AI App: An AI-Assisted Mobile Platform for Emergency Reporting and Decision Support

Safwan Abd Razak[1], Fazrina Majhardeen[2], Muhammad Fuad Abdullah[3], Muhammad Sharilazlan Salleh[4], Muhammad Faheem Mohd Ezani[5]

Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, Durian Tunggal, 76100 Melaka, Malaysia

DOI: https://doi.org/10.47772/IJRISS.2026.101000358

Received: 17 October 2026; Accepted: 22 October 2026; Published: 03 November 2026

ABSTRACT

Emergencies require rapid and accurate communication to support effective response. However, conventional emergency reporting often relies on users manually describing the incident, which may result in incomplete information or reporting delays when users are under stress. This study presents the development and evaluation of the Emergency AI App. This mobile application uses Artificial Intelligence (AI), image validation, GPS location services, emergency guidance, and cloud-based data management to support emergency reporting and decision-making. The application was developed using Flutter and Dart, with Firebase Authentication and Cloud Firestore supporting user authentication and data management. A three-class TensorFlow Lite model was deployed locally on the mobile device to validate uploaded images as AI-generated, non-emergency, or real emergency images. Validated emergency images can subsequently be analysed using Google Gemini AI to provide emergency category, severity, incident summary, suggested assistance, and safety guidance. The system was developed iteratively using the Agile methodology and evaluated through functional testing, AI model evaluation, and User Acceptance Testing (UAT) with usability assessment. Results showed that all 29 defined functional test cases passed after corrective testing. The TensorFlow Lite model achieved an overall accuracy of 82.56% across 86 independent test images, with macro-average precision, recall, and F1-score of 83.43%, 82.94%, and 82.31%, respectively. However, recall for the real-emergency class was 67%, indicating limitations in detecting some genuine emergency images. UAT involving 11 smartphone users recorded 132 task attempts, all of which were eventually completed, although 33 required assistance. The overall usability mean was 4.79 out of 5. The findings demonstrate the feasibility of integrating local image validation, online AI analysis, emergency reporting, GPS support, and emergency guidance within a single mobile platform. Nevertheless, the system requires larger, more diverse datasets; broader usability evaluation; improved offline capabilities; and stronger security controls before it can be considered for real-world emergency deployment.

Keywords: Artificial Intelligence, Emergency Reporting, Mobile Application, Emergency Decision Support, Usability Evaluation

INTRODUCTION

Emergencies such as road accidents, fires, crimes, and medical emergencies require timely communication and accurate information to support effective response. Road traffic crashes alone claim more than 1.35 million lives and cause up to 50 million injuries each year, which makes improving the efficiency of emergency management systems a key measure for reducing deaths and injuries [1]. The quality and completeness of information provided during the initial reporting stage can influence subsequent emergency-response activities, and mobile-device applications have been proposed specifically as a means of improving data-collection accuracy and emergency response time [2]. However, conventional emergency reporting methods often rely on users to manually report incidents through phone calls or text messages; traditional emergency systems frequently face delays, inaccurate tracking, and limited real-time risk

assessment [3]. During an emergency, victims or witnesses may experience panic or confusion, making it difficult to accurately communicate the emergency type, severity, and location.

The increasing capabilities of smartphones and Artificial Intelligence provide opportunities to improve this process. Modern smartphones integrate cameras, GPS, Internet connectivity, and other device services that can facilitate the collection and transmission of emergency information. AI technologies can further support the analysis of collected information and provide decision-support information to users. Machine learning algorithms are highly suitable for handling the large volumes of spatiotemporal data generated during emergency situations and have been used to aid first responders and decision-makers across the prevention, preparedness, response, and recovery phases [4].

Existing emergency applications commonly provide functions such as SOS alerts, emergency contact notifications, direct emergency calls, and location sharing. A review of mobile-based emergency intervention apps found that although a growing body of knowledge addresses usability, safety, and privacy, studies that map emergency intervention strategies and propose assessment indicators for evaluating such apps remain limited [5]. While these functions are useful for communication, they generally require users to determine and communicate the emergency situation manually. The project review identified a need for an emergency-support platform capable of combining reporting, image validation, AI analysis, location information, safety guidance, and additional assistance features.

To address this gap, this study developed the Emergency AI App, a mobile application that integrates manual emergency reporting with AI-assisted image processing and emergency decision support. The application incorporates an offline TensorFlow Lite image-validation model, optional Google Gemini AI analysis, GPS location support, Firebase-based cloud services, emergency guidance, and local bystander-assistance functions. The study focuses on three main objectives:

1. To develop a mobile application that supports emergency reporting using image submission and GPS location information.

2. To implement an AI-assisted image-validation mechanism within a mobile environment.

3. To provide emergency decision-support information including emergency classification, severity information, suggested assistance, and safety guidance.

The contribution of this study is therefore not the development of a novel AI algorithm, but the integration and empirical evaluation of multiple technologies within a single emergency-support mobile platform, an integrative approach consistent with recent AI-driven emergency response applications that combine risk categorisation, live GPS tracking, smart routing, and first-aid guidance [3].

LITERATURE REVIEW

Mobile Technologies for Emergency Management

Mobile technologies have increasingly been applied to emergency and disaster-management activities. Smartphones provide communication, GPS, camera, and Internet capabilities that can support the collection and sharing of emergency information.

Aisyah et al. [6], as reviewed in the project, examined mobile-enabled technologies used in disaster management and reported the increasing adoption of mobile applications across prevention, preparedness, response, and recovery activities. GPS-based location services were identified as an important capability for providing incident-location information. Location accuracy remains a central technical consideration for such services: smartphone GPS has been shown to achieve average worldwide accuracy of 2 m ± 3 m, whereas Wi-Fi positioning averages 7 m ± 24 m and location-based-service methods average 214 m ± 321 m [7]. Accuracy degrades considerably in GNSS-denied environments such as urban canyons and indoor

settings, which has driven regulatory requirements such as the demand for 50 m horizontal accuracy for 80% of indoor emergency calls [8].

Similarly, Devi et al. [9] proposed a Flutter-based SOS application incorporating real-time location tracking and volunteer assistance. Their work demonstrates the potential of combining mobile applications, GPS, cloud services, and emergency communication within a single system.

Beyond alerting, mobile and crowdsourced approaches have been used to collect incident reports and photographs from citizens to build situational awareness. Dedicated applications can capture reports, photographs, and videos that are then mapped to identify areas where structures or critical infrastructure are most at risk, allowing response resources to be directed before personnel are deployed into unsafe areas [10]. Frameworks such as Public Engagement in Emergency Response provide online and mobile crowdsourcing platforms for situation reporting and resource volunteering [11], while applications such as Crowdalert have been developed specifically to gather and publicise volunteered disaster-related information [12].

Machine learning has also been applied to the analysis of emergency data. Kyrkou et al. [4] surveyed machine-learning applications across all emergency management phases and concluded that such algorithms can provide a unified algorithmic framework for a large set of emergency problems. In a related direction, El Barachi et al. [1] proposed a crowdsensing-based solution that employs AI for the automatic analysis of accident scene data, generating accident intelligence reports to support fast and accurate situational assessment by police officers and rescue teams. These studies demonstrate the value of mobile platforms for emergency communication and coordination. However, their primary focus remains emergency alerting, location sharing, and responder coordination rather than integrated AI-based image validation and emergency decision support

Existing Emergency Applications

Several existing emergency applications were reviewed in the original project.

Table 1. Emergency Applications and their Purposes.

Emergency Application	Purpose
SaveME 999	Focuses on emergency alerts and location sharing but does not provide AI-based emergency detection or classification.
SOS Alert Widget App	Provides rapid-access emergency messages and location information, but users remain responsible for determining the emergency.
SOS Alert Basic App	Provides emergency messaging and contact notification but does not provide AI-assisted reporting or emergency recommendations.
Emergency Call App	Provides rapid access to emergency-service telephone numbers but does not provide integrated emergency reporting, location tracking, or intelligent decision support.
Call For Help	Emergency SOS provides SOS alerts, location sharing, and emergency-contact notification.

The comparison suggests that existing applications generally focus on communication and alerting, whereas the proposed system attempts to combine communication support with image validation, AI analysis, emergency guidance, reporting, and bystander assistance. This distinction is consistent with assessments of mobile emergency apps, which note that although a growing number of applications support emergency intervention, systematic indicators for evaluating their role as emergency agents remain limited [5].

Table 2. Functional Positioning of Existing and Proposed Systems

Capability	Existing Emergency Apps	Proposed Emergency AI App
SOS / emergency reporting	✓	✓
GPS location support	✓	✓

Manual emergency reporting	Limited	✓
Image submission	Limited	✓
Local image validation	-	✓
AI-assisted image analysis	-	✓
Emergency severity information	Limited	✓
Emergency guidance	Limited	✓
Report history/status	Limited	✓
Bystander assistance	-	✓
Guest emergency access	-	✓
Bilingual support	-	✓

The proposed system therefore addresses a broader combination of emergency-support functions rather than replacing existing emergency-service mechanisms.

Proposed System

System Overview

The Emergency AI App was designed as a mobile emergency-reporting and decision-support platform. The application provides two main user pathways: guest users and registered users.

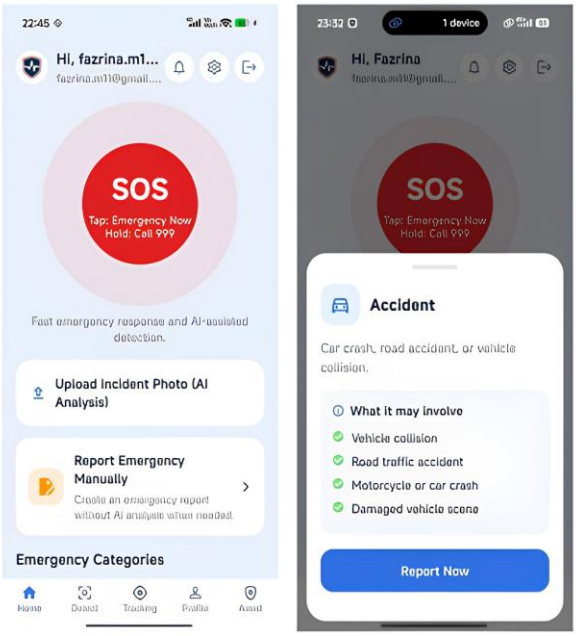

Figure 1. Home Screen

Guest users can access basic emergency functions without registration. Registered users have access to additional features including AI Analysis, emergency-profile management, report history, notifications, report status, application settings, and Bystander Assist tools.

The architecture combines local device processing with cloud-based services. Local TensorFlow Lite inference provides offline image validation, while Firebase and Gemini AI provide online services when connectivity is available.

Hybrid AI Architecture

One of the key characteristics of the proposed system is its hybrid AI architecture. The first stage uses TensorFlow Lite to perform local image validation, classifying uploaded images as AI-generated, non-

emergency, or real emergencies. TensorFlow Lite is a collection of tools and libraries designed for inference in low-resource environments, providing an interpreter with optimised operation kernels together with a converter that produces a compact FlatBuffer model suitable for mobile deployment [13]. Because the model runs locally, this validation stage does not require continuous Internet connectivity. On-device inference avoids transmitting data to a server, which reduces bandwidth consumption and latency while providing privacy protection [14]. TensorFlow Lite has been shown to support real-time inference on constrained smartphone platforms, including through GPU and NNAPI delegation that improves inference time and resource consumption [15], and post-training quantisation can further reduce the model footprint for memory-constrained devices [16].

The validation stage classifies uploaded images as AI-generated, non-emergency, or real emergencies. Distinguishing AI-generated from authentic photographs has become an active research problem, since synthetic images have reached a level of realism that human observers cannot reliably detect [17], and a substantial body of work now addresses the detection, disruption, and authentication of AI-generated visual media [18].

Images that satisfy the validation requirements can subsequently be processed by Google Gemini AI. Gemini AI provides higher-level decision-support information, including emergency category, severity level, incident summary, suggested assistance, and safety guidance. Recent work has shown that large language models can generate valid plans of action for disaster response within seconds when response and planning guidelines are incorporated into the prompt [19], and that they can support real-time information processing, risk assessment and prediction, and decision-support systems during emergencies [20]. Retrieval-augmented generation combined with LLMs has also been applied to generate emergency plans that incorporate region-specific knowledge [21], while LLM-based interventions have been examined for early warning systems, situational awareness, and emergency call triage in crisis scenarios [22].

This architecture separates image validation from higher-level AI analysis, reducing the dependency of the entire reporting process on a single online AI service.

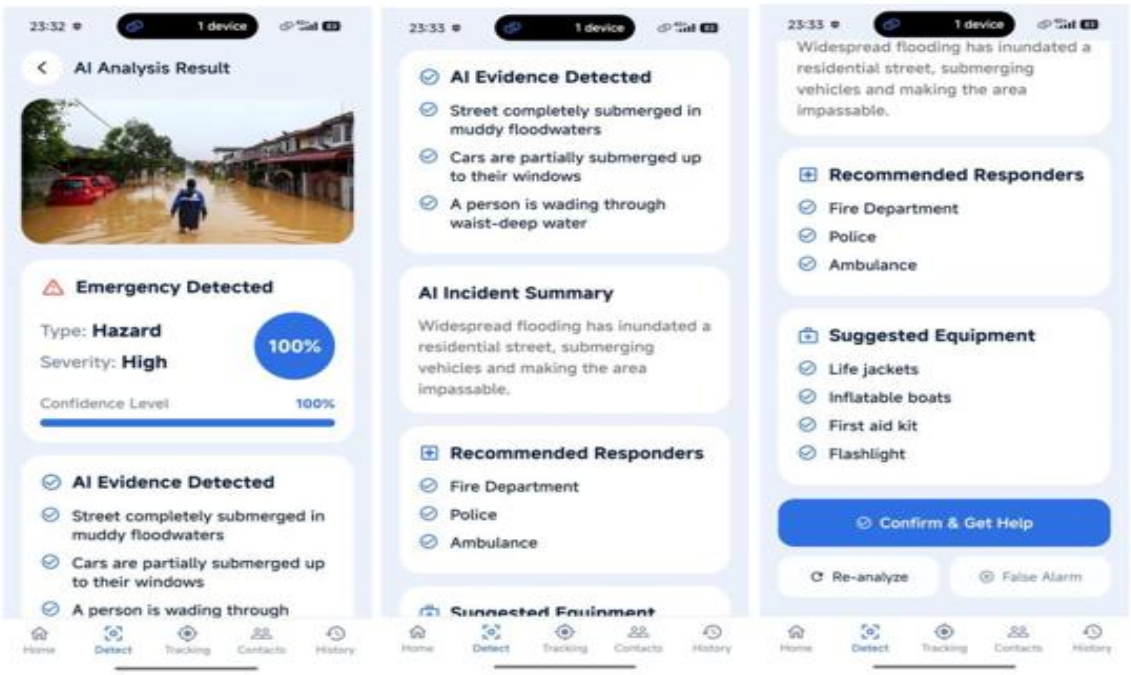

Figure 2. AI Analysis

Manual Reporting and Emergency Guidance

The application does not depend exclusively on AI analysis. Users can also submit emergency information manually, including emergency type, victim condition, number of people involved, danger condition,

description, and location-sharing preference. This design reflects the finding that structured, mobile-assisted data collection can improve the accuracy of emergency information and reduce response time compared with unstructured reporting channels [2].

This provides an alternative pathway when an image is unavailable, the image does not pass validation, Gemini AI is unavailable, or the user prefers manual reporting. Emergency guidance is then provided based on the available emergency information. This design is important because it prevents the application from becoming completely dependent on AI availability.

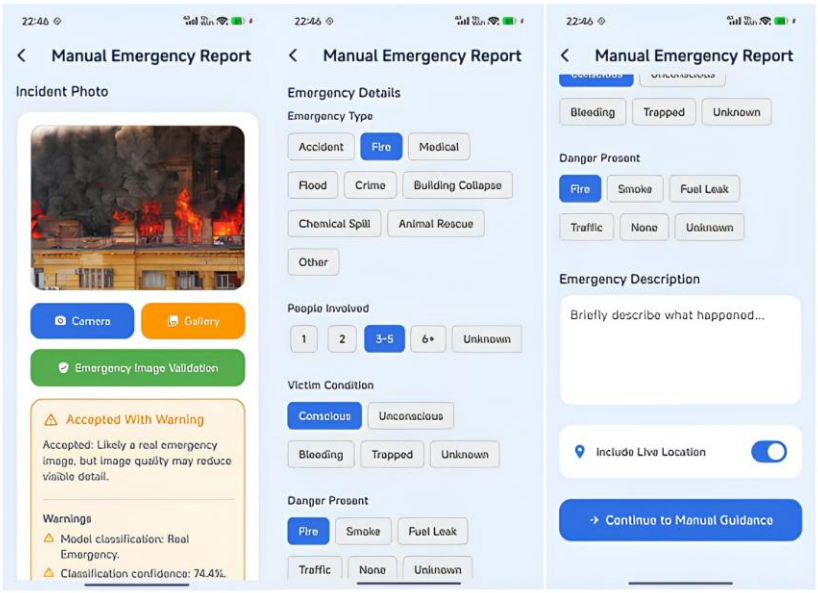

Figure 3. Manual Reporting

Bystander Assistance

The application also includes local bystander-support functions, including an Incident Timer, CPR Beat Guide, Emergency Flashlight, and first-aid guidance. These features operate independently of Gemini AI and therefore provide additional support even when online AI analysis is unavailable. Comparable first-aid functions have been integrated into AI-driven emergency applications, where an AI-powered first-aid companion provides instant guidance when medical help is delayed [3].

METHODOLOGY

Research and Development Approach

This study adopted a system development and empirical evaluation approach. The system was developed using the Agile methodology, followed by three main evaluation activities:

1. Functional system testing;
2. TensorFlow Lite model evaluation; and
3. User Acceptance Testing and usability evaluation.

This approach enables both technical and user-oriented evaluation of the proposed system.

Agile Development

Agile was selected because the application integrates multiple technologies, including Flutter, Firebase, Google Gemini AI, GPS services, notifications, and mobile user-interface components. These components require iterative implementation, testing, and refinement. The development process consisted of:



Figure 4. Development Process

During requirements analysis, functional and non-functional requirements were identified through literature review, existing-system analysis, and project requirements. Functional requirements included emergency image upload, AI analysis, GPS detection, emergency guidance, notifications, emergency history, and related emergency-reporting functions.

During development, Flutter and Dart were used for mobile applications, Firebase Authentication was used for user authentication, and Cloud Firestore was used for cloud data storage. Google Gemini AI was integrated for online emergency-image analysis.

The iterative process allowed improvements to be introduced based on testing and feedback, including interface refinements, AI-response handling, Firebase integration, guest reporting, multilingual support, and emergency-status functionality.

Evaluation Strategy

The evaluation consisted of three complementary components. Those components are Functional Testing, TensorFlow Lite Model Evaluation, and User Acceptance Testing (UAT).

Functional testing examined whether the implemented system functioned according to the defined requirements. Specifically, testing covered authentication, guest emergency reporting, manual reporting, image validation, Gemini AI analysis, GPS handling, Firestore storage, emergency guidance, emergency profile, language switching, dark mode, report status, notifications, and Bystander Assist. This approach reflects established practice in mobile emergency application development, where predefined acceptance criteria are assessed to verify the correct functioning of the application before deployment [23].

To evaluate the TensorFlow Lite model, the image-validation system was tested using 86 independent images. Its overall performance was measured based on accuracy, precision, recall, and F1-score, alongside a confusion matrix detailing the classification results.

Lastly, UAT was conducted with 11 selected smartphone users. Participants performed predefined application tasks covering guest reporting, registered-user reporting, AI analysis, emergency profile, settings, and Bystander Assist. Usability feedback was subsequently collected using a questionnaire. Usability is a major criterion for safe and efficient use of interactive technology in disaster and emergency management, although evaluation in this setting is difficult because of the heterogeneity and unpredictability of operating conditions [24]. Prior emergency applications have been evaluated using established usability instruments, including the System Usability Scale, to obtain rapid and actionable feedback for improving usability [25].

RESULTS AND DISCUSSION

Functional Testing

A total of 29 functional test cases were executed. During the initial testing phase, 28 test cases passed and 1 test case failed. The initial pass rate is 96.55%. The failed test involved GPS location handling. Specifically, GPS information continued to be retrieved after location sharing had been disabled. The defect was corrected, followed by corrective retesting and regression testing. This outcome is consistent with the

emphasis in mobile emergency application development on verifying acceptance criteria through structured functional testing before deployment [23].

TensorFlow Lite Model Performance

The three-class TensorFlow Lite model was evaluated using 86 independent test images.

Table 3. TensorFlow Lite Model Performance

Metric	Result
Independent Test Images	86 images
Accuracy	82.56%
Macro Precision	83.43%
Macro Recall	82.94%
Macro F1-score	82.31%

The model demonstrated its strongest performance on the non-emergency class, with 96% recall, while the AI-generated class achieved 86% recall. However, the real-emergency class achieved only 67% recall. This means that some genuine emergency images were incorrectly classified as either AI-generated or non-emergency.

This finding is particularly important because false rejection of genuine emergency images represents a more serious concern than simply reducing overall classification accuracy. Emergency images may naturally contain difficult characteristics such as darkness, smoke, blur, unusual viewpoints, low resolution, and partial obstruction. Recognizing smoke and fire from images is inherently challenging because of the large variance in colour, texture, and shape, which has motivated dedicated deep architectures for automatic feature extraction in this domain [26]. Efficient models are required for hardware-constrained systems such as mobile devices and embedded platforms, and studies have demonstrated that compact architectures can reach average test accuracies above 84% on fire datasets while using substantially fewer parameters and operations than larger models [27]. CNN-based fire detection research similarly reports average precision of 83.7% with real-time detection speeds of 28 frames per second [28], while energy-efficient architectures inspired by SqueezeNet have achieved accuracy comparable to more complex models for fire detection and localization [29]. These results help contextualize the 82.56% accuracy obtained by the present model, while also highlighting that class-level recall, rather than aggregate accuracy, is the more informative measure for safety-related classification.

Consequently, the current model should be interpreted as a prototype-level validation mechanism, rather than a safety-critical emergency-detection model.

Gemini AI Integration

The integration test demonstrated that Google Gemini AI could be used as an optional online analysis component after image validation. Gemini AI provides emergency category, severity, incident summary, suggested assistance, and safety guidance. The hybrid design provides an important architectural advantage. TensorFlow Lite handles local validation, while Gemini AI provides richer analysis when Internet connectivity is available.

However, the evaluation also revealed an important limitation: Gemini AI response time is dependent on Internet connectivity and external service availability. Two respondents specifically reported that AI Analysis could take too long. This is consistent with the observation that LLM-based crisis-management tools must operate in high-stakes environments where real-time data synthesis and rapid response are critical requirements [22].

Therefore, online AI analysis should be considered an optional decision-support component, rather than a mandatory dependency for emergency reporting.

User Acceptance Testing

The UAT involved 11 smartphone users and 12 predefined application tasks, producing 132 task attempts. These tasks included 99 tasks completed without assistance, 33 tasks completed with assistance, and 0 tasks were recorded as unable to be completed. Therefore, the combined task-completion rate was 100%. However, 25% of task attempts required assistance. This suggests that while users were ultimately able to complete the required tasks, some application functions were not immediately intuitive to first-time users.

This finding is particularly relevant for emergency applications because users may interact with the system under stressful conditions. Therefore, usability should not be assessed solely through successful completion; the amount of assistance required is also an important consideration. This aligns with the argument that the availability and technical reliability of interactive systems alone do not guarantee improved workflows, and that usability evaluation must therefore account for the difficult and usually mobile context in which emergency systems are used [24].

Usability Evaluation

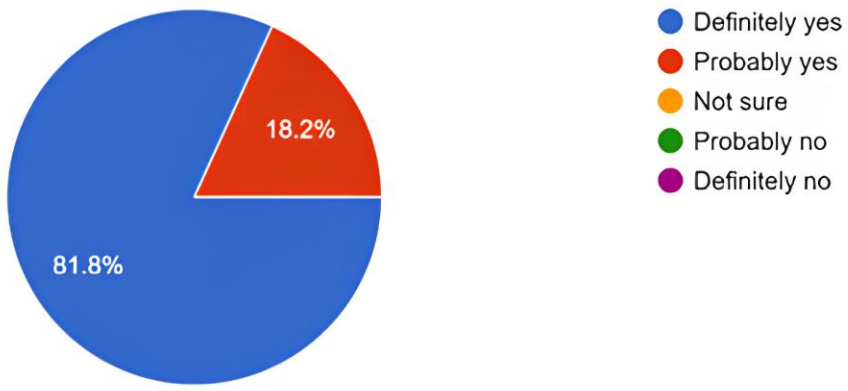

Figure 5. Overall Usability Feedback

The overall usability score was 4.79 / 5.00. All questionnaire responses fell within the Agree or Strongly Agree categories. Participants particularly valued the AI Analysis and Emergency Guidance features. Comparable emergency applications have also reported high usability outcomes when assessed using established instruments, including a mobile prototype for pre-hospital emergency communication that obtained excellent user satisfaction scores [30].

Furthermore, all respondents indicated a positive intention to use the application for emergency-reporting support:

Table 4. Respondents Response

Response	Percentage
Definitely Yes	81.8%
Probably Yes	18.2%
Not Sure	0%
Probably No	0%.
Definitely No	0%

Although these findings are encouraging, they should be interpreted cautiously because the evaluation involved only 11 selected users and was dominated by participants aged 21–25. The results therefore represent preliminary usability evidence rather than population-level generalization.

Overall Discussion

The evaluation demonstrates that the proposed Emergency AI App is technically feasible as an integrated emergency-support prototype. Three findings are particularly significant.

First, the 100% final functional pass rate indicates that the major application workflows operated successfully after defects were corrected.

Second, the 82.56% TensorFlow Lite accuracy demonstrates the feasibility of performing local image validation on a mobile device. However, the 67% recall for genuine emergency images demonstrates that overall accuracy alone is insufficient for evaluating an AI system intended for safety-related applications.

Third, the 4.79/5 usability rating indicates positive user perception, but the 33 assisted task attempts show that usability refinement remains necessary.

Taken together, these results suggest that the strongest contribution of the system is not a standalone AI classifier but rather the integration of local AI validation, online AI decision support, emergency reporting, location services, and user guidance into a unified mobile platform. These findings align with the broader literature, which indicates that machine learning can support emergency management effectively, but that generalization, explainability, and dataset construction remain key open challenges for such systems [4].

CONCLUSION AND FUTURE WORK

This study developed and evaluated the Emergency AI App, an AI-assisted mobile platform designed to support emergency reporting and decision-making.

The developed system integrates Flutter-based mobile development, Firebase services, TensorFlow Lite, Google Gemini AI, GPS location services, emergency guidance, and Bystander Assist tools. The application provides both guest and registered-user workflows, allowing basic emergency-support functions to be accessed without registration while providing additional functionality to authenticated users.

The evaluation produced three major findings. First, all 29 functional test cases passed after corrective testing. Second, the TensorFlow Lite model achieved 82.56% accuracy across 86 independent test images. Third, UAT involving 11 users achieved a 100% combined task-completion rate and a usability mean of 4.79/5.

Despite these positive findings, the system has several limitations. The most significant is the 67% recall for the real-emergency class, which indicates that some genuine emergency images may be incorrectly rejected. The training and testing datasets may also not adequately represent the wide range of emergency conditions, camera devices, image qualities, and environmental conditions encountered in real-world situations. This limitation reflects a recognized challenge in the field, where more efficient ways of addressing the difficulties associated with building appropriate datasets remain an identified need [4].

In addition, Gemini AI and Firebase-dependent functions require Internet connectivity, while GPS functionality depends on device capability and user permissions. The small number of participants also limited the usability evaluation.

Future work should therefore focus on five areas. First, the image dataset should be expanded with more diverse real-emergency images and challenging environmental conditions, addressing the documented need for better generalization of machine-learning algorithms in emergency management [4]. Second, investigate model optimization and alternative architectures to improve real-emergency recall. Third, additional offline

functionality should be introduced to reduce dependence on Internet connectivity. Fourth, larger and more diverse user studies should be conducted across different age groups, devices, and levels of smartphone experience. Finally, security and privacy mechanisms should be strengthened because the application handles emergency reports, location information, emergency-profile data, and potentially sensitive personal information. On-device processing already offers a degree of privacy advantage because data need not be transmitted to a server [14], but this must be complemented by stronger controls for the cloud-based functions. Given the rapid pace of generative-model development, continued attention to AI-generated image detection will also be necessary as the validation model is extended [18].

The Emergency AI App should therefore be considered an academic emergency-reporting and decision-support prototype, rather than a replacement for professional emergency services. It does not directly contact emergency services, dispatch official responders, or guarantee emergency assistance.

ACKNOWLEDGEMENTS

The authors wish to thank Fazrina Majhardeen from Universiti Teknikal Malaysia Melaka (UTeM) for her support and for granting permission to adapt her project into this research article.

REFERENCES

1. M. E. Barachi, F. Kamoun, J. Ferdaos, M. Makni, and I. Amri, “An artificial intelligence based crowdsensing solution for on-demand accident scene monitoring,” Procedia Computer Science, vol. 170, pp. 303–310, Jan. 2020, doi: 10.1016/j.procs.2020.03.044.
2. M. A. Erskine and W. Pepper, “Enhancing Emergency Response Management using Emergency Description Information Technology (EDIT),” International Journal of Electronic Government Research, vol. 11, no. 2, pp. 51–65, Apr. 2015, doi: 10.4018/ijegr.2015040104.
3. S. Krishnendu, S. V. Rithish, S. Niveditha, and R. K. R, “EmergiFy: Development of a Real-Time Mobile Emergency Alert System,” pp. 1–6, Apr. 2025, doi: 10.1109/icict64420.2025.11005046.
4. C. Kyrkou, P. Kolios, T. Theocharides, and M. M. Polycarpou, “Machine Learning for Emergency Management: A Survey and Future Outlook,” Proceedings of the IEEE, vol. 111, no. 1, pp. 19–41, Dec. 2022, doi: 10.1109/jproc.2022.3223186.
5. M. Gaziel-Yablowitz and D. G. Schwartz, “A Review and Assessment Framework for Mobile-Based Emergency Intervention Apps,” ACM Computing Surveys, vol. 51, no. 1, pp. 1–32, Jan. 2018, doi: 10.1145/3145846.
6. D. N. Aisyah et al., “Mobile-Enabled Technologies in Disaster Management: A Systematic Literature Review,” Disaster Medicine and Public Health Preparedness, vol. 20, Jan. 2026, doi: 10.1017/dmp.2025.10252.
7. M. Weinlich, P. Kurz, M. B. Blau, F. Walcher, and S. Piatek, “Significant acceleration of emergency response using smartphone geolocation data and a worldwide emergency call support system,” PLoS ONE, vol. 13, no. 5, May 2018, doi: 10.1371/journal.pone.0196336.
8. H. Lee, T. Kang, S. Jeong, and J. Seo, “Evaluation of RF Fingerprinting-Aided RSS-Based Target Localization for Emergency Response,” arXiv (Cornell University), pp. 1–7, Jun. 2022, doi: 10.1109/vtc2022-spring54318.2022.9860436.
9. V. S. Devi, M. D. Kumar, and C. A. Kumar, “Flutter Based SOS Alert and Location Tracking Application with Volunteer Assist and Rescue,” International Journal of Research and Applied Innovations, vol. 9, no. 3, 2026.
10. H. Gibson, B. Akhgar, I. Hajirasouliha, R. Garcia, and Z. Özdemir, “A SITUATIONAL AWARENESS FRAMEWORK FOR IMPROVING EARTHQUAKE RESPONSE, RECOVERY AND RESILIENCE,” SHURA (Sheffield Hallam University Research Archive) (Sheffield Hallam University), Dec. 2025, doi: 10.71846/16-wcee-0244.
11. D. Lorenzi, S. A. Chun, J. Vaidya, B. Shafiq, V. Atluri, and N. R. Adam, “PEER,” International Journal of E-Planning Research, vol. 4, no. 3, pp. 29–46, Jul. 2015, doi: 10.4018/ijepr.2015070102.
12. J. B. Bernardo, “Crowdalert: An Android Application for Increasing the Awareness and Response Initiatives of the Citizens through Crowdsourcing,” World Academy of Science, Engineering and

Technology, International Journal of Computer and Information Engineering, vol. 2, no. 10, Oct. 2015, Accessed: Oct. 2025. [Online]. Available: https://www.waset.org/abstracts/37771
13. G. Menghani, "Efficient Deep Learning: A Survey on Making Deep Learning Models Smaller, Faster, and Better," arXiv (Cornell University), vol. 55, no. 12, pp. 1–37, Jan. 2023, doi: 10.1145/3578938.
14. Y. Sang, Y. Huang, S. Huang, and H. Cui, "Beyond the Model: Data Pre-processing Attack to Deep Learning Models in Android Apps," May 06, 2023, Cornell University. doi: 10.48550/arxiv.2305.03963.
15. I. Martinez-Alpiste, G. Golcarenarenji, Q. Wang, and J. M. A. Calero, "Smartphone-based real-time object recognition architecture for portable and constrained systems," Journal of Real-Time Image Processing, vol. 19, no. 1, pp. 103–115, Sep. 2021, doi: 10.1007/s11554-021-01164-1.
16. R. Mennilli, L. Mazza, and A. Mura, "Integrating Machine Learning for Predictive Maintenance on Resource-Constrained PLCs: A Feasibility Study," Sensors, vol. 25, no. 2, p. 537, Jan. 2025, doi: 10.3390/s25020537.
17. J. J. Bird and A. Lotfi, "CIFAKE: Image Classification and Explainable Identification of AI-Generated Synthetic Images," IEEE Access, vol. 12, pp. 15642–15650, Jan. 2024, doi: 10.1109/access.2024.3356122.
18. J. Deng et al., "A Survey of Defenses Against AI-Generated Visual Media: Detection, Disruption, and Authentication," ACM Computing Surveys, vol. 58, no. 5, pp. 1–35, Oct. 2025, doi: 10.1145/3770916.
19. V. G. Goecks and N. R. Waytowich, "DisasterResponseGPT: Large Language Models for Accelerated Plan of Action Development in Disaster Response Scenarios," Jun. 29, 2023, Cornell University. doi: 10.48550/arxiv.2306.17271.
20. Y. Jiang, "The Applications of Large Language Models in Emergency Management," pp. 199–202, May 2024, doi: 10.1109/imcec59810.2024.10575031.
21. M. Wu, S. Luo, H. Xu, and W. Sun, "Optimizing Emergency Decision-Making for Natural Disaster Using Light Retrieval Augmented Generation," pp. 2237–2241, Mar. 2025, doi: 10.1109/icaace65325.2025.11019353.
22. O. Odubola, T. O. Adeyemi, O. O. Olajuwon, N. P. Iduwe, A. Inyang, and T. Odubola, "AI in Social Good: LLM powered Interventions in Crisis Management and Disaster Response," Journal of Artificial Intelligence Machine Learning and Data Science, vol. 3, no. 1, pp. 2353–2360, Feb. 2025, doi: 10.51219/jaimld/oluwatimilehin-odubola/510.
23. V. Martínez-Cepeda, M. J. P. Moreira, V. L. B. Tapia, L. Ortiz-Delgado, and D. R. S. Armijos, "ESPE Security: Mobile and Web Application to Manage Community Emergency Alerts," International Journal of Interactive Mobile Technologies (iJIM), vol. 18, no. 5, pp. 77–90, Mar. 2024, doi: 10.3991/ijim.v18i05.46781.
24. T. Mentler, H. Berndt, D. Wessel, and M. Herczeg, "Usability Evaluation of Information Technology in Disaster and Emergency Management," in IFIP advances in information and communication technology, Springer Science+Business Media, 2017, pp. 46–60. doi: 10.1007/978-3-319-68486-4_5.
25. A. Akcakaya, S. Diri, S. Şahın, and R. S. YIKILMAZÇINAR, "Panic Button Mobile Application Usability Study," International Journal of Engineering and Innovative Research, vol. 4, no. 2, pp. 104–113, May 2022, doi: 10.47933/ijeir.1085846.
26. Z. Yin, B. Wan, F. Yuan, X. Xia, and J. Shi, "A Deep Normalization and Convolutional Neural Network for Image Smoke Detection," IEEE Access, vol. 5, pp. 18429–18438, Jan. 2017, doi: 10.1109/access.2017.2747399.
27. A. Ayala, B. Fernandes, F. Cruz, D. Macêdo, and C. Zanchettin, "Convolution Optimization in Fire Classification," IEEE Access, vol. 10, pp. 23642–23658, Jan. 2022, doi: 10.1109/access.2022.3151660.
28. P. Li and W. Zhao, "Image fire detection algorithms based on convolutional neural networks," Case Studies in Thermal Engineering, vol. 19, p. 100625, Mar. 2020, doi: 10.1016/j.csite.2020.100625.
29. K. Muhammad, J. Ahmad, Z. Lv, P. Bellavista, P. Yang, and S. W. Baik, "Efficient Deep CNN-Based Fire Detection and Localization in Video Surveillance Applications," Liverpool John Moores University, vol. 49, no. 7, pp. 1419–1434, Jun. 2018, doi: 10.1109/tsmc.2018.2830099.

30. D. S. Montandon, L. C. S. de Oliveira, A. A. C. Araújo, R. S. E. Sant'Ana, I. A. C. Mendes, and S. de Godoy, "Mobile application for accessible communication with pre-hospital emergency care: e-SU," Acta Paulista de Enfermagem, vol. 37, Jan. 2024, doi: 10.37689/acta-ape/2024ao000026655.