
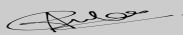




Capstone Project Proposal

Streetlight

Department - Campus	Roll Number	Name	Signatures
IT-OC	BITF22M006	Mariam Qadeem	
IT-OC	BITF22M011	Areeba Tahir	
IT-OC	BITF22M015	Kinz-ul-Eman	
IT-OC	BITF22M044	Shezonias	

Supervisor Name	Dept	Email	Primary/ Secondary	Signatures
Adeel Nisar	IT	adeel.nisar@pucit.edu.pk	Primary	

Department of Information Technology

**Faculty of Computing & Information Technology
University of the Punjab**

Table of Contents

INTRODUCTION	3
SOLUTION DESCRIPTION	4
SYSTEM ARCHITECTURE.....	7
TECHNICAL SPECIFICATIONS.....	9
TEAM INTRODUCTION	10
PROJECT TIMELINE.....	11

Introduction

Title : Streetlight

What is the problem you are going to solve?

The project addresses the absence of an effective, transparent, and centralized platform for citizens to report civic issues (garbage accumulation, potholes, broken roads, faulty streetlights etc). Current reporting methods mostly phone calls, scattered emails, or outdated systems are slow, disorganized, and provide almost no accountability. Citizens often remain unaware of which department is responsible or whether any action is even being taken on their complaint. At the same time, municipal departments do not have clear, up-to-date, or long-term data. This makes it impossible for them to spot recurring problems, decide what needs urgent attention, or move from putting out fires to actually preventing them through smart, forward-looking planning.

Why is this problem chosen as such?

This problem was chosen because it directly impacts Pakistan's fast-growing cities and affects everyday issues like public safety, living conditions, and people's trust in government. As urban areas continue to expand, the lack of a modern complaint system makes it difficult to manage infrastructure issues and creates frustration for citizens who feel unheard. Solving this problem creates measurable impact for communities while addressing Pakistan's smart-city development goals.

Does this problem have a technical impact on your learning?

Yes, this project provides significant technical learning across multiple domains. It covers full-stack development by building both a mobile app and a web dashboard, artificial intelligence to automatically analyze images and detect issue severity, database design to store and manage data efficiently, secure login and authentication systems, blockchain technology for creating transparent and unchangeable records, API development to connect different parts of the system, user-friendly interfaces for both mobile and web platforms, and data analytics to find patterns and predict future problems.

Does this problem have a social impact at large?

Socially, the project creates real impact by giving citizens the power to actively improve their communities. For individuals, it means safer roads, faster problem resolution, and the ability to hold authorities accountable, directly improving daily life for millions. For communities, it brings neighbors together to verify issues collectively, building trust and civic participation. At a larger scale, it addresses Pakistan's infrastructure challenges through data-driven city planning, reduces economic losses from poor roads , and supports smart-city development. By making government operations transparent and accountable, (name of the project) bridges the gap between citizens and authorities, strengthening public participation and improving services across Pakistan's growing urban areas.

Solution Description

Streetlight is a unified digital platform designed to connect citizens and municipal authorities in a transparent, efficient, and accountable manner. The solution replaces fragmented manual complaint channels with a single structured system that enables real-time reporting, verification, tracking, and resolution of civic infrastructure issues.

Type of Application

Streetlight is a cross-platform solution consisting of:

- ◆ **Mobile Application (Flutter)** for citizens to report issues on Android and iOS devices
- ◆ **Web Dashboard (React.js)** for municipal officers to manage and resolve complaints
- ◆ **Cloud-Based Backend System (Python)** that handles complaint processing, automated routing, notifications, and data analytics

How It Works

The platform operates through a simple workflow:

- ◆ **Citizen Reporting:** Citizens report infrastructure problems through the mobile app by capturing photos and confirming GPS location. The system generates a unique complaint ID and sends verification requests to nearby residents.
- ◆ **AI-Based Analysis:** An AI module automatically analyzes the submitted media to classify the issue (e.g., garbage, pothole, sewerage), assess severity, and detect irrelevant or low-quality submissions.
- ◆ **Deduplication Engine:** The system checks spatial, temporal, and visual similarity to remove duplicate complaints and merge multiple reports related to the same issue into a single verified case.
- ◆ **Community Verification:** Multiple neighbors receive notifications to confirm whether the issue actually exists. Once verified by residents, the complaint is marked as "**Verified.**"
- ◆ **Blockchain Integrity Logging:** After verification and deduplication, our system's backend will generate a cryptographic hash (SHA-256) of the complaint data. This hash, along with essential metadata and timestamp, is stored on the blockchain, and the blockchain transaction ID is saved in the system database to ensure immutability, transparency, and tamper-proof records.
- ◆ **Automatic Routing:** Verified complaints are automatically routed to the responsible municipal department based on issue type and location (e.g., garbage to LWMC, roads to highways department).
- ◆ **Department Assignment:** Municipal officers receive instant notifications and view verified complaints on their web dashboard. They assign complaints to field teams and update progress through different stages (Pending → Verified → Assigned → In Progress → Resolved).

- ◆ **Resolution Proof:** Officers upload before-and-after photos as proof of resolution before marking complaints as complete.
- ◆ **Citizen Approval & Escalation:** Citizens track progress in real time through the mobile app and must approve the resolution before a complaint can be closed. If rejected, the system automatically escalates the case to senior officials for review.
- ◆ **Data Analytics & Urban Planning:** Throughout the workflow, the system continuously collects data to generate analytics dashboards showing complaint hotspots, recurring problem patterns, and departmental performance metrics to support informed decision-making and smart city planning.

Major Components

The solution includes several key components:

- ◆ Mobile Reporting Interface with camera and GPS integration for instant complaint submission
- ◆ AI Image Analysis Module that automatically detects issue severity and classifies problem types
- ◆ Community Verification System where nearby residents validate complaints through voting
- ◆ Smart Deduplication Engine that identifies and merges similar complaints using location and visual matching
- ◆ Automated Routing Algorithm that assigns complaints to correct departments based on issue type
- ◆ Real-Time Notification Service for status updates via push notifications and email
- ◆ Web-Based Officer Dashboard with workflow management tools for municipal departments
- ◆ Photo Documentation System requiring before-and-after proof of resolution
- ◆ Analytics Engine generating visual maps, charts, and trend analysis for city planning
- ◆ Blockchain Integration for transparent, tamper-proof audit trails
- ◆ Secure Authentication to protect user data and ensure authorized access

Expected Outcome

The final product will be a fully functional civic management platform that eliminates manual complaint handling, creates accountability through transparent tracking, prevents false reports through community verification, and provides municipalities with actionable data for better city planning. Citizens will have a simple, reliable way to report problems and see real progress, while authorities gain the tools and insights needed to maintain infrastructure proactively rather than reactively.

System Architecture

StreetLight is a network-based client-server application connecting citizens, municipal authorities, and urban planning departments in a transparent and accountable ecosystem. The architecture consists of three main layers ensuring efficiency, security, and scalability.

1. Citizen Interaction Layer (Clients)

Mobile App (Flutter): Citizens report civic issues with photos/videos and GPS location, track complaint status, and verify nearby reports.

Web Dashboard (React.js): Municipal officers view/manage verified complaints, update status, upload resolution proof, and interact with citizen feedback.

Connectivity: Clients communicate with the backend via secure HTTPS APIs for fast, reliable, and encrypted data exchange.

2. Processing & Verification Layer (Server)

Backend Server (Python + FastAPI): Manages authentication, workflows, API requests, and complaint lifecycle.

AI/ML Engine: Classifies issues, detects severity, and identifies potential duplicate complaints from images/videos.

Community Verification Module: Nearby citizens confirm or reject reported issues before forwarding to municipal officers.

Notification & Task Queue (Celery + Redis): Sends timely updates to citizens and officers about status changes and escalations.

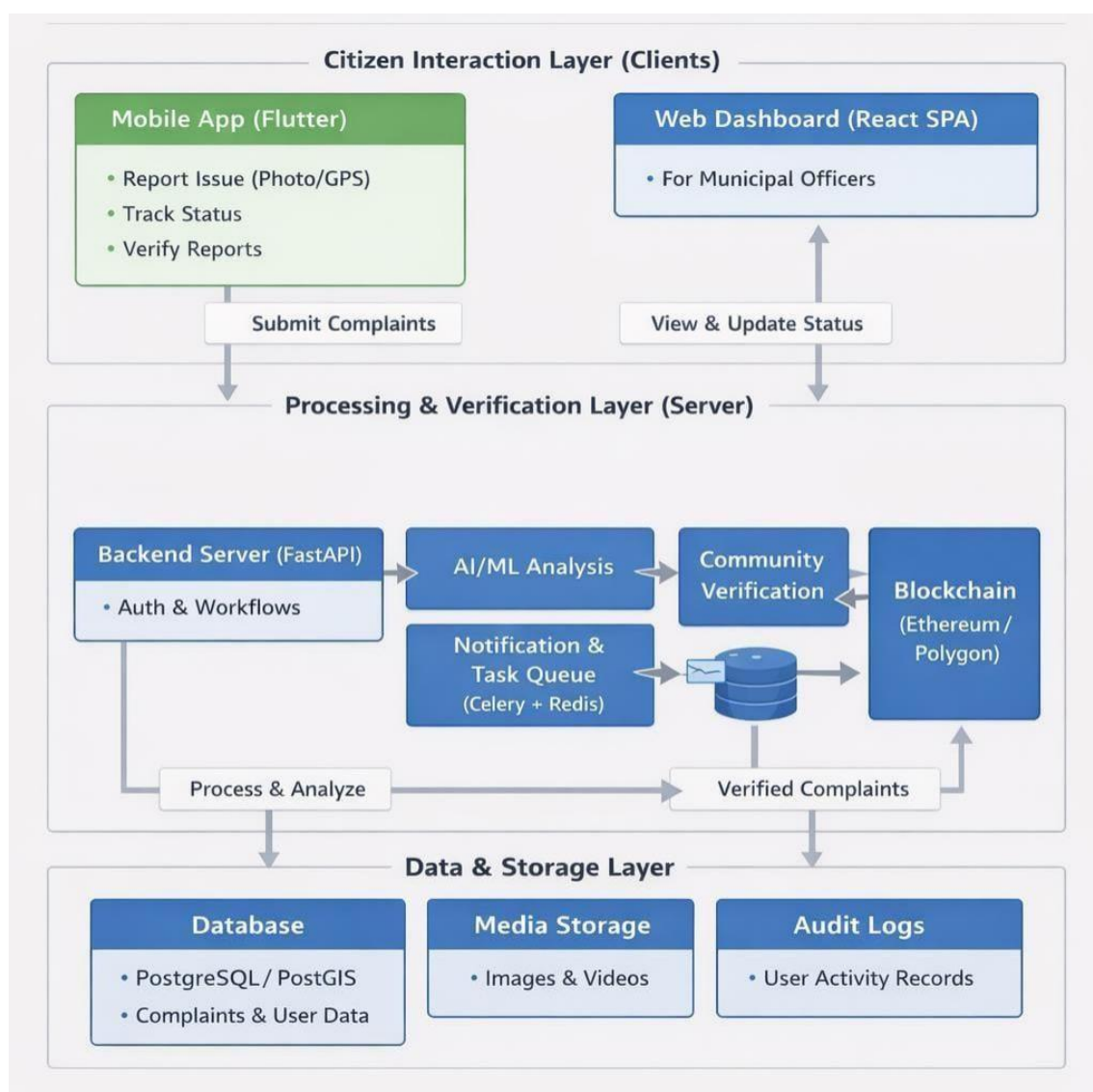
Blockchain Layer (Ethereum / Polygon Testnet): Stores cryptographic hashes of verified complaints for immutable, tamper-proof records.

3. Data & Storage Layer

Database (PostgreSQL + PostGIS): Stores complaint details, user info, votes, and spatial data for duplicates and mapping.

Media Storage (Firebase): Secures images, videos, and resolution proof with controlled access.

Audit Logs: Tracks all user/officer actions, providing a permanent record for accountability.



Technical Specifications

Primary Database	PostgreSQL
------------------	------------

Backend Development	
Programming Language	Python
Framework	FastAPI
Authentication	JWT (Role-based access control)
AI/ML	PyTorch, OpenCV

Frontend Development:	
<i>Mobile Application</i>	
Framework	Flutter (Dart)
<i>Web Dashboard</i>	
Framework	PyTorch, OpenCV
UI Library	Material UI
Styling	Tailwind CSS

APIs and Technologies	
RESTful APIs for client-server communication	
OpenStreetMap API for mapping services	
Firebase APIs for notifications and storage	
GPS services for location tracking	

Blockchain Layer	
Network	Ethereum / Polygon Testnet
Smart Contracts	Solidity

Third-Party Services	
Web Hosting	Vercel
Database Hosting	Supabase
Version Control	GitHub

Team Introduction

Team Member	Registration No.	Role	Technical Skills
Mariam Qadeem	BITF22M006	AI/ML Engineer / Backend Developer	Python, TensorFlow, machine learning, image processing, REST APIs
Areeba Tahir Munir	BITF22M011	Blockchain Developer / Security Engineer	Solidity, smart contracts, cybersecurity, blockchain technology
Kinzul Eman	BITF22M015	AI/ML Engineer / Database Designer	Python, computer vision, SQL/NoSQL databases, data design
Shezonia Idrees	BITF22M044	Project Manager / Frontend Developer	Project management, Flutter, React.js, HTML/CSS, JavaScript

Project Timeline

Milestone	Deliverable	Timeline
1	Project Registration	January 2025
2	Proposal Submission (PPD)	January 2025
3	Analysis & Design Deliverable (ADD1)	February 2025
4	Advanced Design Deliverable (ADD2)	March 2025
5	Prototype Evaluation Deliverable (PED)	April 2025
6	Internal Final Evaluation Deliverable (IF-ED)	May 2025
7	External Final Evaluation Deliverable (EF-ED)	June 2025
8	Project Final Documentation (PFD)	June 2025