# Disaster Management App

## Overview

The Disaster Management app is a comprehensive solution for efficiently handling emergency incidents, coordinating response efforts among emergency responders, and ensuring timely communication with the affected population. This application is built on a microservices architecture to enhance modularity, scalability, and maintainability.

## Features

### User Management

- **User Registration and Authentication:**
    - Citizens and Emergency Responders can register with the system by providing necessary details.
    - Token-based authentication ensures secure access.

### Incident Reporting

- **Citizen Incident Reporting:**
    - Citizens can report incidents by providing a title, description, and location.
    - Reporting Microservice stores and manages incident details.

### Alerting

- **Automatic Alert Generation:**
    - Alerts are automatically generated based on the severity of reported incidents.
    - Alerting Microservice dispatches alerts to relevant Emergency Responders.

### Resource Management

- **Resource Registration and Allocation:**
    - Emergency Responders can register available resources (vehicles, equipment).
    - Resources are allocated based on incident type and severity.

### Emergency Response

- **Real-time Updates:**
    - Emergency Responders provide real-time updates on their status and actions.
    - Coordination and communication tools enhance effective response.

**Historical Data Logging**

- **Data Logging and Analysis:**
  - Historical Data Microservice logs data related to incidents and emergency responses.
  - Enables historical analysis for improvement and reporting.

**Communication**

- **Communication Channels Setup:**
  - Communication Microservice sets up channels for public alerts and responder communication.
  - Enables real-time communication among Emergency Responders.

## System Architecture

The Disaster Management app is structured as a set of interconnected microservices:

1. **User Management Microservice:**
   - Handles user registration and authentication.
2. **Reporting Microservice:**
   - Manages incident reporting and severity determination.
3. **Alerting Microservice:**
   - Generates and dispatches alerts to Emergency Responders.
4. **Resource Management Microservice:**
   - Handles the registration, allocation, and status updates of resources.
5. **Emergency Response Microservice:**
   - Facilitates real-time updates, coordination, and communication among Emergency Responders.
6. **Historical Data Microservice:**
   - Logs data for historical analysis and reporting.
7. **Communication Microservice:**
   - Manages communication channels and facilitates public and private communication.

## Getting Started

To get started with the Disaster Management app, follow these steps:

1. Clone the repository.

   ```
   git clone https://github.com/tomdieu/disasterFlow.git
   ```

2. Install dependencies for each microservice.

3. Configure environment variables.

   Update the `.env` file with your configuration.

4. Run each microservice.

For more detailed instructions, refer to the Getting Started Guide.

## Usage

**Citizen Interaction**

1. **Report an Incident:**
   - Access the Reporting interface.
   - Provide incident details (location,optional image).
   - Submit the report.

**Emergency Responder Interaction**

1. **Receive and Respond to Alerts:**
   - Receive alerts from the Alerting Microservice.
   - Update status and actions through the Emergency Response interface.

## API Documentation

For detailed API documentation, refer to the API Documentation.

## Contributing

If you would like to contribute to the development of the Disaster Management app, please follow the Contribution Guidelines.

## License

This project is licensed under the MIT License.