Integrated Mobile Platform for Emergency Response and AI Chat Support

R. Bhuvaneswari
Department of Computer Science and Engineering,
Rajalakshmi Engineering College,
Chennai, India
bhuvanacheran@gmail.com

Prathyush R.
Department of Computer Science and Engineering,
Rajalakshmi Engineering College,
Chennai, India
prathyushrathnakumar@gmail.com

Naveed Ahmed Basha
Department of Computer Science and Engineering,
Rajalakshmi Engineering College,
Chennai, India
naveed200505@gmail.com

***Abstract*— The rapid and efficient mobilization of emergency services remains a central challenge in urban and rural settings. This paper suggests a mobile application to provide instant and seamless assistance in emergency situations like medical emergencies, fires, or security issues. The app primarily dispatches call to police, medical or fire emergency services, as well as alerting nearby qualified volunteers or professionals to assist. Users can call for the appropriate service in a timely manner by providing crucial information, and real-time location tracking allows them to follow the responder's route. Verified responder profiles instill confidence and improve communication. The app also supports multimedia inputs, like photos, videos, and voice messages, to accurately describe the situation. Automatic alerts are also sent to emergency contacts. There's also a Gemini AI chatbot that provides instant first-aid guidance and emotional support until professional help arrives. The combination of mobile accessibility, AI-based guidance, and real-time tracking proved essential in delivering timely support. This approach aims to empower communities with a responsive and intelligent emergency management tool.**

Keywords*— *Emergency Response, Mobile App, Real-Time Tracking, AI Chatbot, Geolocation, First Aid

I. Introduction

Rapid response during emergency situations is crucial for saving lives and mitigating damage. Traditional systems often suffer from delayed communication, inadequate dispatch units, and inefficient prioritization of requests, resulting in suboptimal outcomes during time-sensitive events. These issues are especially prominent in rural areas, where emergency services face longer response times and limited resources compared to urban counterparts [1]. Furthermore, existing systems often lack adaptive decision-making and struggle with coordination under high demand, especially in chaotic or semi-structured environments [2].

However, with advancements in mobile technology and artificial intelligence, there is vast potential for improving the availability and delivery of emergency services. Emerging research highlights that intelligent traffic systems, mobile apps, and algorithmic coordination tools can significantly reduce delays and improve responder allocation in both urban and rural contexts [3]. Features such as real-time communication, multimedia reporting, and automated alerts have made mobile-based solutions increasingly practical as support systems for emergency management.

This paper presents an Emergency Services App that integrates AI, mobile interfaces, and cloud-based tracking to facilitate timely help. The application employs a map-based user interface, which provides real-time location tracking for both users and responders. The proposed system aims to reduce response times, improve resource allocation, and build a more connected and responsive emergency support network.

The paper is organized as follows: Section II reviews related literature on emergency response systems and AI model applications. Section III describes the methodology. Section IV outlines the implementation flow and user interface structure. Section V concludes with potential future enhancements and research directions.

II. LITERATURE REVIEW

Compared to existing emergency response systems, the proposed model ensures faster dispatch, improved user clarity, and better coordination through its map-based interface and AI-driven features. It supports real-time location sharing, verified responder tracking, and selection of emergency services with support for multimedia inputs and chatbot assistance. By integrating geolocation, Firebase communication, and AI-based triage logic, the system reduces response latency with a scalable and reliable solution in various emergency situations.

A cross-platform emergency reporting system for urban environments was introduced by Swapnil R. Rajput, Mohd Sohel Deshmukh, and Karbhari V. Kale [4], which focused on GIS-based location tracking using Google Maps APIs. The system allows users to report medical, fire, or police emergencies, while also sharing geocoded messages with family and emergency services. PhoneGap and HTML5 were used to ensure cross-platform compatibility, and the Haversine formula was applied to compute the nearest emergency centres.

Caliston and Tabia [5] introduced "Help Me," an Android-based emergency response mobile application designed for real-time alerts and monitoring. Key features included a panic button, push notifications through Firebase, live location tracking via leaflet maps, and video calling through agora.io. A separate web-based command centre was developed to manage users, rescuer details, and response status. The system followed ISO 9126 software evaluation standards and was rated highly in terms of functionality, reliability, and efficiency by users from various emergency departments in the Philippines.

Kalyani Pendke, Sakshi Shegokar, Rutika Malghade, Sakshi Datir, Rakshita Adhau, and Poonam Thakre [6] proposed a mobile emergency response app integrated with command centres, using geolocation to send user information like name, contact, and coordinates. It includes emergency buttons for police, ambulance, and fire, organized into tabs—Home, Info, and Hotlines. It leverages GPS-based localization and Google Maps integration to help authorities pinpoint emergencies and take timely action. The app also supports SMS-based fallback communication for devices with limited internet connectivity.

ELERTS is a mobile application widely used across the United States and is available in both mobile and web formats. It operates as a centralized platform that enables users at an emergency site to capture photos, which are instantly uploaded to a shared system. These alerts, along with the location and images, are then broadcast to other users of the application. As a result, individuals nearby can act as first responders and assist during emergencies [7]. One of the notable features of ELERTS, is its two-way communication system. When users receive an emergency alert on their phones, they can respond with additional information, helping authorities or other users better understand the situation. The platform also allows real-time reporting, enabling any smartphone user with the app to send immediate updates that include text and images. These firsthand reports act as the initial trigger for emergency notifications. Additionally, the app supports photo sharing with security personnel and other users in the vicinity, enhancing situational awareness. Leveraging GPS and mapping technologies like Google Maps, the application accurately identifies the emergency location and provides detailed information and directions to nearby responders.

While existing systems primarily focus on emergency reporting, location tracking, and low-level service notifications, our proposed approach extends further and integrates AI-powered triage assistance, smart request prioritization, and authenticated responder tracking. In contrast to existing systems based on static reporting or limited user-responder interaction, our system integrates real-time trust management, multimedia input handling, and chatbot-facilitated support. These enhancements improve situation awareness, reduce response time, and offer a more scalable and intelligent solution tailored for modern emergency response needs.

III. METHODOLOGY

The proposed mobile-based emergency response system is designed to address the limitations of traditional emergency services and improve the speed, accessibility, and reliability of assistance during critical situations [8]. The Home screen displays a map-view of the user’s current location. Users select one of the services-Police, Fire, or Medical-by clicking the button. Then, they request service after describing the situation along with photos, videos (if any). Location is shared with emergency response units as well as nearby volunteers.

If the request is accepted, the user receives ride details, including the volunteer's information, estimated time of arrival, and live route tracking. If the request is declined, users need to try again to get services.

The Account section, users can manage their personal information such as name, age, and contact number along with additional details (medical history, blood type). It connects to your emergency contacts and allows the app to alert nearby users during large-scale emergencies. The app also includes a built-in chatbot that offers first aid instructions and emotional support until help arrives, further enhancing the app's crisis response capabilities. Fig.1. depicts the proposed system.

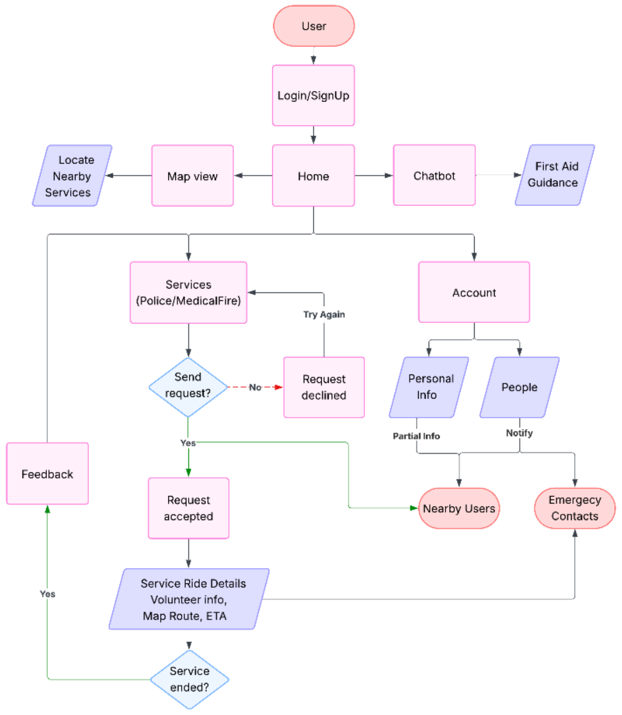

Fig.1. Block Diagram of the system

Key Features:

A. *Service Selection*

The Emergency Services App offers users a clean and intuitive interface to choose the required emergency service: Medical, Fire, or Police. Upon selection, the app gathers incident details and initiates backend processes for prioritization and dispatch. This modular interface simplifies the request process, ensuring users can act quickly under stress.

B. *Location Sharing*

Using GPS-enabled geolocation, the app captures and transmits the user's real-time coordinates is shared securely with emergency responders and nearby volunteers. Accurate location data ensures rapid deployment and reduces confusion during emergencies [9].

C. *Real-Time Responder Tracking*

The app includes a real-time tracking that allows users to track the live location of dispatch units. This feature increases user confidence, reduces panic, and helps individuals prepare for the arrival of help [10]. The tracking system also enables backend systems to log estimated response times for future performance analysis.

D. *Verified Responder Details*

Each responder or volunteer is verified through identity authentication. Upon assignment, the responder’s name, ID, profile picture, and credentials are displayed to the user. This feature establishes trust and ensures safety by guaranteeing that only authorized personnel are dispatched to assist [11].

E. Smart Request Handling

The application utilizes Random Forest and Decision Tree models to analyse incoming requests and assign priority based on urgency, severity, and proximity. The system's decision-making algorithm considers multiple factors, ensuring that life-threatening situations are addressed first. The model continuously learns from historical data to improve triage accuracy over time [12].

F. Automated Alerts to Emergency Contacts

Upon initiating a request, the system automatically sends real-time alerts to the user's pre-registered emergency contacts via SMS, in-app notifications, or email [13]. These alerts include the nature of the emergency and the user's live location, allowing family members or designated contacts to act quickly in parallel with official responders.

G. AI-Based Triage and First-Aid Guidance

An integrated Gemini AI chatbot provides immediate first-aid suggestions based on the user's inputs. Leveraging rule-based logic and classification algorithms, the system offers real-time guidance on procedures such as CPR, wound dressing, or choking response. This feature ensures basic care before professional help arrives [14].

H. Nearby User Alerts

During major emergencies, the app sends notifications to nearby users, enabling them to provide early assistance or stay informed about local risks. This enhances community involvement and situational awareness [15].

IV. IMPLEMENTATION

Developing a real-time emergency response application requires the integration of various tools and technologies to ensure seamless communication, data processing, AI functionality, and system reliability. The selection of these tools is based on scalability, cross-platform support, and performance under high-stress conditions.

A. Mobile Development Framework:

To ensure a responsive and user-friendly interface across both Android and iOS platforms, the application is developed using the Flutter SDK with Dart programming language. This cross-platform framework allows for faster development and consistent UI/UX on multiple devices from a single codebase.

B. Database Management Systems:

To manage and store user data, emergency requests, and service-related information, the app utilizes Cloud Firestore for real-time remote storage, Hive for efficient local caching, and Path Provider for accessing and managing device-level file storage.

C. Location Services & Mapping:

The application integrates Google Maps Flutter for displaying user and responder locations on an interactive map. It uses the Geolocator package for accurate real-time GPS tracking and supports location-based service discovery to identify nearby responders or facilities.

D. Voice Recognition:

To support hands-free emergency reporting, the system incorporates Google Speech-to-Text APIs. This enables users to initiate emergency service requests through voice commands, improving accessibility and usability in high-stress or injury-restrictive scenarios.

E. Cloud Services:

The backend is powered by Firebase, which provides scalable cloud infrastructure, including Authentication for secure login, Firestore for real-time data storage, and Firebase Cloud Messaging for efficient communication between devices.

F. Artificial Intelligence & Generative Models:

The application leverages Google Generative AI (Gemini models) to power its conversational chatbot. These models handle real-time user queries, offer first-aid guidance, and support emergency detection through text and vision-based inputs, enhancing situational response.

G. Messaging & Notifications:

To ensure timely communication during emergencies, the app uses Firebase Cloud Messaging (FCM) for push notifications, along with in-app alert systems to notify users, responders, and emergency contacts in real time.

H. Data Persistence & Caching:

Offline functionality and improved app performance are achieved through Hive, a lightweight local database that stores user data and request history on the device, ensuring availability even during network outages.

I. Security:

User data and app communication are secured through Firebase Security Rules and API key management. These mechanisms help enforce access control, encrypt sensitive data, and prevent unauthorized access to backend services.

J. Performance Monitoring & Analytics:

The app integrates Firebase Analytics to monitor user behaviour, feature usage, and engagement trends. Additionally, the Logger package is used to track application events and identify crashes or anomalies for continuous system improvement.

The selection of these tools is based on scalability, cross-platform support, and performance under high-stress conditions. Together, they ensure that the application remains reliable, responsive, and secure during real-time emergencies, while also offering a seamless user experience across varied network and device environments.

A. *User Registration and Authentication:*

User onboarding begins with the account creation screen, where users provide basic personal information including name, date of birth, phone number, email, and password. This information is securely stored using Firebase Authentication, enabling both email/password and social login options. Fig 2. shows the registration screen, which collects user credentials in a structured form with input validation and navigation prompts for existing users.

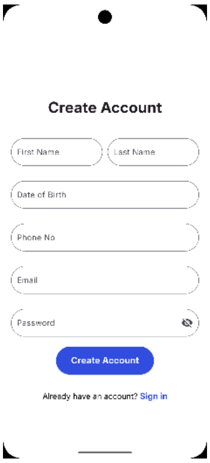

Fig. 2. SignUp Page

Users can choose to log in using either their phone number or email/password, with third-party sign-in support for Google and Apple accounts. Fig 3.a. depicts the phone number login interface, offering OTP-based verification. Fig 3.b. illustrates the email and password login screen, with options for social logins and user-friendly toggles between login modes.

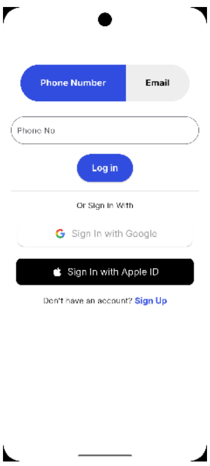

Fig. 3.a. Phone Login

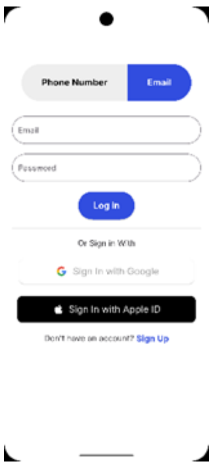

Fig. 3.b. Email Login

B. *Home Screen:*

Upon successful login, users are directed to the Home Screen, which serves as the primary interface for users to view their current location and request immediate emergency assistance. Upon launching the app, the user's real-time location is retrieved via geolocation services and displayed on an interactive map powered by Google Maps.

The screen provides quick-access buttons for major emergency services—Police, Ambulance, and Fire Department—allowing users to request help instantly with a single tap. Additional features include Notify Nearby Users, which sends alerts to others within the area, and Automatic Detection, which uses AI-driven mechanisms to identify emergencies without user input.

At the bottom, the Immediate AID section allows users to specify the nature of their emergency—Medical, Fire, or Other—and either type a request or launch the chatbot via the Open button. This acts as a gateway to AI-driven assistance, seamlessly guiding users in high-stress situations through conversation or direct text input.

The Home interface is shown in Fig 4., highlighting the integrated emergency tools, real-time map, and chatbot launch options within a clean and responsive layout. Fig 5. shows the expanded map view, highlighting nearby critical infrastructure such as hospitals, police stations, and fire stations.

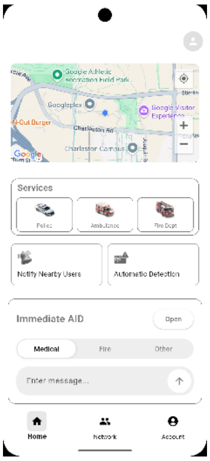

Fig. 4. Home Page

Fig. 5. Map-view

C. *AI Chatbot Screen:*

This chatbot is powered by Google's Gemini AI and provides emergency-specific responses. Users can type queries or situations, and the chatbot returns safety instructions or protocols. The AI can handle both text-based and context-aware queries. A chat history option lets users revisit past conversations.

Fig 6.a. presents the chatbot's home interface, greeting the user and prompting message input. Fig 6.b. depicts the chat history, allowing users to revisit past interactions for consistent support.

Fig. 6.a. Chat Screen

Fig. 6.b. Chat History

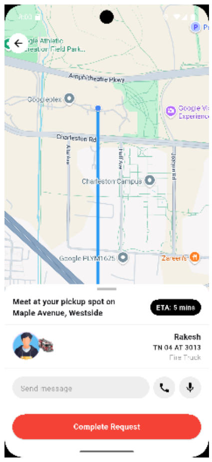

Fig. 7.c. Responder Details

D. Emergency Service Request Flow (Fire Service Use Case):

The emergency service flow is demonstrated through a fire department request sequence. Users are prompted to describe their situation and share their location. Upon request submission, the system begins matching nearby aid resources or responders.

Fig 7.a. shows the fire service input screen, where users provide details and optionally share GPS location. Fig 7.b. illustrates the real-time request matching process, highlighting nearby support. Fig 7.c. confirms the final request stage, showing ETA, responder information, and communication options like messaging or calling the responder.

E. User Profile and Account Management:

The Profile tab enables users to manage their identity, security, data privacy, and app interactions. Basic profile information includes name, contact details, and account metadata.

Fig 8.a. displays the user profile landing page. Fig 8.b. provides access to personal information, including date of birth, email, and phone number. Fig 8.c. outlines the People and Sharing screen, which includes contact synchronization, friend management, and location-sharing controls. Fig 8.d. features the security settings, where users can manage passwords, enable two-factor authentication, and view login activity logs.

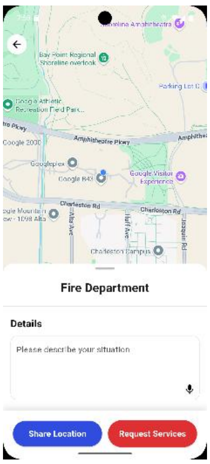

Fig. 7.a. Service Input

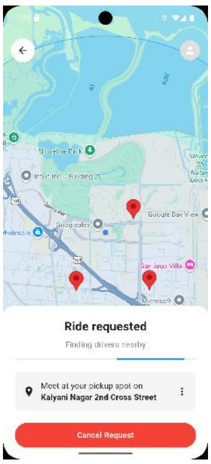

Fig. 7.b. Request Service

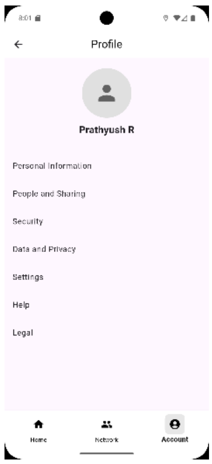

Fig. 8.a. Profile Page

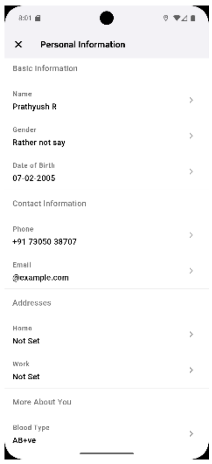

Fig. 8.b. Personal Info

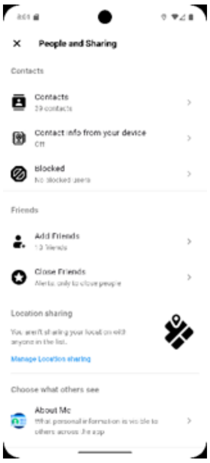

Fig. 8.c. People & Sharing

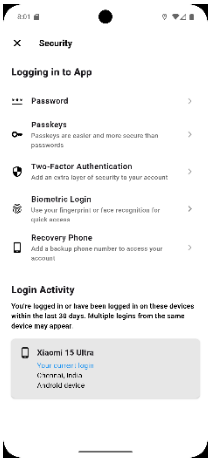

Fig. 8.d. Security

V. CONCLUSION

This paper presents a smart emergency reporting and response system using mobile technologies, real-time GPS tracking, and cloud-based services to assist users during critical situations. By leveraging smartphone capabilities, our system enables users to request help from police, medical, or fire services with a single tap while automatically sharing their live location with verified responders. The integration of Gemini AI for first-aid guidance and emotional support further strengthens the app's ability to assist users before professional help arrives.

This system addresses the need for rapid, reliable, and accessible emergency communication, especially in situations where traditional media may fail or be delayed. With the support of Google Maps-based geolocation and real-time data sync via Firebase, command centers and emergency units can easily detect and plot the user's position on a live map and dispatch the nearest available responders. The aim is not to replace existing emergency protocols, but to complement them by utilizing modern mobile and AI technologies to improve response time and public safety. In the future, this system can be expanded to include analytics for predicting emergency trends based on historical data, improved responder coordination through AI optimization, and enhanced media sharing using geo-tagged images and videos for better situational awareness.

REFERENCES

[1] [A. Alanazy, Stuart Wark, John Fraser, Amanda Nagle, "Factors Impacting Patient Outcomes Associated with Use of Emergency Medical Services Operating in Urban Versus Rural Areas: A Systematic Review," Int. J. Environ. Res. Public Health, vol. 16, no. 10, 2019.

[2] N.C. Simpson and P.G. Hancock, "Fifty years of operational research and emergency response," J. Oper. Res. Soc., vol. 60, no. 1, pp. 126–139, 2009.

[3] G. Pettet, Ayan Mukhopadhyay, Mykel J. Kochenderfer, Abhishek Dubey, "On Algorithmic Decision Procedures in Emergency Response Systems in Smart and Connected Communities," Adaptive Agents and Multi-Agent Systems, 2020.

[4] S. R. Rajput, M. S. Deshmukh, and K. V. Kale, "Cross-platform Smartphone Emergency Reporting Application in Urban Areas using GIS Location-based and Google Web Services," Int. J. Comput. Appl., vol. 130, no. 12, pp. 1–6, Nov. 2015.

[5] N. P. Caliston and G. C. Tabia, "HELP ME: An Emergency Response Mobile Application," J. Eng. Sci. Technol. Special Issue on ICITE2021, pp. 47–63, Oct. 2021, © School of Engineering, Taylor's University.

[6] K. Pendke, S. Shegokar, R. Malghade, S. Datir, R. Adhau, and P. Thakre, "Mobile Application for Emergency Services," JETIR, vol. 8, no. 5, May 2021.

[7] ELERTS Corporation, "Official ELERTS website," Jan. 2013. [Online]. Available: http://elerts.com

[8] R. Repanovici, A. Rusu, A. Vasilache, and R. Oltean, "Improvement of emergency situation management through an integrated system using mobile alerts," Sustainability, vol. 14, no. 24, pp. 1–15, 2022.

[9] A. Khan, F. Muhammad, M. Ullah, H. Khan, M. Shoaib, and M. A. Khan, "Accident detection and smart rescue system using Android smartphone with real-time location tracking," in Proc. Int. Conf. on Advances in Computing, Communication Control and Networking (ICACCCN), 2018, pp. 1–5.

[10] L. K. Nuevas, J. L. C. Velarde, J. B. A. Javellana, and J. R. E. Torlao, "Real-time incident reporting and emergency response: A mobile GPS app for Abuyog, Leyte, Philippines," in Proc. 7th Int. Conf. Vocational Education and Electrical Engineering (ICVEE), 2024, pp. 37–43

[11] S.-Y. Chiou and J.-W. Lin, "A real-time, automated and privacy-preserving mobile emergency-medical-service network for informing the closest rescuer to rapidly support mobile-emergency-call victims," IEEE Access, vol. 6, pp. 68962–68974, 2018.

[12] H. Mutegeki, R. S. Rajamani, D. Thirumalai, and M. Kumar, "Interpretable Machine Learning-Based Triage for Decision Support in Emergency Care," Proc. 7th Int. Conf. Trends Electron. Inform. (ICOEI), 2023, pp. 1234–1239.

[13] F. Sposaro and G. S. Tyson, "iFall: An android application for fall monitoring and response," in Proc. 2009 Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC), Minneapolis, MN, USA, 2009, pp. 6119–6122

[14] N. Ouerhani, M. Ben Halima, and A. M. Alimi, "Towards a Chatbot Based Smart Pervasive Healthcare Medical Emergency Cases," in Proc. 2020 Int. Conf. on Advanced Systems and Emergent Technologies (IC_ASET), 2020, pp. 288–293.

[15] A. Valeriano, M. Van Heer, C. Perel, and D. Prieto Merino, "Crowdsourcing to Save Lives: A Scoping Review of Bystander Alert Technologies for Out-of-Hospital Cardiac Arrest," Resuscitation, vol. 152, pp. 164–172, 2020.