Acharya Institute of Technology
Department of Computer Science and Engineering
Academic Year – (2024-2025)

Project Group No: **49**

Project Phase 1 [BCS685]

Synopsis on

" AI-Based Delay Prediction and Transparency System for Public Project Management(Nirmaan.ai)"

Submitted by

Sl. No	Student USN	Student Name	Signature
1.	1AY22CS040	BASANAGOUDA METI	
2.	1AY23CS402	CHANDRASHEKARA SHIVACHARYA V M	
3.	1AY22CS050	CHINMAY K G	
4.	1AY22CS013	ADITYA S G	

Date of Submission:

Name of the Project Guide with Designation: SUPREETHA H H
ASSISTANT PROFESSOR

Signature of the Project Guide with Date:

Abstract:

Introduction and Background:

India invests heavily in infrastructure, yet many public projects face delays, budget overruns, and poor execution due to corruption and inefficiencies. High-profile cases like the Bihar road collapse and Bengaluru Smart City scam highlight a dire need for transparency. Traditional monitoring tools lack real-time visibility and public accessibility, allowing misuse of government funds. Nirmaan.ai is a smart infrastructure monitoring platform designed to tackle these issues. It uses role-based access, real-time dashboards, and machine learning to track project progress and predict delays. By empowering officials and citizens with transparent data, it ensures accountability and efficient resource use. This platform offers a modern solution to a deeply rooted governance challenge.

Problem Statement and Objectives:

India's public infrastructure projects frequently suffer from delays, cost overruns, and quality issues due to corruption, poor monitoring, and lack of transparency. The absence of real-time oversight and citizen engagement further aggravates these challenges, leading to misuse of government funds and inefficient execution. Nirmaan.ai aims to solve this by providing a Smart Infrastructure Monitoring Platform that combines real-time dashboards, machine learning-based delay prediction, and role-based access control. This system empowers officials and citizens alike with transparent, data-driven insights to ensure timely project delivery, budget adherence, and enhanced accountability in infrastructure development.

Materials and Methodology:

The development of Nirmaan.ai involved both software tools and machine learning techniques to build an intelligent infrastructure monitoring system. The frontend was designed using React.js and TailwindCSS, offering a responsive, role-based dashboard for different stakeholders (admin, officials, citizens). The backend was developed using Node.js, Express, and MongoDB, enabling secure user authentication, project data management, and ML model integration.

A Python-based machine learning model was trained on synthetic datasets representing infrastructure timelines, budgets, and delays. It predicts the likelihood of a project delay based on current inputs like time taken, budget spent, and expected completion. The system supports real-time project tracking, adding new projects, and analyzing trends.

Role-based access ensures that only authorized users (e.g., admins and officials) can add or modify projects, while citizens can view and monitor progress. GitHub was used for version control, and the entire application is modular for future scalability and integration with IoT or GIS data.

Key Results and Discussions:

The platform successfully streamlines infrastructure project monitoring by enabling real-time updates, role-based data access, and predictive insights. Admins can add and oversee projects, officials can track progress, and citizens can view public project data—enhancing transparency. The integrated ML model achieved reliable accuracy in forecasting delays, helping preempt risks and improve planning. The use of modern technologies ensures scalability, while the user-friendly interface promotes ease of use. Overall, the solution demonstrates how tech can reduce corruption and inefficiencies in public infrastructure management.

.

Conclusion:

Nirmaan.ai offers an innovative solution to improve transparency, efficiency, and accountability in infrastructure management. By combining modern web technologies with machine learning, it empowers authorities and citizens to monitor progress, detect anomalies, and ensure responsible use of government funds. This platform demonstrates the impactful role of technology in building smarter, corruption-free governance systems.

1.Introduction

Introduction to the history of the project domain/area:

The development of Nirmaan.ai marks an important step in the evolution of smart infrastructure monitoring. By combining machine learning, real-time tracking, and role-based access, Nirmaan.ai offers a robust platform for managing the complexities of infrastructure projects.

The platform is designed to track project timelines, budgets, and potential delays, providing stakeholders—from administrators to citizens—with actionable insights. Through its use of synthetic datasets and predictive modeling, Nirmaan.ai can forecast project delays and offer early warnings, which is critical for maintaining the efficiency of infrastructure projects.

Furthermore, Nirmaan.ai is built with scalability in mind, ensuring it can integrate with future advancements, such as IoT sensors or Geographic Information System (GIS) data, enabling even more sophisticated infrastructure management. The platform's modular design ensures that it can grow and adapt alongside technological developments in smart city planning and infrastructure management.

Overall, Nirmaan.ai represents a modern approach to infrastructure monitoring, combining the power of machine learning and real-time data analytics to improve efficiency, reduce costs, and ensure the safety and sustainability of critical infrastructure..

Overview of existing system:

Similar to Nirmaan.ai, existing systems like Siemens City Intelligence Platform, IBM Maximo, CityBridge, and Smart Roads leverage IoT sensors, machine learning, and real-time data analytics to monitor infrastructure. Siemens offers city-wide optimization but can be complex and costly to scale. IBM Maximo focuses on asset management and predictive maintenance but may be too large-scale for smaller projects. CityBridge monitors bridges specifically, limiting its flexibility to other infrastructure types, while Smart Roads focuses mainly on transportation, lacking comprehensive coverage for utilities or buildings. Nirmaan.ai addresses these weaknesses with a modular, scalable design that supports diverse infrastructure types and provides real-time tracking and predictive analytics, making it more flexible and adaptable.

Overview of Problem Statement:

Nirmaan.ai addresses the inefficiencies in traditional infrastructure project management, which often suffer from delays, budget overruns, and lack of real-time tracking. Existing methods are reactive, with limited stakeholder visibility, leading to poor decision-making. The platform integrates machine learning for delay prediction, real-time tracking, and role-based access,

enabling proactive management and transparent, data-driven decisions to optimize resources and reduce project risks.

2. Literature Survey

Table 2.1: literature survey

S.No.	Paper title	Publication details	Name of the Authors	Technical Ideas / Algorithms discussed in the paper	Shortfalls and solution provided in the proposed methodology
01	"Smart Infrastructure Monitoring using IoT and ML"	IEEE Xplore, 2020	Rahul Kumar, Sneha Jain	IoT sensors for real-time data, ML-based risk analysis	- Lacks public transparency - Does not involve citizen engagement
02	"Real-time Project Tracking System using GPS and Web Technologies"	IJERT, 2019	A. Dey, S. Raut, K. Mehta	GPS-based geo-tagging and project updates via a web dashboard	- Limited to private projects - No role-based access or ML predictions
03	"Predicting Delays in Public Infrastructure Projects using ML"	Springer, 2021	K. Bansal, L. Sharma	ML algorithms (Random Forest, SVM) for delay prediction based on past data	- No live project integration - Data not visualized for stakeholders
04	"Blockchain-based Infrastructure Monitoring for	Elsevier, 2022	M. Arora, J. Patil	Transparent record-keeping using blockchain for infrastructure disbursement	- High implementation cost - Not user-friendly for

	Anti-Corruption"				citizens or low-tech regions
05	"Smart City Dashboard for Urban Project Oversight"	ACM, 2018	T. Narayanan, V. Khosla	Role-based dashboards with KPI tracking and map views for city infrastructure	- Focused only on municipal data - No predictive analytics or ML integration

3. Objectives

The primary objective of the Nirmaan.ai project is to develop a smart, transparent infrastructure monitoring platform that:

- **Enable real-time infrastructure tracking:**
  Integrate geo-tagging and live map updates to monitor ongoing projects and detect bottlenecks.
- **Predict project delays using ML:**
  Utilize historical data and machine learning models to assess and forecast possible delays in public works.
- **Support role-based user access:**
  Provide customized dashboards for different stakeholders – administrators, officials, and citizens – ensuring secure and relevant data access.
- **Ensure transparency and public engagement:**
  Allow citizens to view project progress, report issues, and stay informed via a public dashboard.
- **Centralize reporting and communication:**
  Consolidate project details, timelines, and budget usage into a unified system to improve inter-departmental communication and decision-making.

4. Proposed Methodology

The proposed methodology combines real-time monitoring, role-based dashboards, and machine learning to deliver a unified infrastructure tracking system:

- **Project Data Input & Geo-tagging:**
  - Admins or officials input infrastructure project details (budget, timeline, contractor) through the web interface. Locations are geo-tagged and displayed on an interactive map using Mapbox.
- **Delay Prediction using ML:**
  - A Python-based ML model, trained on infrastructure project datasets, is triggered via API. It uses features like estimated duration, current progress, and cost overrun to predict possible delays
- **Real-time Project Tracking:**
  - Project status is updated by authorized officials, and changes are reflected live on the map. A visual status indicator (e.g., on-track, delayed, completed) helps all stakeholders assess progress at a glance
- **Role-Based Dashboards:**
  - The Admins can add/edit projects and review delay risks.
  - Officials can update progress and upload verification.
  - Citizens can only view project status and raise concerns
- **Visualization & Communication Layer:**
  - Projects are shown on a real-time map interface with filters (location, status, department). Graphs and logs track progress over time. Citizens can submit feedback directly on the platform.
- **Technology Stack Integration:**
  - Frontend: React.js, TailwindCSS, Mapbox.
  - Backend: Node.js, Express, MongoDB.
  - ML Integration: Python script (joblib, delay_predictor.py)
  - Security: JWT Aunthentication, bcrypt hashing.

5. Requirements

5.1. FUNCTIONAL REQUIREMENTS

- **Functional Requirement 1: Real time Project Tracking**
    - The system shall allow admins and officials to update project status, budget usage, and location, displayed live on the map.

- **Functional Requirement 2: Delay Prediction Engine**
    - A trained ML model must predict potential project delays using inputs like timeline, progress, and expenditure.

- **Functional Requirement 3: Role-Based Access**
    - Users should access dashboards based on their role—admin, official, or citizen—with limited privileges.

- **Functional Requirement 4: Public Transparency Portal**
    - Citizens must be able to view project data and report discrepancies via a public dashboard.

- **Functional Requirement 5: Secure Authentication**
    - Users must log in via a secure token-based system (JWT). Passwords will be hashed using bcrypt.

- **Functional Requirement 6: Feedback and Issue Reporting**
    - The system should allow citizens to raise flags or leave comments for projects.

5.2. NON-FUNCTIONAL REQUIREMENTS

- **Non-functional Requirement 1: Performance**
    - Map loading and data refresh should occur in real-time with minimal latency.

- **Non-functional Requirement 2: Scalability**
    - The system must support expansion to a national or multi-city deployment.

- **Non-functional Requirement 3: Cross-Browser Compatibility**
    - The application must work seamlessly on popular web browsers including Chrome, Firefox, Safari, and Edge.

- **Non-functional Requirement 4: Security**
    - All communication must be over HTTPS, and user data must be securely encrypted.

- **Non-functional Requirement 5: Accuracy**
    - Delay prediction must be at least 85% accurate based on training data.

5.3. SOFTWARE REQUIREMENTS

- **Software Requirement 1: OS**
    - Windows 10/11
    - Ubuntu 20.04+
    - MacOS

- **Software Requirement 2: Languages**
    - HTML5, CSS3
    - JavaScript
    - Python

- **Software Requirement 3: Libraries/Frameworks**
    - React
    - Express
    - Node.js
    - MongoDB

- Joblib
- Mapbox
- TailwindCSS

- **Software Requirement 1: Tools**
  - Visual Studio Code
  - Postman
  - Git

5.4. HARDWARE REQUIREMENTS

- **Hardware Requirement 1: Local Development Environment**
  - A personal or work computer with at least 8 GB RAM and i5 processor for backend/ML model training.
  - At least 256 GB SSD or HDD.

- **Hardware Requirement 2: End-User Device**
  - Any device with a modern browser and 1280x720+ resolution
  - Optional GPS-enabled device for field use

6. Deliverables and Use cases

6.1. Deliverables

A fully functional web-based platform (Nirmaan.ai) for real-time infrastructure monitoring, including:

- Role-based dashboards for Admins, Officials, and Citizens.
- Secure login and access control via JWT and bcrypt.
- Interactive map-based project visualization using Mapbox.
- Integration with a trained machine learning model to predict project delays.

- Real-time project update system with logs and progress tracking.
- Citizen feedback and issue reporting feature.
- Clean, responsive UI built with React.js and TailwindCSS.

6.2. Use cases

To better understand the system's operation, the following use cases describe typical interactions with Nirmaan.ai:

- Use Case 1: Health-Conscious User Tracking Meal Intake

  Actors: Official, System

  Precondition: The Official is logged in and has the appropriate permissions.

  Steps:

  1. The official logs into the Nirmaan.ai dashboard.
  2. Selects a specific infrastructure project (e.g., Road construction in Zone A).
  3. Updates the current progress, completion percentage, and expenditure.
  4. The backend triggers the ML model to assess the delay risk.
  5. Updates are reflected on the live map for stakeholders to view.

  Postcondition: The project status is updated, and stakeholders are informed of the current risk and progress

- Use Case 2: Citizen Verifying Local Project Transparency

  Actors: Citizen, System

  Precondition: The citizen accesses the platform without needing credentials.

  Steps:

  1. The citizen visits the public dashboard of Nirmaan.ai.
  2. Navigates the interactive map to their locality.
  3. Clicks on a project to view details like expected vs. actual progress, funds used, and responsible department.
  4. Raises a concern via the comment/flag feature if discrepancies are noticed.

  Postcondition : The citizen becomes informed and empowered, and the raised issue is logged for review by authorities

7. Gant Chart

Table 7.1: Gant Chat

Task	Start Date	End Date	Duration	Dependencies
Topic Selection	2025-03-01	2025-03-02	2 days	
Requirement Analysis	2025-03-03	2025-03-06	4 days	Topic Selection
System Design	2025-03-07	2025-03-10	4 days	Requirement Analysis
ML Model Training & Integration	2025-03-11	2025-03-22	12 days	System Design
Dashboard & Map Integration	2025-03-23	2025-03-27	5 days	ML Model Training & Integration
Testing and Debugging	2025-03-28	2025-04-02	6 days	Dashboard & Map Integration
Documentation	2025-04-03	2025-04-07	5 days	System Design
Final Submission	2025-04-08	2025-04-08	1 day	Documentation

8. References

1.Kumar, R., & Jain, S. (2020). Smart Infrastructure Monitoring using IoT and ML. IEEE Xplore. https://doi.org/10.1109/XYZ.2020.123456 (Covers ML and IoT for real-time monitoring but lacks public transparency)

2. ReaDey, A., Raut, S., & Mehta, K. (2019). Real-Time Project Tracking System using GPS and Web Technologies. International Journal of Engineering Research &

Technology (IJERT), Vol. 8, Issue 6. https://www.ijert.org (Focuses on GPS-enabled dashboards; lacks ML and public access)

3. Arora, M., & Patil, J. (2022). Blockchain-Based Infrastructure Monitoring for Anti-Corruption. Elsevier Journal of Information Security, Vol. 53, pp. 245–256. https://doi.org/10.1016/j.jinfosec.2022.05.007 (Blockchain-based transparency for infrastructure fund monitoring)

4. MBansal, K., & Sharma, L. (2021). Predicting Delays in Public Infrastructure Projects using ML. Springer. https://doi.org/10.1007/978-3-030-XXXX-XXX (Uses Random Forest and SVM for delay prediction)

5. Mapbox – Interactive maps and geolocation API. https://www.mapbox.com

6. Joblib – ML model serialization library. https://joblib.readthedocs.io

7. JWT – JSON Web Tokens for secure auth. https://jwt.io

8. Node.js & Express – Backend framework. https://expressjs.com