Mantik Journal, 8 (2) (2024), ISSN 2685-4236 (Online)

Published by: Institute of Computer Science (IOCS)

Jurnal Mantik

Journal homepage: www.iocscience.org/ejournal/index.php/mantik

Accuracy and usability of a geofencing-enabled truancy monitoring system in educational institution

Rizky Rachman Judhie Putra[1], Ani Anisyah[2], Ghifari Octaverin[3]

[1,2,3]Program Studi Ilmu Komputer, Fakultas Pendidikan Matematika dan Ilmu Pengetahuan Alam, Universitas Pendidikan Indonesia, Indonesia

ARTICLE INFO

Article history:

Received Aug 05, 2024
Revised Aug 20, 2024
Accepted Aug 30, 2024

Keywords:

Geofencing;
Modification;
Polygon;
Truancy;
Winding Number Algorithm

ABSTRACT

Truancy among school-going adolescents remains a persistent issue in education systems globally, impacting academic careers, social relationships, and future prospects. Research indicates that truancy is linked to adverse outcomes, including academic underperformance and significant curricular gaps, which affect both individuals and society. Students who skip school are more susceptible to negative influences that can harm society. To address this issue, we propose a system for monitoring students' locations, enabling parents and teachers to track their whereabouts effectively. This study developed an Android-based student location monitoring application with geofencing to assist parents and teachers during school hours. The geofence is polygon-shaped to accurately represent geographical conditions, utilizing the winding number algorithm to detect a point's position relative to the polygon. However, the algorithm was modified with cross product and dot product calculations to accurately detect points on the polygon's boundary. The waterfall methodology was employed throughout this study. Implemented at PGII 1 Bandung Junior and Senior High School, the application effectively monitored students from a distance. The modification of the winding number algorithm improved accuracy, achieving 100% accuracy in 36 testing scenarios. Evaluation using the System Usability Scale (SUS) yielded an average score of 76.17 from 64 respondents, placing the application in the acceptable category for the acceptability range, grade C on the grade scale, and good for adjective ratings. This indicates that the application is well-accepted and can be effectively used by users.

Corresponding Author:

Rizky Rachman Judhie Putra,
Program Studi Ilmu Komputer, Fakultas Pendidikan Matematika dan Ilmu Pengetahuan Alam,
Universitas Pendidikan Indonesia,
Jl. Dr. Setiabudhi No. 229 Bandung 40154 Jawa Barat - Indonesia.
Email: rizky_rjp@upi.edu

1. INTRODUCTION

Truancy among school-going adolescents has persisted as an unresolved issue within education systems worldwide for many years (Abdullah et al., 2020; Mahmud et al., 2019; Virtanen et al., 2021; Wardhani et al., 2023). Depending to the attendance regulations of the specific nation, any unexcused or unreported absence from school is referred to as truancy. and take multiple forms, ranging from consistently arriving late to missing one or more school days (Joacim Ramberg Sara Brolin Låftman & Modin, 2019).

This behaviour has a significant impact on academic career, social relationships, and future prospects. Research shows that truancy is linked to adverse outcomes both for individuals and society as a whole, affecting academic achievement and contributing to underperformance (Fahiroh, 2020; Joacim Ramberg Sara Brolin Låftman & Modin, 2019; Lázaro et al., 2020). Based on research conducted at SMA Negeri 1 Plumang Tuban and SMP Negeri 2 Semen Puhsaran, the impact of truancy on students includes feelings of fear and anxiety, falling behind in lessons, failing to submit assignments, receiving reprimands from teachers, failing exams, poor academic performance, and being ostracized by peers (Murdianti, Yurika Tri and Nursalim, 2018; Sari & Muis, 2018).

Based on research conducted in large city in the sample, 22.73% of 110 senior high school students skipped classes (Damayanti & Setiawati, 2013). In another study, 5.56% of a total of 216 junior high school students in Yogyakarta were found to have falsified attendance records (Saliman, 2016). This indicates that the truancy behaviour occurs from junior high school through senior high school. Truancy issues should not be overlooked since students who skip school are more likely to be used by negative elements to harm society. To address this issue, we suggest a system for monitoring students' locations, allowing parents and teachers to properly monitor their whereabouts.

The developed system utilizes GPS functionality on smartphones, allowing parents and school authorities to determine students' locations in real-time. Previous research has demonstrated that using GPS is effective in assisting parents with remotely tracking their children (Juansyah, 2015). Other researches utilize GPS to develop applications that monitor children's locations using radius-based geofencing techniques (Adam et al., 2022; Agustian et al., 2018; Beny et al., 2017; Jaya et al., 2021). This allows parents to receive notifications when their children enter or exit a circular virtual boundary that has been established (Raflesia et al., 2021). However, the use of GPS and radius-based geofencing each have their respective drawbacks. The average error distance for mobile devices is approximately 10 meters under challenging conditions (Chen et al., 2019; Merry & Bettinger, 2019; Uradziński & Bakuła, 2020). Additionally, the limitation of radius-based geofencing is that circular shapes cannot accurately represent actual geographic forms (Ryoo et al., 2012; Tuveri et al., 2022) and there isn't always a clear process for calculating the geofences' diameter or radius, which can impact the precision of the monitoring boundaries (Kharisma et al., 2021). Since the Android operating system natively supports only circular geofences, there is a need for other shapes that can better represent actual geographic conditions while accounting for the tolerance distance of the error margin in mobile devices.

A polygon shape can provide a better representation of the real geographic conditions compared to a circle (Ryoo et al., 2012; Tuveri et al., 2022). Polygon-shaped geofencing has been used in previous research to monitor classroom spaces (Babatunde et al., 2022). This demonstrates that polygon shapes can accurately represent small sections of actual geographic conditions. The winding number approach was employed in that study to ascertain whether or not students were inside the polygon that represented the classroom. For determining if the point is assumed to be in a closed polygon, the winding number is often utilized (Farajallah, 2017). By using this algorithm, the position of a point can be determined to be inside or outside the polygon (Deshmukh et al., 2018). However, the winding number algorithm is imprecise in determining points that are near the polygon's boundary (Conde Curuchet, 2021). Therefore, modifications to the winding number algorithm are necessary to accurately detect points that are close to the polygon's boundary to implement geofencing.

Based on this explanation, this research focuses on developing an Android-based student location monitoring application using geofencing with the winding number algorithm. Polygon-based geofencing was chosen because the polygon shape has been proven effective in monitoring small areas such as classrooms. Modifications to the winding number algorithm are necessary to apply geofencing more precisely, detecting points near or on the polygon boundary, compared to previous research. This application's usability will be evaluated using the System Usability Scale (SUS). It is intended that this software will benefit educators and parents to monitor their students to prevent truancy more accurately by modifying the winding number algorithm used.

2. RESEARCH METHOD

The method used in developing the student location monitoring application is the implementation of the waterfall methodology (Adenowo & Adenowo, 2013; Bassil, 2012; Sommerville, 2010), with the steps described in Figure 1 below.

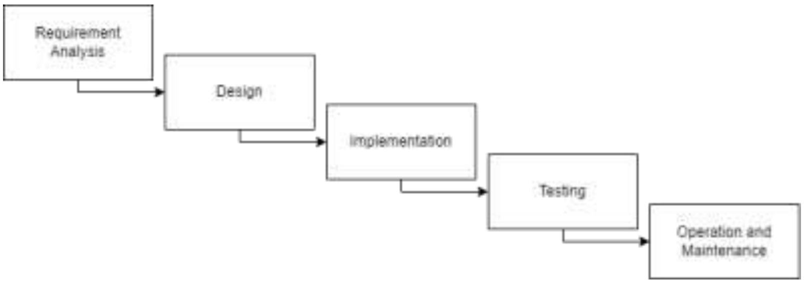

Figure 1. Waterfall Methodology

The stages in the waterfall methodology consist of requirement analysis, design, implementation, testing, and operation and maintenance (Sommerville, 2010). As part of requirement analysis, one must ascertain the features, constraints, and the system's objectives. This stage involved conducting a literature review and resulted in features focused on the principles of geofencing, specifically notifications to parents or teachers when a student enters or exits the virtual boundary created by the geofence. The design stage involves allocating the identified requirements into hardware or software components by determining the overall system architecture. Implementation is the stage where the software design is realized into an actual program. The application was developed on the Android platform using the Kotlin programming language, Google Maps API for mapping services, and Firebase Realtime Database for storage to utilize real-time data exchange features. Testing entails verifying whether the system meets the design specifications and is free of bugs or errors. Finally, the operation and maintenance stage are where the system is in use and maintenance activities are performed. Additionally, the System Usability Scale (SUS) used for evaluation is conducted during this phase.

3. RESULTS AND DISCUSSIONS

3.1 Implementation

In this section, we will explain the core program code, specifically the winding number algorithm. The winding number algorithm code is based on the pseudo code created by (Deshmukh et al., 2018). According to this pseudo code, a point is considered to be inside a polygon if its winding number is not zero. This is determined by examining each edge of the polygon. The first step is to determine whether the edge crosses upward or downward past the point. If the point is exactly to the left of an upward crossing edge, the winding number increases by one. If the point is exactly to the right of a downward crossing edge, the winding number decreases by one. After examining all edges, it can then be determined whether the point is inside or outside the polygon. However, as noted by (Conde Curuchet, 2021), this algorithm requires modification to accurately determine points located on the boundary lines of the polygon, thus enhancing geofencing accuracy.

The modification of the winding number algorithm involves adding a function to check whether a point is on the edge of the polygon. This function determines if a point is on the polygon's edge by using cross product and dot product calculations. The cross product is used to determine if three points lie on the same line by calculating the cross product of two vectors. The dot product is used to measure whether the projection of one vector onto another vector lies within the length of the second vector. The pseudo code for this function can be seen in Figure 2 below.

Algorithm 1: Checks if a point lies exactly on the edge of a polygon.

```
Function is_point_on_edge, P1, P2, P3
  | pos1 ← P2.x − P1.x
  | pos2 ← P3.y − P1.y
  | pos3 ← P3.x − P1.x
  | pos4 ← P2.y − P1.y
  | crossProduct ← pos1 × pos2 − pos3 × pos4
  | if crossProduct ≠ 0.0 then
  |   | return false
  | end if
  | dot1 ← P3.x − P1.x
  | dot2 ← P2.x − P1.x
  | dot3 ← P3.y − P1.y
  | dot4 ← P2.y − P1.y
  | dotProduct ← dot1 × dot2 + dot3 × dot4
  | if dotProduct < 0 then
  |   | return false
  | end if
  | length1 ← P2.x − P1.x
  | length2 ← P2.y - P1.y
  | squaredLengthBA ← length1 × length1 + length2 × length2
  | if dotProduct > squaredLengthBA then
  |   | return false
  | end if
  | return true
End
```

Figure 2. Pseudo Code is_point_on_edge Function

The function will be called at the beginning of the iteration to check if the point is exactly on the edge of the polygon. First, the cross product is calculated to ensure that the point is collinear with the polygon edge. If the cross product is not zero, the point is not on that edge. If it is collinear, the dot product is then calculated to ensure the point lies between the two endpoints of the edge. If the dot product is negative or greater than the edge length, the point is outside the edge's range. If all conditions are met, the point is considered to be on the edge. The next function is is_left, which can be seen in Figure 3.

Algorithm 2: Calculates the relative position of the point to the edge to determine if it's left or right.

```
Function is_left, P1, P2, P3
  | pos1 ← P2.x − P1.x
  | pos2 ← P3.y − P1.y
  | pos3 ← P3.x − P1.x
  | pos4 ← P2.y − P1.y
  | return pos1 × pos2 − pos3 × pos4
End
```

Figure 3. Pseudo Code is_left Function

The is_left function is used to determine whether a point P3 is to the left, right, or exactly on the line formed by two points P1 and P2. This function returns a positive value if P3 is to the left of the line, a negative value if P3 is to the right of the line, and zero if P3 is exactly on the line. By using the is_point_on_edge and is_left functions, the modified winding number algorithm is obtained as shown in Figure 4 below.

Algorithm 3: Winding number after modification

```
Function winding_number P, Polygon, n
  | winding_number ← 0
  | for i = 0 → all edges in Polygon do
  |   | currentPoint ← Polygon[i]
  |   | nextPoint ← Polygon[(i + 1) % n]
  |   | if is_point_on_edge(P, currentPoint, nextPoint) is true then
  |   |   | return 1
  |   | end if
  |   | if currentPoint.y ≤ P.y then
  |   |   | if nextPoint.y > P.y and is_left(currentPoint, nextPoint, P) > 0 then
  |   |   |   | winding_number ← winding_number + 1
  |   |   | end if
  |   | else if nextPoint.y ≤ P.y and is_left(currentPoint, nextPoint, P) < 0 then
  |   |   | winding_number ← winding_number − 1
  |   | end if
  | end for
  | return winding_number
End
```

Figure 4. Pseudo Code Winding Number Algorithm After Modification

In the winding_number function, the returned value is either zero or non-zero. If the value obtained is non-zero, Within the polygon is the point; otherwise, if the value is zero, the polygon does not contain the point. The simplified pseudo code of the modified winding number algorithm is shown in Figure 5.

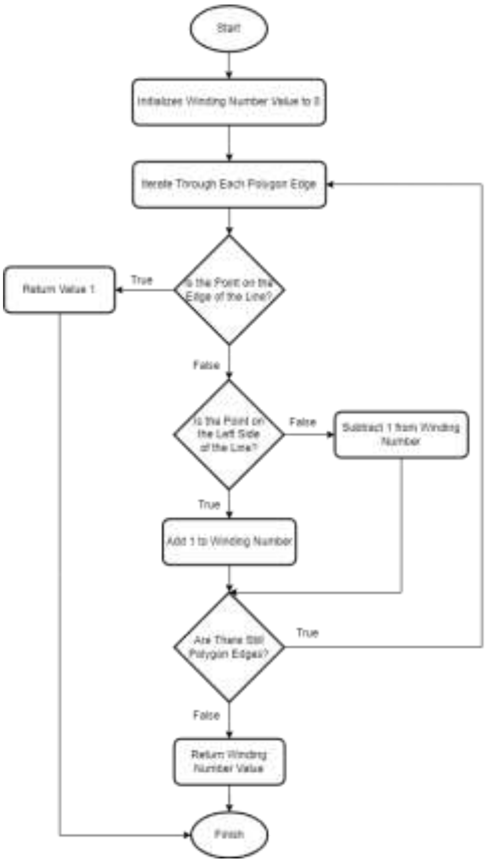

Figure 5. Flowchart Winding Number Algorithm

First, the winding number is initialized to 0. The program then iterates through each edge of the polygon to check the position of the point relative to the polygon's edges. During each iteration, the point is checked. The first check is whether the point is on the edge of the polygon. If the point is on the edge, the function returns 1 and ends the process. If not, the second check is whether the point is to the left of the edge. If the point is to the left, the winding number is incremented by 1. If the point is to the right, the winding number is decremented by 1. This process continues until all edges of the polygon have been checked. After the iteration ends, the winding number is returned, completing the process.

3.2 Testing

The accuracy testing of the winding number algorithm before and after modification was conducted by evaluating several polygon shape scenarios. The algorithm was tested using JUnit4 as the tool for unit testing. Four polygon shapes were tested in this study: convex, concave, complex, and self-intersecting polygons. For each polygon shape, nine points were tested: three points inside the polygon, three points outside the polygon, and three points on the polygon's edge. From a total of 36 testing scenarios, the results are presented in Table 1.

Table 1. Unit Test Result

Polygon Shape	Before Modification	After Modification
Convex	7 / 9	9 / 9
Concave	6 / 9	9 / 9
Complex	6 / 9	9 / 9
Self-Intersecting	9 / 9	9 / 9
	Accuracy (77,7%)	Accuracy (100%)

Before modification, the winding number algorithm passed 28 testing scenarios, achieving an accuracy of 77.7%. However, after modifying the algorithm to include checking whether a point is on the polygon's edge by calculating the cross product and dot product, the modified winding number algorithm passed all testing scenarios with 100% accuracy.

3.3 Operation

This application was implemented for monitoring students at PGII 1 Bandung Junior and Senior High School on July 16, 2024. PGII 1 Bandung Junior and Senior High School covers an area of 658 m^2 with a building area of 432 m^2. The application successfully mapped the school to create safe zones and danger zones with various polygon shapes. The mapped areas include the entire PGII 1 Bandung Junior and Senior High School area, the administrative office, classrooms of X-12, the canteen, and the motorcycle parking area as safe zones, and areas commonly used for skipping classes as danger zones. These areas are shown in Figure 6 below.

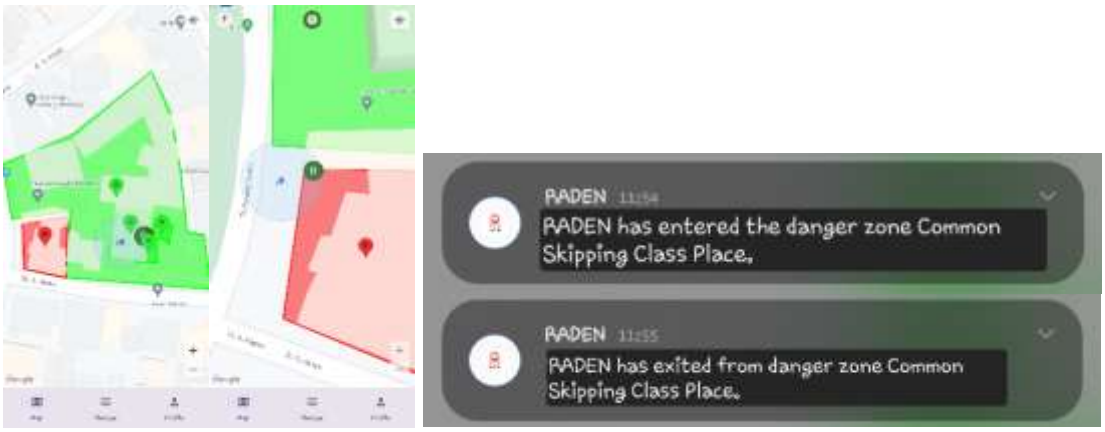

Figure 6. Application Interface

As seen in Figure 6, the mapping results in the five school areas include various polygon shapes, such as convex, concave, complex, and self-intersecting polygons. The application successfully detects students entering and leaving the predefined zones by providing warning notifications to parents/teachers and students.

3.4 Evaluation

The evaluation of the Android-based student location monitoring application with geofencing was conducted using the System Usability Scale (SUS). The System Usability Scale is a usability evaluation method that provides reliable results based on several considerations, such as a small sample size, cost, and time. The results from this method are converted into a score that serves as a reference to determine whether a system is acceptable or not (Soejono et al., 2018). The System Usability Scale consists of ten statements, with odd-numbered statements phrased positively and even-numbered statements phrased negatively (Lewis & Sauro, 2009). The System Usability Scale uses a Likert scale with point ranges from one to five. A score of one indicates "strongly disagree," two indicates "disagree," three indicates "neutral," four indicates "agree," and five indicates "strongly agree.". The SUS questionnaire was distributed over five days, from July 18, 2024, to July 22, 2024, and received responses from 64 participants.

The total SUS score calculation is 4875, which is then divided by the number of respondents to obtain the average score. The average SUS score obtained is 76.17. The System Usability Scale has three perspectives used in determining its evaluation results: acceptability range, grade scale, and adjective rating (Brooke, 2013; Wahyuni & Hamzah, 2024). The obtained results are then correlated with the SUS score scale, yielding the outcomes shown in Fig. 9. below.

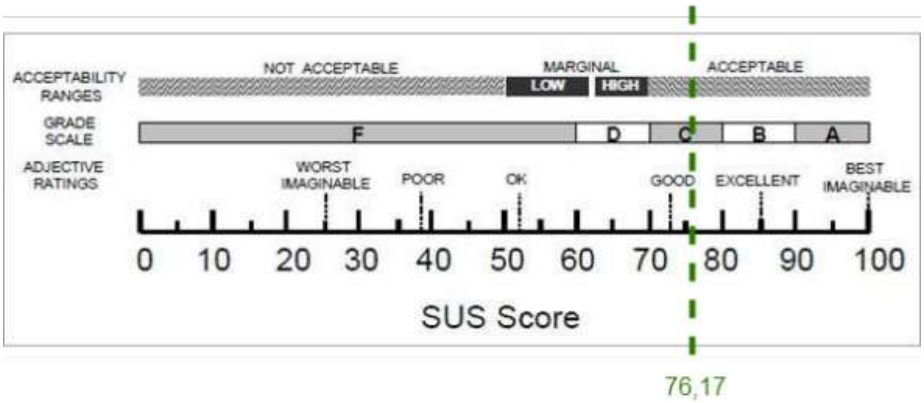

Figure 7. Results of the SUS Score Scale

Based on Figure 7, the SUS score falls into the acceptable category, grade C, and receives a rating of good. This indicates that the respondents have positively received the developed application.

3.5 Discussion

The modification of the winding number algorithm, which enhances its precision in determining the position of a point on the boundary of a polygon by incorporating cross product and dot product calculations, has significantly improved the performance of the student location monitoring application. This improvement has led to increased accuracy, ensuring that points exactly on the edges of polygons are correctly identified, which is crucial for accurately defining safe and danger zones within the application. The modified algorithm's ability to pass all 36 testing scenarios with 100% accuracy demonstrates robust performance across various polygon shapes, including convex, concave, complex, and self-intersecting polygons. The results provided by the modified winding number algorithm are superior to those of the original winding number algorithm, as developed by (Deshmukh et al., 2018). This accuracy is essential for reliably monitoring student's locations and providing timely alerts to parents and teachers.

Enhanced reliability, achieved through precise detection capabilities, helps prevent false positives and negatives, reducing the likelihood of incorrect alerts and building trust among users. Accurate mapping and detection capabilities also contribute to improved user confidence and satisfaction, as users are less likely to experience frustration or confusion due to incorrect notifications. This improved user experience likely contributes to higher satisfaction and confidence in the application's effectiveness. The increased accuracy and reliability directly enhance the perceived usability of the application, which is likely reflected in higher scores in usability assessments. The reported SUS score of 76.17, falling into the acceptable category with a grade C and a "good" adjective rating, reflects a positive reception from users. These results are superior to those of previous studies conducted by (Adam et al., 2022; Agustian et al., 2018; Babatunde et al., 2022; Beny et al., 2017; Jaya et al., 2021). The improvement in handling edge cases and ensuring accurate zone mapping aligns with users' expectations of a well-integrated and consistent system, further boosting the SUS score.

4. CONCLUSION

This study successfully developed an Android-based student location monitoring application utilizing geofencing. The use of polygon-shaped geofencing facilitates accurate mapping by conforming to actual geographical conditions, thereby aiding parents and schools in monitoring students during school hours. Parents or teachers can be informed if a student is truant through notifications provided when the student interacts with the virtual boundary. The winding number algorithm was modified to enhance precision by incorporating cross product and dot product calculations to check a point's position relative to the polygon's edges. This modification significantly improved the algorithm's ability to detect points on the polygon boundaries. The modified algorithm was rigorously tested using unit testing on four polygon shapes—convex, concave, complex, and self-intersecting—across 36 scenarios. The original algorithm passed 28 scenarios with an accuracy of 77.7%, failing primarily at detecting points on polygon edges. In contrast, the modified algorithm achieved a perfect accuracy of 100% in all scenarios. Additionally, the System Usability Scale (SUS) evaluation, involving 64 respondents, yielded an average SUS score of 76.17, placing the application in the acceptable category, grade C on the grade scale, and achieving a good rating on adjective ratings, indicating that the application is well-received and considered functional and useful by its users. Based on the results of the operation stage, a factor that posed a challenge and influenced geofencing was the instability of GPS accuracy on smartphones. Therefore, for future research, it is recommended to use smartwatches, IoT devices, or other GPS devices that can be worn by students to address this issue. Additionally, developing the application on other platforms, such as iOS, is necessary to reach a broader user base. Lastly, more comprehensive testing scenarios for the algorithm are required, particularly for points near the polygon's edge and points exactly on the polygon's edge.

REFERENCES

Abdullah, N., Bakar, A. Y. A., Mahmud, M. I., & others. (2020). School Refusal or Truancy Challenges: A Special Need for the Collaboration? *Creative Education*, *11*(11), 2199.

Adam, S. I., Lengkong, O. H., Pungus, S. R., & Kollabathula, S. R. (2022). Geofencing Application for Parents Tracking Children using Push Notification in Universitas Klabat based on mobile. *2022 4th International Conference on Cybernetics and Intelligent System (ICORIS)*, 1–6. https://doi.org/10.1109/ICORIS56080.2022.10031487

Adenowo, A. A. A., & Adenowo, B. A. (2013). Software Engineering Methodologies: A Review of the Waterfall Model and Object-Oriented Approach. *International Journal of Scientific & Engineering Research*, *4*(7).

Agustian, A., Trisnadoli, A., & Lestari, I. (2018). Analisis Hasil Implementasi Konsep Context-Aware Pada Aplikasi Mobile Family Tracking Untuk Platform Android. *Teknika*, *7*(1), 34–41.

Babatunde, A. N., Oke, A. A., Babatunde, R. S., Ibitoye, O., & Jimoh, E. R. (2022). Mobile Based Student Attendance System Using Geo-Fencing With Timing and Face Recognition. *J. Adv. Math. Comput. Sci.*

Bassil, Y. (2012). *A Simulation Model for the Waterfall Software Development Life Cycle*. https://arxiv.org/abs/1205.6904

Beny, B., Budiman, J., & Nugroho, A. (2017). Implementasi Geofencing Pada Aplikasi Layanan Pemantau Anak Berbasis Lokasi. *Prosiding 2nd Seminar Nasional IPTEK Terapan (SENIT) 2017*, *2*(1), 63–66.

Chen, B., Gao, C., Liu, Y., & Sun, P. (2019). Real-time Precise Point Positioning with a Xiaomi MI 8 Android Smartphone. *Sensors*, *19*(12). https://doi.org/10.3390/s19122835

Conde Curuchet, J. P. (2021). *Addressing the Issues in the Winding Number Algorithm Related to the Floating Point Representation.* https://purl.lib.fsu.edu/diginole/2021_Fall_CondeCuruchet_fsu_0071N_16915

Damayanti, F. A., & Setiawati, D. (2013). Studi tentang perilaku membolos pada siswa SMA swasta di Surabaya. *Jurnal BK Unesa*, *3*(1), 454–461.

Deshmukh, P., Bhajibhakre, A., Gambhire, S., Channe, A., & Deshpande, N. (2018). Survey of geofencing algorithms. *International Journal of Computer Science Engineering Techniques*, *3*(2), 1–5.

Fahiroh, S. A. (2020). The Meaning of Truant Behavior for Junior and Senior High School Students in Indonesia. *Proceedings of the 1st Borobudur International Symposium on Humanities, Economics and Social Sciences (BIS-HESS 2019)*, 1140–1143. https://doi.org/10.2991/assehr.k.200529.239

Farajallah, W. (2017). *Winding Numbers and Solid Angles*.

Jaya, M. I., Tong, G. X., Razak, M. F. A., Zabidi, A., & Hisham, S. I. (2021). Geofence Alerts Application With GPS Tracking For Children Monitoring (CTS). *2021 International Conference on Software Engineering & Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM)*, 222–226. https://doi.org/10.1109/ICSECS52883.2021.00047

Joacim Ramberg Sara Brolin Låftman, E. F., & Modin, B. (2019). School effectiveness and truancy: a multilevel study of upper secondary schools in Stockholm. *International Journal of Adolescence and Youth*, *24*(2), 185–198. https://doi.org/10.1080/02673843.2018.1503085

Juansyah, A. (2015). Pembangunan aplikasi child tracker berbasis assisted--global positioning system (a-gps) dengan platform android. *Jurnal Ilmiah Komputer Dan Informatika (KOMPUTA)*, *1*(1), 1–8.

Kharisma, A. P., Jonemaro, E. M. A., & Arwani, I. (2021). Paratransit Trip Data Collection System with Smartphone GPS and REST Web Service in Malang, Indonesia. *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, *9*(1), 245–255.

Lázaro, S., Urosa, B., Mota, R., & Rubio, E. (2020). Primary education truancy and school performance in social exclusion settings: The case of students in Cañada real Galiana. *Sustainability*, *12*(20), 8464.

Mahmud, N. A., Awaluddin, S. M., Yoep, N., Hasani, W. S. R., Yn, J. L. M., & Kuay, L. K. (2019). Association of truancy and health risk behaviours among school-going adolescents in Malaysia. *Open Journal of Social Sciences*, *7*(7), 228–237.

Merry, K., & Bettinger, P. (2019). Smartphone GPS accuracy study in an urban environment. *PLOS ONE*, *14*(7), 1–19. https://doi.org/10.1371/journal.pone.0219890

Murdianti, Yurika Tri and Nursalim, M. (2018). Studi Tentang Perilaku Membolos Siswa di SMP Negeri 2 Semen Puhsarang Kabupaten Kediri. *Jurnal BK Unesa*, *9*(1), 109–116.

Raflesia, S. P., Taufiqurrahman, T., Lestarini, D., & Bardadi, A. (2021). A Context-Aware Mobile-Based System for Crime Prevention and Emergencies. *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, *9*(1), 153–162.

Ryoo, J., Kim, H., & Das, S. R. (2012). Geo-fencing: Geographical-fencing based energy-aware proactive framework for mobile devices. *2012 IEEE 20th International Workshop on Quality of Service*, 1–9.

Saliman, S. (2016). KENAKALAN REMAJA PADA SISWA SMP DI KOTA YOGYAKARTA. *JIPSINDO (Jurnal Pendidikan Ilmu Pengetahuan Sosial Indonesia)*, *2*(2), 179–201. https://doi.org/10.21831/jipsindo.v2i2.7781

Sari, W. P., & Muis, T. (2018). Studi Kasus Tentang Perilaku Membolos di SMA Negeri 1 Plumpang Tuban. *Jurnal BK Unesa*, *3*(1), 23–30.

Sommerville, I. (2010). *Software Engineering (9th Edition).* Addison-Wesley.

http://www.amazon.com/Software-Engineering-9th-Edition-Sommerville/dp/0137035152
Tuveri, G., Garau, M., Sottile, E., Pintor, L., Atzori, L., & Meloni, I. (2022). Beep4me: automatic ticket validation to support fare clearing and service planning. *Sensors*, *22*(4), 1543.
Uradziński, M., & Bakuła, M. (2020). Assessment of Static Positioning Accuracy Using Low-Cost Smartphone GPS Devices for Geodetic Survey Points' Determination and Monitoring. *Applied Sciences*, *10*(15). https://doi.org/10.3390/app10155308
Virtanen, T. E., Räikkönen, E., Lerkkanen, M.-K., Määttä, S., & Vasalampi, K. (2021). Development of Participation in and Identification With School: Associations With Truancy. *The Journal of Early Adolescence*, *41*(3), 394–423. https://doi.org/10.1177/0272431620919155
Wardhani, C. K., Zulkarnain, W., Nurabadi, A., & Kusumaningrum, D. E. (2023). Development of Student Discipline in Controlling Truancy Behavior (Case Study at SMPN 11 Malang). *Proceedings of the International Conference on Educational Management and Technology (ICEMT 2022)*, 519–531. https://doi.org/10.2991/978-2-494069-95-4_61