**SafeRoutes Web App - MVP Roadmap**

I'll provide a simplified roadmap for building your safety app as a web application with core features. This approach will help you create a functional MVP for the GDG Solution Hackathon within 14 days.

**Project Overview**

SafeRoutes will be a web application that helps users:

1. View crime hotspots on a map
2. Find safe routes between locations
3. Report incidents anonymously
4. See crime analytics for their area

**Tech Stack**

To keep things simple and focused on the MVP, we'll use:

**Frontend**

* **React.js** - For building the user interface
* **Google Maps API** - For map integration and routing
* **Chart.js** - For analytics visualization
* **Material-UI** - For pre-built UI components
* **Firebase SDK** - For connecting to backend services

**Backend**

* **Firebase Authentication** - For user accounts
* **Firestore** - For database needs
* **Firebase Hosting** - For deploying the web app
* **Firebase Cloud Functions** - For any backend processing

**Project Structure**

Copy

saferoutes/

├── public/

│ ├── index.html

│ └── favicon.ico

├── src/

│ ├── components/

│ │ ├── Map/

│ │ ├── Navigation/

│ │ ├── Reports/

│ │ └── Analytics/

│ ├── pages/

│ │ ├── Home.js

│ │ ├── SafeRoutes.js

│ │ ├── ReportIncident.js

│ │ └── Dashboard.js

│ ├── services/

│ │ ├── firebase.js

│ │ ├── mapService.js

│ │ └── analyticsService.js

│ ├── App.js

│ └── index.js

├── firebase.json

└── package.json

**14-Day Roadmap**

**Days 1-2: Setup & Planning**

* Set up project repository
* Configure Firebase project
* Create basic React app structure
* Set up Google Maps API
* Design database schema

**Days 3-5: Core Map Features**

* Implement map display
* Add crime hotspot visualization
* Create safe route calculation algorithm
* Build basic UI components

**Days 6-8: User Features**

* Implement anonymous authentication
* Create incident reporting form
* Add location selection functionality
* Build report submission flow

**Days 9-11: Analytics Dashboard**

* Create analytics components
* Implement data visualization
* Add filtering capabilities
* Connect to Firestore data

**Days 12-14: Integration & Polishing**

* Connect all components
* Add responsive design
* Optimize performance
* Prepare for submission
* Create documentation

**Core Features Focus**

1. **Interactive Map**
   * Crime hotspot visualization (heatmap)
   * Current location display
   * Location search functionality
2. **Safe Route Generation**
   * Route calculation between two points
   * Safety score display
   * Alternative route suggestions
3. **Incident Reporting**
   * Anonymous reporting form
   * Location selection
   * Incident categorization
4. **Analytics Dashboard**
   * Crime frequency visualization
   * Area safety comparison
   * Time-based trend analysis

**Day-by-Day Task Breakdown**

**Day 1: Project Setup**

* Create GitHub repository
* Set up Firebase project (Authentication, Firestore, Hosting)
* Initialize React app
* Configure Google Maps API
* Create project structure

**Day 2: Map Implementation**

* Add Google Maps component
* Implement basic map controls
* Add location search functionality
* Create map marker components

**Day 3: Hotspot Visualization**

* Design crime data structure
* Add sample crime data to Firestore
* Implement heatmap visualization
* Add crime category filtering

**Day 4: Safe Routes Algorithm**

* Create routing service
* Implement route calculation
* Add safety scoring logic
* Display routes on map

**Day 5: Route Optimization**

* Enhance route safety algorithm
* Add alternative routes display
* Implement route comparison
* Create route details panel

**Day 6: User Authentication**

* Set up Firebase Authentication
* Create login/signup components
* Implement anonymous authentication
* Add user profile management

**Day 7: Incident Reporting Form**

* Design report submission form
* Create form validation
* Implement location selection
* Add category selection

**Day 8: Report Submission Flow**

* Connect form to Firestore
* Add anonymous submission option
* Implement confirmation flow
* Create user report history

**Day 9: Analytics Framework**

* Design analytics dashboard layout
* Create data aggregation service
* Implement basic chart components
* Connect to Firestore data

**Day 10: Data Visualization**

* Enhance chart components
* Add filtering capabilities
* Implement time-based analysis
* Create area comparison view

**Day 11: Dashboard Refinement**

* Add interactive elements to charts
* Implement trend analysis
* Create user-friendly visualizations
* Add export functionality

**Day 12: Integration & Testing**

* Connect all components
* Implement global state management
* Add loading states and error handling
* Conduct cross-browser testing

**Day 13: UI Polishing**

* Enhance visual design
* Add responsive design
* Implement dark/light mode
* Create onboarding flow

**Day 14: Submission Preparation**

* Final bug fixes
* Create documentation
* Record demo video
* Prepare submission materials