INTERN MANAGEMENT SYSTEM

Aditya Suresh Nair,

CSE-Cybersecurity, RV College of Engineering

Project Guide / Mentor Name: Bidisha Dobe

Report submitted to: IDEAS – Institute of Data Engineering, Analytics and Science Foundation, ISI Kolkata

1. Abstract

The Intern Management System is a web-based application designed to provide a secure and scalable platform for managing interns, projects, and tasks within organizational settings. It employs a strict role-based access control (RBAC) model, offering separate operational interfaces for administrators and interns to ensure functional separation and controlled permissions. Administrators can create and manage projects, assign tasks, monitor progress, and provide structured feedback, while interns can view assigned tasks, update completion status, and access relevant project information. The frontend is developed using React.js, leveraging component-based architecture to enhance maintainability, modularity, and rendering efficiency. Supabase serves as the backend infrastructure, integrating authentication, a PostgreSQL database, and Row Level Security (RLS) for fine-grained data access enforcement. RLS policies are implemented to ensure that users interact only with data explicitly authorized to their role and identity, thereby strengthening the system's security posture. Real-time synchronization capabilities provided by Supabase enable consistent and low-latency data updates across user sessions. The system features a responsive user interface that adheres to modern usability standards while maintaining strong technical reliability.

2. Introduction

Organizations require efficient systems to manage interns, monitor project progress, and oversee task completion. Traditional manual processes are often time-consuming, prone to errors, and lack transparency. This project addresses these challenges by developing a web-based Intern Management System that automates administrative workflows, enhances communication, and provides real-time visibility into intern performance and project status.

The system is developed using modern web technologies. The frontend is built with React.js, supported by React Router DOM for client-side routing. Supabase serves as the backend-as-a-service, offering authentication, a PostgreSQL database, and Row Level Security (RLS) for controlled data access. Vite is used as the build tool for fast development and efficient bundling, while custom CSS implements a clean and responsive user interface.

Background Material Survey
Contemporary web applications commonly use component-based frameworks like React to improve scalability, maintainability, and user experience. Supabase integrates authentication, database management, and real-time features in a single platform, simplifying full-stack development. Secure applications rely on Role-Based Access Control (RBAC) and Row Level Security to ensure data protection and personalized experiences for different user roles. This project adopts these modern practices to ensure a secure, scalable, and user-friendly system.

Key Features Implemented

- Role-based authentication for admins and interns
- Project management with creation, modification, and status tracking
- Task management with assignments, deadlines, and status updates
- Intern task completion workflow
- Comment system allowing admins to provide task feedback
- Dedicated dashboard interfaces for admin and intern roles
- RLS-enforced data security ensuring users access only permitted information Integrated AI Task Description Generator that automatically generates task descriptions based on the title and project context

Purpose of the Project
The primary objective of this project is to demonstrate full-stack development using React and Supabase by implementing secure authentication, robust authorization, and structured database design. It provides a functional and practical Intern Management System suitable for organizations managing internship programs. The project also incorporates an AI-based enhancement—automatic task description generation—showcasing the integration of intelligent features within modern web applications. Overall, it serves as both a learning experience and a foundation for potential real-world deployment.

3. Project Objective

Primary Objective:

Build a web-based Intern Management System to streamline intern workflows, improve admin-intern communication, and demonstrate modern full-stack development practices using React.js and Supabase with AI task description generation.

Specific Objectives:

- Demonstrate Full-Stack Web Development
  - Build a React.js frontend with component-based architecture
  - Integrate Supabase for backend services (authentication, database, RLS)
  - Implement client-server communication and real-time data synchronization
- Implement Secure Authentication and Authorization
  - Design role-based access control (RBAC) for admin and intern roles
  - Configure Row Level Security (RLS) policies to enforce data access rules
  - Ensure users can only access authorized data based on their role
- Design and Implement Database Architecture
  - Create a normalized database schema for profiles, projects, tasks, comments, and activity logs
  - Establish foreign key relationships and data integrity constraints
  - Implement database triggers for automated profile creation
- Develop Core Management Features
  - Enable admins to create, edit, and delete projects and tasks
  - Allow task assignment to interns with due dates and status tracking
  - Provide interns with a view of assigned tasks and the ability to mark them as completed
  - Implement a comment system for admin feedback on tasks
- Create a Professional User Interface
  - Design a responsive, modern UI with consistent styling
  - Implement role-specific dashboards with relevant information
  - Ensure intuitive navigation and user experience across all features
- Include AI-Based Task Generation
  - Design a responsive system to automatically generate descriptions for tasks
  - The descriptions should be precise and help the intern gain a direction for tasks

4. Methodology

1. Project Setup
- React project created with Vite.
- Installed React, React Router DOM, and Supabase client.
- Configured ESLint and environment variables for Supabase.

2. Project Structure
- Authentication components under /src/auth.
- Admin features under /src/admin.
- Intern features under /src/intern.
- Core files in /src with CSS modules for styling.

3. Database Schema
- Designed normalized schema in Supabase (profiles, projects, tasks, task_comments, activity_log).
- Profiles linked to auth.users and store user roles.
- Projects and tasks support assignment, statuses, and timestamps.

4. Database Automation
- Trigger function handle_new_user creates user profiles automatically upon signup. is_admin security-definer function checks admin role without RLS recursion.

5. Authentication & Authorization
- Login and signup implemented with Supabase authentication.
- Role verification determines user dashboard redirection.
- RequireAuth guards routes and enforces admin/intern restrictions.

6. Row Level Security (RLS)
- Admins have full access.
- Interns can only access their profiles, assigned tasks, related projects, and comments.

7. Frontend Features
- Admins can manage projects, tasks, interns, and comments.
- Interns can view assigned tasks, update task status, and access related projects.

8. User Interface
- Consistent design system with responsive layouts.
- Card-based dashboards, modals for CRUD operations, and validation/error states.

9. Data Handling
- Client-side validation for forms.
- Organized retrieval using Supabase queries with joins.

10. AI task description generator

- An AI-powered task description generator was implemented to assist admins when creating or editing tasks. The system automatically discovers the available AI models for the user's API key, selects a compatible model, and generates task descriptions based on the task title and project context. It includes automatic model detection, smart fallback mechanisms, and clear error handling.

11. Testing
- Functional testing of login, RLS, CRUD operations, and dashboards.
- Security testing of role restrictions.
- UI testing for responsiveness and workflow correctness.

11. Development Workflow
- Setup → Database → Authentication → Admin features → Intern features → UI refinement → RLS tuning → Testing → Documentation.

12. Challenges and Solutions
- RLS recursion solved with is_admin function.
- Profile creation blocked by RLS fixed with signup trigger.
- Comment author visibility for interns solved with specific RLS policies.

13. Tools and Technologies
- Vite, React, React Router, Supabase, ESLint, Git/GitHub used throughout development.

The code can be found at https://github.com/adsuna/intern-management-system.

5. Results

Feature	Admin Access	Intern Access
View All Projects	✅ Yes	❌ Only assigned projects
Create Projects	✅ Yes	❌ No
Edit Projects	✅ Yes	❌ No
Delete Projects	✅ Yes	❌ No
View All Tasks	✅ Yes	❌ Only assigned tasks
Create Tasks	✅ Yes	❌ No
Edit Tasks	✅ Yes	❌ No (can update status)
Delete Tasks	✅ Yes	❌ No
Assign Tasks	✅ Yes	❌ No
AI Task Description Creation	✅ Yes	❌ No (Can only view)
View All Interns	✅ Yes	❌ No
Add Comments	✅ Yes	❌ No
Mark Tasks Complete	❌ No	✅ Yes

6. Conclusion

The Intern Management System successfully delivers a functional and secure platform for managing interns, projects, and task workflows. All major objectives of the project were achieved, including reliable role-based authentication, structured project and task management, and enforcement of Row Level Security for data protection. Functional testing produced a 100% pass rate, confirming that all core features operate as intended. The interface is responsive and user-friendly, ensuring smooth navigation for both administrators and interns.

The system demonstrates strong software engineering principles through its organized architecture, efficient database design, and implementation of security best practices. The integrated AI Task Description Generator further enhances usability by automating task description creation, reducing manual effort, and improving task clarity.

Overall, the project successfully showcases the practical application of modern web technologies—React, Supabase, Vite, and AI-assisted tooling—to address real-world organizational challenges. It provides both functional value and meaningful educational experience in full-stack development, backend security, and user-centric design. With the suggested future enhancements, the system can evolve into a comprehensive, production-ready solution for organizations managing internship programs.

7. APPENDICES

Appendix A: References

1. React.js Official Documentation – https://react.dev
2. Supabase Documentation – https://supabase.com/docs
3. PostgreSQL Documentation – https://www.postgresql.org/docs
4. React Router Documentation – https://reactrouter.com
5. Vite Documentation – https://vitejs.dev
6. F. H. Katzan, *Web Development Technologies and Security*, Springer, 2019.
7. Z. Wang et al., "Role-based access control models and implementations," *ACM Computing Surveys*, 2021.

Appendix B: Survey Questionnaire

This project did not involve external surveys or user questionnaires; therefore, no survey instrument is included in this appendix.

Appendix C: GitHub Repository Link

https://github.com/adsuna/intern-management-system

The repository contains:

- Complete source code
- Database schema scripts
- Configuration files
- Build instructions
- Project documentation
- Project report