

ResQAI – Intelligent Emergency Resource Allocation and Crisis Response Using Artificial Intelligence

Dr. B. Meena Preethi, Aswath.T

¹Associate Professor, Department of Software System, Sri Krishna Knowledge Foundation, Deemed to Be University, Coimbatore, India

²Student, Department of Software System, Sri Krishna Knowledge Foundation, Deemed to Be University, Coimbatore, India.

meenapreethib@skasc.ac.in, aswatht23mss005@skasc.ac.in

Abstract

During an emergency, getting the right help at the right time is very important. However, many existing emergency reporting methods mainly depend on manual communication, which can cause delays in identifying the type and severity of an emergency and assigning suitable resources. This paper presents ResQAI, an intelligent emergency response application designed to improve the process of reporting, analyzing, and responding to emergency situations.

ResQAI uses artificial intelligence to identify the emergency category from the user's description and predict its severity. Based on the emergency type, severity, location, and availability of resources, the system helps assign suitable resources and volunteers. The application also uses GPS to capture the emergency location and Firebase Cloud Messaging (FCM) to send notifications to the concerned users and organizations. A voice-based reporting option allows users to describe an emergency using speech, which is converted into text before submitting the report. To support situations where internet connectivity is unavailable, ResQAI stores emergency requests locally using SQLite and synchronizes them with the server when connectivity is restored.

The proposed approach combines artificial intelligence, location services, mobile technology, and offline synchronization in a single emergency response platform. The system is intended to reduce manual effort, improve response coordination, and provide a more organized approach to emergency resource allocation.

Keywords: Artificial Intelligence, Emergency Response, Resource Allocation, Severity Prediction, GPS, Voice Recognition, Offline Synchronization, Firebase Cloud Messaging.

I. INTRODUCTION

Emergency situations such as floods, fires, accidents, medical emergencies, and other disasters require quick and coordinated action. In many situations, people report emergencies through phone calls or messages, and the responsible organization has to manually understand the situation, determine its severity, and decide which resources should be sent. This process can take additional time, especially when multiple emergency requests are received at the same time.

The availability of smartphones and artificial intelligence provides an opportunity to improve this process. A mobile application can collect important information such as the emergency description and location, while an AI model can help identify the emergency category and severity. This information can then be used to support the assignment of appropriate resources and volunteers.

The main objective of **ResQAI** is to provide a single platform for emergency reporting and response coordination. The application allows users to report emergencies using either text or voice. The reported description is processed

by AI to identify the emergency category. A separate severity prediction model is used to determine the level of the emergency. GPS coordinates are collected to identify the location of the incident.

Based on the emergency details and available resources, the system supports intelligent resource assignment. Volunteers can also be assigned according to their skills and availability. Notifications are delivered through Firebase Cloud Messaging so that relevant users can receive updates about the emergency and assignment.

Another important feature of ResQAI is offline emergency reporting. Emergency situations may occur in locations where internet connectivity is weak or temporarily unavailable. Therefore, the application stores the emergency request locally and synchronizes it with the server after the network connection becomes available.

The proposed system combines these features to provide a practical approach to emergency reporting and response management.

II. RELATED WORK

Several emergency management applications and research approaches have been developed to improve communication during disasters. Traditional emergency systems generally depend on telephone communication and manual coordination. Although these methods are useful, they may not provide structured information about the emergency or automatically connect the request with suitable resources.

Recent research has explored the use of machine learning and natural language processing for classifying emergency-related text. Text classification can help identify whether a reported situation is related to a medical emergency, accident, fire, flood, or other category. Machine learning techniques can also be used to estimate the severity of an incident based on the information provided by the user.

Location-based emergency applications have also been developed using GPS technologies. These systems can provide the location of an incident to emergency organizations and help responders identify nearby resources. However, some applications focus mainly on reporting and location sharing without combining AI-based classification, severity prediction, resource allocation, volunteer management, and offline synchronization in one platform.

ResQAI attempts to address these requirements by combining AI-based analysis with location services, resource assignment, notifications, and offline support.

TABLE I — FEATURE COMPARISON

FEATURE	TRADITIONAL APPROACH	RESQAI
Emergency Reporting	Manual	Digital
Emergency Classification	Manual	AI-assisted
Severity Prediction	Manual	AI-assisted
GPS Location	Limited	Supported
Resource Assignment	Manual	Intelligent matching
Volunteer Assignment	Manual	Skill-based
Voice Reporting	Not available	Supported
Offline Reporting	Limited	SQLite-based
Notifications	Phone/SMS	FCM

III. PROBLEM STATEMENT

During emergency situations, several problems can occur in the existing reporting and coordination process.

- ❖ First, emergency information may be provided in an unstructured form. The responsible organization may need to manually read the description and determine the type of emergency.
- ❖ Second, the severity of an emergency may not be immediately clear. Without proper classification, highly critical cases may not receive appropriate priority.

- ❖ Third, assigning resources manually can consume time. Different emergencies require different resources, such as ambulances, fire trucks, shelters, food, blood, or medicines.
- ❖ Fourth, volunteer assignment may not always consider the skills required for a particular emergency.
- ❖ Finally, internet connectivity cannot always be guaranteed during disasters. An emergency application that depends completely on a continuous internet connection may not be useful in areas affected by network problems.

ResQAI addresses these challenges by providing AI-assisted classification and severity prediction, location-based resource assignment, volunteer assignment, real-time notifications, and offline synchronization.

IV. PROPOSED METHODOLOGY

ResQAI follows a sequence of steps from emergency reporting to response coordination.

When a user reports an emergency, the application collects the emergency description and current GPS location. The user can either type the description or use the voice reporting feature. In voice reporting, the user's speech is converted into text, and the user can review or edit the generated text before submitting the emergency.

The emergency description is sent to the backend for processing. The AI module predicts the emergency category and severity. The system then uses the emergency type, location, resource availability, and other assignment conditions to identify suitable resources.

The selected resource can be assigned to the emergency, and the concerned organization receives a notification. Where volunteer assistance is required, suitable volunteers can also be assigned based on their skills and availability.

The overall process can be represented as:

Emergency Reporting → Text/Voice Input → AI Classification → Severity Prediction → GPS Location → Resource Matching → Volunteer Assignment → Notification → Response Tracking

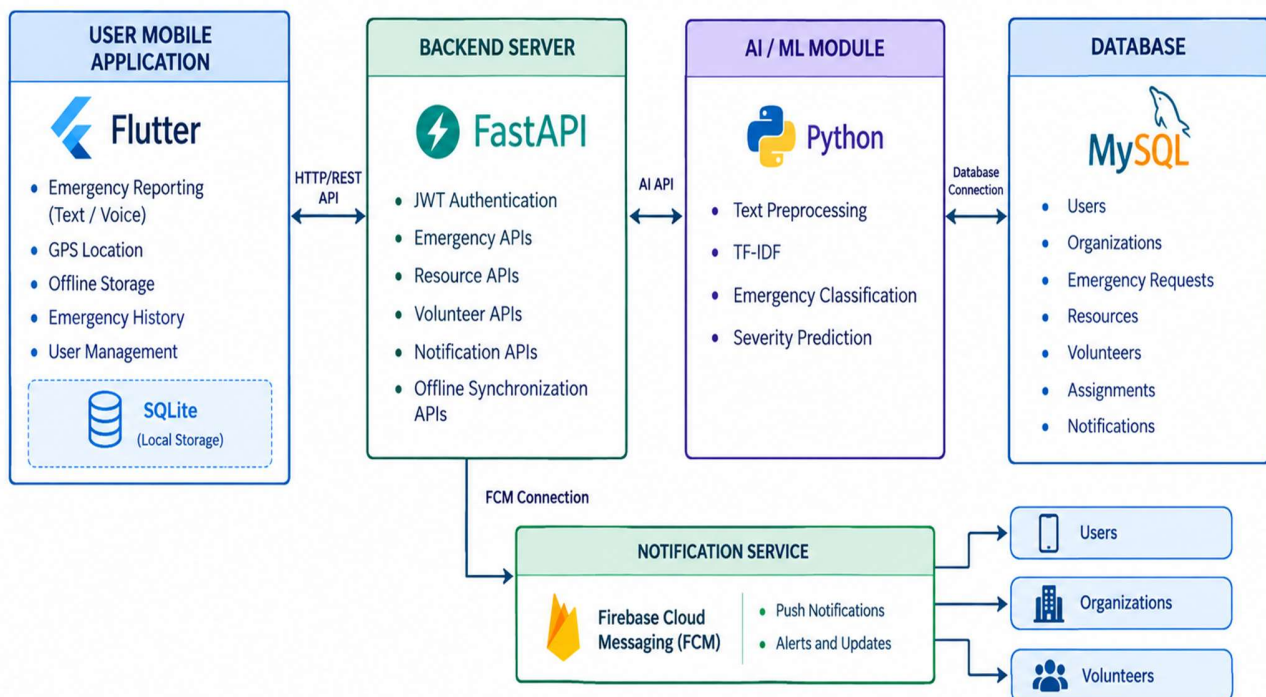


Fig. 1. ResQAI system architecture

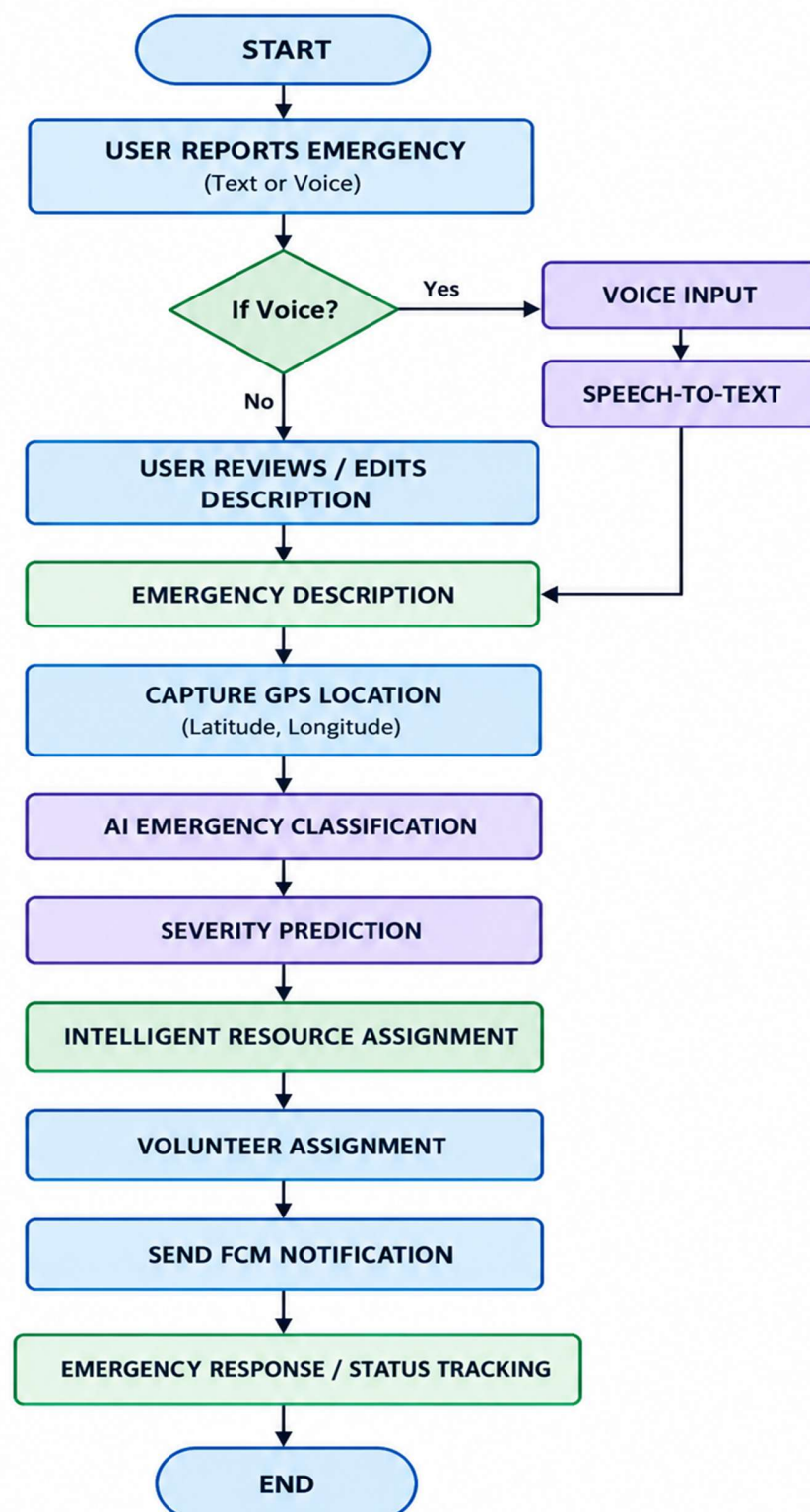


Fig. 2. Emergency reporting and AI prediction workflow

If the device is offline, the emergency request is temporarily stored in SQLite. Once connectivity is restored, the stored request is synchronized with the backend and processed normally.

V. SYSTEM ARCHITECTURE

The ResQAI architecture consists of four major layers: the mobile application, backend services, AI processing, and database/notification services.

The mobile application is developed using Flutter. It provides interfaces for emergency reporting, emergency history, profile management, resource information, volunteer-related operations, and offline emergency reporting. The backend is developed using FastAPI. It provides REST APIs for authentication, emergency requests, resources, assignments, volunteers, notifications, and offline synchronization.

The AI component is implemented using Python-based machine learning techniques. The emergency description is processed using text-based features, including TF-IDF, and passed to the classification model. Severity prediction is handled separately using the trained severity model.

MySQL is used for storing application data such as users, emergencies, resources, assignments, organizations, and notifications. SQLite is used on the mobile device to temporarily store emergency requests when the device is offline.

Firebase Cloud Messaging is used to deliver notifications to relevant users.

VI. AI-BASED EMERGENCY CLASSIFICATION

One of the important features of ResQAI is automatic identification of the emergency category.

The user provides a description such as:

"Heavy flood water has entered the house and people need rescue team."

Instead of requiring the user to manually select the category, the system processes the description and predicts the appropriate emergency type.

Text preprocessing is performed before classification. The text is converted into numerical features using **Term Frequency-Inverse Document Frequency (TF-IDF)**. TF-IDF helps represent important words in the emergency description while reducing the influence of frequently occurring words.

A machine learning classification model is then used to predict the emergency category.

The predicted category can be used for further processing, including resource matching. For example, a medical-related emergency may require an ambulance, while a fire emergency may require a fire truck.

This approach reduces the amount of manual classification required from emergency coordinators.

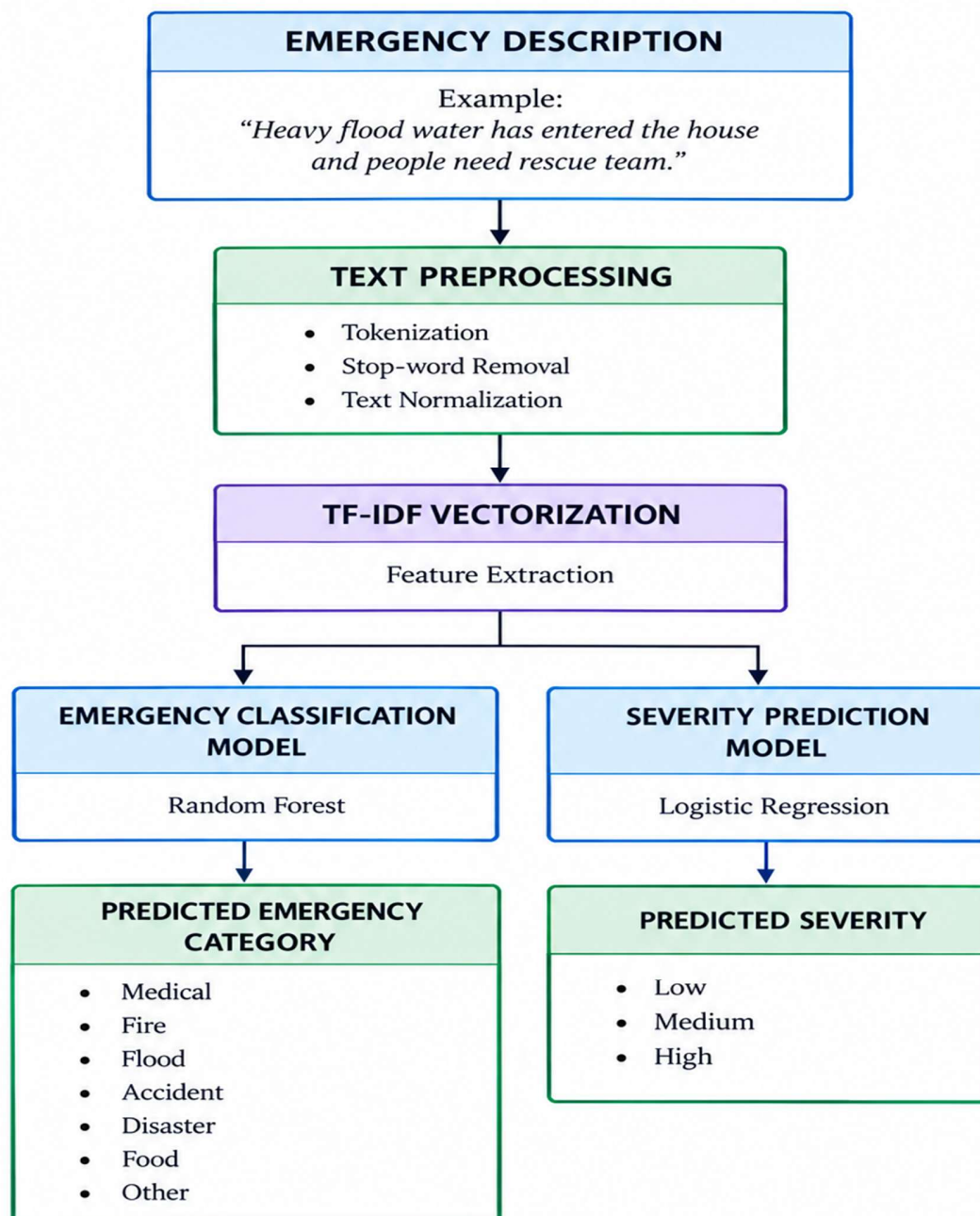


Fig. 3. AI-based emergency classification and severity prediction

Table II — Emergency Type and Resource Mapping

EMERGENCY TYPE	SUITABLE RESOURCE
Medical	Ambulance
Accident	Ambulance
Injury	Ambulance
Fire	Fire Truck
Flood	Shelter / Rescue Resource
Disaster	Shelter / Rescue Resource

Evacuation	Shelter
Food Emergency	Food Resource
Blood Requirement	Blood Resource
Medicine Requirement	Medicine Resource

VII. SEVERITY PREDICTION

Identifying the type of emergency alone is not sufficient because two emergencies belonging to the same category may have different levels of seriousness.

ResQAI therefore includes a separate severity prediction component. The system analyzes the emergency information and predicts a severity level such as **Low, Medium, or High**.

The severity model uses the emergency information provided by the user. The prediction helps the response team understand the urgency of the request and supports further decision-making.

The use of a separate severity model also allows emergency classification and severity estimation to be handled as two different tasks. This makes the system easier to maintain and improve independently.

VIII. INTELLIGENT RESOURCE AND VOLUNTEER ASSIGNMENT

After identifying the emergency type, ResQAI helps determine an appropriate resource.

The resource assignment process considers factors such as:

- Emergency type
- Resource type
- Resource availability
- Resource status
- Quantity
- Location of the resource
- Distance from the emergency location

For example, medical emergencies can be matched with ambulances, fire-related emergencies with fire trucks, and flood or evacuation situations with suitable shelters or rescue resources.

The system also supports volunteer assignment. Volunteers are considered based on their skills and availability. This is useful because different emergencies require different types of assistance. For example, a volunteer with medical-related skills may be more suitable for a medical emergency than a volunteer without those skills.

This approach reduces unnecessary manual searching and provides structured support for assigning available resources and volunteers.

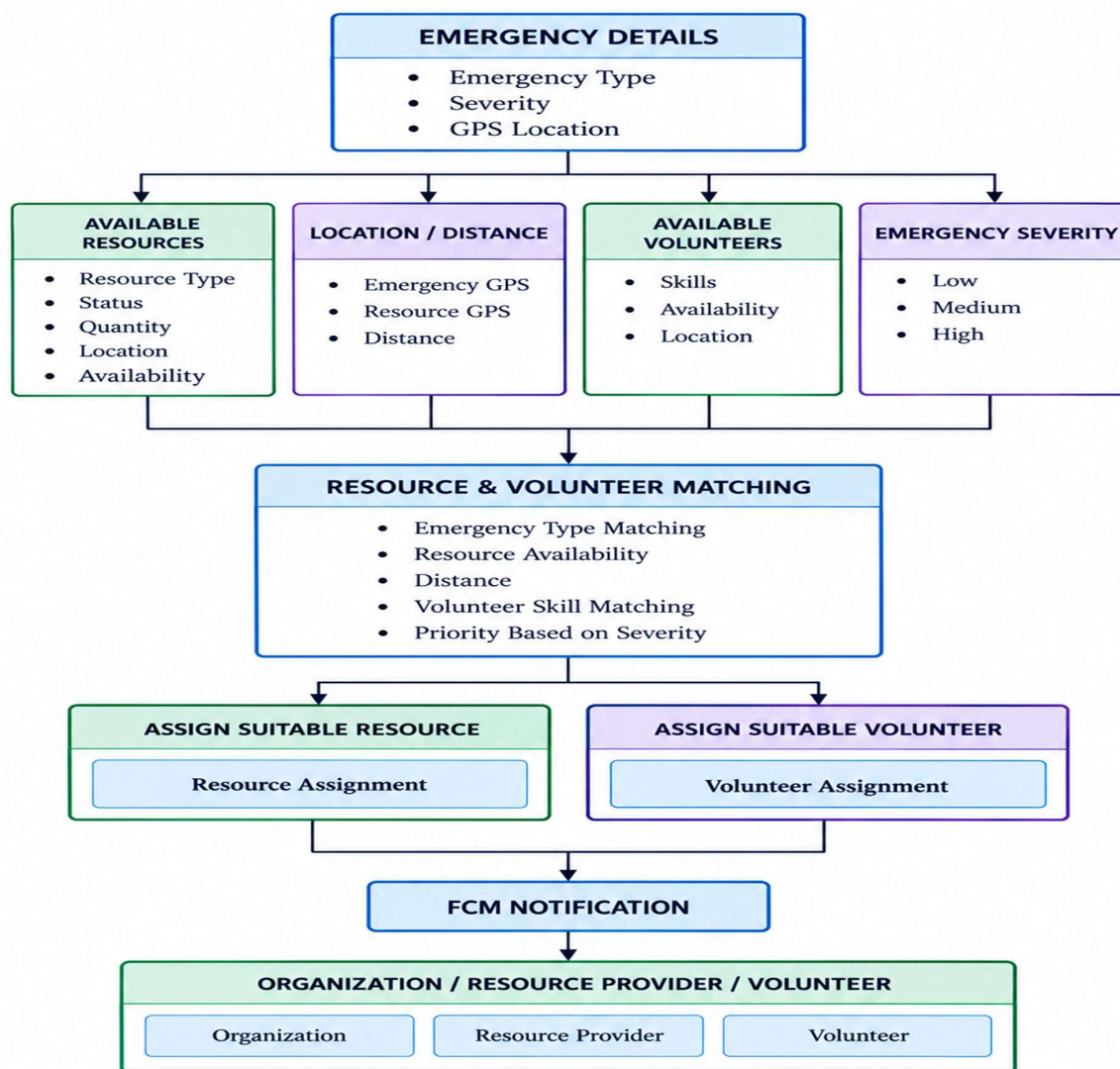


Fig. 4. Intelligent resource and volunteer assignment process

IX. VOICE-BASED EMERGENCY REPORTING

During an emergency, typing a detailed description may not always be convenient. To address this situation, ResQAI provides a voice-based emergency reporting option.

The user can speak about the emergency through the mobile application. The speech is converted into text using speech recognition. The generated text is displayed in the emergency description field so that the user can check and edit it before submitting the request.

The process is:

Speak → Speech-to-Text → Review/Edit → Report Emergency → AI Processing

The voice feature does not replace the existing text-based reporting method. Users can continue to type their emergency description when they prefer to do so.

X. OFFLINE EMERGENCY REPORTING AND SYNCHRONIZATION

Internet connectivity can become unreliable during disasters. Therefore, ResQAI provides an offline reporting mechanism.

When the device is not connected to the internet, the emergency request is stored locally using SQLite. A unique identifier is generated for the offline request so that it can be tracked during synchronization.

When network connectivity becomes available again, the application sends the pending requests to the backend. The backend processes the synchronized emergency request and performs the required AI prediction and assignment operations.

The synchronization process can be summarized as:

Offline Request → SQLite Storage → Network Restored → Server Synchronization → AI Processing → Resource/Volunteer Assignment → Notification

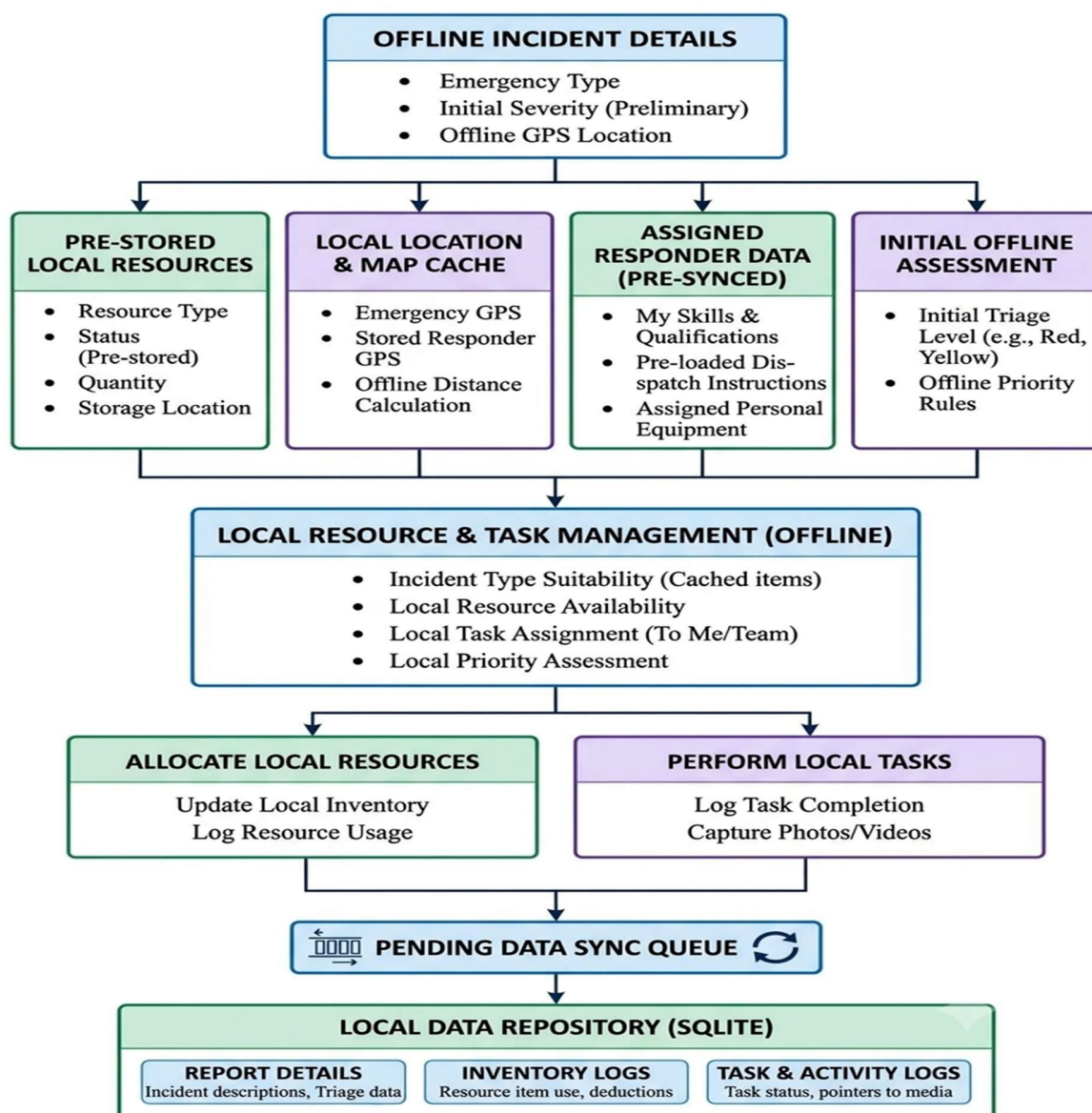


Fig. 5. Offline emergency reporting and synchronization workflow

This feature allows users to record emergency information even when a stable internet connection is temporarily unavailable.

XI. REAL-TIME LOCATION AND NOTIFICATION SERVICES

GPS plays an important role in ResQAI. When an emergency is reported, the application collects the user's latitude and longitude. The location information is stored along with the emergency request.

Location information helps the system identify resources that are geographically suitable for the emergency. It also provides the response organization with information about where the incident occurred.

Firebase Cloud Messaging is used for notifications. Notifications can be sent to relevant users, organizations, and volunteers based on the assignment.

This allows important updates to reach the concerned parties without requiring them to continuously check the application.

XII. IMPLEMENTATION

The ResQAI mobile application was developed using **Flutter**, which provides a cross-platform framework for building the user interface.

The backend was implemented using **FastAPI**, providing REST-based APIs for communication between the mobile application and server.

MySQL was used as the primary database for persistent application data. **SQLAlchemy** was used for database interaction on the backend.

The AI components were implemented using Python machine learning libraries. TF-IDF was used for converting text into numerical features, while machine learning models were used for emergency classification and severity prediction.

The mobile application uses **SQLite** for local offline storage. Network connectivity is monitored so that pending emergency requests can be synchronized when the connection becomes available.

Firebase Cloud Messaging was integrated for delivering notifications.

The main technologies used are shown below.

TABLE III — TECHNOLOGIES USED IN RESQAI

COMPONENT	TECHNOLOGY
Mobile Application	Flutter
Backend	FastAPI
Programming Language	Python
Database	MySQL
Local Storage	SQLite
ORM	SQLAlchemy
AI/NLP	TF-IDF and Machine Learning
Notifications	Firebase Cloud Messaging
Location	GPS
Authentication	JWT

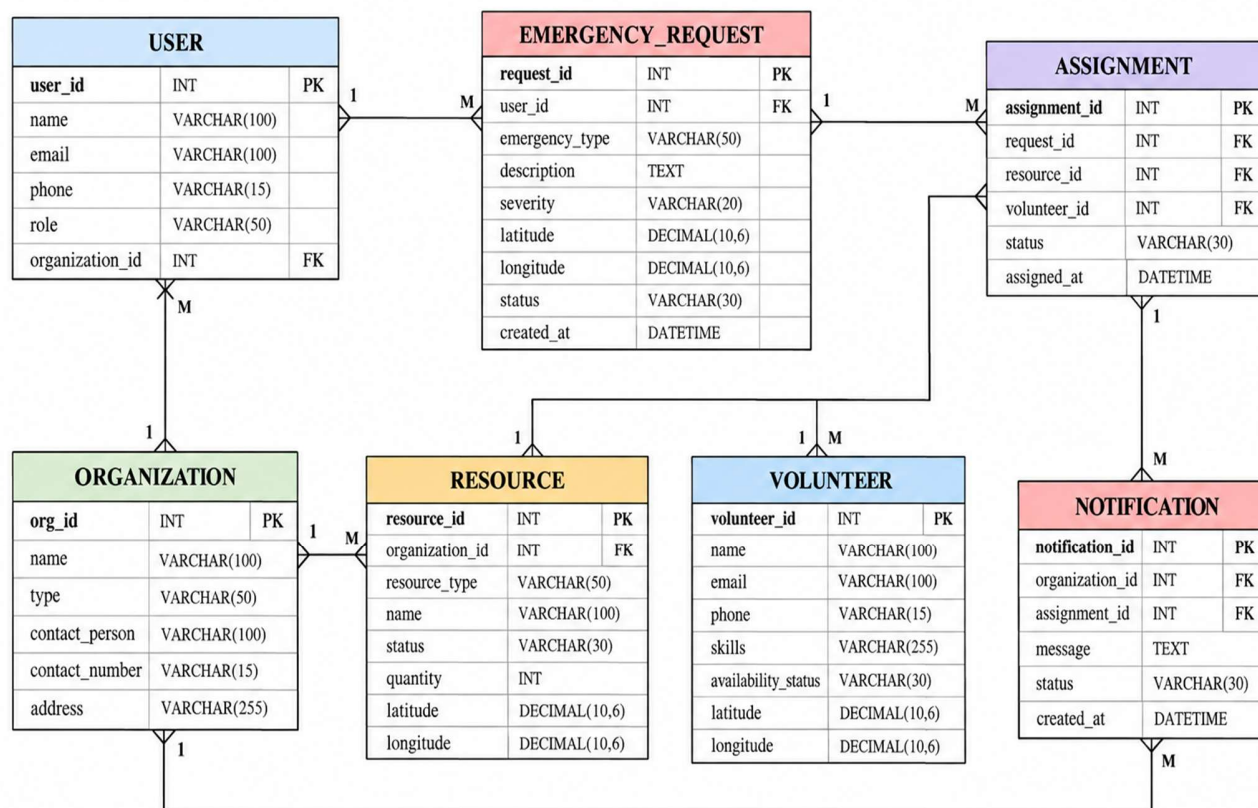


Fig. 6. ResQAI mobile application interfaces

XIII. RESULTS AND DISCUSSION

The implemented ResQAI application was tested for the major functions involved in emergency reporting and response coordination.

The emergency reporting module successfully accepts both typed and voice-generated descriptions. The AI module processes the emergency description and produces an emergency category. The severity module provides a severity prediction that can be used during response coordination.

The resource assignment module matches emergency requests with suitable resources based on emergency type, resource status, availability, and location-related information. Volunteer assignment also considers the skills required for the emergency.

The offline module was tested by storing emergency requests locally when the device had no active network connection. After network connectivity was restored, the pending requests were synchronized with the backend. The synchronized requests were then available for further processing.

FCM notifications were also integrated to communicate assignment and emergency-related information to the concerned users.

These results show that the proposed application can combine emergency reporting, AI-based analysis, resource coordination, and offline support within a single platform.

XIV. ADVANTAGES AND LIMITATIONS

Advantages

The main advantages of ResQAI include:

1. Automatic emergency category prediction.
2. AI-assisted severity prediction.
3. GPS-based emergency location.
4. Intelligent resource matching.
5. Skill-based volunteer assignment.
6. Voice-based emergency reporting.
7. Offline emergency storage and synchronization.
8. Real-time notifications using FCM.
9. Centralized emergency and assignment management.

Limitations

The current implementation also has some limitations. The accuracy of AI predictions depends on the quality and diversity of the training data. GPS accuracy may vary depending on the environment and device. Resource availability also depends on the information maintained by the organization.

Offline synchronization requires the device to reconnect to the network before the stored emergency can be processed by the server. In addition, the current AI models can be further improved using larger and more diverse emergency datasets.

XV. FUTURE ENHANCEMENT

The system can be improved in several ways in the future. More emergency-related datasets can be collected and used to improve the classification and severity prediction models.

A more advanced AI model can be explored for handling complex emergency descriptions. The system can also be extended with multilingual voice input so that users can report emergencies in different regional languages.

Future versions can include more advanced resource optimization methods that consider multiple emergency requests simultaneously. Predictive analysis could also be used to estimate resource requirements during large-scale disasters.

The notification system can be further improved by introducing priority-based alerts for critical emergencies. Additional offline capabilities can also be developed so that more application features remain available during network failures.

XVI. CONCLUSION

ResQAI was developed to provide a more organized approach to emergency reporting and response coordination. The system combines artificial intelligence, mobile technology, GPS, notification services, and offline data synchronization to support the emergency response process.

The application allows users to report emergencies using text or voice. AI-based processing helps identify the emergency category and predict its severity. The system then supports the assignment of suitable resources and volunteers using emergency information, availability, skills, and location.

The offline reporting feature is particularly useful in situations where network connectivity is temporarily unavailable. Emergency requests can be stored locally and synchronized with the backend when connectivity is restored.

Overall, ResQAI demonstrates how AI and mobile technologies can be combined to support emergency management. The current implementation provides a foundation that can be further improved with larger datasets, more advanced AI models, multilingual support, and optimized resource allocation techniques.