A LOCATION BASED ALARM REMINDER SYSTEM FOR MOBILE APPLICATION

[1] Kasi Vinesh [2] Kattekola Sai Srikar Rao [3] Abhram Thon Ayuen [4] Shankar RajSoni

[1, 2, 3,] UG Scholars, [4] Assistant Professor

[1,2,3, 4] Department of Computer Science and Engineering

[1,2,3,4]Guru Nanak Institutions Technical Campus, Hyderabad,Telanagana, India.

ABSTRCT

This Android application is designed to provide location-based reminders and friendly notifications. Italerts users when a friend enters a specified 1 km radius from their location. Additionally, users can mark a location on the map, and the app will trigger an alarm when they enter that defined area. **NearbyRegistered Shop Offers Notification** – When the user enters an area, the app checks if any shops within that area are registered in the database. If so, the user will receive notifications about the availableoffers at those shops. The user's location is constantly sent to a server using GPS and Location-Based Services (LBS) for real-time tracking.

The application also features an **Admin Panel**, where administrators can log in to manage the shop database by adding or removing shops. Shop managers, through their own login, are able to add and manage special offers on the platform.

INTRODUCTION

With the widespread use of smartphones, users now have access to highly advanced and efficient hardware. A key feature of smartphones is their ability to provide precise, real-time location data through technologies like GPS, Wi-Fi, and mobile networks. This capability opens up a range of possibilities, including the ability to offer location-based reminders to users.login, are able to add and manage special offers on the platform.

A location-based reminder operates within the context of the user's location. It notifies the userabout a task or activity when they arrive at a specific place. Modern smartphones can track theuser’s location in real-time, sending reminders when they reach a designated spot. This type ofreminder helps users by alerting them about tasks ahead of the scheduled time, saving both time and effort. Additionally, the data gathered from a user's location and reminders can be utilized to recommend new places where they might want to complete their activities.

RELATED WORK

Geofencing Techniques: Geofencing involves establishing a virtual 2boundary around a physical location, and the system monitors a user’s movement in and out of this area using GPS, Wi-Fi, or cellular signals. When a user crosses this predefined boundary, an alert or reminder is activated. Geofencing is typically powered by algorithms that calculate the user’s proximity to specific coordinates. GPS Tracking and Positioning Algorithms: GPS-based systems determine a user’s precise location by utilizing satellite signals. 1In areas where GPS signals are weak, these systems may rely on Wi-Fi or Bluetooth-based triangulation, which estimates the user's position based on the proximity and strength of nearby networks. These algorithms are crucial for accurate location tracking, especially in dense urban environments with limited satellite visibility.

METHODOLOGY-ALGORITHM USED

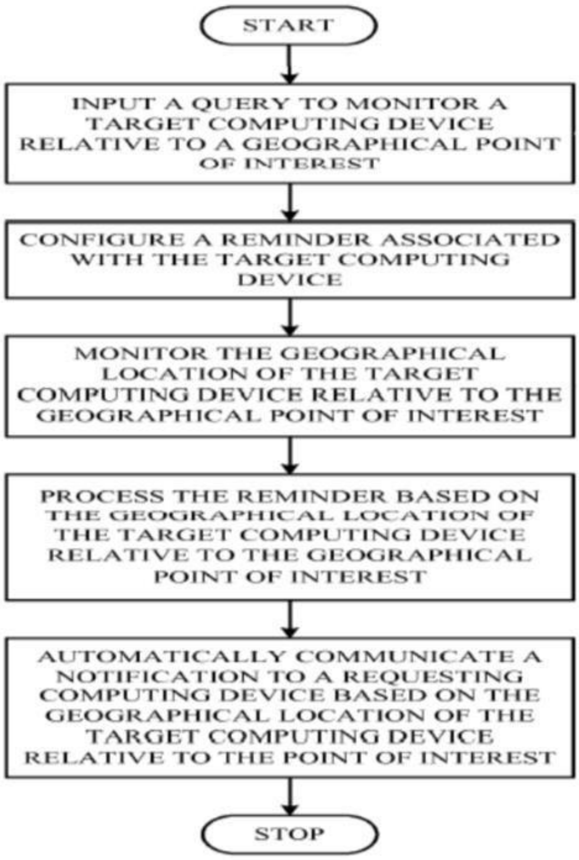
Geofencing Algorithm: A virtual boundary, known as a geofence, is set around a specified location. The system tracks the user's movement using GPS, Wi-Fi, or cellular data. When theuser enters or exits the defined geofenced area, the system triggers a notification or reminder. This process relies on calculating the user's distance from predefined geographic coordinates to ensure timely alerts.

GPS Tracking and Triangulation: GPS tracking determines the user's precise location using satellite signals. In areas with poor GPS reception (such as indoors or urban environments), the system switches to Wi-Fi or Bluetooth triangulation, estimating the user's position based on the strength and proximity of nearby signals. This combination of GPS and triangulation ensures reliable location tracking in diverse settings.

Location Prediction Model: To enhance the timing of notifications, machine learning models (such as decision trees or neural networks) predict the user's future location based on historicalmovement data and real-time GPS input. By forecasting the user's movements, this model improves the accuracy of reminders and helps optimize when they are triggered.

Together, these algorithms work in tandem to deliver accurate, timely location-based notifications, improving the user experience and reducing delays or errors in reminders.

SYSTEM ARCHITECTURE



EXPLANATION OF SYSTEM ARCHITECTURE

system enables users to input a specific target device and a geographical point of interest. Once configured, the system continuously tracks the target device's location relative to the designated point. When the target device enters a predefined proximity to the point of interest, the system activates a previously set reminder. Subsequently, a notification is dispatched to the requesting device, informing them about the target device's current location.

To bring this system to life as a mobile application, developers must meticulously consider various aspects. These include crafting a user-friendly interface, seamlessly integrating location services, implementing a robust notification system, and potentially incorporating secure authentication mechanisms. On the backend, a server-side component is essential to handle location tracking, process reminders, and deliver notifications efficiently. A well-structured database is necessary to store user data, reminders, and notification history. Additionally, a robust RESTful API must be developed to facilitate communication between the mobile app and the server.

To ensure optimal performance and user experience, developers must prioritize battery efficiency, adhere to strict privacy standards, implement robust error handling mechanisms, and conduct thorough testing under diverse conditions. By carefully addressing these considerations, a reliable and user-friendly location-based alarm system can be realized for mobile devices.

CONCLUSION

The objective of this project was to introduce a new dimension to traditional date- and time-based reminders by incorporating location as a key factor. To date, we have successfully developed a functional prototype for an Android application that offers location-based reminders. However, it is important to recognize that there is still potential for further refinement and expansion of this model. Additionally, a more thorough analysis of the real-world implications and impact of this application is necessary for a complete evaluation.

REFERENCE

[1] C. Romero and S. Ventura, ‘‘Educational data science in massive open online courses,’’ Wiley Interdiscipl. Rev., Data Mining Know. Discovery, vol. 7, no. 1, p. e1187, Jan. 2017. [Online]. Available: https://onlinelibrary.wiley.com/doi/full/10.1002/widm.1187 and https:// onlinelibrary.wiley.com/doi/abs/10.1002/widm.1187 and https://wires. onlinelibrary.wiley.com/doi/10.1002/widm.1187

[2] R. S. Baker and P. S. Invent ado, ‘‘Educational data mining and learning analytics,’’ in Learning Analytics: From Research to Practice. Springer, Jan. 2014, pp. 61–75. [Online]. Available: https://link.springer. com/chapter/10.1007/978-1-4614-3305-7_4

[3] M. I. Baig, L. Shuib, and E. Yadegaridehkordi, ‘‘Big data in education: A state of the art, limitations, and future research directions,’’ Int. J. Educ. Technol. Higher Educ., vol. 17, no. 1, pp. 1–23, Dec. 2020. [Online]. Available: https://educationaltechnologyjournal.springeropen.com/articles/10. 1186/s41239-020-00223-0 [4] B. Bakhshinategh, O. R. Zaiane, S. ElAtia, and D. Ipperciel, ‘‘Educational data mining applications and tasks: A survey of the last 10 years,’’ Educ. Inf. Technol., vol. 23, no. 1, pp. 537–553, Jul. 2017. [Online]. Available: https://link.springer.com/article/10.1007/s10639-017-9616-z