

Pembatasan Jarak Presensi Dalam Sistem Presensi Daring Menggunakan Formula Haversine

Bagas Pardana Ilham¹, Fidi Supriadi²

^{1,2}Fakultas Teknologi Informasi, Universitas Sebelas April

e-mail: ibagaspardana@gmail.com, *²fidi@unsap.ac.id

Abstrak

Transformasi sistem presensi manual menjadi sistem daring sudah banyak dilakukan berbagai pihak, mulai dari instansi pemerintah, perkantoran, dan sekolah. Sebagai contoh suatu sekolah yang telah menerapkan sistem manajemen pembelajaran berbasis elektronik dapat menggunakan fitur formulir kehadiran sebagai salah satu cara mendata kehadiran siswanya dalam proses pembelajaran. Akan tetapi fitur ini dirasa memiliki kekurangan apabila diterapkan pada saat pembelajaran tatap muka di kelas sebab siswa yang tidak hadir masih dapat mengisi darimana saja dia berada asalkan dapat mengakses form tersebut. Adanya batasan dalam radius diperlukan guna membatasi hanya siswa yang berada dalam jangkauan area kelas saja yang dapat mengisi form kehadiran. Tujuan dari penelitian ini ialah merancang sebuah sistem presensi daring dengan menerapkan Formula Haversine menggunakan Metode *Extreme Programming*. Formula ini dipilih sebab cocok untuk menghitung jarak antara dua titik koordinat pada permukaan bumi dengan anggapan bumi berbentuk bulat. *Extreme Programming* merupakan sebuah metode pembangunan sistem perangkat lunak cepat yang terdiri atas fase perencanaan, perancangan, koding, dan pengujian. Melalui percobaan sebaran *user* sebanyak 30 titik akses terhadap sistem dengan titik batasan "berhasil" harus kurang dari sama dengan 20 meter dan "gagal" jika berada lebih dari batasan tersebut, didapatkan prosentase keberhasilan percobaan 96,6% yang menunjukkan bahwa Formula Haversine dapat diterapkan guna membatasi jarak pada suatu sistem presensi.

Kata Kunci: Formula Haversine, Presensi, Radius, Extreme Programming, DSRM

Restricting Attendance Distance in the Online Attendance System Using The Haversine Formula

Abstract

The transformation of the manual attendance system into an online system has been carried out by various parties, including government agencies, offices, and schools. For example, a school that has implemented an electronic-based learning management system can use the attendance form feature as a way to record students' attendance in the learning process. However, this feature is considered to have shortcomings when applied during face-to-face learning in class because absent students can still fill in from wherever they are as long as they can access the form. The existence of a radius limit is necessary to limit only students who are within the range of the class area to be able to fill out the attendance form. This research aims to design an online presence system by applying the haversine formula using the Extreme Programming method. This formula was chosen because it is suitable for calculating the distance between two coordinate points on the earth's surface assuming the earth is round. Extreme Programming is a method for building fast software systems that consists of planning, design, coding, and testing phases. By conducting user distribution experiments on 30 access points to the system with a "success" limit of less than 20 meters and "failure" if it is more than that limit, the percentage of successful experiments was obtained at 96.6%. , shows that the Haversine Formula can be applied to limit the distance in an attendance system.

Keywords: Haversine Formula, Attendance, Radius, Extreme Programming, DSRM

1. Pendahuluan

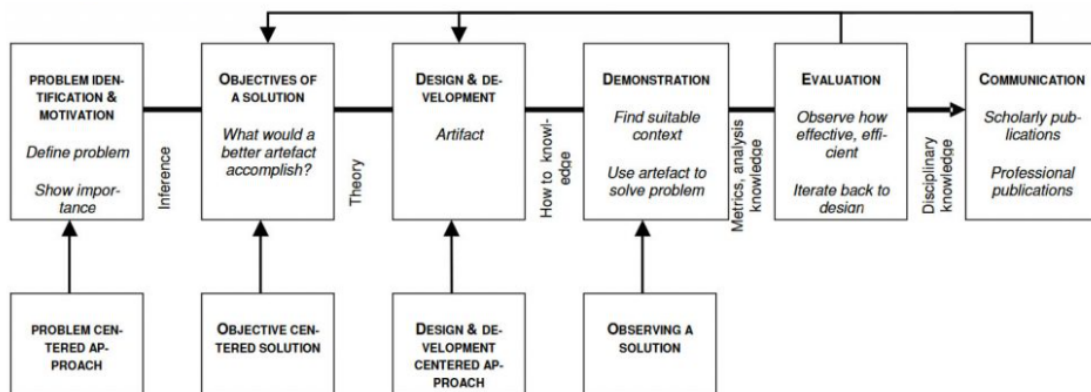
Banyak variasi teknologi presensi daring yang berkembang saat ini dan biasanya digunakan mulai dari sistem presensi berbasis *biometric* (*finger print, face recognition*), *qr code, barcode*, dan juga IoT dengan memanfaatkan Kartu RFID [1]). Namun pemanfaatan teknologi geolokasi pada sistem presensi berbasis *mobile* masih jarang sekali digunakan di lingkungan pendidikan, yang mana dengan memanfaatkan teknologi geolokasi presensi hanya bisa dilakukan didalam radius tertentu saja [2], [3]. Bahkan dalam pengimplementasiannya, presensi dengan memanfaatkan teknologi geolokasi dapat dikombinasikan dengan teknologi yang lain. Metode ini sangat cocok khususnya untuk lingkungan perkantoran yang menerapkan sistem kerja WFO (*Work From Office*), sehingga rasanya perlu juga diterapkan di lingkungan Pendidikan seperti kampus.

Untuk menghitung batasan radius presensi, maka diperlukan suatu formula untuk menghitung jarak antara titik lokasi *user* pada saat melakukan presensi dengan titik pusat presensi bisa dilakukan. Ada beberapa rumus untuk menyelesaikan permasalahan tersebut, diantaranya adalah Formula Haversine, yaitu suatu metode yang memungkinkan bagaimana cara menghitung jarak antara dua titik di permukaan bumi dengan menggunakan variabel masukan berupa garis lintang (*latitude*) dan garis bujur (*longitude*)[4]. Formula Haversine juga dikenal sebagai jarak lingkaran yang besar, rumus ini menghitung jarak dari titik utama ke titik target menggunakan fungsi trigonometri dengan menggunakan garis lintang dan garis bujur dan pada akhirnya memberikan data jarak antara dua titik dalam meter [5].

Dengan demikian dalam penelitian ini akan membuktikan bahwa teknologi geolokasi dapat diterapkan pada sistem presensi dengan Formula Haversine sebagai rumus untuk menghitung jarak garis lurus antara dua titik lokasi dengan menggunakan nilai garis lintang (*latitude*) dan garis bujur (*longitude*) sebagai parameter masukannya dengan tujuan untuk membuat radius lingkaran dari area presensi. Lokasi yang akan menjadi tempat uji coba di Kampus FTI UNSAP Sumedang. Kemudian data koordinat yang digunakan akan bersumber dari *Fused Location Provider* yaitu API lokasi yang disediakan oleh Layanan *Google Play* yang secara cerdas menggabungkan sinyal untuk menyediakan informasi lokasi yang dibutuhkan aplikasi pengguna [6]. Penelitian ini juga bertujuan untuk mengukur tingkat akurasi keberhasilan yang dihasilkan dari sistem presensi daring berbasis radius dengan Formula Haversine itu sendiri.

2. Metode Penelitian

Tahapan pada penelitian ini menggunakan Metodologi DSRM (*Design Science Research Method*) yaitu dimulai dari *Problem Identification, Define Objectives Of Solution, Design & Development, Demonstration, Evaluation, dan Communication* [7], [8], [9], [10]:



Gambar 1 Tahapan DSRM

a) Problem Identification

Pada tahap ini melakukan penelitian sangat diperlukan dan penting untuk memahami permasalahan yang mendasari penelitian dan mengumpulkan bukti-bukti dibalik permasalahan tersebut. Alur tahap penelitian dimulai dari mengidentifikasi masalah dengan mengambil studi kasus sistem presensi mahasiswa di Fakultas Teknologi Informasi Universitas Sebelas April.

b) Define Objectives Of Solution

Tahap ini dilakukan untuk menemukan solusi atas permasalahan yang ada dengan cara mencari penelitian terdahulu yang berkaitan dengan permasalahan yang diangkat serta mencari dan mendalami teori yang berhubungan dengan sistem yang akan dibangun. Selain itu studi literatur juga dilakukan untuk mendalami teori yang berhubungan dengan metode pengembangan sistem yang akan digunakan.

c) Design & Development

Tahapan ini merupakan tahapan untuk melakukan desain dan implementasi dari sebuah penelitian. Desain yang disebutkan antara lain merupakan desain alur kerja sistem, ruang lingkup sistem yang dibuat dalam bentuk diagram UML atau DFD, kemudian desain *interface* sistem yang akan dibangun beserta teknologi atau *tools* yang akan digunakan pada saat pengembangan. Tahap ini juga menentukan kebutuhan minimum *software* dan *hardware* untuk dapat menjalankan aplikasi yang diteliti. Pada tahap pengembangan dilakukan pengkodean sistem dengan produk akhir berupa *prototype* sistem sederhana sebagai bukti dan media pengujian.

d) Demonstration

Pada tahap ini dilakukan uji instalasi sistem dengan cara memaketkan projek aplikasi yang dibangun ke dalam format (.apk). Format ini merupakan bentuk umum untuk aplikasi *mobile*. Setelah dilakukan pemaketan aplikasi, aplikasi dengan format (.apk) tersebut didistribusikan kepada beberapa *user* sebagai percobaan instalasi sistem.

e) Evaluation

Pada tahap ini dilakukan pengujian *Blackbox* yaitu pengujian pada fungsi-fungsi yang terdapat pada sistem. Apabila pada tahap ini ditemukan kesalahan maka proses akan kembali ke tahap perancangan untuk memperbaiki kesalahan tersebut. Selain pengujian *Blackbox* juga dilakukan pengujian akurasi keberhasilan dari *prototype* sistem yang dibangun.

f) Communication

Setelah sistem sudah berjalan sesuai dengan yang diharapkan dan siap untuk diimplementasikan, maka yang dilakukan adalah mengkomunikasikan hasil penelitian tersebut dengan cara publikasi dalam bentuk jurnal

3. Hasil

a) Problem Identification

Alur tahap penelitian dimulai dari mengidentifikasi masalah yang terjadi dalam sistem presensi mahasiswa. Proses perekaman kehadiran yang dilakukan masih menggunakan sistem konvensional dengan cara mengisi tanda tangan pada buku kehadiran atau daftar hadir mahasiswa. Selain itu terdapat beberapa dosen yang telah memberlakukan sistem verifikasi kehadiran mahasiswa yang terdapat pada *website e-learning*. Bentuk presensi seperti ini masih menyisakan celah yang memungkinkan mahasiswa yang tidak mengikuti perkuliahan tatap muka di kelas masih bisa mengisi form kehadiran.

b) Define Objectives of Solution

Untuk menemukan solusi terbaik guna menyelesaikan permasalahan yang diangkat, maka perlu dilakukan pengumpulan data dan informasi terkait penelitian terdahulu yang memiliki kemiripan dalam

studi kasus, teori-teori yang berhubungan dengan penelitian, serta teknologi yang akan digunakan pada saat pengembangan. Untuk melakukan hal tersebut maka pada penelitian ini digunakanlah metode pengumpulan data Kualitatif.

c) *Design and Development*

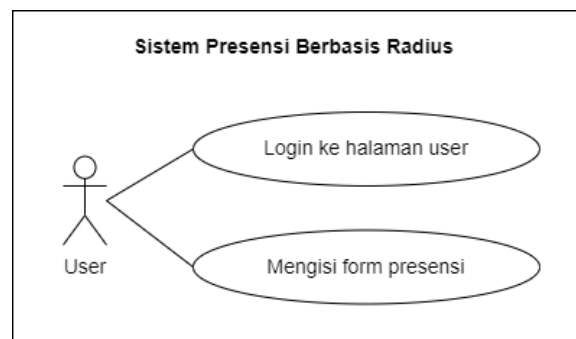
Pada tahap ini dilakukan pengembangan aplikasi dengan menggunakan pendekatan *Model Extreme Programming* [11], dengan uraian sebagai berikut :

1) *Planning*

Tahap pertama melakukan analisis kebutuhan sistem yang akan dibangun, seperti kebutuhan perangkat lunak, perangkat keras, serta teknologi yang akan digunakan selama tahap pengembangan sistem tersebut.

2) *Design*

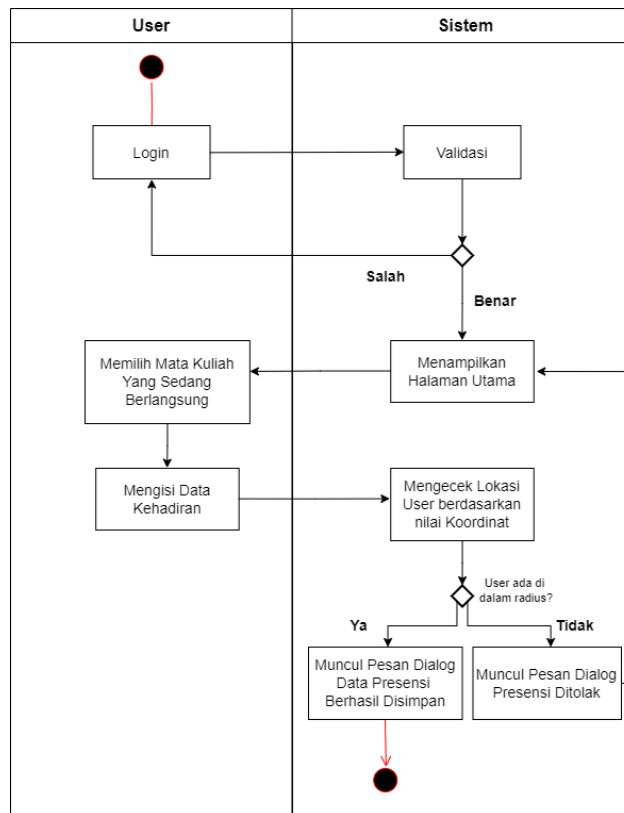
a. *Use Case Diagram*



Gambar 2 *Use Case Diagram*

Sistem yang dibuat hanya berfokus pada penerapan Formula Haversine pada fitur pembatasan jarak presensi. Oleh karena itu, dalam diagram yang berperan sebagai aktor ialah *user* dalam hal ini mahasiswa. *User* dapat mengisi Formulir presensi dan nantinya sistem akan mendeteksi lokasi perangkat apakah berada dalam radius presensi atau tidak

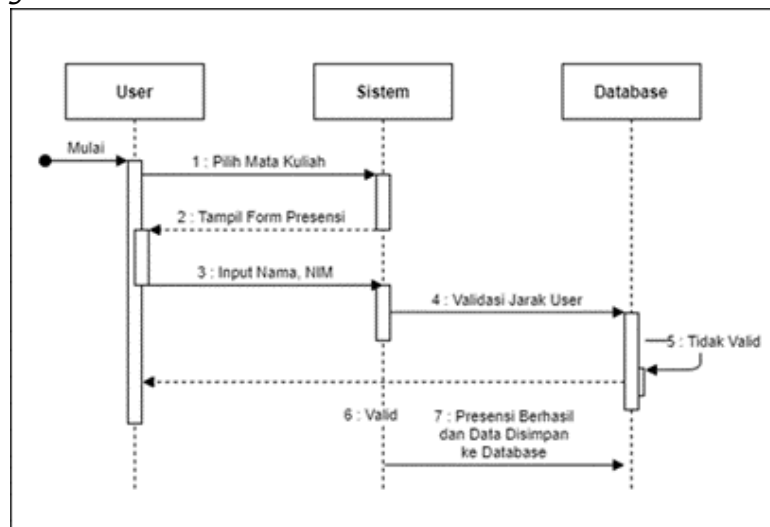
b. *Activity Diagram*



Gambar 3 Activity Diagram

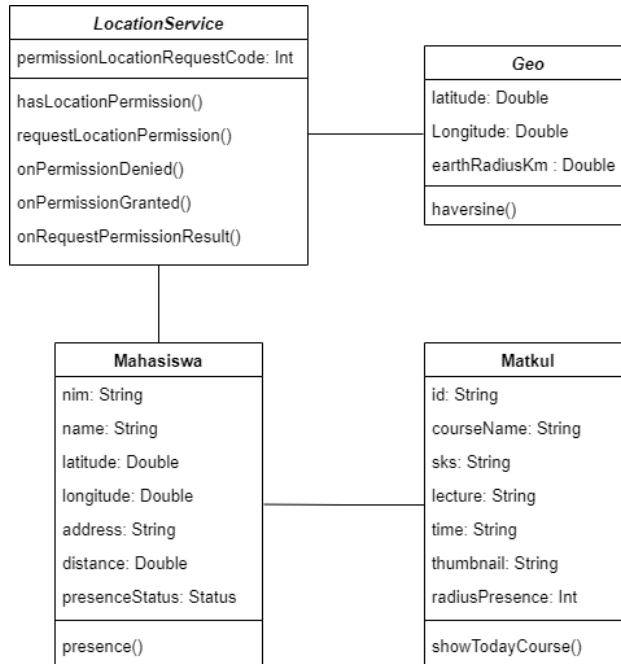
Diagram di atas menggambarkan aliran aktivitas *user* pada saat melakukan presensi daring. Dari mulai sistem mendeteksi lokasi koordinat perangkat, kemudian *user* mengisi formulir presensi sesuai dengan mata kuliah yang dipilih, dan ketika *user* menekan tombol submit maka akan muncul dialog atau popup yang memberi tahu apakah presensi tersebut berhasil atau ditolak oleh sistem karena berada diluar radius presensi.

c. Sequence Diagram



Gambar 4 Sequence Diagram

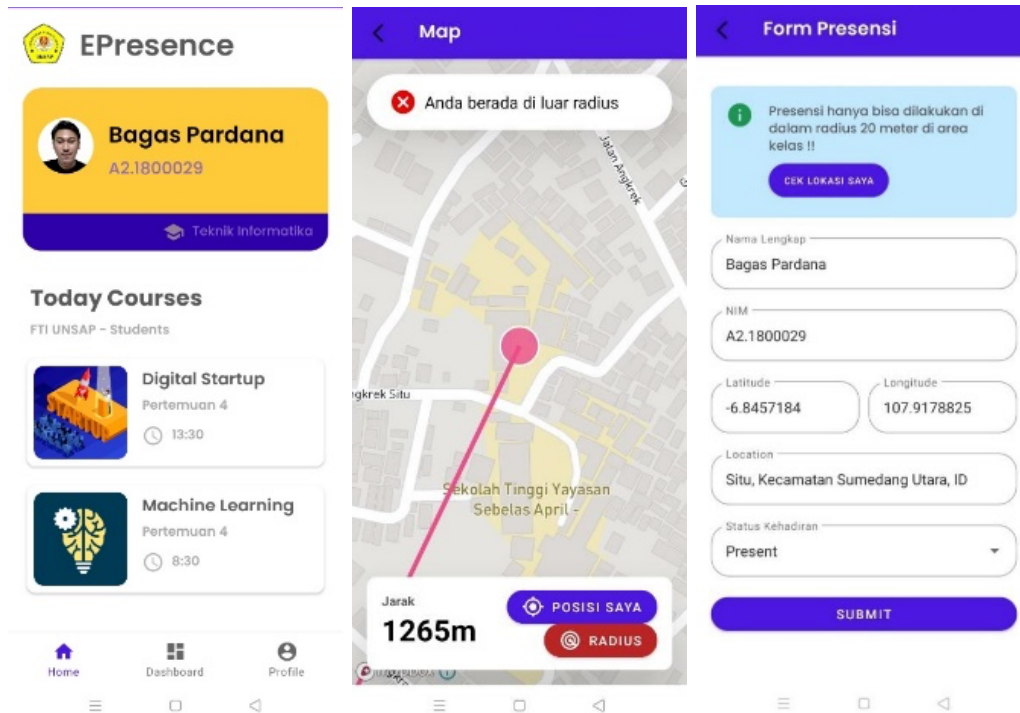
d. *Class Diagram*



Gambar 5 *Class Diagram*

Class Diagram di atas menggambarkan dengan jelas struktur serta deskripsi *class*, *atribut*, *metode*, dan relasi dari setiap objek yang ada pada sistem presensi daring berbasis radius dengan menggunakan Formula Haversine yang dibangun.

e. *User Interface*



Gambar 6 *User Interface* Aplikasi

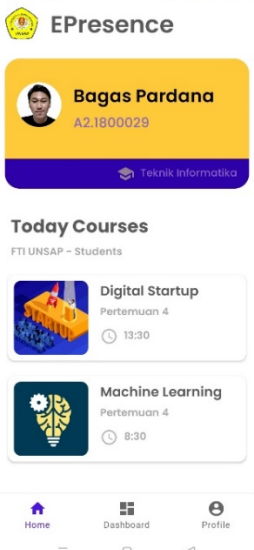
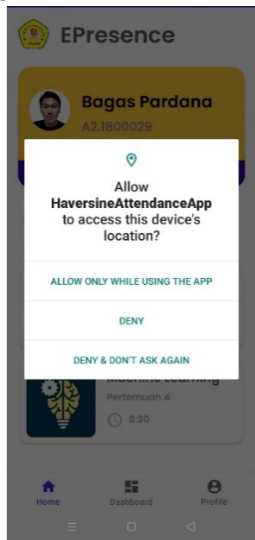
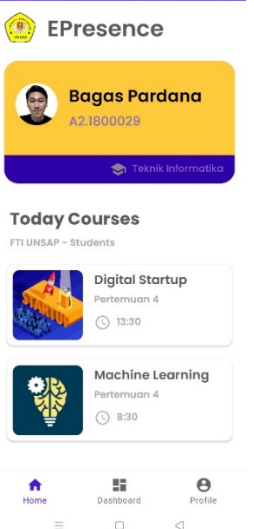
3) *Coding*

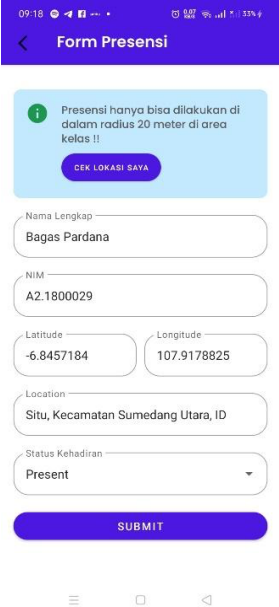

Pada tahap ini sistem dibangun dengan menggunakan Software Android Studio dan Kotlin sebagai bahasa pemrogramannya. Untuk penyimpanan data presensi mahasiswa menggunakan *Firestore Database*.

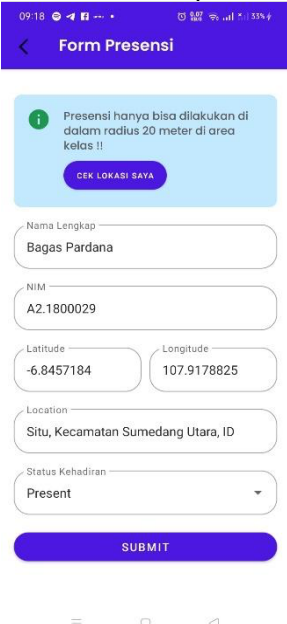
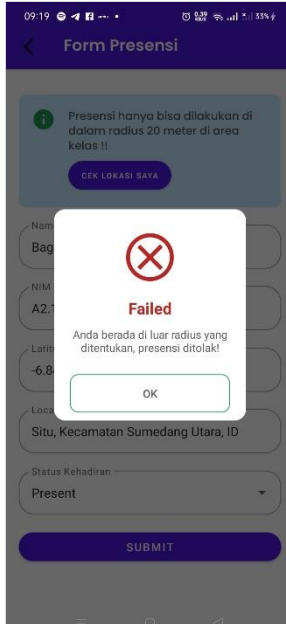
4) *Testing*

Tahap ini dilakukan dua bentuk pengujian yaitu Uji *Blackbox* [12], Tabel 1 memperlihatkan bentuk pengujian yang dilakukan.

Tabel 1 Uji *Blackbox* Aplikasi

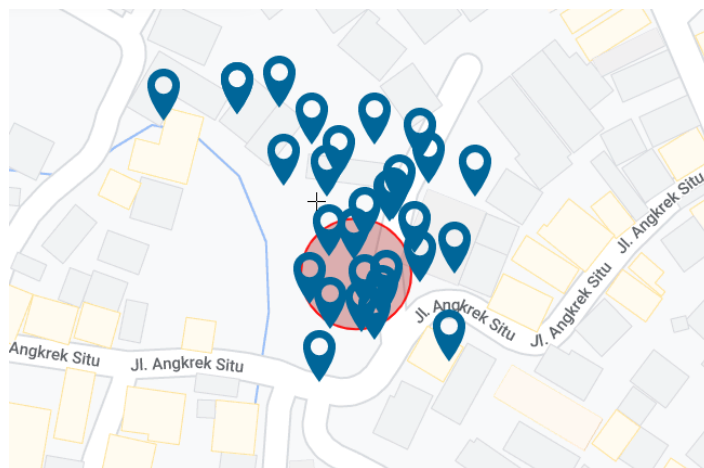
No	Skenario Pengujian	Hasil Yang Diharapkan	Kesimpulan
1	<p>Pertama kali mengunduh aplikasi, kemudian menekan item mata kuliah</p> 	<p>Muncul dialog permintaan akses lokasi</p> 	Valid
2	<p>Menekan mata kuliah setelah akses lokasi diaktifkan</p> 	Pindah ke halaman form presensi	Valid

No	Skenario Pengujian	Hasil Yang Diharapkan	Kesimpulan
			
3	<p>Menekan tombol cek lokasi</p> 	<p>Pindah ke halaman peta yang menunjukkan posisi <i>user</i> terhadap radius presensi</p> 	Valid

No	Skenario Pengujian	Hasil Yang Diharapkan	Kesimpulan
4	<p>Menekan tombol submit di halaman form presensi dengan kondisi <i>user</i> berada di luar radius presensi</p> 	<p>Muncul dialog bahwa presensi yang dilakukan <i>user</i> gagal karena berada di luar radius</p> 	Valid

d) Demonstration & Evaluation

Tahap ini dilakukan pengujian dengan cara melakukan percobaan mengisi presensi dari 30 titik lokasi yang berbeda di wilayah terdekat Kampus FTI UNSAP dengan kondisi *user* berada di dalam radius dan di luar radius presensi dengan titik terjauh kurang dari 1,5 km dengan batasan presensi harus dalam radius 20 meter. *User* akan dinyatakan “berhasil” mengakses sistem apabila jarak berada di bawah 20 meter dan dinyatakan “Gagal” apabila melebihi 20 meter. Sebaran 30 titik lokasi tersebut diperlihatkan pada Gambar 7 seperti berikut:



Gambar 7 Sebaran titik lokasi uji presensi

Berdasarkan kondisi sebaran yang terlihat pada Gambar 7, didapatkan hasil seperti terlihat pada Tabel 2 di bawah ini.

Tabel 2 Uji Akurasi Keberhasilan

No	Koordinat Perangkat <i>User</i> Dari Fused Location Provider		Jarak Dengan Formula Haversine (m)	Status Presensi	Status Pengujian
	Latitude	Longitude			
1	-6.8356833	107.923796	41.574	Gagal	Valid
2	-6.835715	107.9237808	38.460	Gagal	Valid
3	-6.8357189	107.9237728	38.255	Gagal	Valid
4	-6.8357627	107.9237111	36.147	Gagal	Valid
5	-6.8359274	107.9237089	21.768	Gagal	Valid
6	-6.8360108	107.9237345	14.723	Berhasil	Valid
7	-6.8358133	107.9236833	32.987	Gagal	Valid
8	-6.8457234	107.9178837	1262.178	Gagal	Valid
9	-6.8359848	107.9236252	10.759	Berhasil	Valid
10	-6.835992	107.9237017	8.414	Berhasil	Valid
11	-6.8359157	107.923764	7.687	Berhasil	Valid
12	-6.8359556	107.9237514	7.638	Berhasil	Valid
13	-6.8354415	107.9234991	57.085	Gagal	Valid
14	-6.836113	107.923596	24.388	Gagal	Valid
15	-6.835922	107.923573	13.419	Berhasil	Valid
16	-6.839825	107.988264	7143.994	Gagal	Tidak Valid
17	-6.835472	107.9232107	72.800	Gagal	Valid
18	-6.8356596	107.9239832	42.836	Gagal	Valid
19	-6.8357965	107.923832	20.240	Gagal	Valid
20	-6.8360634	107.9239199	29.772	Gagal	Valid
21	-6.83584775	107.923932437	27.383	Gagal	Valid
22	-6.83586906	107.923846607	17.624	Berhasil	Valid
23	-6.835805147	107.923621301	14.802	Berhasil	Valid
24	-6.835610738	107.923645441	34.457	Gagal	Valid
25	-6.835530843	107.923733954	43.139	Gagal	Valid
26	-6.835568127	107.923846607	42.257	Gagal	Valid
27	-6.835626716	107.923868064	37.525	Gagal	Valid
28	-6.835632043	107.923508648	37.726	Gagal	Valid
29	-6.835530843	107.923578386	44.789	Gagal	Valid
30	-6.835659821	107.923616154	29.849	Gagal	Valid

Dari hasil 30 percobaan pada Tabel 2 diatas, dapat terlihat kondisi beberapa lokasi *user* yang berada dalam radius kurang dari 20 meter status presensi yang dihasilkan ialah "Berhasil" dan lokasi *user* yang melebihi 20 meter status presensi yang dihasilkan ialah "Gagal". Namun terdapat satu anomali pada satu lokasi *user* yang dinyatakan bahwa jaraknya berada sekitar 7,1 km dari kampus, padahal dalam kondisi batasan di atas dinyatakan jarak terjauh percobaan kurang dari 1,5 km meski pada status presensi dinyatakan "Gagal". Dari keseluruhan kondisi percobaan dapat dihitung tingkat akurasi keberhasilan sistem presensi daring berbasis radius dengan Formula Haversine yaitu :

$$\begin{aligned}
 \text{Akurasi} &= \frac{\text{Jumlah Pengujian Berhasil}}{\text{Total Percobaan}} \times 100\% \\
 &= \frac{29}{30} \times 100\% = \mathbf{96.66\%}
 \end{aligned}$$

4. Kesimpulan

Dari hasil penelitian yang dilakukan dapat disimpulkan bahwa Formula Haversine sebagai metode atau rumus untuk menghitung jarak garis lurus antara dua titik koordinat dapat diimplementasikan untuk membuat fitur radius pada sistem presensi daring. Berdasarkan hasil percobaan sebaran *user* sebanyak 30

titik percobaan mengakses sistem presensi dengan kondisi batasan presensi berada dalam radius kurang dari 20 meter dari lokasi kampus FTI UNSAP dinyatakan "berhasil" dan lebih dari itu dinyatakan "gagal", menghasilkan akurasi percobaan sebesar 96,66%. Selain itu *Fused Location Provider* sebagai API untuk mendapatkan nilai koordinat posisi perangkat *user* kadang tidak akurat sampai 100% karena bisa jadi dipengaruhi faktor kekuatan sinyal perangkat *user* itu sendiri.

5. Daftar Pustaka

- [1] S. Sultana, A. Enayet, and I. J. Mouri, "A Smart, Location Based Time and Attendance Tracking System using Android Application," *International Journal of Computer Science, Engineering and Information Technology*, vol. 5, no. 1, pp. 01–05, Feb. 2015, doi: 10.5121/ijcseit.2015.5101.
- [2] A. P. Aldya, A. Rahmatulloh, and M. Fachurroji, "HAVERSINE FORMULA UNTUK MEMBATASI JARAK PADA APLIKASI PRESENSI ONLINE," vol. 4, no. 2, 2019.
- [3] A. Fali Oklilas, D. Siswanti, and M. Dieka Rachman, "Akurasi Pembacaan GPS pada Android untuk Location Based Service (Studi Kasus: Informasi Lokasi SMA di Palembang) Reading Accuracy GPS in Android for Location Based Service (case study: Location Information of Senior High School in Palembang)," [Online]. Available: <http://journal.ipb.ac.id/index.php/jika>
- [4] E. Yulsilviana, P. Adytia, and I. N. Riandika, "PENCARIAN DRIVER DRY CLEAN TERDEKAT DENGAN METODE HAVERSINE FORMULA," *Sebatik*, vol. 25, no. 1, Jun. 2021, doi: 10.46984/sebatik.v25i1.1210.
- [5] C. N. Alam, K. Manaf,) Aldy, R. Atmadja,) Digital, and K. Aurum, "Implementation of Haversine Formula for Counting Event Visitor in The Radius Based on Android Application 1)."
- [6] D. R. Amalia, "PENINGKATAN AKURASI PENENTUAN LOKASI PADA APLIKASI ANDROID MENGGUNAKAN ASSISTED GPS (A-GPS) DAN KALMAN FILTER," 2020.
- [7] A. Hevner and J. Park, "Design Science in Information Systems Research," 2004. [Online]. Available: <https://www.researchgate.net/publication/201168946>
- [8] Q. Deng and S. Ji, "A Review of Design Science Research in Information Systems: Concept, Process, Outcome, and Evaluation," *Pacific Asia Journal of the Association for Information Systems*, pp. 1–36, 2018, doi: 10.17705/1pais.10101.
- [9] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A design science research methodology for information systems research," *Journal of Management Information Systems*, vol. 24, no. 3, pp. 45–77, 2007, doi: 10.2753/MIS0742-1222240302.
- [10] Fidi Supriadi, Y. Yan Sofiyan, and Muhammad Agreindra Helmiawan, "A Model of Virtual Learning Environments Using Micro-Lecture."
- [11] L. Sadath, K. Karim, and S. Gill, "Extreme programming implementation in academia for software engineering sustainability," in *2018 Advances in Science and Engineering Technology International Conferences, ASET 2018*, Institute of Electrical and Electronics Engineers Inc., Jun. 2018, pp. 1–6. doi: 10.1109/ICASET.2018.8376925.
- [12] S. Nidhra, "Black Box and White Box Testing Techniques - A Literature Review," *International Journal of Embedded Systems and Applications*, vol. 2, no. 2, pp. 29–50, Jun. 2012, doi: 10.5121/ijesa.2012.2204.