

Project Title:

Title - BroomNav: The Safe Route Companion

Synopsis:

BroomNav is a spooky-themed, safety-focused navigation web app designed to help users find the safest possible route between two locations instead of just the fastest. Traditional maps only focus on travel time and distance, often overlooking safety factors like lighting, accident-prone areas, or poorly crowded streets. BroomNav fills this gap by integrating real-world and user-contributed data to compute a Safety Score for each route, guiding users through paths that are safer, better lit, and more reliable- especially during late-night travel.

The app visually represents safety levels on an interactive map, using green, orange, and red color codes to indicate safe, moderate, and risky zones respectively. It also features a community-driven layer where users can report “spooky” or unsafe locations, helping others make informed decisions. Designed to combine functionality with a fun, Halloween-inspired interface, BroomNav ensures that safety can also be engaging and collaborative.

Tech Stack:

Frontend:

HTML, CSS, JavaScript

Bootstrap or Tailwind CSS for responsive UI

Leaflet.js for interactive map integration

OpenStreetMap API for route and map data

Backend:

Node.js with Express.js for server and API handling

Database

Firebase Realtime Database or MongoDB Atlas for storing user reports and safety data

APIs / Data Sources:

OpenStreetMap for route data

Mock or open datasets for crime and lighting data (JSON files or APIs)

User-reported safety and spooky zone data




Key Features:

1. Safety-Based Navigation:

Instead of the shortest route, BroomNav calculates the safest route based on multiple safety parameters such as lighting conditions, nearby accidents, crime reports, and user feedback.

2. Safety Visualization:

Each route is color-coded according to its safety score:

-  Safe (well-lit, low-risk areas)
-  Moderate (some risk factors present)
-  Risky (poor lighting or high accident/crime zones)

3. User Reporting System:

Users can mark and describe “spooky” or unsafe areas on the map (e.g., dimly lit roads, isolated areas, places with frequent accidents), helping build a community-driven safety layer.

4. Spooky and Engaging Interface:

The UI uses a Halloween-inspired dark mode with ghostly icons, subtle animations, and a fun tone while maintaining usability and readability.

5. Real-Time Updates (Prototype):

Reports added by users get reflected for others instantly (via Firebase’s real-time database sync).

6. Mobile-Friendly Design:

Designed to work seamlessly on both mobile and desktop browsers, ensuring accessibility and convenience for all users.

Workflow:

1. User Input:

The user enters their starting point and destination.

2. Route Fetching:

The app fetches possible routes using the OpenStreetMap API and overlays them on an interactive map.

3. Safety Score Calculation:

For each route, the backend calculates a Safety Score based on:

Lighting data (mock or open dataset)

Historical crime or accident reports

Crowd density (if available or approximated)

User-contributed “spooky” zones

4. Route Display:

The app displays all possible routes with color-coded overlays to indicate safety levels.

5. User Feedback Loop:

Users can tap on any location to report it as unsafe or spooky. This feedback updates the database and improves route scoring over time.