

PROJECT MANAGEMENT SYSTEM

Kishan Das

Abdulrahman Imam

Shubhendu Sekhar Sahoo

Assistant Professor,

Department of Computer Science and Engineering
(IOT) GIFT Autonomous, Bhubaneswar,
Odisha, India

Department of Computer Science and Engineering
(AI) GIFT Autonomous, Bhubaneswar,
Odisha, India

Department of Computer Science and Engineering
(Artificial Intelligence) GIFT

ssahoo@gift.edu.in

kishand2023@gift.edu.in

abdul20222023@gift.edu.in

1. Introduction

A Project Management System is a software application designed to help organizations and teams plan, manage, monitor, and complete projects efficiently. It provides a centralized platform where project managers and team members can collaborate, assign tasks, track progress, manage deadlines, and generate reports in real time. In modern organizations, managing projects manually becomes difficult due to increasing project complexity, communication gaps, and large amounts of data. Traditional methods such as paperwork and spreadsheets are time-consuming and may lead to errors, delays, and inefficient resource management. To overcome these problems, a digital Project Management System is required. The proposed Project Management System is developed using modern technologies such as ReactJS for the front-end user interface, Django Framework for the back-end server, and PostgreSQL for database management. The system provides a secure, responsive, and scalable environment for managing projects and tasks effectively. The application allows administrators and project managers to create projects, assign tasks to employees, monitor project progress, manage deadlines, and generate reports. Team members can update task status and communicate project-related information through the system. The system improves coordination, transparency, and productivity within the organization. Project management consists of several important phases commonly known as the project life cycle. These phases include project initiation, planning, execution, monitoring and controlling, and project closure. During the initiation phase, the purpose, objectives, feasibility, and stakeholders of the project are identified. In the planning phase, project managers develop schedules, allocate resources. Project management has become an essential requirement across organizations and academic environments due to the increasing complexity of tasks, team coordination, and workflow management. Traditional methods such as manual record keeping, spreadsheets, and verbal communication are often inefficient because they make project tracking, task monitoring, and team collaboration difficult to manage effectively. With the rapid growth of web technologies and collaborative platforms, digital project management systems have become a practical solution for improving productivity and communication. However, many existing systems are either difficult to use, lack real-time task monitoring, or do not provide centralized management for projects and team activities.

Keywords: Project Management System, React.js, Vite, Django Framework, PostgreSQL, Task Management, Team Collaboration, Project Tracking, Role-Based Authentication, Web Application.

OBJECTIVES OF THE PROJECT

- i. To develop an efficient Project Management System for managing projects and tasks effectively.
- ii. To provide secure user authentication and role-based access control.
- iii. To create and manage projects with proper task allocation features.
- iv. To monitor project progress and track task completion status in real time.
- v. To maintain employee and team member details securely in the system.
- vi. To improve communication and collaboration among project team members.
- vii. To generate project reports and performance analysis automatically.
- viii. To maintain records of project activities, updates, and deadlines.
- ix. To provide an administrative dashboard for monitoring user activities and project performance.
- x. To build a scalable and user-friendly platform for future enhancements and integrations.

1. LITERATURE SURVEY

The Project Management System has become an important software solution for organizations to manage projects, resources, tasks, and team collaboration effectively. Various researchers and developers have proposed different techniques and methodologies to improve project planning, monitoring, and execution. This literature survey reviews existing studies and systems related to project management applications.

Traditional project management methods mainly relied on manual documentation, spreadsheets, and physical communication. These approaches were time-consuming, less secure, and difficult to maintain for large-scale projects. Researchers identified several limitations such as lack of real-time monitoring, inefficient communication, poor task tracking, and data redundancy.

Many modern project management systems introduced web-based platforms to improve accessibility and collaboration among team members. These systems allow users to create projects, assign tasks, set deadlines, and monitor progress from any location. Studies show that web-based systems significantly improve productivity and reduce

project delays.

Several researchers focused on task management and scheduling techniques. Techniques such as Gantt Charts, Critical Path Method (CPM), and Agile methodologies were integrated into project management applications to improve project planning and resource utilization. Agile-based.

Project management is a structured approach used to plan, organize, execute, monitor, and successfully complete projects within defined goals, time limits, and budgets. It plays a vital role in modern organizations by ensuring that resources are effectively utilized and objectives are achieved efficiently. A project can be defined as a temporary endeavor undertaken to create a unique product, service, or result. Since projects often involve multiple tasks, stakeholders, risks, and constraints, project management provides systematic methods and tools to handle complexity and improve the likelihood of success. The concept of project management has evolved significantly over time due to advancements in technology, globalization, and increasing business competition.

Organizations across industries such as construction, information technology, healthcare, manufacturing, education, and finance rely heavily on project management practices to meet strategic goals. Effective project management helps organizations reduce costs, improve quality, enhance customer satisfaction, and achieve sustainable growth. Project management consists of several important phases commonly known as the project life cycle. These phases include project initiation, planning, execution, monitoring and controlling, and project closure. During the initiation phase, the purpose, objectives, feasibility, and stakeholders of the project are identified. In the planning phase, project managers develop schedules, allocate resources, estimate budgets, identify risks, and define communication strategies. Proper planning is considered one of the most critical aspects of successful project management because it provides a roadmap for the entire project. The execution phase involves implementing the project plan by coordinating team members, managing resources, and performing project activities. Effective leadership, communication, and teamwork are essential during this phase to ensure that project tasks are completed efficiently. Monitoring and controlling occur simultaneously with execution,

where project managers track progress, measure performance, identify deviations, and take corrective actions when necessary. Tools such as Gantt charts, Critical Path Method (CPM), Program Evaluation and Review Technique (PERT), and project management software assist managers in maintaining control over project activities.

2. EXISTING SYSTEM

The existing project management system mainly depends on manual methods or basic software tools for managing projects and tasks. Many organizations still use spreadsheets, paperwork, emails, and traditional communication methods to handle project-related activities. These methods are time-consuming, difficult to maintain, and prone to human errors.

In the existing system, project managers manually assign tasks to employees and monitor project progress through meetings or reports. Data related to projects, deadlines, and employee performance is often stored in separate files, making it difficult to access and manage information efficiently. Lack of centralized data management creates communication gaps among team members and affects productivity.

Most traditional systems provide limited functionality for task tracking and real-time monitoring. Team members may face difficulties in updating project status, sharing files, and communicating with other members. As a result, project delays and improper resource management frequently occur.

Another limitation of the existing system is the lack of proper security and access control. Sensitive project information may not be protected effectively, leading to risks of unauthorized access and data loss. In many cases, report generation and performance analysis are performed manually, which increases workload and reduces efficiency.

3. PROPOSED SYSTEM

The proposed Project Management System is a web-based application designed to manage projects, tasks, employees, and project activities efficiently. The system aims to overcome the limitations of the existing manual and traditional project management methods by providing an automated, secure, and user-friendly platform. It automatically notifies whenever an emergency occurs.

The core functionality of the system is the **Safety**

Check feature. When the user performs a safety check, the application analyzes several parameters such as traffic density, crowd level, road condition, street-light availability, time of day, and crime history of the current area. Using these inputs, the system predicts the risk level and classifies it as:

- **Safe (Risk0)** – The surroundings are considered secure.
- **Moderate Risk (Risk1)** – The area requires caution.
- **High Risk (Risk2)** – Immediate attention is required.

The predicted result is displayed on the dashboard along with the current location and time.

The proposed system provides a centralized environment where administrators, project managers, and team members can access project-related information in real time. It allows users to create projects, assign tasks, set deadlines, monitor project progress, and generate reports efficiently. The system improves communication and coordination among team members, which helps in completing projects within the specified time.

The system includes secure user authentication and role-based access control to protect sensitive project information. Different users are provided with specific permissions according to their roles, such as administrator, manager, or employee. This improves system security and ensures proper management of project data.

The proposed system also provides task tracking and progress monitoring features. Team members can update task status regularly, while project managers can monitor project performance through dashboards and reports. Automated report generation reduces manual work and helps management in decision-making.

Key Features of the Proposed System

- i. **User Authentication and Authorization,**
- ii. **Project Creation and Management.**
- iii. **Task Assignment and Tracking.**
- iv. **Real-Time Project Monitoring.**
- v. **Dashboard Management**
- vi. **Team Collaboration.**
- vii. **Deadline and Schedule Management.**

viii. **ReportGeneration.**

4. SYSTEM REQUIREMENTS

4.1 Hardware Requirements

- Intel Core i3 Processor or above
- 4GB RAM or higher
- 500GB Hard Disk
- GPS-enabled smartphone
- Stable Internet Connection

5. SYSTEM ARCHITECTURE

The Project Management System follows a three-tier architecture consisting of the Presentation Layer, Application Layer, and Database Layer. This architecture helps in maintaining scalability, security, and efficient communication between users and the system.

1. Presentation Layer (User Interface)

The presentation layer acts as the front-end interface between the user and the system. It allows administrators, project managers, and employees to interact with the application through web pages or dashboards. Users can perform activities such as login, project creation, task assignment, progress tracking, and report viewing through this layer.

Main Functions:

- User Login and Registration
- Dashboard Display
- Project and Task Management
- Report Viewing
- User Interaction

2. Application Layer (Business Logic Layer)

The application layer processes user requests and implements the business logic of the system. It handles project management operations, task assignments, progress tracking, authentication, and report generation. This layer acts as a bridge between the presentation layer and the database layer.

Main Functions:

- Authentication and Authorization
- Project Processing
- Task Allocation
- Progress Monitoring
- Report Generation
- Notification Management

3. Database Layer

The database layer stores all project-related information securely. It maintains records of users,

projects, tasks, deadlines, reports, and employee activities. The database ensures data consistency, security, and efficient retrieval of information.

Main Functions:

- User Data Storage
- Project and Task Records
- Activity Logs
- Report Storage
- Data Backup and Recovery

Working of the System

1. The user logs into the system using valid credentials.
2. The request is processed by the application layer.
3. The system retrieves or stores data in the database layer.
4. The processed information is displayed to the user through the presentation layer.
5. Managers and administrators monitor project progress and generate reports when required.

Advantages of the Architecture

- Improved system security
- Easy maintenance and scalability
- Faster data processing
- Better user interaction
- Efficient project and task management

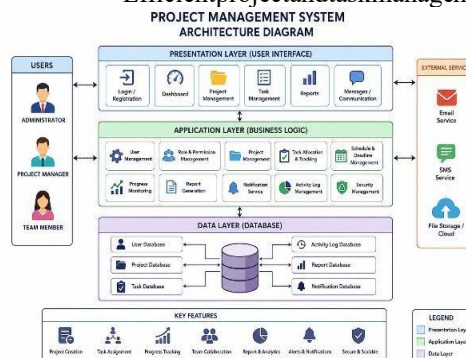


Fig1: System Architecture Diagram

6. DATA FLOW DIAGRAM

The Data Flow Diagram (DFD) of the Project Management System describes how data moves between users, processes, and databases within the system. It helps in understanding the overall working structure and interaction among different modules of the application. The DFD provides a graphical and logical representation of the system processes, data stores, and external entities involved in project management activities. In the Level 0 DFD, also known as

the Context Diagram, the entire Project Management System is represented as a single process. The main external entities interacting with the system are the Administrator, Project Manager, Team Member, Notification Services, and Report Dashboard. The Administrator manages users, projects, and system settings. The Project Manager creates projects, assigns tasks, and monitors progress, while team members update task status and upload project-related information. The system processes these inputs and generates reports, notifications, and project updates. The Level 1 DFD divides the system into major functional modules such as User Management, Project Management, Task Management, Progress Monitoring, and Report Generation. The User Management module handles user registration, login, authentication, and role management. The Project Management module manages project creation, deadlines, milestones, and project updates. The Task Management module is responsible for task assignment, task tracking, and status updates. The Progress Monitoring module tracks project progress and employee performance, while the Report Generation module creates project reports, activity summaries, and performance analysis reports.

Each process interacts with its respective database for storing and retrieving information. The User Database stores user details and login credentials. The Project Database maintains project-related information such as project titles, deadlines, and milestones. The Task Database stores task details, task assignments, and status updates. The Progress Database maintains records of project progress and work logs, while the Report Database stores generated reports and analytics data.

In the Level 2 DFD, the Task Management module is further divided into sub-processes such as View Tasks, Assign Tasks, Update Task Status, Upload Documents, and Add Comments or Feedback. The Project Manager assigns tasks to team members, and team members update their task status regularly. All task-related information is stored in the Task Database and can be accessed for monitoring and report generation purposes.

The Data Flow Diagram helps developers and users understand the system functionality clearly. It simplifies system analysis, improves communication between stakeholders, and

provides a structured approach for designing and maintaining the Project Management System efficiently.

The SOS Alert Module generates an emergency message containing the user's name, current time, risk level, and live Google Maps location link. This message is sent to all registered emergency contacts through WhatsApp. The alert details are also recorded in the database for future reference.

The Emergency Contacts receive the alert and can use the live location link to track the user's exact position and provide immediate assistance.

The Admin Module continuously interacts with the database to monitor registered users, safety checks, and alert activities.

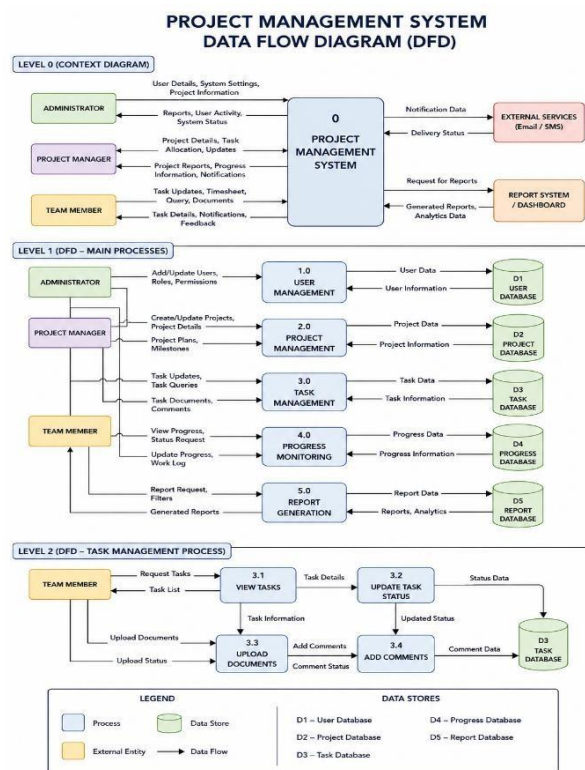


Fig2:DataFlowDiagram

7. DATABASE DESIGN

The database design of the Project Management System is an important part of the system architecture that helps in storing, managing, and retrieving project-related information efficiently. The database is designed using a relational model where data is organized into multiple tables connected through relationships. This design helps maintain data integrity, security, and efficient data

handling within the system.

The system database contains several tables such as User Table, Project Table, Task Table, Team Member Table, ProgressMonitoringTable, ReportTable, and Notification Table. Each table stores specific information related to the functioning of the Project Management System.

The User Table stores details of administrators, project managers, and team members. It contains fields such as User ID, Name, Email, Password, Role, Contact Number, and Login Status. This table is mainly used for user authentication and role-based access control within the system.

The Project Table stores all project-related information including Project ID, Project Name, Description, Start Date, End Date, Project Status, and Created By.

This table helps manage multiple projects and their details systematically.

The Task Table maintains records of tasks assigned to employees or team members. It includes fields such as Task ID, Project ID, Task Name, Task Description, Assigned To, Priority, Deadline, and Task Status. This table is used for task allocation and progress tracking.

The Team Member Table contains information about employees working on different projects. It includes Member ID, Name, Role, Contact Details, and Assigned Projects. This table helps manage team participation and employee responsibilities.

The Progress Monitoring Table stores work updates and progress details of ongoing projects. It contains fields such as Progress ID, Project ID, Task ID, Progress Status, Work Update, and Updated Date. This table helps project managers monitor project development and employee performance.

The Report Table stores generated reports and analytics data related to projects and tasks. It contains Report ID, Project ID, Report Type, Generated Date, and Report Status. This table helps administrators and managers generate performance and activity reports.

The Notification Table stores notifications and alerts generated by the system. It includes Notification ID, User ID, Message, Notification Date, and Status. This table is useful for sending reminders, updates, and deadline notifications to users.

The tables in the database are connected through primary keys and foreign keys. One user can manage multiple projects, one project can contain multiple tasks, and one team member can work on multiple tasks. These relationships help

maintain proper data organization and improve system efficiency.

The database design reduces data redundancy, improves security, supports fast data retrieval, and ensures smooth operation of the Project Management System. It also provides flexibility for future enhancements and additional features.

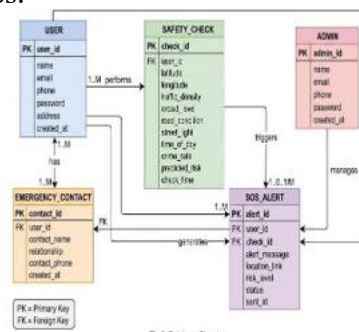


Fig 3: Database Structure

Fig3:DatabaseStructure

24. MODULE DESCRIPTION

24.1 UserRegistrationModule

This module allows users to create an account in the application by entering personal details such as name, mobile number, email address, and password. During registration, users can also add trusted emergency contacts who will receive alerts during emergencies.

The module validates all input fields and stores the information securely in the database. It ensures that each user is registered successfully and can access the application features after authentication.

24.2 SafetyCheckModule

The Safety Check Module is the core component of the system. It collects the user's current location and analyzes important environmental parameters such as traffic density, crowd level, road condition, street-light availability, time of day, and local crime history.

Based on these inputs, the module predicts the current risk level and classifies the situation as Safe, Moderate Risk, or High Risk.

24.3 SOSAlertModule

This module is responsible for sending emergency alerts when the user manually activates the SOS button or when the system detects a High Risk situation.

Front-End Implementation
he Django server handles:

- Business logic

• API management

- Authentication
- Data processing
- Communication with the PostgreSQL database

Features of Django Back-End

1. REST API Development

Django REST Framework (DRF) is used to create APIs for communication between React JS and the server.

2. User Authentication

The system implements:

- Secure login authentication
- JWT token authentication
- Role-based authorization
- Session handling

3. Project and Task Processing

The Django server processes:

- Project creation requests
- Task assignments
- Task updates
- Report generation

4. Data Validation

All user inputs are validated before storing data in the database to maintain data integrity.

The front-end of the Project Management System is developed using **React JS**. React JS is a JavaScript library used for building interactive and responsive user interfaces.

The user interface is designed to provide easy navigation and better user experience for administrators, project managers, and team members.

Features of Front-End UI

1. Dashboard Interface

The dashboard provides an overview of:

- Total projects
- Active tasks
- Completed tasks
- Team performance

- Project progress status

2. Project Management Pages

User scan:

- Create projects
- Edit project details
- Assign team members
- Track project progress

25. RESULTS AND DISCUSSION

The Project Management System was successfully developed and implemented using **React JS** for the front-end, **Django Framework** for the back-end, and **PostgreSQL** for database management. The system was tested under different conditions to evaluate its performance, usability, reliability, and efficiency. The obtained results show that the system effectively manages projects, tasks, employees, and reports while reducing manual work and improving communication among team members.

26. ADVANTAGES OF THE SYSTEM

- Secure login and registration system implemented
- Role-based access for Admin, Manager, and Employees
- Unauthorized access prevented using authentication tokens

2. Project Management

- Projects can be created, updated, and monitored easily
- Project deadlines and status tracked efficiently
- Improved project organization and management

3. Task Management

- Tasks assigned to team members successfully
- Task priorities and deadlines managed properly
- Real-time task status updates available

4. Front-End Performance (React JS)

- Fast and responsive user interface
- Smooth navigation between pages
- Interactive dashboard for users
- Mobile-friendly and responsive design

5. Back-End Performance (Django)

- Efficient API handling using Django Framework
- Secure data processing and authentication

- Stable server-side performance during testing

REFERENCES

[1] References

[2] React Documentation Team, *ReactJS Official Documentation*, Meta Platforms Inc. Available at:

[React Official Website](https://reactjs.org/docs/getting-started.html)

[3] Django Software Foundation, *Django Documentation*, Django Project. Available at: [Django Official Website](https://docs.djangoproject.com/en/3.2/)

[9] Edition, 2019.

[10] Learning React, O'Reilly Media, 2020.

[11] RESTful Web APIs, O'Reilly Media, 2013.

[12] Research papers and online journals related to:

[13] Project Management Systems

[19] on for Web Development

[4] PostgreSQL Global Development Group, *PostgreSQL Documentation*. Available at:

[PostgreSQL Official Website](https://www.postgresql.org/docs/)

[5] Software Engineering, Ian Sommerville, 10th Edition, Pearson Education, 2016.

[6] Database System Concepts, McGraw Hill Education, 6th Edition, 2019.

[7] Web Technologies, Oxford University Press, 2018.

[8] Python Crash Course, No Starch Press, 2nd

[14] Web Application Development

[15] Task Management Systems

[16] Database Management Systems

[17] Software Development Methodologies

[18] Python Documentation