BANGLADESH UNIVERSITY OF BUSINESS & TECHNOLOGY (BUBT)

Department of Computer Science & Engineering

SafePath

Community-Driven Harassment Reporting & Heatmap Platform

Submitted to:
Md Ashiqur Rahman
Lecturer, Department of CSE

Date of Submission:
November 26, 2025

Chapter 1
Introduction

1.1 Project Overview

SafePath is a web-based platform designed to address the critical issue of public harassment in Bangladesh. The system enables users to report incidents anonymously, which are then aggregated to create real-time "heatmaps" of unsafe zones. By leveraging modern web technologies, SafePath aims to bridge the gap between victims and law enforcement, providing data-driven insights to improve public safety.

1.2 Problem Statement

Harassment in public spaces is a pervasive issue, yet it remains severely underreported.

- **High Prevalence:** Statistics indicate that 87% of women have faced harassment.
- **Public Transport Risks:** 36% of women face harassment frequently while using public transport.
- **Lack of Legal Action:** Only 0.5% of victims take formal legal action.
- **The Silence Gap:** Approximately 34.8% of victims remain silent due to fear of judgment and complicated reporting systems.

This lack of reporting results in zero visibility for authorities, meaning unsafe areas remain unidentified and preventive actions are rarely taken.

1.3 Project Objectives

The primary objectives of the SafePath project are:

1. To provide a secure, anonymous platform for reporting harassment in under 10 seconds.
2. To visualize unsafe zones using real-time heatmaps based on crowdsourced data.
3. To suggest safer travel routes based on incident density.
4. To provide an administrative dashboard for NGOs and Police to monitor and verify reports.

1.4 Team Contribution

The development was distributed among the team members as follows:

Name	**ID**	**Specific Responsibilities**
Shad Mohammad	...115	UI/UX Design + Basic React Structure
Md Nafijur Rahaman	...116	Admin Panel, API Integration, User Authentication
Vasha Quddus	...118	Backend Logic(Report & Location)
Md Alhanullah Sajib	...201	API Integration + Dynamic Components+ Leaflet Map Integration
Sadia Afrin Khan	...306	Backend Logic(Report & Location)

Table 1.1: Team Responsibilities Breakdown

Chapter 2
System Analysis & Design

2.1 Requirement Analysis

2.1.1 Functional Requirements

The functional requirements outline the essential operations that the SafePath system must support. These requirements ensure that users can report incidents, explore safety insights, and receive guidance, while administrators manage verification and maintain data quality. The system also automates geolocation capture and scoring to streamline overall workflow.

- **User Module:** Anonymous reporting, View heatmap, Get safer route suggestions, View safety tips.
- **Admin Module:** Login authentication, Verify pending reports (Accept/Reject), View dashboard statistics.
- **System Module:** Auto-capture Geolocation, Calculate route safety score.

2.1.2 Non-Functional Requirements

Non-functional requirements define the quality attributes of the system. These characteristics ensure smooth performance, the ability to scale under increased load, and a user-friendly interface suitable even for low-end mobile devices.

- **Performance:** The system must load the map and heat layers within 2 seconds.
- **Scalability:** The backend must handle concurrent report submissions.
- **Usability:** The UI must be mobile-responsive and accessible on low-end devices.

2.2 System Architecture

SafePath follows a standard **Client-Server Architecture**, ensuring a clean separation between user interfaces, backend processing, and data management.

- **Frontend:** React.js Single Page Application (SPA).
- **Backend:** Spring Boot REST API.

- **Database:** MySQL Relational Database.

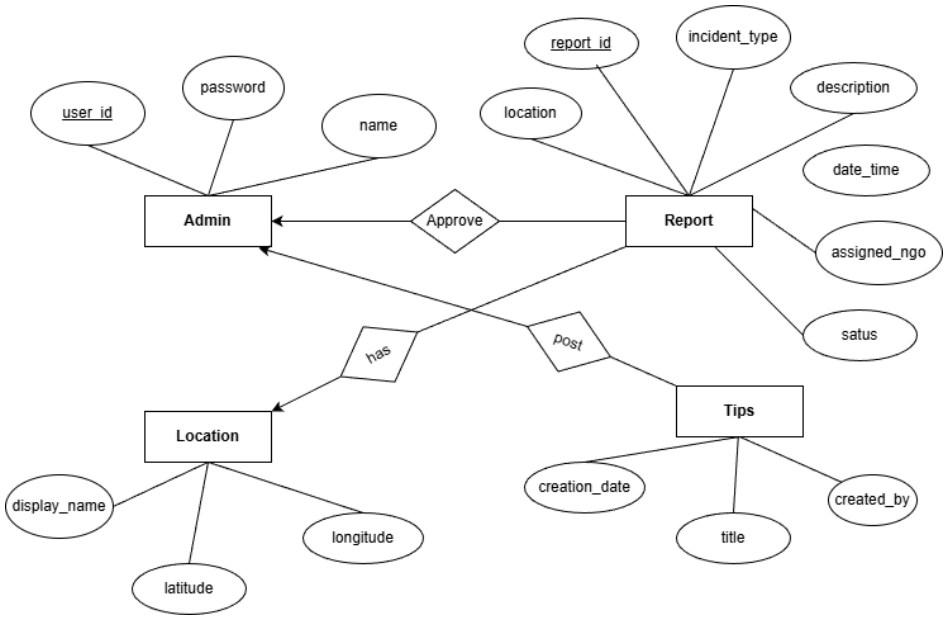

Figure 2.1: Entity Relationship (ER) Diagram of SafePath

2.3 Entity Relationship (ER) Diagram Description

The ER diagram presents the conceptual structure of the SafePath database. It outlines the main entities—Admins, Reports, Locations, and Safety Tips—along with their interactions. This structure ensures consistency, supports data validation, and enables efficient map-based safety visualizations.

2.3.1 Entities and Attributes

The system consists of four main entities, each serving a specific purpose within the platform.

- **Admin:** Represents the system moderators responsible for reviewing reports and publishing tips.
  - `user_id` (Primary Key)
  - `name`
  - `password`
- **Report:** Captures details of harassment or safety incidents submitted by users.
  - `report_id` (Primary Key)
  - `incident_type`
  - `description`
  - `date_time`
  - `assigned_ngo`

  - `status`

- **Location:** Stores geographical information associated with reports.
  - `latitude`
  - `longitude`
  - `display_name`

- **Tips:** Contains safety advice published by administrators.
  - `title`
  - `creation_date`
  - `created_by`

2.3.2 Relationships

The relationships define how the entities interact within the SafePath ecosystem. These connections help maintain data integrity and support core functionalities such as report verification and mapping.

1. **Admin approves Report.**
2. **Admin posts Tips** (One-to-many relationship).
3. **Report has Location.**

Chapter 3

User Interface & Results

3.1 Home Page & Map View

The Home Page is designed as the primary interaction point for users. It displays an interactive map integrated with a dynamic heatmap layer that visualizes real-time incident density. Users can zoom, drag, and explore different areas to understand risk levels before navigating. The intuitive interface ensures quick access to essential safety insights without requiring a login, keeping the experience anonymous and accessible. The heatmap adapts as new reports are submitted, reflecting the most recent data.

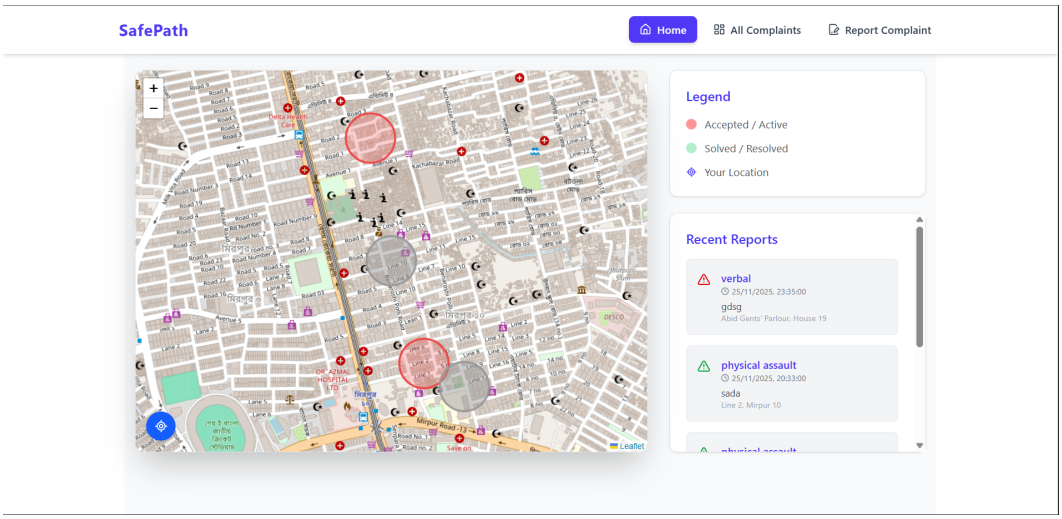

Figure 3.1: SafePath Homepage showing the interactive Heatmap.

3.2 Reporting Interface

The Reporting Interface allows users to submit harassment or safety-related incidents anonymously. The form collects essential details such as the type of incident, a short description, and the automatically detected location. A minimalist layout ensures that the process is fast, reducing hesitation for users in stressful situations. The system validates the data and forwards it to the admin panel for verification. This streamlined flow helps maintain both ease of use and data reliability.

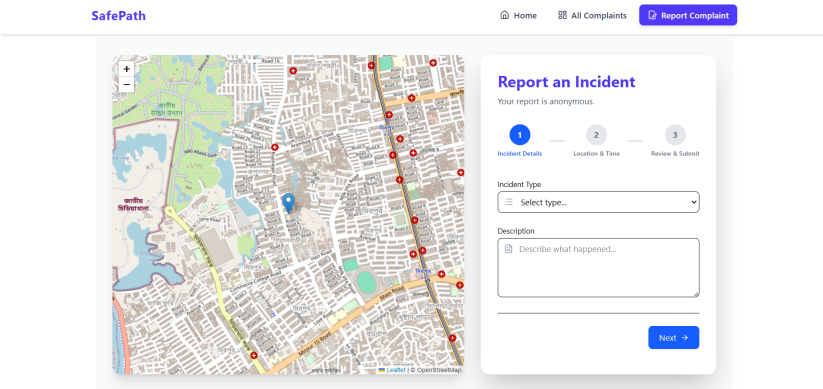

Figure 3.2: Anonymous Reporting Form.

3.3 Admin Dashboard

The Admin Dashboard is designed for authorized moderators to manage and verify submitted reports. It presents a clean table view with filters, timestamps, incident categories, and action buttons for approval or rejection. The dashboard also displays summary metrics that help administrators track daily report volume, system activity, and trends. This centralized interface ensures that all reports are reviewed efficiently, maintaining the accuracy and credibility of the platform's heatmap and route-safety features.

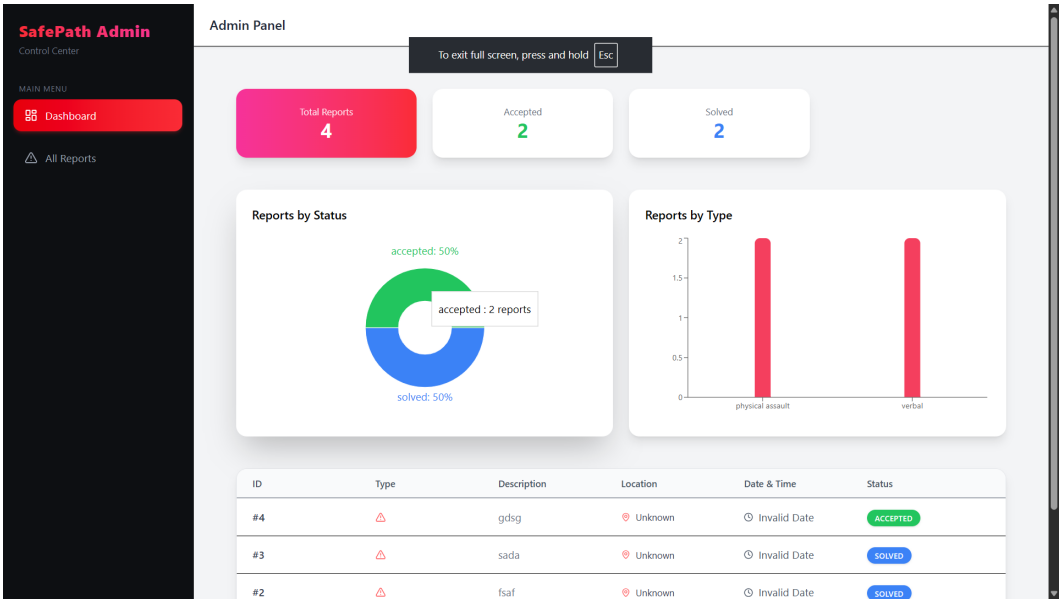

Figure 3.3: Admin Dashboard for verifying reports.

Chapter 4
Conclusion & Future Scope

4.1 Challenges Faced

Developing SafePath required addressing several technical and practical hurdles that affected both accuracy and usability. As the platform relies heavily on user-generated data, maintaining reliability while preventing misuse was a consistent challenge. Additionally, location-based systems often struggle with precision due to environmental factors, which required thoughtful design decisions to ensure consistent performance.

- **Spam Prevention:** Ensuring truthful and authentic reports was a major challenge. To reduce spam, the system integrates admin verification and automated consistency checks.
- **Geolocation Accuracy:** Urban high-rise areas often introduce GPS drift. This issue was minimized by providing users with an option to manually adjust the map pin before submission.

4.2 Future Improvements

While SafePath already provides a functional platform for reporting and visualizing unsafe areas, several enhancements can significantly strengthen its usability, intelligence, and accessibility. These improvements aim to expand the system's predictive capabilities, increase offline reliability, and reach a broader user base through native mobile applications.

1. **AI Prediction Models:** Integrating machine learning would enable forecasting of high-risk zones based on historical trends and temporal patterns.
2. **Offline Mode:** Allowing reports to be saved locally and synchronized once connectivity is restored would make the system more accessible in low-network areas.
3. **Mobile App:** Building a native Android application would improve performance, allow background location features, and enhance user engagement.

4.3 Conclusion

SafePath demonstrates the potential of community-driven data in improving urban safety and awareness. By combining real-time reporting, heatmap visualization, and admin verification, the system provides a scalable and practical solution for identifying unsafe zones. Although challenges such as data reliability and geolocation accuracy persist, the platform lays a strong foundation for future development. With additional intelligence, offline support, and mobile integration, SafePath can evolve into a comprehensive safety assistance tool for everyday users.