IVARS - Incident Verification and Reporting System

Workflow Block Diagram

```
flowchart TB
    Start([User Accesses Application]) --> Home[Home Page]

    Home --> AuthCheck{Authenticated?}

    AuthCheck -->|No| GuestActions[Guest Actions]
    AuthCheck -->|Yes| UserActions[Authenticated Actions]

    GuestActions --> ViewMap[View Live Incident Map]
    GuestActions --> ReportIncident[Report Incident]
    GuestActions --> Login[Login/Register]

    Login --> AuthAPI[/auth API/]
    AuthAPI --> Register[POST /register]
    AuthAPI --> LoginReq[POST /login]

    Register --> TokenGen[Generate JWT Token]
    LoginReq --> TokenGen
    TokenGen --> AuthSuccess[Authentication Success]

    AuthSuccess --> UserActions

    UserActions --> RoleCheck{User Role?}

    RoleCheck -->|Citizen| CitizenActions[Citizen Actions]
    RoleCheck -->|Responder| ResponderActions[Responder Actions]
    RoleCheck -->|Admin| AdminActions[Admin Actions]

    CitizenActions --> MyReports[My Reports Page]
    CitizenActions --> CreateReport[Create Incident Report]

    CreateReport --> IncidentForm[Incident Reporting Form]
    IncidentForm --> LocationPicker[Select Location on Map]
    IncidentForm --> UploadImages[Upload Images - Cloudinary]
    IncidentForm --> IncidentDetails[Enter Incident Details]

    LocationPicker --> GoogleMaps[Google Maps API]
    UploadImages --> CloudStorage[Cloudinary Storage]

    IncidentDetails --> SubmitIncident[POST /api/incidents]
    SubmitIncident --> SaveDB[(MongoDB Database)]
    SaveDB --> NotifySystem[Notification System]

    ResponderActions --> Dashboard[Dashboard Page]
    ResponderActions --> ViewIncidents[View All Incidents]
    ResponderActions --> UpdateStatus[Update Incident Status]
```

```
    Dashboard --> LiveMap[Live Incident Map]
    Dashboard --> IncidentList[Incident List]

    UpdateStatus --> UpdateAPI[PUT /api/incidents/:id]
    UpdateAPI --> SaveDB

    AdminActions --> Analytics[Analytics Page]
    AdminActions --> ManageIncidents[Manage All Incidents]
    AdminActions --> ViewStats[View Statistics]

    Analytics --> StatsAPI[GET /api/incidents/stats/overview]
    StatsAPI --> ProcessData[Process Data]
    ProcessData --> DisplayCharts[Display Charts & Metrics]

    ManageIncidents --> DeleteIncident[DELETE /api/incidents/:id]
    DeleteIncident --> SaveDB

    MyReports --> GetMyIncidents[GET /api/incidents/my-reports]
    ViewIncidents --> GetAllIncidents[GET /api/incidents]

    GetMyIncidents --> SaveDB
    GetAllIncidents --> SaveDB

    SaveDB --> RenderData[Render Data]

    RenderData --> DisplayMap[Display on Map]
    RenderData --> DisplayList[Display in List]

    DisplayMap --> GoogleMaps

    subgraph "Frontend - React/TypeScript"
        Home
        ReportIncident
        MyReports
        Dashboard
        Analytics
        IncidentForm
        LocationPicker
        LiveMap
        IncidentList
        DisplayCharts
        DisplayMap
        DisplayList
    end

    subgraph "Backend - Node.js/Express"
        AuthAPI
        SubmitIncident
        UpdateAPI
        StatsAPI
        GetMyIncidents
        GetAllIncidents
        DeleteIncident
```

```
    end

    subgraph "External Services"
        GoogleMaps
        CloudStorage
    end

    subgraph "Database"
        SaveDB
    end

    subgraph "Authentication & Authorization"
        TokenGen
        RoleCheck
    end

    style Start fill:#4CAF50
    style SaveDB fill:#2196F3
    style GoogleMaps fill:#FF9800
    style CloudStorage fill:#FF9800
    style AuthSuccess fill:#4CAF50
    style RoleCheck fill:#9C27B0
```

Detailed Workflow Components

1. **User Authentication Flow**

```
sequenceDiagram
    participant User
    participant Frontend
    participant Auth API
    participant Database
    participant JWT

    User->>Frontend: Access Application
    Frontend->>User: Show Home/Login
    User->>Frontend: Submit Credentials
    Frontend->>Auth API: POST /auth/login
    Auth API->>Database: Verify Credentials
    Database->>Auth API: User Data
    Auth API->>JWT: Generate Token
    JWT->>Auth API: JWT Token
    Auth API->>Frontend: Token + User Info
    Frontend->>User: Redirect to Dashboard
```

2. **Incident Reporting Flow**

```
sequenceDiagram
    participant User
```

```
    participant Frontend
    participant Google Maps
    participant Cloudinary
    participant Incident API
    participant Database
    participant Email Service

    User->>Frontend: Navigate to Report Page
    Frontend->>Google Maps: Load Map
    Google Maps->>Frontend: Display Map
    User->>Frontend: Select Location & Fill Form
    User->>Frontend: Upload Images
    Frontend->>Cloudinary: Upload Images
    Cloudinary->>Frontend: Image URLs
    Frontend->>Incident API: POST /incidents
    Incident API->>Database: Save Incident
    Database->>Incident API: Incident Saved
    Incident API->>Email Service: Send Notifications
    Email Service->>User: Confirmation Email
    Incident API->>Frontend: Success Response
    Frontend->>User: Show Success Message
```

3. **Incident Management Flow (Responder/Admin)**

```
sequenceDiagram
    participant Responder
    participant Dashboard
    participant Incident API
    participant Database
    participant Live Map

    Responder->>Dashboard: Access Dashboard
    Dashboard->>Incident API: GET /incidents
    Incident API->>Database: Fetch Incidents
    Database->>Incident API: Incident Data
    Incident API->>Dashboard: Return Incidents
    Dashboard->>Live Map: Display on Map
    Dashboard->>Responder: Show Incidents
    Responder->>Dashboard: Update Incident Status
    Dashboard->>Incident API: PUT /incidents/:id
    Incident API->>Database: Update Status
    Database->>Incident API: Updated
    Incident API->>Dashboard: Success
    Dashboard->>Responder: Status Updated
```

4. **Analytics Flow (Admin Only)**

```
sequenceDiagram
    participant Admin
```

```
    participant Analytics Page
    participant Stats API
    participant Database

    Admin->>Analytics Page: Access Analytics
    Analytics Page->>Stats API: GET /incidents/stats/overview
    Stats API->>Database: Aggregate Data
    Database->>Stats API: Statistics
    Stats API->>Analytics Page: Stats Data
    Analytics Page->>Admin: Display Charts & Metrics
```

System Architecture

Frontend (React + TypeScript)

- **Pages**: Home, Report, Dashboard, Analytics, My Reports
- **Components**: Map, Header, Login Modal, Protected Routes
- **Services**: Auth, Incident, User APIs
- **Routing**: React Router with role-based access

Backend (Node.js + Express)

- **Controllers**: Auth, Incident, User, Distance, Places
- **Models**: User, Incident (MongoDB/Mongoose)
- **Middleware**: Authentication, Authorization, File Upload
- **Services**: Email notifications
- **Routes**: RESTful API endpoints

External Integrations

- **Google Maps API**: Location selection and visualization
- **Cloudinary**: Image storage and management
- **MongoDB**: Database for users and incidents

User Roles & Permissions

Role	Permissions
Guest	View map, Report incidents (optional auth)
Citizen	All guest + View my reports, Full reporting
Responder	All citizen + View dashboard, Update incident status
Admin	All responder + Analytics, Delete incidents, Manage users

API Endpoints Summary

Authentication

- `POST /api/auth/register` - Register new user

- `POST /api/auth/login` - Login user
- `GET /api/auth/me` - Get current user

Incidents

- `POST /api/incidents` - Create incident (optional auth)
- `GET /api/incidents` - Get all incidents
- `GET /api/incidents/my-reports` - Get user's incidents (protected)
- `GET /api/incidents/:id` - Get specific incident
- `PUT /api/incidents/:id` - Update incident (responder/admin)
- `DELETE /api/incidents/:id` - Delete incident (admin only)
- `GET /api/incidents/stats/overview` - Get statistics (protected)

Additional Services

- Distance calculation API
- Places/nearby resources API
- User management API

Data Flow

```
User Input → Frontend Validation → API Request →
Backend Validation → Database Operation → Response →
Frontend Update → User Notification
```

Key Features

1. ☑ Real-time incident mapping
2. ☑ Role-based access control
3. ☑ Image upload with Cloudinary
4. ☑ Google Maps integration
5. ☑ Email notifications
6. ☑ Analytics dashboard
7. ☑ Mobile responsive design
8. ☑ API rate limiting
9. ☑ JWT authentication