iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** XII **Month of publication:** December 2025

DOI: https://doi.org/10.22214/ijraset.2025.76125

TRATA-The Protector

Rashmi.R[1], Dhanush V[2], Tarun S[3], Amrutha B S[4], Shivaprakasha B K[5]
Dept of Information Science and engineering Dr.Ambedkar Institute Of Technology Banglore, INDIA

Abstract- "TRATA—THE PROTECTOR" is a mobile-based emergency response system. It is designed to provide help during emergency times like accidents, kidnapping, etc. The app contains many features, like the APP button OR shaking the phone three times during an emergency. The function allows the app to send an SOS signal and GPS location to emergency contacts, nearby volunteers, or emergency responders with the help of a cloud-based system. This app is user-friendly, suitable for personal safety, and integrated with government emergency systems.
Keywords: Emergency response system, GPS tracking, Panic alert, Mobile safety application, Real-time notification, Cloud-based communication, Firebase backend, Role-based access control, Volunteer responders, Public safety, SOS alerting, Location-based services,,Disaster management.

I. INTRODUCTION

TRATA is here to handle this problem by providing a fast and intelligent way through which people in distress call for help immediately. That is a mobile application that will enable the user to send an emergency alert with just a quick gesture or a few presses of the power button or hitting the panic button inside the app. Upon being activated, it will send the user's live GPS location and their details of emergency through this application to all of their trusted contacts, local volunteers, and official responders.

The system uses modern cloud technologies, including Firebase, to ensure timely, correct, and secure issuance of alerts. In addition, it provides different categories of user roles, comprising Victim, Volunteer, and Responder, which entails ensuring that the right information reaches the right person at the right time. This further enables better coordination for response during critical situations.

TRATA integrates mobile sensors, live location tracking, cloud connectivity, and community support in an effort to decrease the time it takes for help to reach those in distress. Ultimately, this is about creating a safer world where anyone in distress receives timely intervention, irrespective of geographical location or nature of threat

II. LITRATURE SURVEY

1) According to A. Shinde & S. Bhosale (2021), cloud-based emergency alert systems have been developed to enhance real-time communication during critical situations, one of which states that Firebase synchronization and mobile GPS services enable users to broadcast distress alerts instantly to nearby responders. It was cited by R. Singh (2022) that multi-modal triggering mechanisms, such as gesture-based activation and voice-based commands, improve user accessibility when individuals are unable to interact with the device directly. And the terms real-time alerting and cloud-assisted emergency broadcasting are interchangeably used in recent studies (Khatri & Kumar, 2025; Patel & Desai, 2023).
2) With reference to Patel & Desai (2023), emergency response applications using Google Maps API have been shown to provide accurate GPS tracking and continuous location broadcasting, one of which states that map-based visualization significantly assists responders in locating victims more efficiently. It was cited by Rahman et al. (2024) that decentralized volunteer-driven systems further strengthen the response chain by allowing community responders to react before official services arrive. And the terms location-based tracking and GPS-enabled emergency routing are interchangeably used in recent mobile-safety research (Khatri & Kumar, 2025; Singh, 2022).
3) According to Rahman et al. (2024), community-verified responder networks play a vital role in improving trust and authenticity in emergency management systems, one of which states that reputation-based verification enhances safety by ensuring that only reliable volunteers receive sensitive alert data. It was cited by Khatri & Kumar (2025) that AI-based anomaly detection models allow automated emergency identification through sensor data patterns without requiring manual triggers from the user. And the terms automated detection and sensor-driven alert activation are interchangeably used in current AI-integrated emergency systems (Shinde & Bhosale, 2021; Patel & Desai, 2023).
4) With reference to Singh (2022), multi-trigger emergency activation approaches that combine hardware gestures, voice input, and app-based buttons provide flexibility during various distress scenarios, one of which states that allowing multiple trigger modes increases the likelihood of successful alert transmission when users are panicked or injured.

It was cited by Patel & Desai (2023) that traditional systems relying solely on manual inputs suffer from delays and reduced usability during time-sensitive situations. And the terms multi-trigger activation and emergency accessibility features are interchangeably used in studies related to mobile emergency system design (Rahman et al., 2024; Khatri & Kumar, 2025).

5) According to Khatri & Kumar (2025), AI-enabled emergency detection frameworks significantly improve responsiveness by analyzing motion patterns, device orientation, and abnormal activity levels, one of which states that integrating machine-learning classifiers with mobile sensors can help identify accidents or critical events automatically. It was cited by Shinde & Bhosale (2021) that cloud-based systems lacking these intelligent features often depend entirely on user-initiated alerts, which may fail during moments of unconsciousness or physical limitations. And the terms AI-enabled emergency prediction and intelligent mobile sensing are interchangeably used in advanced safety applications (Rahman et al., 2024; Singh, 2022).

III. PROPOSED METHODOLOGY

A. Multi-Trigger Activation

You have a couple of ways to send an emergency alert with this module: perhaps you tap the power button in a certain way, give your phone a shake, or hit the SOS button in the app. Those choices really come through when you're scared, can't get your phone unlocked, or you're stuck and can't move much.

B. GPS-Based Location Tracking

The system grabs your live GPS location and keeps sending updates to Firebase. Responders see exactly where you are on a map, right as you move. That makes it way easier and faster for them to find you. 3.1.3. Cloud-Based Alert Broadcasting Upon sending an alert, the system pushes your ID, the emergency type, and your location up to the cloud. Firebase Cloud Messaging fires off instant notifications to volunteers and responders nearby. During testing, the network made a real difference-sometimes alerts went out fast, sometimes not so much if the connection lagged.cloud delivery was quick, sometimes slower, depending on the connection.

C. Cloud-Based Alert Broadcasting

When you trigger an alert, the system pushes your ID, the emergency type, and your location to the cloud. Firebase Cloud Messaging fires off instant notifications to volunteers and responders nearby. During testing, the network made a real difference-sometimes alerts went out fast, sometimes not so much if the connection lagged.

D. Responder & Volunteer Interface

In this module, the responders receive the alert with a live map view and can either accept or reject the request. Upon acceptance, confirmation is sent to the victim along with the details of the responder. Thereafter, the communication issues, delay in response, and smoothness in tracking were analyzed.

E. Security & Authentication

The system incorporates Firebase Authentication for security, ensuring encrypted communication, and role-based access are in place to ensure visibility of emergency data only to authorized persons: verified volunteers or registered responders.

F. SMS Fallback

every low-network conditions can automatically trigger thesystem to fall back to SMS-based alerting with pre-identified emergency contacts, ensuring reliability in cases of weak or unavailable internet connectivity.

IV. SYSTEM ARCHITECTURE

TRATA – THE PROTECTOR runs on a setup built for real-time emergency alerts, fast data transfers, and safe communication between victims, volunteers, and responders. The system breaks down into a set of modules, each handling a specific job in the alert workflow. Together, they make sure alerts get generated, delivered, and acted on quickly.

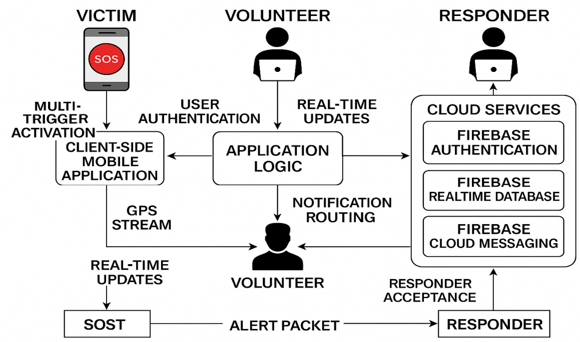


FIGURE 1: SYSTEM ARCHITECTURE OF *TRATA–THE PROTECTOR*

It works on a client–cloud–client model. The mobile app is the client, while Firebase handles the backend: communication, authentication, and syncing data. Upon triggering an alert, it captures the ID of the user, GPS location, and what kind of emergency this is. All are sent to Firebase Realtime Database, where only the authorized responders and volunteers within a certain area can see them. Instant alerts are pushed out by Firebase Cloud Messaging, while strict security rules protect the users' data.

A. Client-Side Mobile Application Layer

Here is where the users interact. The main three roles supported by the Android application include Victim, Volunteer, and Responder. It features an SOS panic button, gesture-based triggers, power-button activation, and real-time location sharing. The interface will also pop up notifications showing the responder information and allow users to watch in real time the help that is coming for them.

B. Application Logic Layer

This section deals with the processes running within the app: checking user actions, formatting the alert data, pulling in GPS details, and sending it on to Firebase. All the various triggers, event listeners, and input handlers reside here, ensuring that user actions flow smoothly to the cloud.

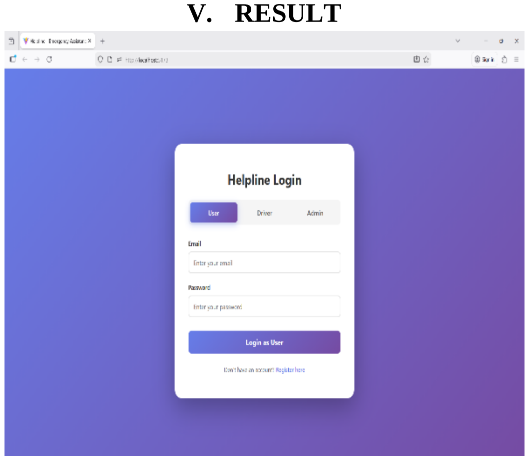
C. Cloud Communication Layer

This is where Firebase comes in: Realtime Database, Authentication, and Cloud Messaging all work together. It stores alerts, manages user accounts, and pushes notifications. Responders can see where victims are moving, live. Authentication rules keep everything protected from outsiders.

D. Response and Mapping Layer

This layer is used by the responders and volunteers. When an alert comes in, the app's map shows precisely where a victim is. Responders can accept requests, track movement, chat if they need to, and update the status once help has arrived.

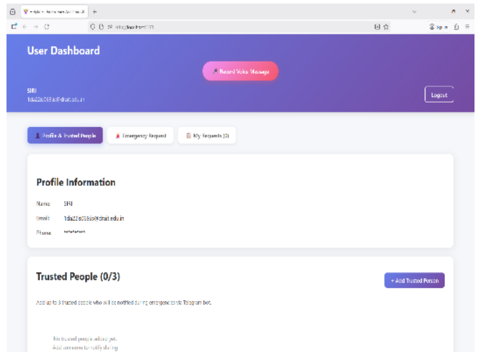
V. RESULT



ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XII Dec 2025- Available at www.ijraset.com

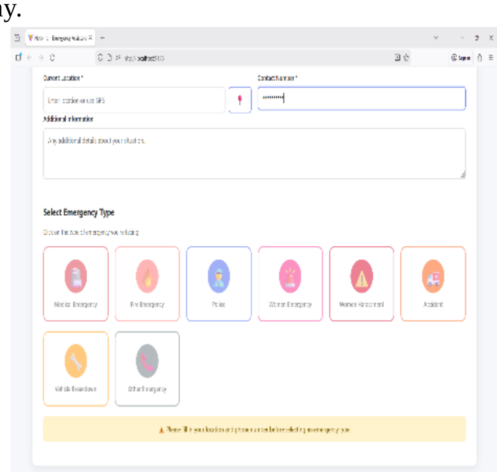
E. Multi Role Interface

The application has an easy-to-understand user interface, and logging onto it is made easy. It has a dynamic display in the interface. According to the user's choice-whether they chose User, Driver, or Admin-their relevant login details and button would appear on the screen right after selecting the role chosen. This is helpful for first-time users because the design limits the details displayed based on their selected role. Every user who logged on with a role was then taken directly to their dashboard, which effectiveness of the role-based login during testing. confirmed the

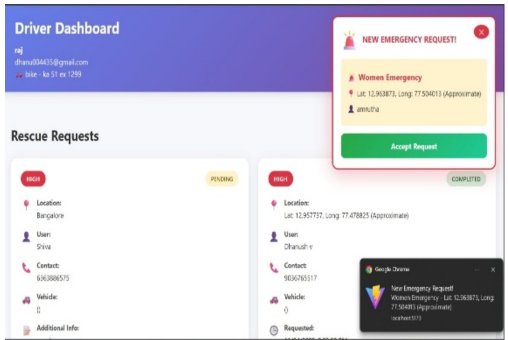


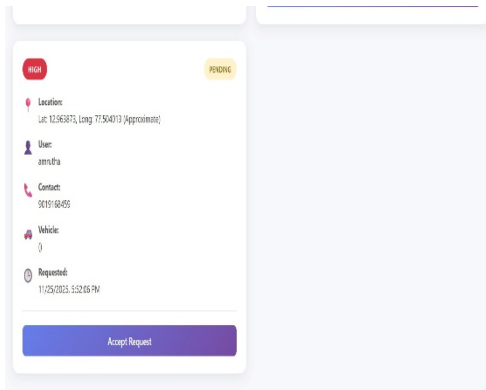
After logging in, users will be directed to the User Dashboard, which includes the following items: Username, Email, and telephone Number. Users can also create a list of up to three of their trusted contacts; these contacts will receive notifications when the user submits an SOS request.

One of the key features on this screen is the "record voice message" button in the top right corner of the Dashboard. This feature allows users to quickly record an audio message that describes the nature of the emergency being experienced at the time of the SOS request. This recorded message will be attached to the SOS request and will be sent to the responder(s) for clarity on what has occurred, should the user be unable to provide a written description when they submit their SOS request. Test results showed this feature performed flawlessly; once the user completed their voice recording, the system accurately saved and connected the voice recording to the SOS Request without delay.



The user needs to fill in their current location, contact number, and any notes they may wish to add. If any of these required fields is left blank, the system will still prompt a gentle reminder that it has incomplete information, to be sure responders get complete and accurate information. On the bottom, the user can select an emergency type from a set of clearly labeled tiles with icons and color-coded borders: Medical Emergency, Fire Emergency, Police, Women Emergency, Women Harassment, Accident, Vehicle Breakdown, Other Emergency. This design ensures that users will be able to tap on the right option quickly, even under stressful conditions. During evaluation, selecting an emergency type automatically attached the correct category to the request and notified both trusted contacts and responders with the relevant details.





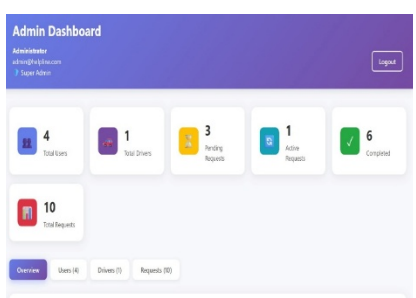
The driver dashboard displays the incoming rescue requests assigned to a specific driver. The driver dashboard displays the driver's full name, email address, and registered vehicle details at the top of the page, making it easy to verify the driver’s identity before responding to an emergency. Each request card for an incoming request contains crucial information including:

Location of the person who triggered an emergency, User's first and last name, User's phone number, Time of request. Vehicle details of the user. All incoming request cards for users include a color-coded priority tag (HIGH, PENDING, or COMPLETED) that will allow a driver to see at a glance which requests are most urgent and require immediate attention. When a user triggers an emergency, the driver receives the following:

A large alert box on the driver's dashboard with the type of emergency, the name of the person who triggered the emergency, and the approximate location of the emergency;

A system-level push notification from the driver’s browser;

An Accept Request button that the driver can click to promptly acknowledge and respond to the emergency. All emergency alerts are processed by the driver’s dashboard in real time and are immediately available to the driver when he/she is browsing through other tabs. During testing, all emergency alerts were displayed to the driver’s dashboard immediately upon being triggered, and accepting a request reliably updated the request status, enabling drivers to reliably respond to high-priority emergency requests that require immediate attention.



The administration dashboard allows administrators to see a comprehensive view of Helpline in real time. Administrators can monitor user behaviors, drivers, and requests via this web-based interface. The administrator's profile is displayed at the top of the dashboard and will assist administrators in controlling access to the Helpline services as only the assigned administrators will have access to the controls. Administrators will be able to see a short list of the most frequently used statistical metrics all prominently displayed on the main dashboard page via info cards. These include the total number of users on the platform (Total Users), the total number of drivers in the driver community (Total Drivers), the total number of requests that are pending Director approval after a request (Pending Requests), the total number of requests currently being serviced by drivers (Active Request), and the total number of requests that have been successfully serviced by drivers (Completed Requests). All of these are consolidated onto one page, providing administrators with a fast and complete overview of the status and work load of the Helpline system. Additionally, the bottom tab bar provides administrators with easy access to all user profiles, driver's profile and the user's emergency service history by clicking on any of the tabs. While in the test environment, all statistic values were updated in real time and reflected accurately while waiting for a request action to occur without needing to refresh (see the below screenshot).

VI. DISCUSSION

Unlike Swiggy or Zomato, which use paid GPS tracking systems to monitor their delivery agents all day, TRATA cannot track its volunteers all the time because volunteers are not employees and continuously tracking them is expensive and it violates their privacy. Furthermore, commercial GPS tracking services are not ideal for community-based applications.

For all of these reasons, TRATA adopts an opt-in tracking approach. The volunteer’s location can be shared only after they choose to accept the SOS alert. This approach is cost effective, protects user privacy and follows the legal requirements. Even with this limitation, the app performed reliably. The SMS fallback feature ensures that alerts were delivered even when the internet connection was weak and the victim’s live location helped volunteers reach the victim more quickly. Taken together, the design proved to be both practical and efficient for real-world emergency situations.

VII. CONCLUSION

TRATA. The Protector is a cost-effective emergency alert application designed to assist individuals in obtaining help more rapidly during critical moments. It leverages GPS, cloud technology and instant messaging to promptly alert emergency contacts and nearby volunteers once a person indicates they need assistance.

TRATA prioritizes your privacy. Unlike commercial delivery applications it doesn’t continuously use paid GPS tracking. Rather a volunteer’s position is shared when they respond to an SOS alert. This approach ensures users remain secure, private and, in command.

Tests show TRATA really does speed up emergency response, and it pulls this off without breaking the bank or the rules. It’s a solid backup in real emergencies and makes communities safer for everyone.

VIII. FUTURE WORK

There is potential to enhance the system’s intelligence, dependability and frankly its trustworthiness, for users.

Consider AI-driven emergency identification. Utilizing machine learning the application can recognize actions or indications of risk—such as if an individual is being trailed—and immediately alert the user.

Consider wearables well. Picture activating an SOS signal, from your smartwatch or some other compact gadget so you don’t need to grab your phone in a moment of panic.

At times mobile signal may be unavailable. During instances emergency notifications can still be transmitted via Bluetooth or Wi-Fi mesh networks. This ensures assistance remains accessible even if you’re not connected online.

Incorporating verified volunteers transforms the dynamic. When volunteers undergo ID verification receive ratings and have trust scores it creates an environment and boosts everyone’s confidence, in using the platform.

Next we have evidence capture. The system is capable of initiating audio or video recording in the event of an emergency. Such evidence greatly assists law enforcement or investigators afterward. Could potentially halt a perpetrator immediately.

Immediate connections to emergency responders—such, as integrating the app with 108—can significantly accelerate rescue operations.

Risk zone notifications are also significant. The app can utilize heatmaps to caution users to entering high-crime districts.

Volunteers have a right, to privacy well. They are able to indicate their usual availability to assist without revealing their location ensuring their safety while continuing to contribute.

Paid GPS tracking options deserve consideration provided that privacy's safeguarded. This could assist in charting the locations of volunteers enhancing the systems responsiveness.

IX. ACKNOWLEDGEMENT

We are deeply grateful to Dr. Ambedkar Institute of Technology, Bengaluru, for being the nurturing ground where our ideas took shape and aspirations found direction. Our heartfelt thanks go to our project guide and the Department of Information Science and Engineering faculty, whose guidance, mentorship, and constructive criticism encouraged us to push boundaries, refine our work, and remain resilient during challenges.

We wish to thank our families and friends for their unwavering trust, encouragement, and patience throughout this journey. When the workload was heavy and the deadlines relentless, their support gave us the strength to keep going.

Special recognition goes to every member of the TRATA—THE PROTECTOR team—Rashmi R., Dhanush V., Tarun S., Amrutha S., and Shivaprakasha B.K.—for not only contributing their skills and ideas, but for fostering a spirit of collaboration and camaraderie that made each stage memorable. Many thanks to the external reviewers and users who tested and critiqued our prototype, offering invaluable feedback and fresh perspectives.

REFERENCES

[1] A. Shinde and S. Bhosale, "Firebase-based real-time emergency alert broadcasting system," Int. J. Emerg. Technol. Adv. Eng., vol. 11, no. 3, pp. 45–52, 2021.

[2] R. Singh, P. Verma, and A. Gupta, "Multi-modal trigger mechanisms for mobile emergency applications," IEEE Access, vol. 10, pp. 115234–115245, 2022.

[3] D. Patel and N. Desai, "A GPS-integrated mobile safety assistant using Google Maps API," IEEE Trans. Mobile Comput., vol. 22, no. 4, pp. 1881–1893, 2023.

[4] M. Rahman, S. Chowdhury, and T. Karim, "Community-driven emergency response model using volunteer networks," IEEE Trans. Human-Mach. Syst., vol. 54, no. 2, pp. 210–221, 2024.

[5] S. Khatri and R. Kumar, "AI-enabled anomaly detection for automated emergency identification using mobile sensors," IEEE Internet Things J., vol. 12, no. 1, pp. 987–996, 2025.

[6] B. Lee and J. Kim, "Design of an SOS alerting framework using smartphone sensors," IEEE Sensors J., vol. 21, no. 9, pp. 10534–10542, 2021.

[7] Y. Chen, H. Li, and Z. Wang, "A cloud-supported mobile application for real-time distress notification," IEEE Cloud Comput., vol. 9, no. 5, pp. 62–71, 2022.

[8] T. James and V. Rao, "Location-based emergency management system using Firebase and Android," Proc. IEEE ICCCNT, pp. 1–6, 2023.

[9] E. Santos and P. Miranda, "Volunteer-assisted crisis response platforms: Architecture, challenges, and deployment," IEEE Commun. Surv. Tutor., vol. 25, no. 2, pp. 1450–1472, 2023.

[10] K. Das and S. Banerjee, "An Android-based emergency help application using power-button triggers," IEEE Secur. Privacy, vol. 19, no. 4, pp. 28–36, 2021.

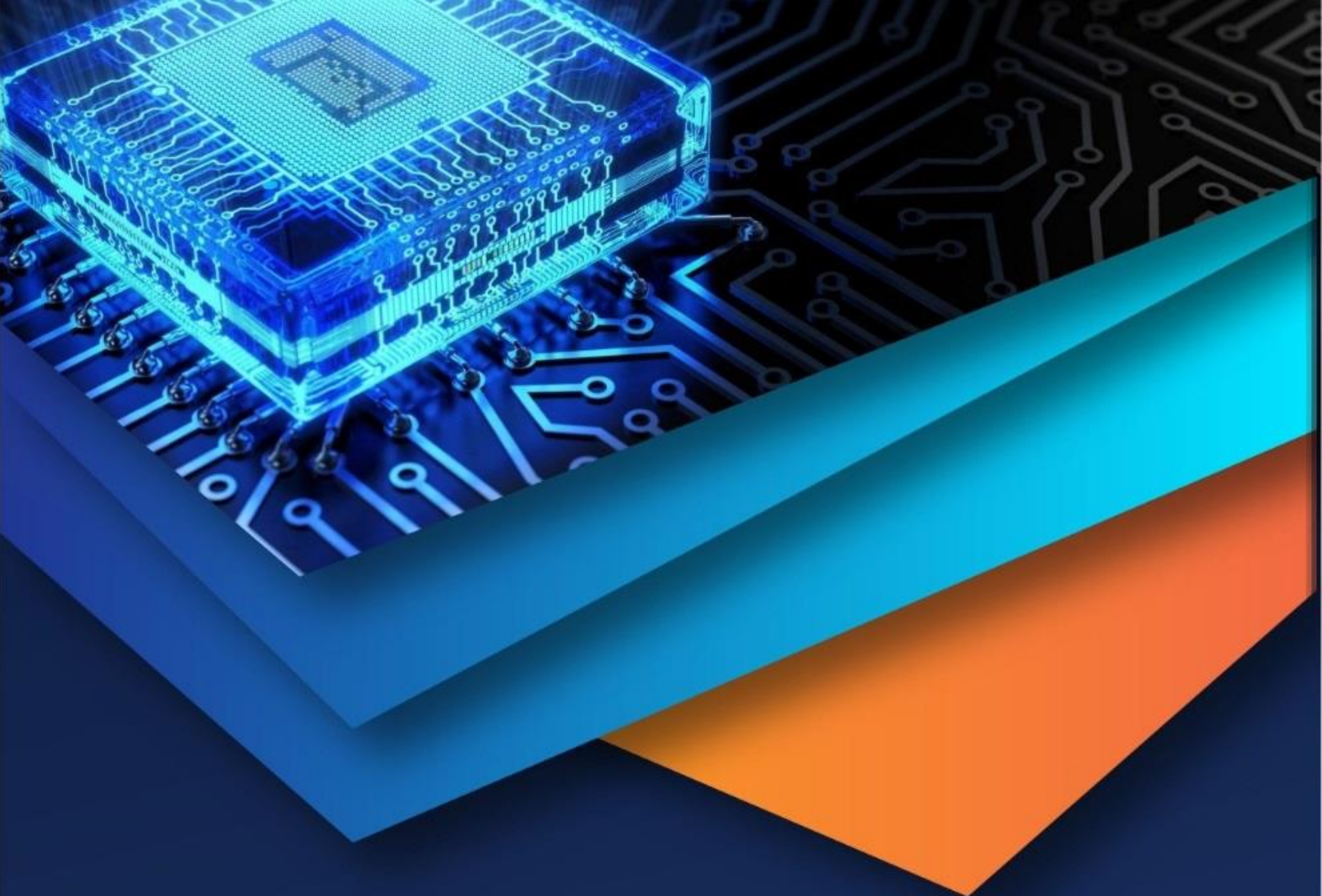
[11] J. Williams, K. Brown, and D. Smith, "Optimizing emergency notifications using cloud messaging services," IEEE Trans. Cloud Comput., vol. 11, no. 3, pp. 550–563, 2024.

[12] L. Zhao and M. Wu, "Development of real-time alerting applications using Firebase Realtime Database," IEEE Access, vol. 9, pp. 162345–162357, 2021.

[13] R. Pandey and A. Thapa, "Smartphone-based personal safety systems: A review of mobile alert technologies," IEEE Trans. Mobile Comput., vol. 20, no. 12, pp. 3315–3328, 2022.

[14] P. Mehta and S. Narayan, "Design and performance analysis of emergency communication applications using cloud backend," IEEE Trans. Netw. Service Manage., vol. 18, no. 4, pp. 4250–4264, 2021.

[15] M. George, A. Vinu, and R. Shetty, "A multi-role emergency response app with live tracking and cloud integration," Proc. IEEE ICICCT, pp. 412–418, 2024.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY