status, and upload necessary documents. This systematic approach reduces miscommunication and improves task completion rates. Additionally, the **Client Management Module** enables businesses to maintain organized client data, track project interactions, and generate detailed reports.

Another critical aspect of **Smart Worksphere** is its **Attendance Management Module**, which provides a curated view of employee attendance data. Filters based on employee name, month, or year allow easy tracking and reporting. Similarly, the **Payroll Management Module** ensures accurate salary generation based on attendance data, deductions, and allowances. Generated payslips can be securely accessed by employees, reducing administrative workload.

**Localization** plays a significant role in the system, allowing companies to customize settings such as **language**, **currency**, **date formats**, and **time zones**. This makes **Smart Worksphere** adaptable for use across different geographical regions. Furthermore, the **Backup Management Module** ensures data safety by providing regular backups of application code and database information.

Organizations today face multiple challenges in managing their workforce effectively. Traditional HR management methods often involve extensive paperwork, manual attendance tracking, and inefficient payroll processing. Moreover, the absence of a unified platform leads to fragmented data management, causing delays in decision-making and operational inefficiencies. Smart Worksphere addresses these challenges by offering a digital solution that automates routine HR tasks, ensuring seamless communication and collaboration between employees, management, and clients.

The platform is equipped with various modules designed to cater to the specific needs of organizations. It offers a user-friendly interface that simplifies day-to-day HR operations. From managing employee profiles and tracking attendance to generating payroll reports and handling project tasks, Smart Worksphere provides a complete ecosystem for HR management. Additionally, it includes features such as real-time chat for internal communication, ticketing systems for issue management, and role-based access controls to ensure data security.

**Objectives**

The development of Smart Worksphere revolves around three primary objectives:

1. **Employee Management, Attendance Tracking, and Payroll Generation:**
   * The platform enables organizations to manage employee records efficiently, storing information such as personal details, educational background, work experience, and emergency contacts.
   * The attendance tracking module ensures accurate and automated recording of employee working hours, reducing errors and ensuring transparency.
   * The payroll generation feature automates salary calculations, including allowances and deductions, simplifying the payroll management process.
2. **Project Management, Asset Handling, and Client Interactions:**
   * The system provides robust project management features, including task assignments, progress tracking, and collaboration through taskboards.
   * Organizations can efficiently manage their assets, track asset usage, and resolve asset-related issues using the asset management module.
   * The client management module ensures smooth interactions with clients by maintaining detailed client records and facilitating effective communication.
3. **User Accessibility and System Customization:**
   * Smart Worksphere offers role-based access and permission management, ensuring that users have access only to the information relevant to their responsibilities.
   * Localization settings allow organizations to customize the system based on their regional preferences, including language, time zones, currency, and date formats.
   * Customizable themes and personalized dashboards provide users with a comfortable and efficient working experience.

**Conclusion**

Smart Worksphere is designed to bridge the gap between traditional HR processes and modern technological advancements. By providing a robust and scalable solution, it enhances operational efficiency and decision-making within organizations. The platform’s emphasis on automation, collaboration, and customization makes it a versatile choice for businesses aiming to streamline their HR operations. Through its implementation, organizations can expect improved productivity, reduced administrative burdens, and better employee engagement.

**SYSTEM REQUIREMENTS**

The development and deployment of **Smart Worksphere** require a combination of software and hardware resources to ensure efficient operation, scalability, and reliability. This section outlines the essential software and hardware requirements for both the development environment and the production environment.

**Software Requirements**

**1. Development Tools and Frameworks**

* **PHP**: Used as the primary backend language for server-side scripting and logic.
* **Laravel**: A PHP framework used to develop a robust and scalable backend with built-in security features.
* **React.js**: For building a dynamic and responsive user interface.
* **Node.js**: To enable real-time data communication and handle server-side processes.
* **Bootstrap**: For creating modern, responsive designs with minimal effort.
* **JavaScript**: For client-side interactivity and functionality.

**2. Database Management**

* **MySQL**: A relational database management system used for storing and managing application data.
* **phpMyAdmin**: For easy database management and administration.

**3. Operating System**

* **Windows 10/11** or **Linux (Ubuntu/CentOS)**: Preferred for development and deployment.
* **macOS**: Optional for development purposes.

**4. Web Servers**

* **Apache HTTP Server**: Suitable for local development using XAMPP or WAMP.
* **Nginx or Apache**: Recommended for production deployment on cloud or dedicated servers.

**5. Development Environment**

* **Visual Studio Code (VS Code)**: Recommended code editor for developing front-end and back-end components.
* **Postman**: For API testing and debugging.
* **Git & GitHub**: For version control and collaboration.
* **XAMPP/WAMP**: For creating a local server environment during development.

**6. Third-Party Tools and Libraries**

* **Axios**: For API requests and data fetching.
* **JWT (JSON Web Tokens)**: For secure authentication and authorization.
* **Socket.IO**: For real-time chat functionality.
* **Lodash and Moment.js**: For utility functions and date-time management.

**Hardware Requirements**

**1. Development Environment**

* **Processor**: Intel Core i5/i7 or AMD equivalent (8th Gen or higher)
* **RAM**: Minimum 8GB (16GB recommended for large-scale development)
* **Storage**: SSD with at least 500GB storage capacity.
* **Graphics Card**: Integrated or discrete GPU (not mandatory but recommended for UI rendering).
* **Internet Connection**: Stable internet connection for API testing and version control.

**2. Production Server Requirements**

* **Processor**: Intel Xeon or AMD Ryzen equivalent (Multi-core)
* **RAM**: 16GB to 32GB depending on the user base.
* **Storage**: 1TB SSD with scalable storage options.
* **Operating System**: Linux-based server for secure and efficient hosting.
* **Database Server**: Separate database server recommended for large datasets.
* **Backup Solution**: Cloud-based storage with automated backup options.

**3. Client Requirements**

* **Device**: Desktop, Laptop, or Mobile Device
* **Browser**: Google Chrome, Firefox, Safari, or Microsoft Edge (latest versions)
* **Operating System**: Windows, macOS, Linux, iOS, or Android.
* **RAM**: Minimum 4GB for desktops and laptops.
* **Internet Connection**: Required for accessing the web-based application.

**Additional Considerations**

* **Security**: SSL certificates for secure communication between the server and client.
* **Load Balancer**: For distributing incoming network traffic across multiple servers to ensure reliability and performance.
* **Monitoring Tools**: Tools like Prometheus or Grafana for system monitoring.
* **Cloud Services**: AWS, Azure, or Google Cloud for scalable hosting and data storage.

By meeting these software and hardware requirements, **Smart Worksphere** can function efficiently, providing a reliable and user-friendly experience for both employees and administrators.

**SOFTWARE REQUIREMENT ANALYSIS**

The development and deployment of **Smart Worksphere** require a combination of software and hardware resources to ensure efficient operation, scalability, and reliability. This section outlines the essential software and hardware requirements for both the development environment and the production environment.

**Software Requirement Analysis**

**Problem Definition**

Organizations often struggle with fragmented HR management systems that require manual interventions and lack centralized control. Common challenges include inefficient employee management, inaccurate attendance tracking, delayed payroll processing, ineffective communication, and poor project coordination. **Smart Worksphere** addresses these issues by providing an integrated, automated, and scalable solution that enhances organizational productivity and ensures seamless operations.

**Modules and Their Functionalities**

**1. Employee Management**

* Manage employee profiles with personal details, education, work experience, and emergency contacts.
* Maintain a structured record of employee data for easy access.
* Enable admins to add, edit, or delete employee information.

**2. Attendance Tracking**

* Track employee attendance using an automated system.
* Generate monthly attendance reports.
* Implement filters for employee name, month, and year for customized reports.

**3. Payroll Management**

* Automate payroll generation based on attendance records.
* Manage employee salary details, allowances, and deductions.
* Generate and download payslips in PDF format.

**4. Project Management**

* Create and manage projects using taskboards.
* Assign project leads and team members.
* Track project progress using a visual task management interface.

**5. Asset Management**

* Maintain a database of company assets.
* Assign assets to employees and track usage.
* Raise and manage asset-related issues.

**6. Client Management**

* Maintain detailed client records.
* Facilitate efficient client communication.
* Provide a card and tabular view for quick client management.

**7. Tickets Management**

* Enable employees to raise tickets for issues.
* Assign tickets to employees for resolution.
* Track ticket progress and allow conversations between relevant parties.

**8. User Management**

* Provide CRUD functionality for managing system users.
* Ensure role-based access control with defined permissions.

**9. Backup Management**

* Create and manage application and database backups.
* Allow admins to download, restore, or delete backups.

**10. Localization**

* Customize system settings like language, time zone, currency, and date format.
* Provide region-specific configurations for seamless usability.

**11. Real-Time Chat**

* Facilitate one-to-one chats for internal communication.
* Ensure secure and private conversations.

**Software Requirements**

**1. Development Tools and Frameworks**

* **PHP**: Used as the primary backend language for server-side scripting and logic.
* **Laravel**: A PHP framework used to develop a robust and scalable backend with built-in security features.
* **React.js**: For building a dynamic and responsive user interface.
* **Node.js**: To enable real-time data communication and handle server-side processes.
* **Bootstrap**: For creating modern, responsive designs with minimal effort.
* **JavaScript**: For client-side interactivity and functionality.

**2. Database Management**

* **MySQL**: A relational database management system used for storing and managing application data.
* **phpMyAdmin**: For easy database management and administration.

**3. Operating System**

* **Windows 10/11** or **Linux (Ubuntu/CentOS)**: Preferred for development and deployment.
* **macOS**: Optional for development purposes.

**4. Web Servers**

* **Apache HTTP Server**: Suitable for local development using XAMPP or WAMP.
* **Nginx or Apache**: Recommended for production deployment on cloud or dedicated servers.

**5. Development Environment**

* **Visual Studio Code (VS Code)**: Recommended code editor for developing front-end and back-end components.
* **Postman**: For API testing and debugging.
* **Git & GitHub**: For version control and collaboration.
* **XAMPP/WAMP**: For creating a local server environment during development.

**6. Third-Party Tools and Libraries**

* **Axios**: For API requests and data fetching.
* **JWT (JSON Web Tokens)**: For secure authentication and authorization.
* **Socket.IO**: For real-time chat functionality.
* **Lodash and Moment.js**: For utility functions and date-time management.

**SOFTWARE DESIGN**

The Smart Worksphere project follows a structured software design approach to ensure efficiency, scalability, and ease of use. The system is developed using PHP (Laravel), React, Node.js, MySQL, HTML, CSS, Bootstrap, and JavaScript, ensuring a full-stack web application with a robust backend and dynamic frontend. The design consists of architectural design, database design, module design, and user interface design to ensure seamless functionality and integration.

**1. Architectural Design**

The **Smart Worksphere** system follows a **three-tier architecture**, which includes:

1. **Presentation Layer (Frontend)**
   * Developed using **React.js** for a responsive and interactive user interface.
   * Uses **Bootstrap, CSS, and JavaScript** for styling and layout.
   * Handles user interactions, such as employee management, attendance tracking, and project handling.
2. **Application Layer (Backend)**
   * Built with **PHP (Laravel)** for efficient server-side processing.
   * Uses **Node.js** for handling real-time operations, such as the one-to-one chat system.
   * Implements **RESTful APIs** for seamless communication between the frontend and backend.
3. **Data Layer (Database)**
   * Uses **MySQL** for structured data storage, ensuring efficient management of employees, attendance, payroll, and projects.
   * Ensures **data integrity and security** with proper indexing and relational database design.

**2. Database Design**

The database is structured to maintain **relational integrity** and ensure **efficient data retrieval**. Key tables include:

* **Users Table**: Stores admin and employee credentials, roles, and permissions.
* **Employees Table**: Manages employee details, designations, and assigned assets.
* **Attendance Table**: Tracks employee attendance records with timestamps.
* **Payroll Table**: Maintains payroll details, salary breakdowns, and payment history.
* **Projects Table**: Stores project details, assigned employees, and task progress.
* **Clients Table**: Holds client information and interaction history.
* **Tickets Table**: Manages support tickets, assigned employees, and resolution status.

The **ER (Entity-Relationship) Diagram** is designed to ensure data normalization, reducing redundancy and improving efficiency.

**3. Module Design**

The system is divided into various modules, each handling a specific set of functions. The key modules are:

1. **Employee Management Module**
   * Allows CRUD (Create, Read, Update, Delete) operations on employee profiles.
   * Manages personal information, emergency contacts, and work experience.
2. **Attendance Tracking Module**
   * Provides a tabular view of attendance with filters for employee name, month, and year.
   * Automates attendance recording using timestamps.
3. **Payroll Management Module**
   * Generates employee payslips with allowances and deductions.
   * Maintains payroll history for financial transparency.
4. **Project Management Module**
   * Allows project creation, task assignment, and progress tracking.
   * Implements a **task board** for better visualization of ongoing projects.
5. **Client Management Module**
   * Stores client details in tabular and card views.
   * Provides client interaction history for better communication.
6. **Asset Management Module**
   * Assigns assets to employees and tracks usage.
   * Allows employees to raise issues regarding assets, notifying the admin.
7. **Ticketing System Module**
   * Enables employees to create tickets for issues, which are assigned to HR or IT personnel.
   * Ensures that only admins receive ticket notifications via email.
8. **User & Role Management Module**
   * Admins can manage users and their access permissions.
   * Role-based access control ensures security and compliance.
9. **Backup & Security Module**
   * Allows data backup and restoration (both database and full application).
   * Implements role-based restrictions for sensitive operations.
10. **Localization & Settings Module**

* Allows customization of language, currency, date format, and timezone.
* Enables invoice customization with company details and logo upload.

**4. User Interface (UI) Design**

The UI design focuses on **simplicity, responsiveness, and accessibility**. The key principles include:

* **Dashboard Interface**: Provides a summarized view of employees, projects, attendance, and payroll.
* **Navigation Bar**: Includes quick links to all modules for seamless navigation.
* **Forms & Modals**: Used for adding/editing data in a structured format.
* **Filters & Search**: Allows users to filter and search for employees, projects, and clients efficiently.
* **Real-Time Notifications**: Displays updates on attendance, tickets, and payroll status.

**5. Security & Performance Considerations**

* **User Authentication**: Secure login using Laravel authentication.
* **Role-Based Access Control (RBAC)**: Ensures only authorized users can access specific modules.
* **Data Encryption**: Sensitive data such as employee payroll is encrypted for security.
* **Performance Optimization**: Uses caching mechanisms and optimized database queries for faster loading.

**Design Patterns and Principles**

* **Model-View-Controller(MVC)**:  
  The system uses the **MVC architecture** in the Laravel backend to separate concerns, making the codebase organized and maintainable. Models handle data operations, Views manage UI representation, and Controllers process requests and logic.
* **Component-BasedDesign**:  
  The frontend in **React.js** follows a component-based architecture. Reusable components for tables, forms, and dashboards ensure faster development and maintainability.
* **RESTfulAPIDesign**:  
  The backend provides APIs that follow **RESTful principles**, allowing easy data communication between the frontend and backend. APIs ensure seamless data exchange for employee management, attendance tracking, and project handling.
* **StateManagement**:  
  The project uses **React’s Context API** and **Redux** for efficient state management. This helps in managing user data, authentication status, and real-time chat messages without unnecessary re-rendering.
* **AsynchronousOperations**:  
  **Axios** is used for asynchronous HTTP requests, ensuring smooth communication with APIs. Error-handling mechanisms are integrated to manage failed requests.

**Security Measures in Design**

* **Role-BasedAccessControl(RBAC)**:  
  The system ensures restricted access using **RBAC**. Users are assigned specific roles (e.g., Admin, Employee, Client), and permissions are granted accordingly. Sensitive modules like payroll management are accessible only to authorized users.
* **DataEncryption**:  
  Sensitive information, including passwords and payroll data, is encrypted using **Laravel’s encryption library**. Additionally, data transmission between the client and server is secured using **HTTPS**.
* **Token-BasedAuthentication**:  
  **JWT (JSON Web Tokens)** is used for user authentication. Tokens ensure secure and scalable login sessions, preventing unauthorized access.
* **InputValidationanSanitization**:  
  All user inputs are validated and sanitized using Laravel’s built-in validation system to prevent SQL injection and cross-site scripting (XSS) attacks.

**Error Handling and Logging**

* **ErrorManagement**:  
  The backend uses **Laravel’s exception handling mechanism** to catch and manage errors. Errors are logged for troubleshooting using the **Monolog** library.
* **FrontendErrorDisplay**:  
  Errors encountered during API calls are displayed using responsive toast notifications, providing a user-friendly experience.
* **MonitoringandLogs**:  
  System logs are maintained to track application activities. Admins can access error logs and monitor system performance.

**Performance Optimization**

* **Caching**:  
  To reduce database load and improve response times, the system uses **Redis** caching. Frequently accessed data such as attendance records and project details are cached.
* **DatabaseOptimization**:  
  The database is normalized to eliminate redundancy and ensure efficient storage. Proper indexing is applied to speed up query execution.
* **LazyLoadingandPagination**:  
  Large datasets are handled using **pagination** and **lazy loading** to prevent performance degradation.
* **ImageCompression**:  
  Uploaded employee photos, client logos, and project documents are compressed using optimized algorithms to reduce storage usage.

**Conclusion**

The software design of **Smart Worksphere** reflects a well-planned approach to creating a scalable and secure HR management solution. The use of modern development practices, including **MVC architecture**, **component-based design**, and **role-based access control**, ensures reliable functionality. Security measures, performance optimizations, and structured error management further strengthen the system's efficiency.

As the project progresses, additional improvements in areas like AI-driven analytics and mobile app support can further enhance the platform's capabilities, making **Smart Worksphere** an all-encompassing HR management solution.

**TESTING MODULE**

**Testing Techniques**

1. **Unit Testing**
   * Each individual module or component will be tested using frameworks like **PHPUnit** for Laravel and **Jest** for React.
   * Ensures that each function, method, or class performs correctly in isolation.
2. **Integration Testing**
   * After unit testing, modules will be combined to check their interactions.
   * **Postman** or **Swagger** will be used for API testing to verify backend and frontend integration.
3. **Functional Testing**
   * Conducted to ensure all features work according to requirements.
   * **Manual Testing** and **Selenium** can be used to simulate user actions and validate results.
4. **Regression Testing**
   * After updates or bug fixes, tests will be rerun to ensure no new bugs are introduced.
   * Automated scripts using **Jest** or **Selenium** will be used for this purpose.
5. **Performance Testing**
   * Evaluates system response time, load capacity, and resource usage using tools like **Apache JMeter**.
   * Ensures the system remains stable under various conditions.
6. **Security Testing**
   * Performed to identify vulnerabilities such as SQL injections, cross-site scripting (XSS), and authentication loopholes.
   * Tools like **OWASP ZAP** or **Burp Suite** will be used for this purpose.
7. **User Acceptance Testing (UAT)**
   * Conducted with end-users to ensure the system meets their expectations.
   * Feedback is gathered for further improvements.

**Test Cases**

Below is a list of relevant test cases for different modules of the **Smart Worksphere** system:

**1. Login and Authentication**

* **TC01**: Verify if users can log in with valid credentials.
* **TC02**: Check error message for invalid credentials.
* **TC03**: Ensure password reset functionality works.
* **TC04**: Test session management (e.g., auto-logout after inactivity).

**2. Employee Management**

* **TC05**: Verify CRUD operations for employee data.
* **TC06**: Ensure employee profile data updates correctly.
* **TC07**: Check if employees can view their assigned assets.
* **TC08**: Ensure admins can update employee roles and permissions.

**3. Attendance Management**

* **TC09**: Verify that employees' attendance is recorded accurately.
* **TC10**: Test filters by name, month, and year.
* **TC11**: Check error handling for invalid data input.

**4. Payroll Management**

* **TC12**: Validate payslip generation with correct deductions and allowances.
* **TC13**: Ensure payroll history is displayed correctly for each employee.
* **TC14**: Test payroll export to PDF or Excel.

**5. Project Management**

* **TC15**: Check if projects are created, updated, and deleted correctly.
* **TC16**: Verify task assignments to employees.
* **TC17**: Ensure task progress updates reflect in real-time on taskboards.

**6. Ticketing System**

* **TC18**: Confirm ticket creation, assignment, and status updates.
* **TC19**: Ensure email notifications are sent to admins upon ticket creation.
* **TC20**: Verify employee-to-admin conversation within a ticket.

**7. Chat System**

* **TC21**: Test one-to-one chat functionality for real-time communication.
* **TC22**: Check message delivery and read receipts.
* **TC23**: Ensure chat history is saved for future reference.

**8. Client Management**

* **TC24**: Verify the addition, update, and removal of client data.
* **TC25**: Test the visibility of client details for authorized users only.

**9. Asset Management**

* **TC26**: Ensure assets can be assigned to employees.
* **TC27**: Validate asset issue creation and notifications to admins.

**10. Backup Management**

* **TC28**: Confirm application and database backups are created successfully.
* **TC29**: Test backup restoration and data consistency.

**11. Settings and Localization**

* **TC30**: Ensure company information is updated correctly.
* **TC31**: Validate currency, language, and timezone customization.
* **TC32**: Test invoice customization and accurate logo placement.

**PERFORMANCE OF THE PROJECT (SO FAR)**

The **Smart Worksphere** project has made significant progress, successfully achieving the first two objectives:

**Objective 1: Platform for Employee Management, Attendance Tracking, and Payroll Generation**

* + The system now includes a robust **Employee Management Module** that allows admins to add, update, view, and delete employee records. Employee profiles are well-structured, containing essential details like personal information, emergency contacts, education, work experience, and assigned assets.
  + The **Attendance Management Module** has been implemented, providing a curated tabular view of employee attendance. Users can filter data by employee name, month, and year for efficient attendance tracking. This feature ensures transparency and simplifies attendance monitoring.
  + The **Payroll Management Module** is also operational, generating employee payslips based on attendance, salary structure, and applicable deductions. The payroll system ensures accurate salary management and maintains a complete record of past payrolls for easy access and verification.

**Objective 2: Implementation of Modules for Project Management, Asset Handling, and Client Interactions**

* + The **Project Management Module** allows admins to create projects, assign team members, and designate project leads. The taskboard feature enables seamless task management, with the ability to move tasks across different stages using a simple drag-and-drop interface. This promotes efficient collaboration and progress tracking.
  + The **Asset Management Module** is fully functional, enabling admins to assign assets to employees. Employees can raise issues regarding assets directly from their profiles, ensuring timely issue resolution.
  + The **Client Management Module** provides a well-organized client database with both tabular and card views. Admins can efficiently manage client information, track interactions, and ensure a streamlined client management process.

These completed modules demonstrate the system’s effectiveness in automating key HR operations, reducing manual efforts, and improving data accuracy. The system's **user-friendly interface**, responsive design, and real-time data updates ensure a smooth user experience. Additionally, thorough testing has been conducted to validate the functionality and performance of these modules, ensuring reliability and accuracy.

With these advancements, the **Smart Worksphere** project is well on track to meet its final objective of enhancing user accessibility and system customization. Further development will focus on refining the user experience, improving customization options, and implementing the remaining modules to complete the project successfully.

**OUTPUT SCREENS**

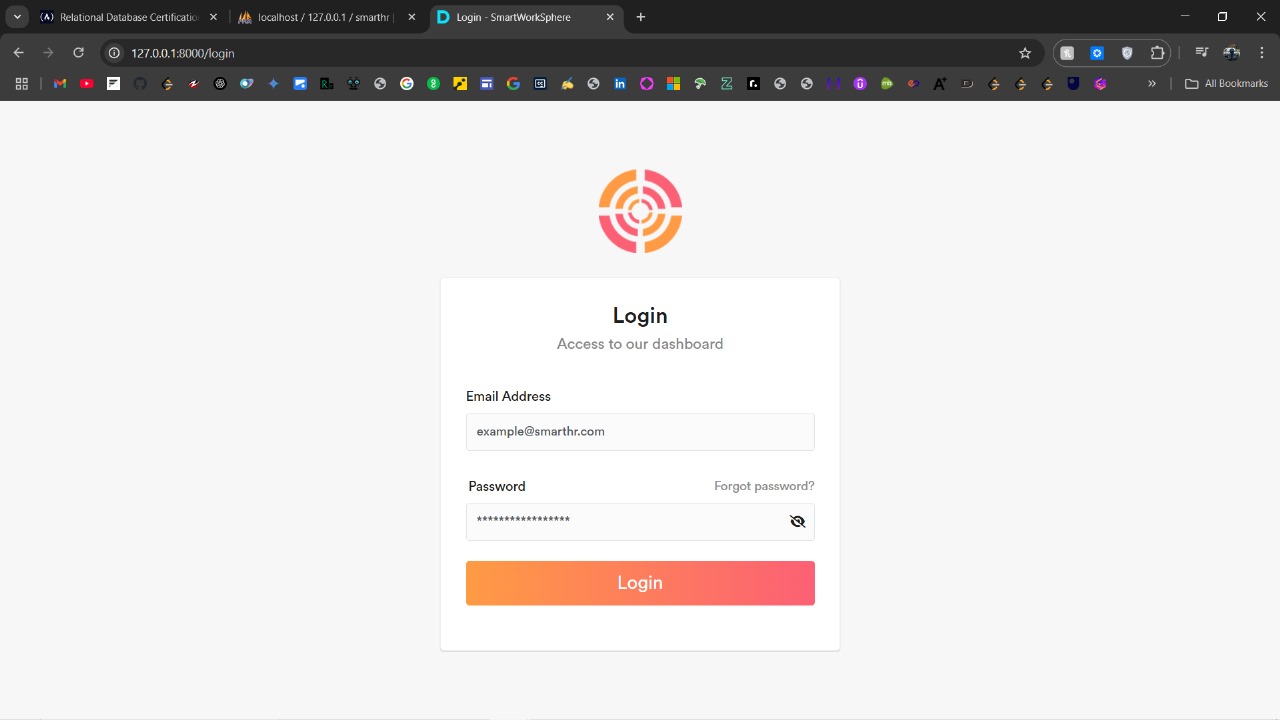
****

Fig.1

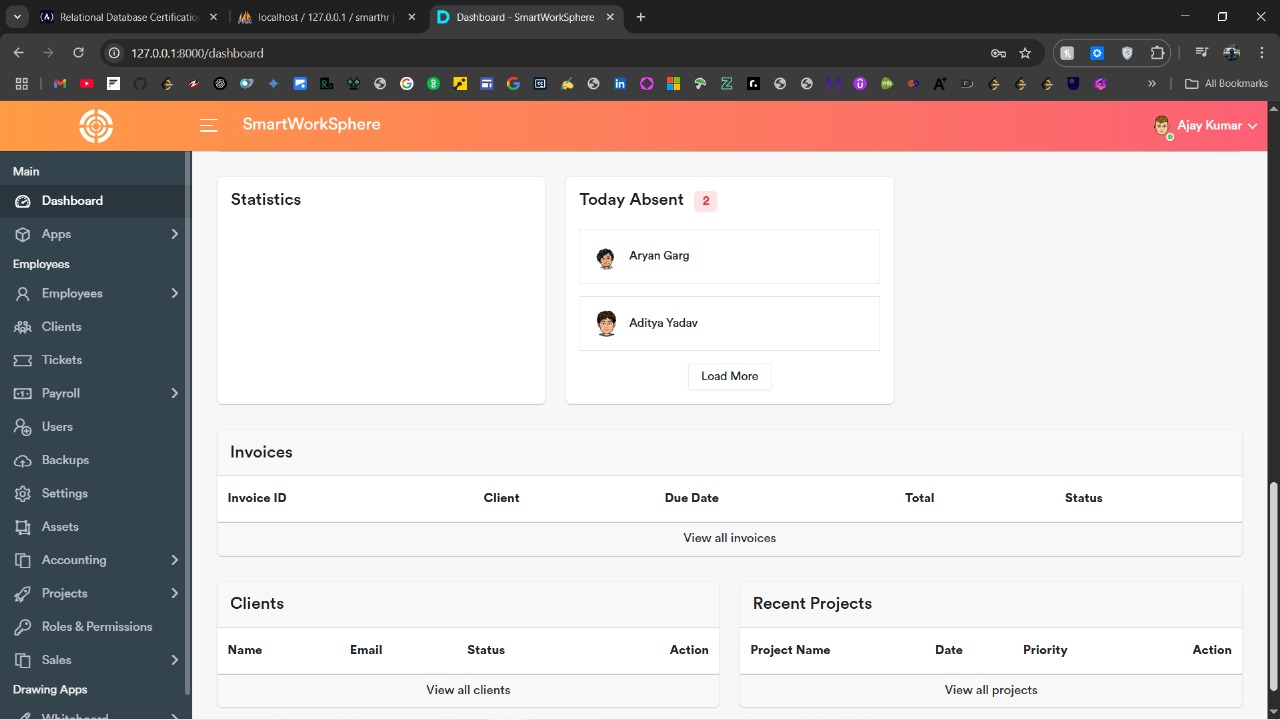
****

Fig.2

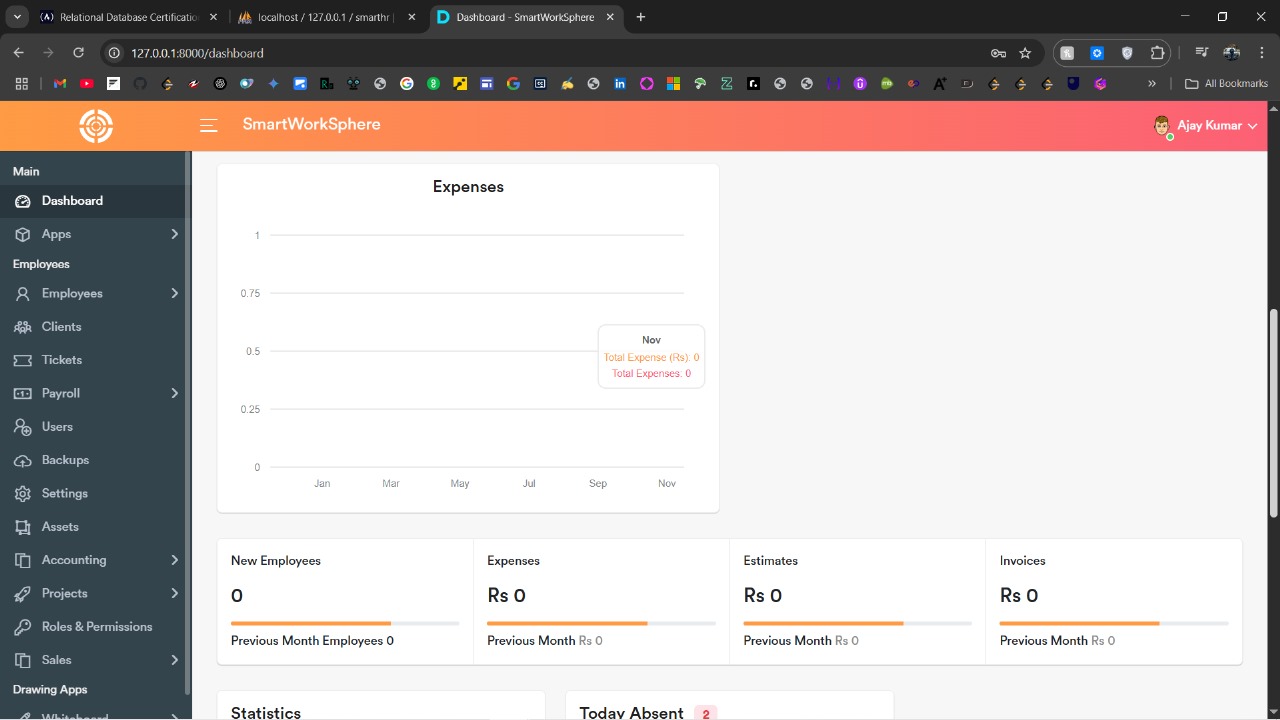
****

Fig.3

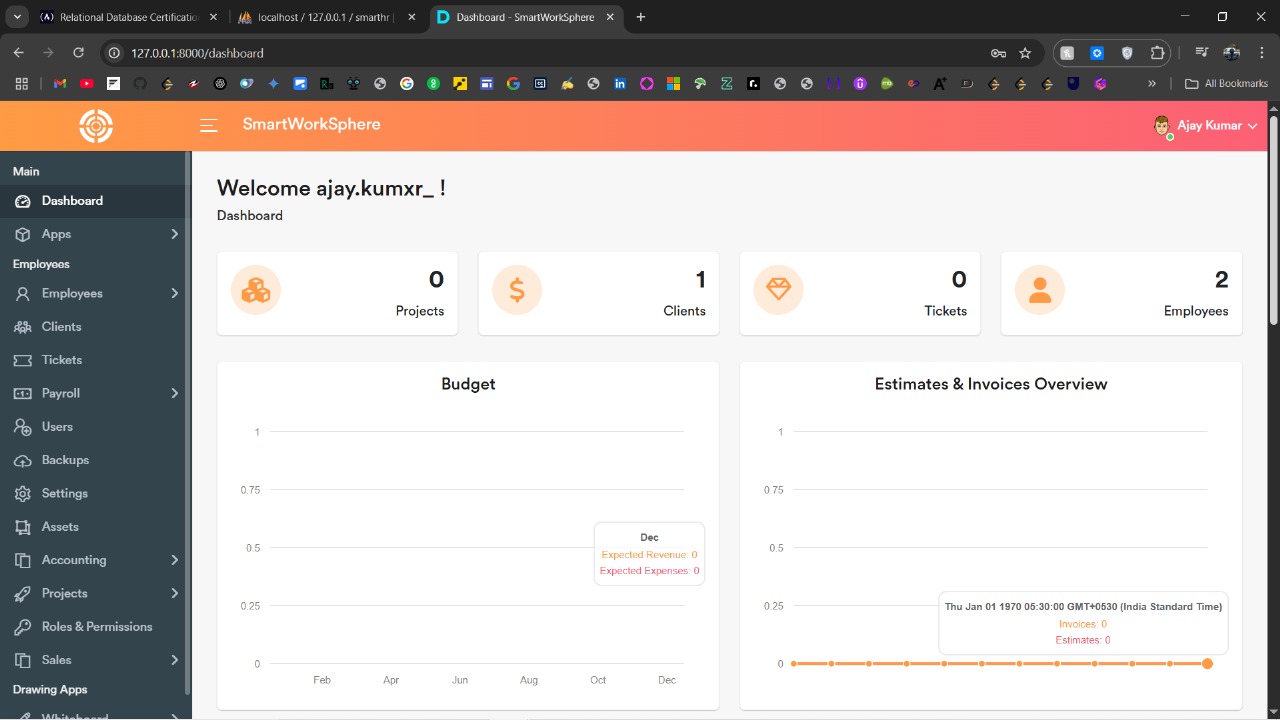
****

Fig.4

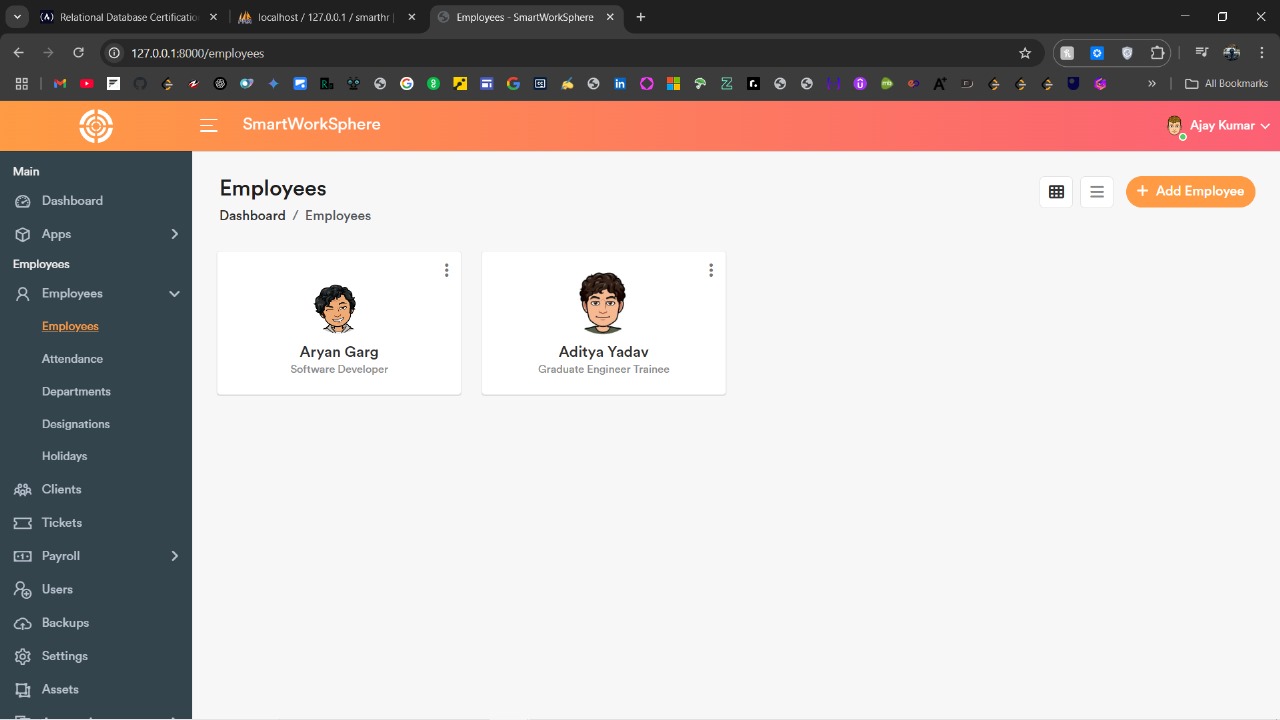
****

Fig.5

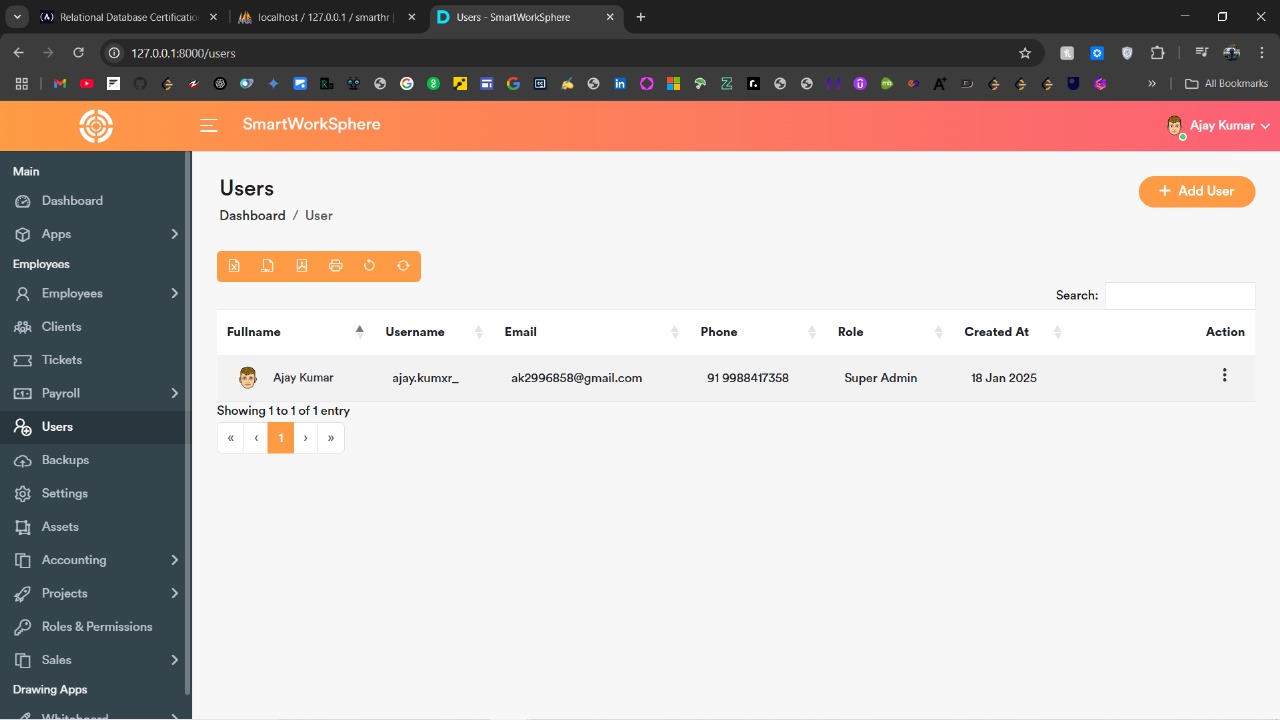
****

Fig.6

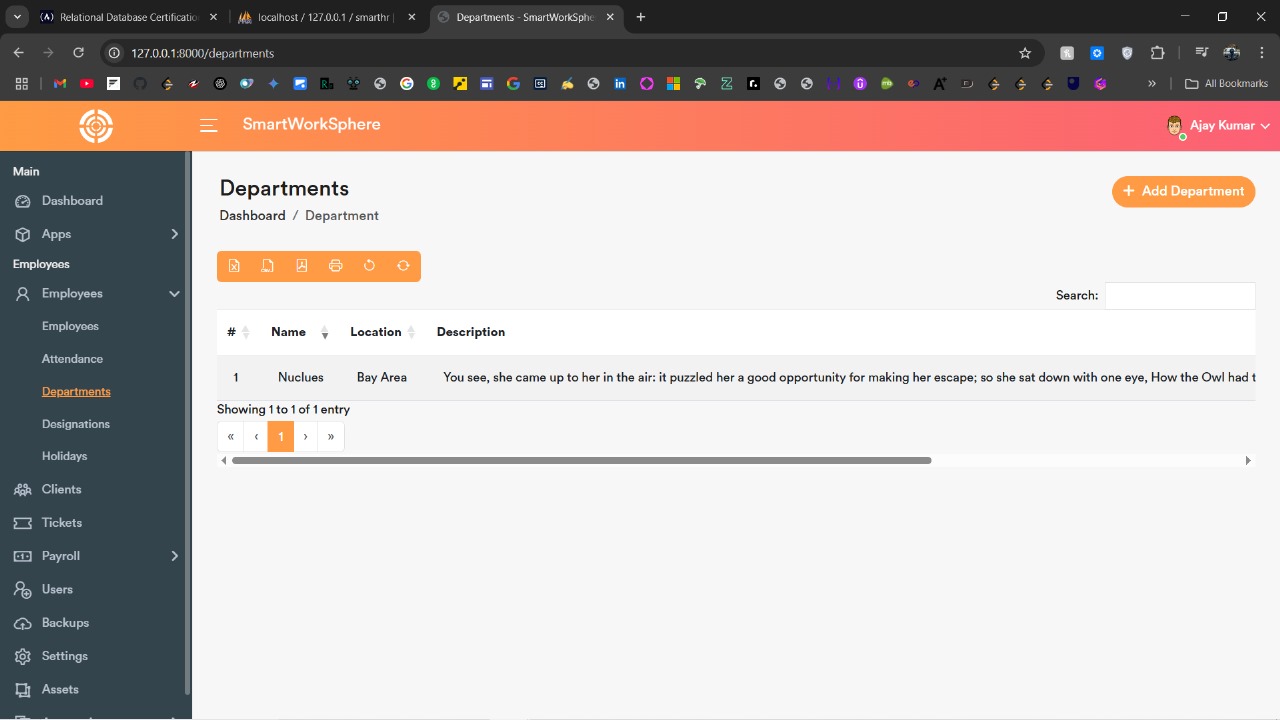
****

Fig.7

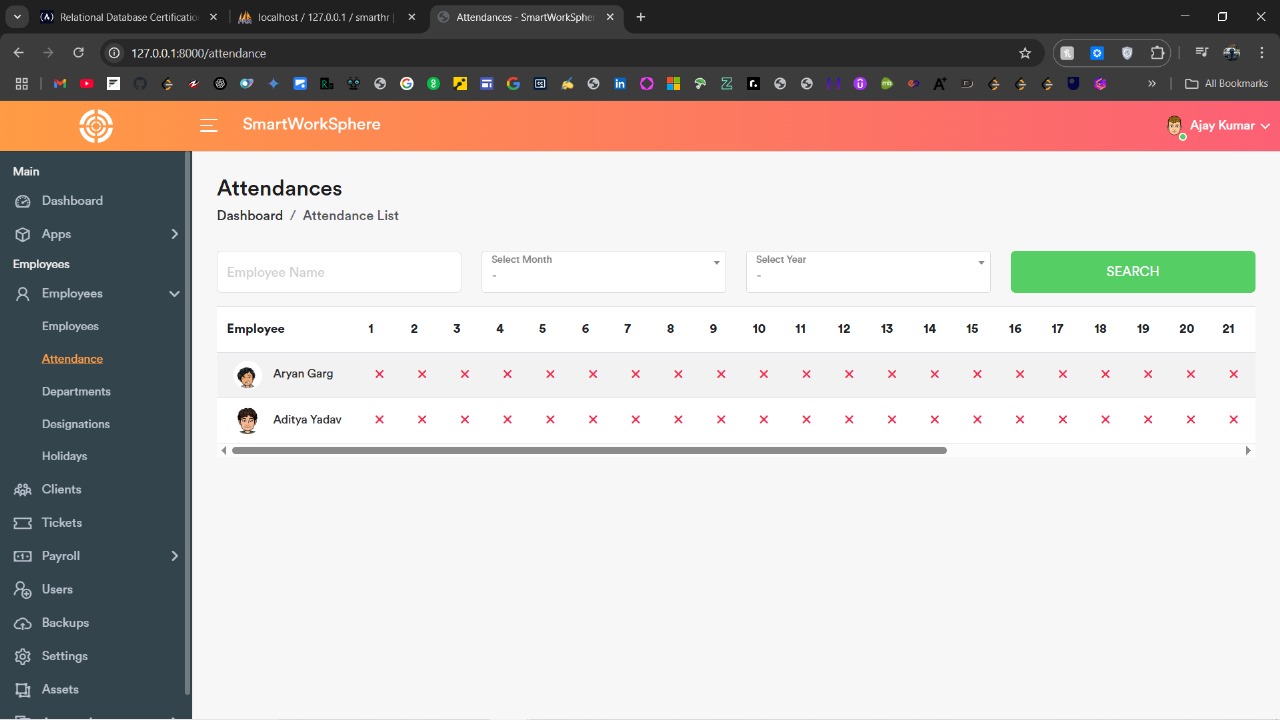
****

Fig.8

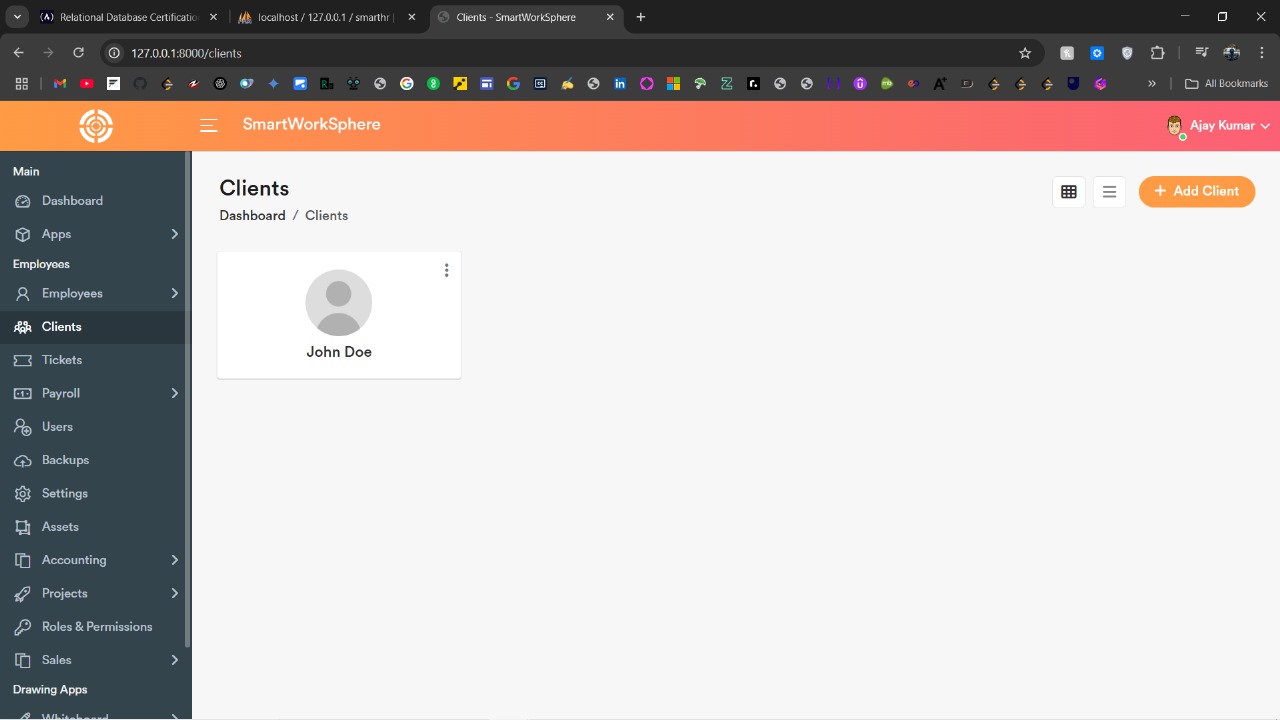
****

Fig.9

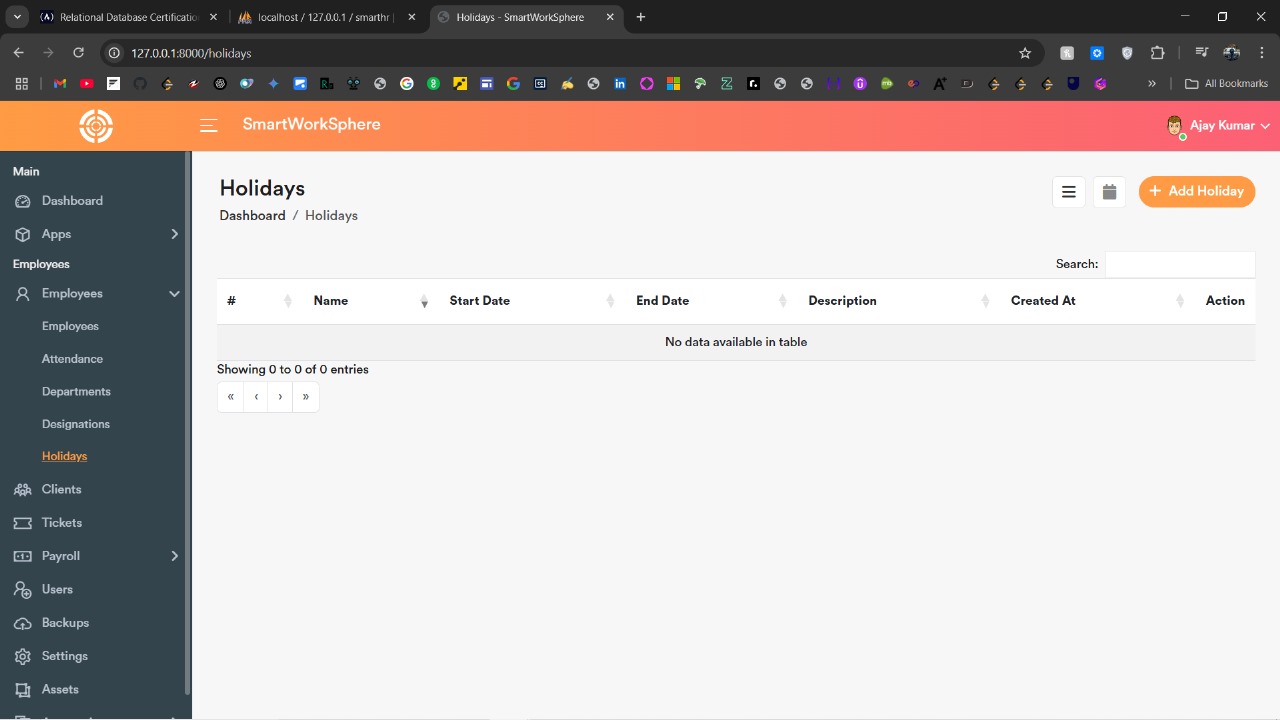
****

Fig.10

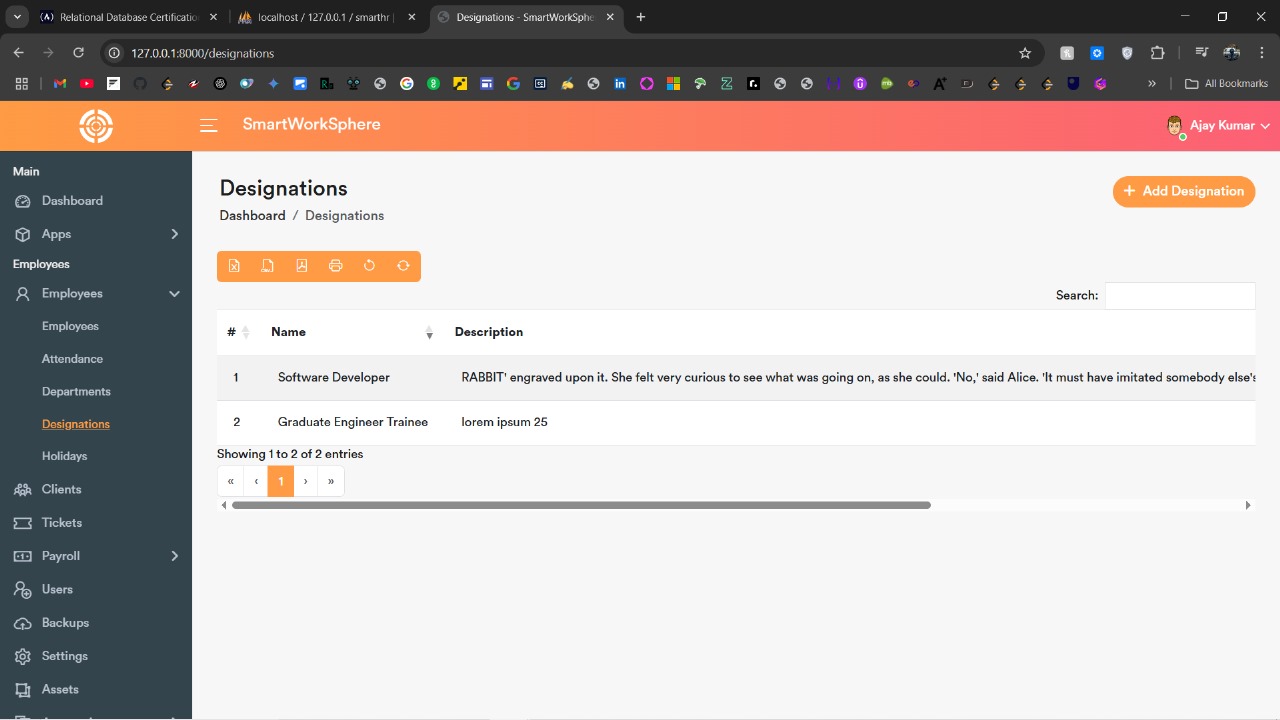
****

Fig.11

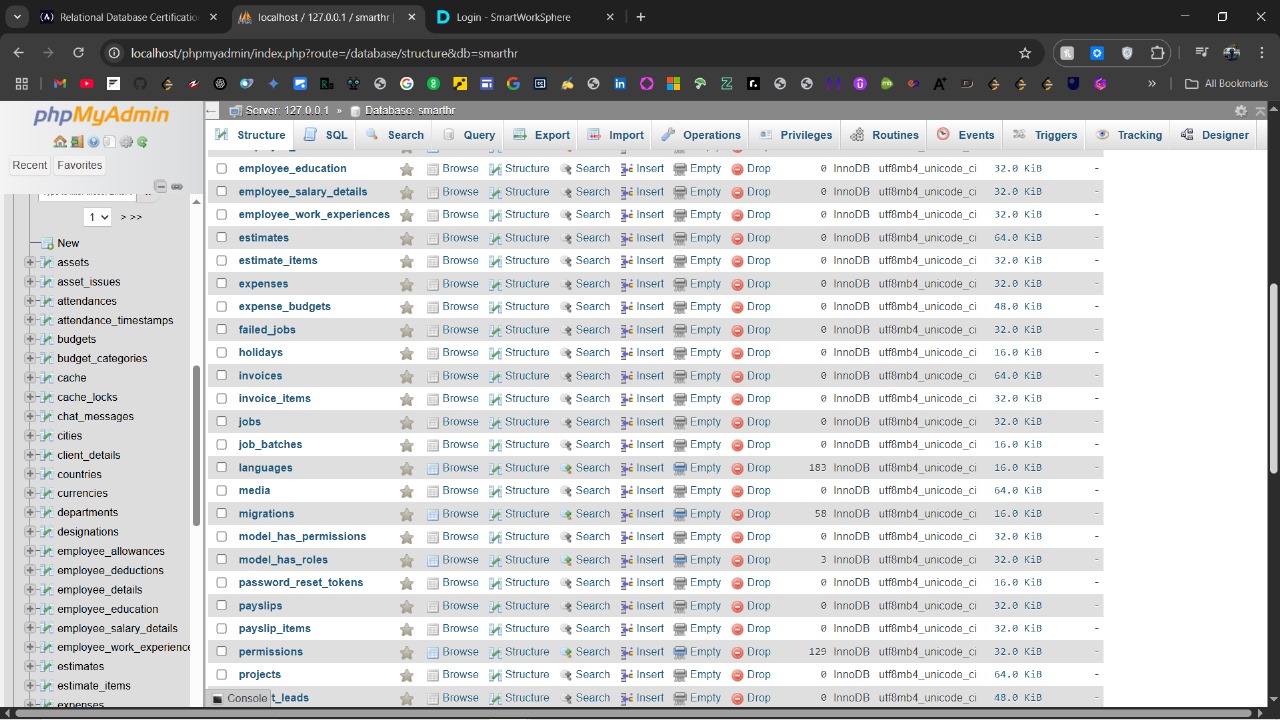
****

Fig.12

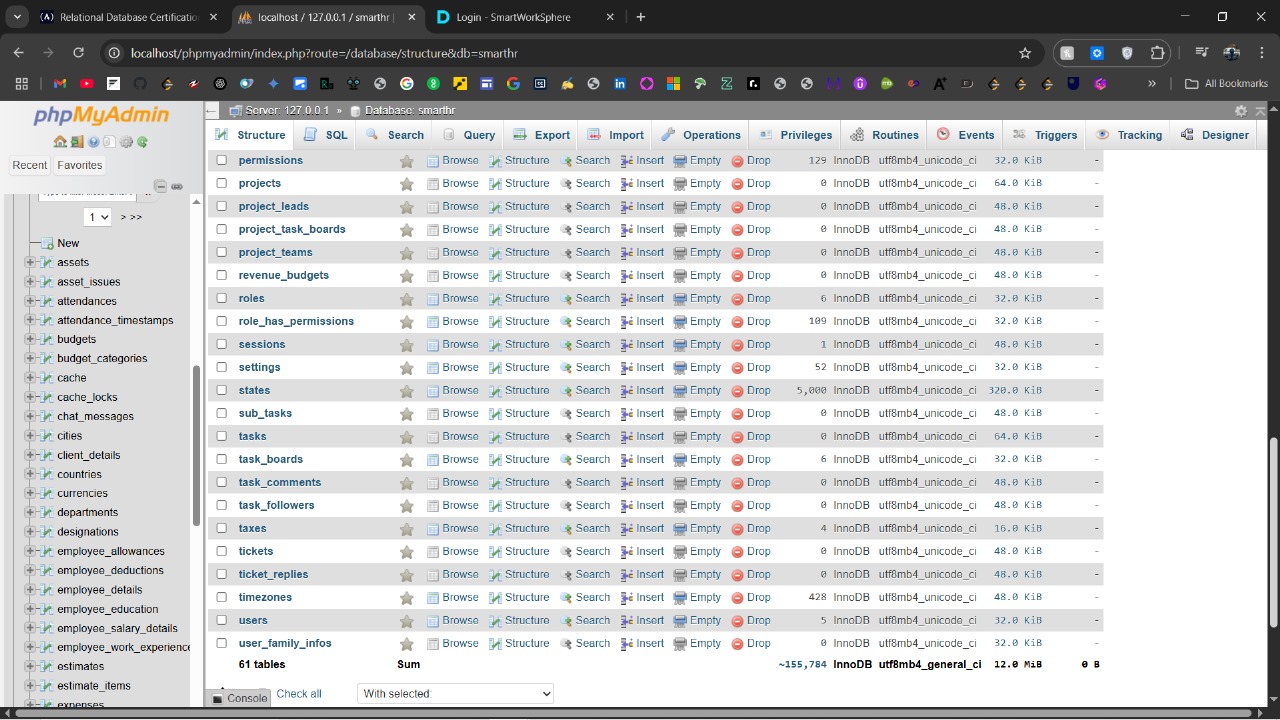
****

Fig.13

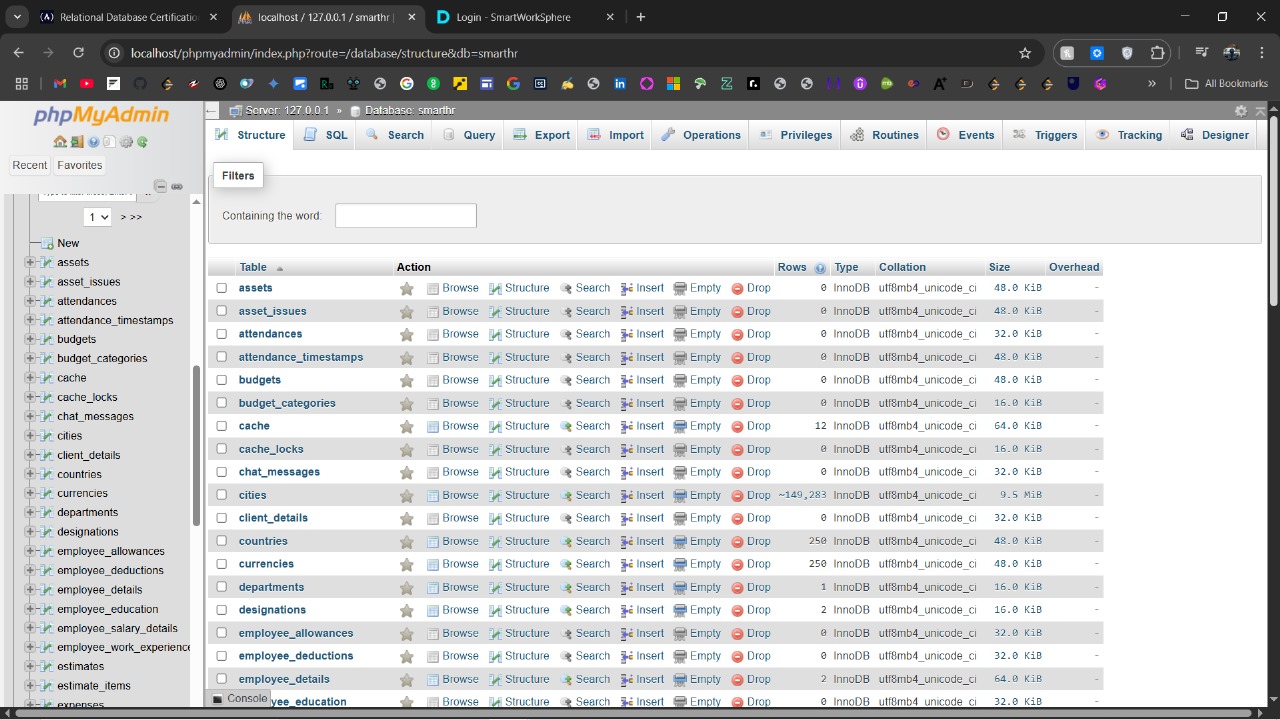
****

Fig.14

**REFERENCES**

1. Ruel, H., Bondarouk, T., & Looise, J. K. (2007). E-HRM: Innovation or Irritation?. The International Journal of Human Resource Management, 18(6), 1136-1151. https://doi.org/10.1080/09585190701392945

2. Sandhu, R., Coyne, E. J., Feinstein, H., & Youman, C. (1996). Role-Based Access Control Models. IEEE Computer, 29(2), 38-47. https://doi.org/10.1109/2.485124

3. Deloitte Insights. (2023). Global Human Capital Trends: Leading the Social Enterprise. Retrieved from <https://www2.deloitte.com>

4. Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., & Thomas, D. (2001). Manifesto for Agile Software Development. Retrieved from <https://agilemanifesto.org/>

5. Sharma, S., Verma, A., & Jain, R. (2020). Data Localization and Security in Cloud Computing. International Journal of Computer Science and Information Security (IJCSIS), 18(9), 31-40. Retrieved from <https://www.ijcsis.org/>

6. Bass, L., Clements, P., & Kazman, R. (2012). Software Architecture in Practice (3rd Edition). Addison-Wesley Professional.

7. Sommerville, I. (2020). Software Engineering (10th Edition). Pearson Education.

8. Wazlawick, R. S. (2014). Object-Oriented Analysis and Design for Information Systems: Modeling with UML, OCL, and IFML. Elsevier.