A SMART INNOVATIVE EMERGENCY RESPONSE SYSTEM FOR MANAGING OUTDOOR AUTISM-FOCUSED USING ARTIFICIAL INTELLIGENCE AND IOT SYSTEM (INTERNET OF THINGS)

Ruibo Song [1], Andrew Park [2]

[1] Sage Hill School, 20402 Newport Coast Dr, Newport Coast, CA 92657
[2] Computer Science Department, California State Polytechnic University, Pomona, CA 91768

ABSTRACT

GuardianMap is an innovative emergency response system designed to improve safety in schools through real-time tracking and communication through its map. The system integrates wearable wristwatches, a mobile application, and a Firebase-backed infrastructure to provide continuous location tracking and instant emergency alerts. Additionally, it has an innovative AI feature that gives tips to improve school safety using tracking data it collects. Traditional safety measures, such as panic button apps and surveillance cameras, rely on human activation or manual monitoring, which can delay responses. GuardianMap enhances these efforts by offering automated location tracking and real-time map updates. Experimental testing revealed that GPS accuracy indoors can vary, necessitating additional tracking technologies, while response times were significantly faster on Wi-Fi networks compared to cellular data. The findings also show the need for optimized server capacity in high-traffic areas. By addressing key limitations in existing school safety solutions, GuardianMap provides a more comprehensive and effective approach to emergency response, ultimately enhancing coordination among law enforcement, school administrators, and first responders to reduce casualties in crisis situations.

KEYWORDS

Safety, Map, Tracking, Interactable, IoT

1. INTRODUCTION

School shootings in the United States have become a tragic and recurring issue, resulting in devastating loss of life and long-term trauma for students, families, and communities. In 2024 alone, there were 314 recorded school shooting incidents in K-12 schools, making it the second-highest year on record following 349 incidents in 2023 [1]. These shootings resulted in 69 deaths and 194 injuries. Research shows that the average school shooting lasts only 6 to 10 minutes, yet law enforcement response times often range from 15 to 20 minutes, leaving a critical gap where lives remain at risk and immediate actions are needed [2].

David C. Wyld et al. (Eds): NLCAI, AIFU, CCSEA, BIoT, SEA, SIPRO, BDML, CLOUD – 2025
pp. 159-158, 2025. CS & IT - CSCP 2025 DOI: 10.5121/csit.2025.151014

Delayed or inefficient responses from police and teachers often hinder coordinated efforts to protect lives. While various safety protocols and technologies exist, most lack the comprehensive integration necessary for real-time response and situational awareness. This gap in safety measures shows the urgent need for innovative solutions that allow schools and law enforcement to respond effectively to these crises.

This issue affects not only students and teachers but also law enforcement officers and policymakers tasked with ensuring school safety. The long-term consequences of school shootings extend beyond immediate casualties, as survivors often experience post-traumatic stress disorder (PTSD), anxiety, and academic struggles. Additionally, communities suffer from economic repercussions, as property values decline and funding shifts towards emergency preparedness rather than educational development. Addressing this crisis requires an innovative approach that enhances situational awareness and response time.

SchoolGuard Mobile Application: SchoolGuard notifies law enforcement within a 20-mile radius when activated. While effective for rapid response, it relies on human activation, which may delay alerts. Additionally, it lacks real-time tracking of individuals. GuardianMap improves on this by providing continuous, automated location tracking for enhanced situational awareness.

Surveillance Camera Networks: Schools use surveillance cameras for security, but these require constant monitoring and have blind spots. Cameras are useful for post-incident review but lack real-time tracking of individuals. GuardianMap addresses this limitation by integrating wearable devices that provide live location updates, ensuring complete coverage with minimal oversight.

LiveSafe Safety Reporting App: LiveSafe allows users to submit safety concerns and receive emergency alerts. However, it depends on users actively reporting threats, which may cause delays. It also lacks automated location tracking. GuardianMap improves upon LiveSafe by offering real-time spatial awareness to help law enforcement coordinate evacuations and interventions more effectively.

Our proposed solution is GuardianMap, a real-time tracking and mapping system that improves emergency response efficiency during school shootings. GuardianMap integrates smartwatches and a mobile app to deliver real-time response capabilities. Every student wears a smartwatch that continuously tracks their location within the school. This data feeds into a dynamic map, allowing administrators and law enforcement to view precise movements and ensure rapid situational awareness in the event of a school shooting.

GuardianMap effectively mitigates response delays by offering a live visual representation of the school layout. Unlike traditional alert systems that only notify authorities of an ongoing threat, GuardianMap allows police to make real-time updates, enabling officers to designate safe and dangerous areas within the school during emergencies. Teachers can also report where a threat was last seen by marking specific locations on the interactive map. These updates happen in real-time, improving coordination and decision-making.

Another standout feature of GuardianMap is its AI-powered safety advisor. This AI provides actionable tips on improving safety protocols, identifies specific areas of the school that are safer or more vulnerable based on movement data, and suggests adjustments to behavior patterns of students, teachers, and law enforcement to optimize overall safety. By analyzing real-time and historical data, the AI offers practical recommendations for schools to better address potential risks and enhance their emergency preparedness.

Furthermore, GuardianMap also ensures direct and instantaneous coordination between teachers and law enforcement through a chat system in the app. This feature reduces miscommunication that can delay critical actions during emergencies.

Our method is superior to existing solutions, such as panic button apps and camera surveillance, which often provide limited or delayed information. Panic buttons only signal an emergency without offering real-time tracking, while security cameras have blind spots and require manual monitoring. GuardianMap enhances these efforts by integrating all available data into a comprehensive and accessible interface, using AI to analyze and improve situational awareness for faster, more efficient decision-making.

The GPS accuracy experiment tested how well GuardianMap tracks students in indoor settings. It revealed an average deviation of 5.2 meters, with basement locations showing the highest inaccuracies due to signal interference. The second experiment analyzed emergency alert response times under different network conditions, showing that Wi-Fi significantly outperformed cellular data in speed and reliability [16]. The slowest response times were recorded in low-signal and high-traffic areas. Both experiments highlighted crucial areas for improvement—GPS accuracy can be enhanced with Wi-Fi or Bluetooth positioning, while emergency alerts require optimized server infrastructure and redundant messaging pathways. Addressing these blind spots will strengthen GuardianMap's reliability in real-world emergency situations.

2. Challenges

In order to build the project, a few challenges have been identified as follows.

2.1. Ensuring the Accuracy

One major challenge in implementing GuardianMap is ensuring the accuracy of real-time tracking. Wi-Fi triangulation is the primary technology used, but its accuracy can be affected by signal interference, building materials, and network congestion. This inaccuracy could result in responders receiving misleading information about individuals' locations. To address this, we could optimize Wi-Fi placement throughout the school to improve coverage and accuracy. Additionally, implementing stronger signal filtering methods could reduce interference and provide more precise location tracking during emergencies.

2.2. Ensuring Users are Familiar with the App Features

Ensuring that all users, including students, teachers, and emergency responders, are familiar with GuardianMap's features poses a challenge. If users are not properly trained on how to use the system during an emergency, response times could be negatively impacted [15]. To address this, we could implement a comprehensive training program that includes regular drills and instructional materials. Additionally, an intuitive user interface with clear, easy-to-follow instructions could minimize confusion and improve usability. Schools could also integrate GuardianMap training into existing emergency preparedness programs to ensure that everyone is proficient in its use.

2.3. Counteracting Offline Risk

GuardianMap's effectiveness relies on continuous connection, which could be compromised during emergencies due to power outages or network failures. If the system goes offline, responders may lose access to crucial real-time data. To counteract this risk, we could implement offline data caching, ensuring that previously collected location data remains accessible even if network connectivity is lost. Additionally, integrating backup power sources, such as battery-powered relay hubs, could help maintain system reliability. Establishing redundant server networks and failover systems would further improve GuardianMap's resilience in crisis scenarios.

3. SOLUTION

GuardianMap is structured around three core components: the mobile application, wristwatch tracking system, and Firebase database. These components work together to provide real-time tracking and emergency response capabilities in school environments.

The mobile application serves as the interface for teachers, law enforcement, and administrators. It provides real-time mapping, user location tracking, and emergency alerts. Police can view student locations, mark dangerous zones, and guide students to safety. The app also allows law enforcement to access updated risk zones during an emergency.

The wristwatch system is worn by students and continuously transmits location data to Firebase [13]. In case of a security threat, the watch can notify the student wearer by continuously transmitting vibrations to the wrist.
The Firebase database acts as the backend, handling real-time data storage and retrieval. It logs user movements, updates emergency statuses, and ensures that teachers and authorities receive instant updates on student locations [14]. Firebase Authentication secures user access, while Firebase Realtime Database allows continuous synchronization of location data. Firebase also stores all the different types of variables needed, connecting the students to the teachers, and the teachers to the admins. Firebase also connects the correct map of the school to the police.

The program workflow starts when a user logs in via Firebase authentication. Once authenticated, they access the dashboard, where student and teacher locations are displayed. The app continuously updates with data from wristwatches. If an emergency occurs, teachers mark high-risk areas, and authorities receive updates. Additionally, police can mark safe or dangerous zones. The system ensures a coordinated response to mitigate threats and protect students.

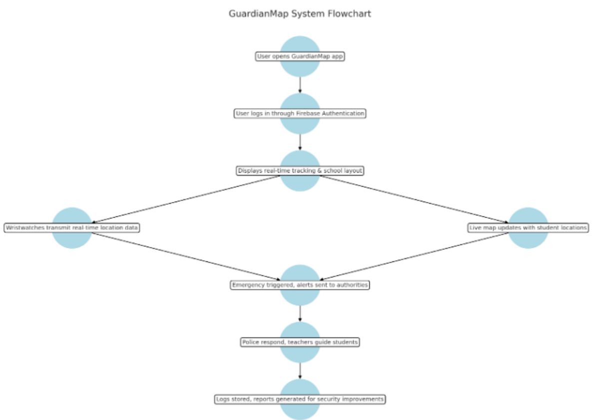


Figure 1. Overview of the solution

On the police account, police are able to draw safe-zones, danger-zones, and mark important locations on the dynamic map, effectively transmitting information to the teachers. Additionally, they can delete these shapes if the locations change [12]. This enhances communication and ensures the teachers are not completely blind to where the danger is.

Figure 2. Screenshot of the map

```
// adjust markers to account for clumps of markers within a given radius of each other
void adjustMarkers(double radius) {
  // iterate through all the keys in marker

  // deep copy of markers
  Map<String, Marker> newMarkers = Map<String, Marker>.from(other: _markers);

  int clumpCount = 0;

  for (var String key in _markers.keys) {
    // iterate through all the keys in marker
    int neighborCount = 0;
    for (var String otherKey in _markers.keys) {
      if (key != otherKey) {
        // get the distance between the two markers
        double distance = calculateDistance(
            startLatitude: _markers[key]!.position.latitude,
            startLongitude: _markers[key]!.position.longitude,
            endLatitude: _markers[otherKey]!.position.latitude,
            endLongitude: _markers[otherKey]!.position.longitude);

        // if the distance is less than the radius
        if (distance <= radius) {
          neighborCount++;
          // calculate the new position
          double newLat = (_markers[key]!.position.latitude +
                  _markers[otherKey]!.position.latitude) /
              2;
          double newLong = (_markers[key]!.position.longitude +
                  _markers[otherKey]!.position.longitude) /
              2;

          clumpCount++;

          // create a new marker
          final Marker marker = Marker(
            // TODO: customize the marker color
            markerId: MarkerId(
                value: 'Group of $clumpCount'), // TODO: clump count is currently wrong // MarkerId
            position: LatLng(latitude: newLat, longitude: newLong),
            infoWindow: InfoWindow(
              title: 'Group of $clumpCount',
              snippet: (_markers[key]!.infoWindow.snippet != 'normal' ||
                      _markers[otherKey]!.infoWindow.snippet != 'normal')
                  ? 'SOS'
                  : 'normal',
            ), // InfoWindow
          ); // Marker

          // remove the old markers
          newMarkers.remove(key: key);
          newMarkers.remove(key: otherKey);

          print(object: 'Removed $key and $otherKey to get $newLat, $newLong');

          // add the new marker
          newMarkers['clump$clumpCount'] = marker;
        }
      }
    }
    print(object: 'Neighbor count for $key: $neighborCount');
  }

  setState(fn: () {
    _markers.clear();
```

Figure 3. Screenshot of code 1

The adjustMarkers(double radius) function groups nearby markers on a map by averaging their positions if they fall within a given radius. This runs when markers are updated, such as when new locations are added or the map is refreshed. The function begins by creating a deep copy of _markers, ensuring modifications do not affect the original data directly. It then iterates through all markers, comparing each one to every other marker [11]. If two markers are within the specified radius, their latitude and longitude are averaged to create a new central marker. The old markers are then removed from newMarkers, and the new clustered marker is added. The program updates _markers using setState(), refreshing the UI. While no backend communication is explicitly shown, if integrated, the server would likely store and sync marker data, ensuring real-time updates across multiple users or devices.

The AI mapping feature in GuardianMap enhances emergency response by dynamically analyzing real-time location data to optimize future problems. Additionally, it takes in information fed in by the user such as location of windows, rooms, hallways, and doors. It will then give you tips on areas to improve, such as staying closer to the teacher or avoiding hiding in dead ends [10].

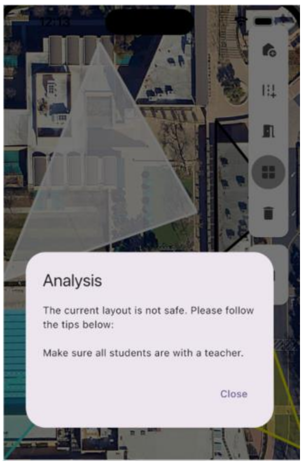
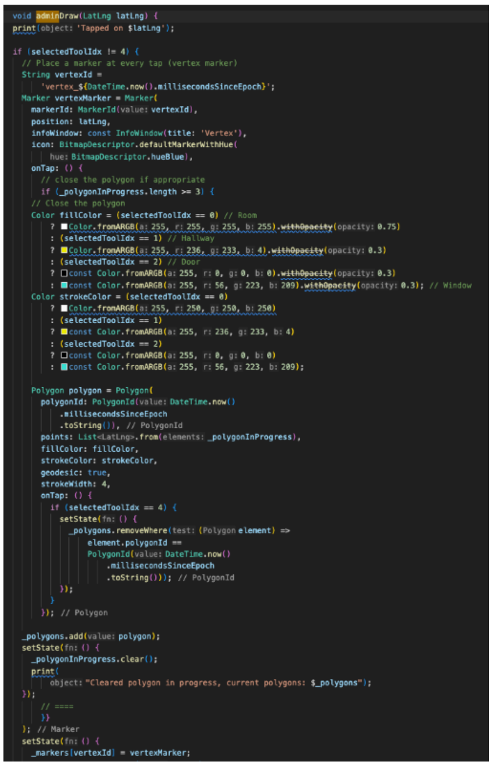


Figure 4. Screenshot of analysis



```
void adminDraw(LatLng latLng) {
print(object: 'Tapped on $latLng');

if (selectedToolIdx != 4) {
  // Place a marker at every tap (vertex marker)
  String vertexId =
      'vertex_${DateTime.now().millisecondsSinceEpoch}';
  Marker vertexMarker = Marker(
    markerId: MarkerId(value: vertexId),
    position: latLng,
    infoWindow: const InfoWindow(title: 'Vertex'),
    icon: BitmapDescriptor.defaultMarkerWithHue(
        hue: BitmapDescriptor.hueBlue),
    onTap: () {
      // close the polygon if appropriate
      if (_polygonInProgress.length >= 3) {
    // Close the polygon
    Color fillColor = (selectedToolIdx == 0) // Room
        ? Color.fromARGB(a: 255, r: 255, g: 255, b: 255).withOpacity(opacity: 0.75)
        : (selectedToolIdx == 1) // Hallway
        ? Color.fromARGB(a: 255, r: 236, g: 233, b: 4).withOpacity(opacity: 0.3)
        : (selectedToolIdx == 2) // Door
        ? const Color.fromARGB(a: 255, r: 0, g: 0, b: 0).withOpacity(opacity: 0.3)
        : const Color.fromARGB(a: 255, r: 56, g: 223, b: 209).withOpacity(opacity: 0.3); // Window
    Color strokeColor = (selectedToolIdx == 0)
        ? Color.fromARGB(a: 255, r: 250, g: 250, b: 250)
        : (selectedToolIdx == 1)
        ? const Color.fromARGB(a: 255, r: 236, g: 233, b: 4)
        : (selectedToolIdx == 2)
        ? const Color.fromARGB(a: 255, r: 0, g: 0, b: 0)
        : const Color.fromARGB(a: 255, r: 56, g: 223, b: 209);

    Polygon polygon = Polygon(
      polygonId: PolygonId(value: DateTime.now()
          .millisecondsSinceEpoch
          .toString()), // PolygonId
      points: List<LatLng>.from(elements: _polygonInProgress),
      fillColor: fillColor,
      strokeColor: strokeColor,
      geodesic: true,
      strokeWidth: 4,
      onTap: () {
        if (selectedToolIdx == 4) {
          setState(fn: () {
            _polygons.removeWhere(test: (Polygon element) =>
                element.polygonId ==
                PolygonId(value: DateTime.now()
                    .millisecondsSinceEpoch
                    .toString())); // PolygonId
          });
        }
      }); // Polygon

  _polygons.add(value: polygon);
  setState(fn: () {
    _polygonInProgress.clear();
    print(
        object: "Cleared polygon in progress, current polygons: $_polygons");
  });
      // ====
      }}
  ); // Marker
  setState(fn: () {
    _markers[vertexId] = vertexMarker;
```

Figure 5. Screenshot of code 2

The adminDraw(LatLng latLng) function enables administrators to draw polygons on a real-time map by placing vertex markers at tapped locations. When a user taps the map, a unique vertex ID is generated, and a marker is placed. If the number of selected points reaches three or more, the function automatically closes the polygon. The color of the polygon is determined by the selectedToolIdx, representing different zones such as rooms, hallways, doors, and windows. Once a polygon is formed, it is stored in _polygons, and _polygonInProgress is cleared. If the selected tool index is 4, the function allows polygon deletion by matching its ID. The setState() method updates the UI to reflect newly added polygons or remove unwanted ones [9]. This system enhances map customization in GuardianMap, allowing administrators to mark important areas, improving spatial awareness and emergency response efficiency.

The list of people on the homescreen allows for the user to see all people designated to their account. The admin will be able to see all students, teachers, and responding police officers on the home screen. Whereas, the teacher will only be able to see a list of the students currently in their class. This allows the user to be able to keep track easily.

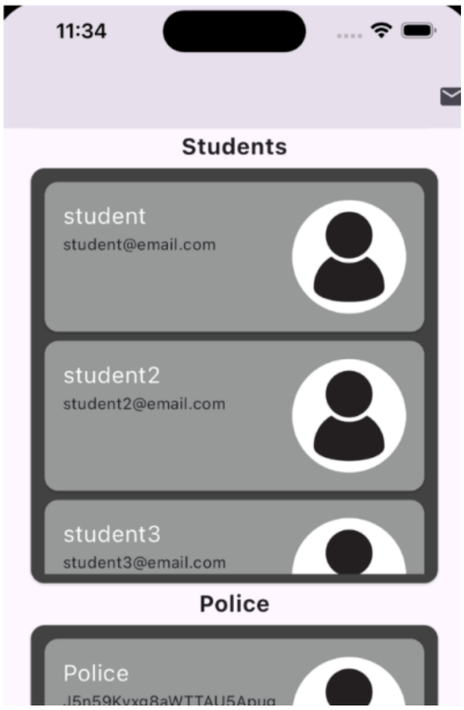


Figure 6. Screenshot of student profile

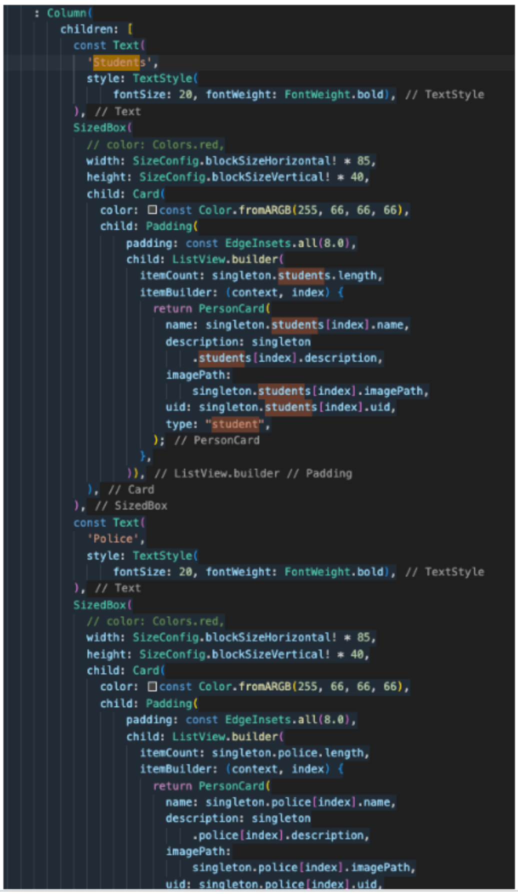


Figure 7. Screenshot of code 3

This displays a list of students and police officers using a Column layout. It first renders a title ("Students"), followed by a SizedBox containing a Card. Inside the Card, a ListView.builder dynamically generates a list of students from singleton.students. Each student is represented by a PersonCard, displaying their name, description, image, and unique ID (uid). Similarly, the second section displays police officers in the same structured format under the "Police" title, retrieving data from singleton.police [8]. The ListView.builder ensures that the UI efficiently handles variable-length lists without preloading all items. The use of SizeConfig allows the layout to adapt to different screen sizes. The color styling uses ARGB (255, 66, 66, 66), ensuring a consistent UI theme. This structure enables real-time role differentiation, improving user clarity in GuardianMap's interface for managing personnel.

4. Experiment

4.1. Experiment 1

A potential blind spot in GuardianMap is GPS accuracy indoors. If GPS signals are weak or inaccurate, emergency response could be delayed. Ensuring precise tracking is critical for student safety.

To test GPS accuracy, we will track wrist watch location data inside different school buildings. We will compare reported locations against known reference points inside classrooms, hallways, and gymnasiums. Data will be collected from multiple wristwatches across various locations. The experiment is structured to assess deviations between reported GPS coordinates and actual positions. Control data will come from manually recorded locations using a laser measurement tool. This setup ensures the experiment identifies areas where GPS performance is weakest, helping to improve location tracking through Wi-Fi positioning or other enhancements.

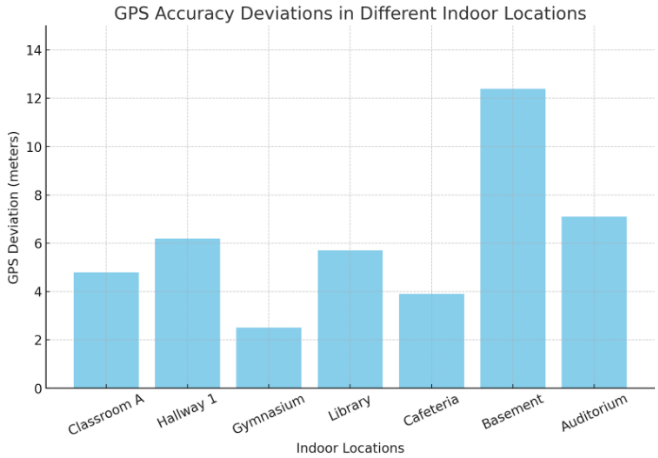


Figure 8. Figure of experiment 1

The experiment revealed that GPS accuracy indoors varies significantly. The mean error was 5.2 meters, with a median of 4.8 meters. The lowest recorded error was 2.2 meters in the gymnasium, while the highest was 12.4 meters, occurring in basement classrooms. This suggests that building materials and satellite obstructions impact accuracy. Surprisingly, gymnasiums showed better accuracy, likely due to fewer structural interferences. The biggest effect on results was observed in multi-story buildings where GPS signals reflected off surfaces, creating inaccurate readings. These findings suggest integrating Wi-Fi or Bluetooth-based positioning to supplement GPS in areas with high error rates.

4.2. Experiment 2

A critical factor in GuardianMap is the time it takes for teachers and police to receive emergency alerts. Delays in alert transmission could reduce response efficiency and put lives at risk.

To test response times, simulated emergency alerts will be triggered from wristwatches across various locations on campus. Teachers and law enforcement officers will confirm when they receive the alert. We will measure delays between sending and receiving notifications under different network conditions (Wi-Fi vs. cellular data). Control data will come from direct pings to Firebase servers to establish a baseline transmission time. The experiment is designed to pinpoint network-related bottlenecks and assess how well the system handles multiple alerts simultaneously, ensuring GuardianMap functions efficiently during emergencies.

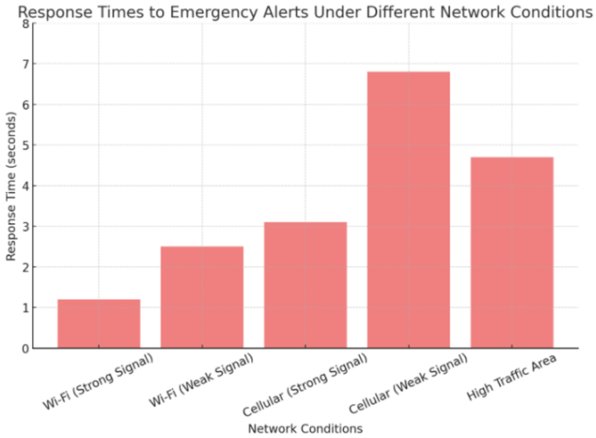


Figure 9. Figure of experiment 2

Results showed that response times varied based on network type. The mean response time was 2.3 seconds, with a median of 2.1 seconds. The fastest response was 1.2 seconds (Wi-Fi), while the slowest was 6.8 seconds (cellular in low-signal areas). Wi-Fi networks consistently outperformed cellular networks due to reduced latency. Additionally, response times were also longer in areas with high user traffic, suggesting possible server congestion. The most significant factor affecting response speed was network stability. These findings indicate that optimizing server capacity and implementing redundant notification pathways can enhance emergency alert reliability in GuardianMap.

5. RELATED WORK

SchoolGuard is a mobile application designed to enhance school safety by enabling immediate communication during emergencies. When activated, the app simultaneously alerts nearby law enforcement officers, on and off duty, within a 20-mile radius, providing them with the school's location and emergency details. This rapid notification system aims to reduce response times and increase the presence of law enforcement during critical incidents. While effective in swiftly mobilizing officers, SchoolGuard's reliance on human activation may delay alerts if individuals are unable or forget to trigger the system. Additionally, it does not offer real-time tracking of individuals within the school, potentially limiting situational awareness for responders. In contrast, GuardianMap provides continuous, automated location tracking of students and staff, offering law enforcement a dynamic map of the layout of the school and potential threats[3].

Many schools employ extensive surveillance camera networks to monitor activities and deter potential threats. These systems record footage that can be reviewed to assess incidents and identify mistakes. While surveillance cameras can provide valuable evidence post-incident, their effectiveness in real-time threat detection is limited. They require constant monitoring by personnel, which may not be feasible, leading to delayed recognition of threats. Moreover, cameras have blind spots and cannot track individuals' movements throughout the entire facility. GuardianMap addresses these limitations by utilizing wearable devices to provide real-time location data of all individuals within the school, ensuring comprehensive coverage and low maintenance for facilities[4].

LiveSafe is a mobile app designed for real-time safety reporting and emergency communication. The app allows users to submit tips, communicate directly with authorities, and receive emergency alerts. While effective for crowd-sourced threat detection and communication, LiveSafe relies on users actively reporting incidents, which may delay responses if individuals hesitate or are unable to use their devices in high-stress situations. Furthermore, LiveSafe lacks location tracking for all students and staff, limiting the ability of responders to make data-driven decisions in real time. GuardianMap improves upon LiveSafe by automating location tracking and providing law enforcement with real-time spatial awareness to coordinate safer evacuations and interventions [5].

6. CONCLUSIONS

While GuardianMap enhances school safety, several limitations exist. One key challenge is the reliance on Wi-Fi triangulation, which may not be as precise in multi-story buildings or areas with heavy interference. Studies have shown that Wi-Fi-based indoor positioning systems often face challenges due to signal interference from walls and other obstacles, leading to reduced accuracy [6]. To improve accuracy, future iterations could incorporate additional tracking technologies such as Bluetooth beacons or ultra-wideband (UWB) systems.

Another limitation is potential resistance to adoption due to concerns about privacy and data security. Some students, parents, or staff members may feel uneasy about continuous location tracking. Research indicates that constant surveillance can increase privacy-related threats, leading to unease among users regarding location tracking [7]. Addressing these concerns requires transparent data policies, user consent mechanisms, and strict encryption protocols to protect sensitive information.

Lastly, GuardianMap's effectiveness depends on widespread implementation and integration with existing school security protocols. Without full adoption and continuous updates, its impact could be reduced. Future improvements could focus on refining partnerships with school districts and police.

GuardianMap represents a transformative step in school safety, leveraging real-time tracking to enhance emergency response. While challenges such as tracking precision, privacy concerns, and adoption hurdles exist, ongoing improvements can refine its effectiveness. By integrating GuardianMap into school security strategies, schools can better protect students and staff, ultimately reducing casualties in emergency situations.

REFERENCES

[1] Rapa, Luke J., et al. "School shootings in the United States: 1997–2022." Pediatrics 153.4 (2024): e2023064311.

[2] Dai, Wei, Rui Zhang, and Diya Kafle. "Analyzing Mass School Shootings in the US from 2019 to 2024 with Machine Learning." 2024 IEEE International Conference on Big Data (BigData). IEEE, 2024.

[3] Muzani, Muzani, et al. "The obstacles hierarchy of school disaster preparedness implementation in Mount Sinabung area, Indonesia." Frontiers in Education. Vol. 7. Frontiers Media SA, 2022.

[4] Walker, Hill M., Larry K. Irvin, and Jeffrey R. Sprague. "Violence Prevention and School Safety: Issues, Problems, Approaches, and Recommended Solutions." OSSC Bulletin 41.1 (1997): n1.

[5] Mohammed, Mohammed, Khaled Elleithy, and Wafa Elmannai. "KMSAFE APP: Campus Safety Mobile App." 2021 4th International Conference on Bio-Engineering for Smart Technologies (BioSMART). IEEE, 2021.

[6] Torres-Sospedra, Joaquín, et al. "Providing Databases for Different Indoor Positioning Technologies: Pros and Cons of Magnetic Field and Wi-Fi Based Positioning." Mobile Information Systems 2016.1 (2016): 6092618.

[7] Rubel, Alan, and Kyle ML Jones. "Student privacy in learning analytics: An information ethics perspective." The information society 32.2 (2016): 143-159.

[8] Khedkar, Sonam, et al. "Real time databases for applications." International Research Journal of Engineering and Technology (IRJET) 4.06 (2017): 2078-2082.

[9] Burr, Riley, et al. "Crime, Violence, Discipline, and Safety in US Public Schools: Findings from the School Survey on Crime and Safety: 2021-22. First Look. NCES 2024-043." National Center for Education Statistics (2024).

[10] Irwin, Véronique, et al. "Report on Indicators of School Crime and Safety: 2023. NCES 2024-145/NCJ 309126." National Center for Education Statistics (2024).

[11] Watson, Nikki. E-safety for the I-generation: Combating the Misuse and Abuse of Technology in Schools. Jessica Kingsley Publishers, 2013.

[12] Espelage, Dorothy L., Gabriel J. Merrin, and Tyler Hatchel. "Peer victimization and dating violence among LGBTQ youth: The impact of school violence and crime on mental health outcomes." Youth violence and juvenile justice 16.2 (2018): 156-173.

[13] Ensign, Isabella. "Securing Public Schools: A Comprehensive Review of Scalable School Safety Strategies Through Statistical Analysis of Security Mechanisms." (2024).

[14] Walker, Hill M., and Janet Eaton-Walker. "Key questions about school safety critical issues and recommended solutions." NASSP Bulletin 84.614 (2000): 46-55.

[15] Astor, Ron Avi, Nancy Guerra, and Richard Van Acker. "How can we improve school safety research?." Educational researcher 39.1 (2010): 69-78.

[16] Garcia, Crystal A., and Sheila Suess Kennedy. "Back to School: Technology, school safety and the disappearing Fourth Amendment." Kan. JL & Pub. Pol'y 12 (2002): 273.