

## SISTEM MONITORING BERBASIS *IoT* MENGGUNAKAN *REST API* DENGAN METODE HTTP DAN *Haversine Formula*

Ersa Zhafrina Febriani<sup>1</sup>, Adianto<sup>2</sup>, Zindu Maulana Ahmad Putra<sup>3</sup>

<sup>1,2,3</sup>Program Studi Teknik Otomasi, Jurusan Teknik Kelistrikan Kapal

Politeknik Perkapalan Negeri Surabaya

Jalan Teknik Kimia, Kampus ITS Sukolilo, Surabaya 60111

E-mail : ersazhafrina@student.ppns.ac.id

### Abstrak

Budaya kejujuran menjadi aspek yang penting bagi sebuah perusahaan. Akan tetapi, pada kenyataannya masih banyak karyawan melakukan kecurangan terhadap perusahaan tempat mereka bekerja. Salah satu contoh kecurangan yang dilakukan oleh karyawan adalah kecurangan *sales* berupa pecah nota dan tidak melakukan kunjungan ke toko - toko. Kecurangan ini membuat perusahaan merugi dan mempunyai data yang tidak valid. Untuk mengatasi hal tersebut diperlukan suatu teknologi yang memudahkan perusahaan untuk memastikan *sales* melakukan kunjungan. Dengan kondisi seperti diatas penulis mempunyai ide untuk menciptakan sebuah aplikasi android untuk memonitor *sales* saat bekerja didukung menggunakan *Rest Api* yang memanfaatkan metode *Hypertext Transfer Protokol* (HTTP) untuk *record data transaction* dan mengirim notifikasi apabila *sales* berada di luar area kerja. Data akan disimpan di *database* MySQL. *Database* ini dapat diakses oleh kantor dengan bantuan *software* lokal yang ada di kantor. Selain HTTP sebagai metode pengiriman data, aplikasi ini juga dilengkapi dengan metode *Haversine Formula Method* untuk menentukan radius kerja dari masing – masing *sales*. Pada penelitian kali ini, akan berfokus pada *record data transaction*. Berdasarkan hasil pengujian, metode HTTP yang dirancang pada sistem ini berjalan dengan baik dan Metode *Haversine Formula* yang digunakan pada sistem ini mampu menentukan jarak dari 2 koordinat yang berbeda dengan nilai *error* yang dihasilkan sebesar 0.18% pada masing – masing metode percobaan. Selain itu, pada sistem ini dapat mengirimkan data dari aplikasi ke *database* dan menampilkannya pada *platform* yang telah dipilih serta terdapat notifikasi apabila *sales* keluar dari area kerja.

**Kata Kunci :** *Haversine Formula Method*, HTTP, *Rest Api*.

### 1. PENDAHULUAN

Budaya kejujuran menjadi aspek yang penting bagi sebuah perusahaan. Tidak hanya kejujuran, ada beberapa aspek dari perilaku karyawan yang tentunya sangat penting bagi perusahaan. Reinout E. de Vries dan Jean-Louis van Gelder dalam jurnalnya yang berjudul *Explaining workplace delinquency: The role of Honesty–Humility, ethical culture, and employee surveillance* tahun 2015 menjelaskan bahwa pentingnya kejujuran, kerendahan hati, kesadaran budaya etis, dan pengawasan karyawan dalam prediksi kejahatan di tempat kerja. Terutama di perusahaan dimana kejahatan atau kecurangan di tempat kerja menimbulkan kerugian bagi perusahaan [2].

Berdasarkan survei integritas dari KPMG pada tahun 2013 menemukan bahwa alasan yang paling sering digunakan oleh para karyawan untuk melakukan kecurangan di tempat kerja adalah tekanan untuk memenuhi target, sinisme terhadap kode etik organisasi,

sistem yang menghargai hasil melebihi kemampuan, dan ketakutan kehilangan pekerjaan jika target tidak tercapai. Lebih sedikit responden dari survei integritas yang dilakukan oleh KPMG pada tahun 2013 menyatakan alasan keuntungan pribadi individu sebagai salah satu faktor untuk kecurangan di tempat kerja [3].

Sebuah perusahaan pasti mempunyai tim untuk mendistribusikan produk maupun jasa yang ditawarkan. *Sales* merupakan salah satu dari bagian pendistribusian produk di sebuah perusahaan. Kecurangan yang dilakukan oleh karyawan perusahaan masih terjadi. Kecurangan yang dilakukan oleh beberapa *sales* perusahaan bisa berupa pecah nota dan tidak melakukan kunjungan ke toko - toko. Claudia Danny Christian dalam jurnalnya yang berjudul *Evaluasi Internal Control dan Desain Pengendalian untuk Mengatasi Pecah Nota di PT. Surya Madistrindo* tahun 2018 menjelaskan bahwa pecah nota sendiri adalah suatu tindakan yang memanfaatkan nota

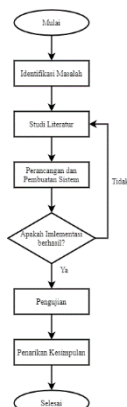
penjualan dari satu toko kemudian dipecah menjadi beberapa nota penjualan [1].

Kecurangan ini membuat perusahaan merugi dan mempunyai data yang tidak valid. Untuk mengatasi hal tersebut diperlukan suatu teknologi yang memudahkan perusahaan untuk memastikan salesnya melakukan kunjungan. Dengan kondisi seperti diatas penulis mempunyai ide untuk menciptakan sebuah aplikasi android untuk memonitor *sales* saat bekerja didukung menggunakan *Hypertext Transfer Protokol* (HTTP) untuk *record data transaction* dan mengirim notifikasi apabila *sales* berada di luar area kerja. Data akan disimpan di *database* MySQL yang ada di komputer perusahaan. Selain HTTP sebagai metode pengiriman data, aplikasi ini juga dilengkapi dengan metode *Haversine Formula Method* untuk menentukan radius kerja dari masing – masing *sales*. Perhitungan jarak antar 2 titik atau lokasi yang berbeda menggunakan formula *Haversine* yang kemudian dari jarak tersebut dinyatakan sebagai radius. Jadi radius setiap *sales* berpindah lokasi akan berbeda – beda. Apabila *sales* berada di daerah dengan radius lebih dari radius yang telah diukur oleh sistem maka *sales* tersebut dinyatakan keluar area kerjanya (radius kerjanya) sehingga notifikasi akan muncul pada aplikasi android. Dengan adanya aplikasi ini, perusahaan dapat melakukan monitoring kinerja dari *sales*.

## 2. METODOLOGI

### 2.1 Diagram alur penelitian

Diagram alur penelitian dari sistem ini secara sistematis merupakan langkah – langkah yang dilakukan dalam penelitian ini. Diagram alur penelitian dapat ditunjukkan pada **Gambar 2.1**.



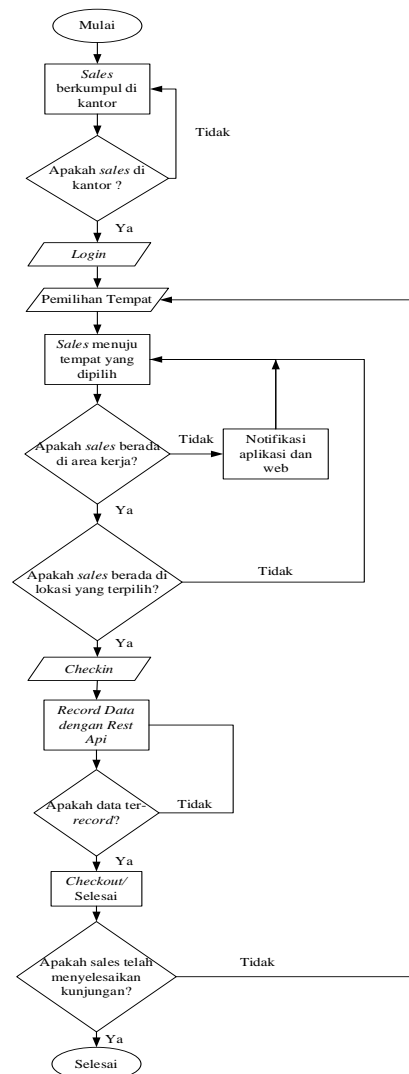
**Gambar 2.1** Diagram Alur Penelitian

Dari diagram alur tersebut, dapat dilihat bahwa tahapan - tahapan penelitian yang akan dilakukan. Dimana setiap tahap akan dijabarkan sehingga dapat membentuk

kerangka penelitian yang sistematis. Tahapan yang akan dilakukan dalam penelitian ini dimulai dengan identifikasi masalah, studi literatur, perancangan dan pembuatan sistem, implementasi sistem, pengujian dan penarikan kesimpulan

### 2.2 Flowchart sistem

Pada tahapan ini terdapat langkah-langkah untuk mengidentifikasi masalah jika terjadi kendala. Perancangan sistem awal pada penelitian ini dapat ditunjukkan pada flowchart dibawah ini.



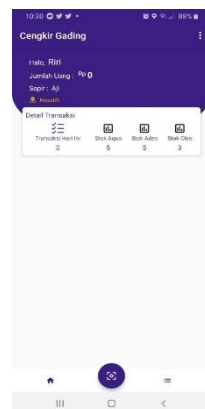
**Gambar 2.2** Flowchart sistem

Perancangan sistem yang dilakukan pada penelitian ini dijabarkan pada **Gambar 2.2**. *Sales* harus berada di kantor terlebih dahulu untuk mendapatkan *device* khusus yang terdapat aplikasi *monitoring sales* ini di dalamnya. *Input* sistem berupa *login sales* ketika berada di kantor. Selanjutnya dilakukan

pemilihan beberapa tempat tujuan. Setelah memilih tempat yang akan dituju, sistem akan menghitung jarak dari 2 titik lokasi yaitu lokasi terkini *sales* dan lokasi yang akan dituju oleh *sales* menggunakan metode *Haversine Formula*. Jarak dari 2 titik yang berbeda ini dinamakan radius, dimana apabila *sales* pergi ke tempat dengan radius melebihi radius yang telah ditentukan oleh sistem maka alarm notifikasi pada aplikasi android *sales*. Saat sampai di tempat tujuan, *sales* baru bisa melakukan *checkin* yang kemudian dilanjutkan dengan transaksi (apabila ada transaksi, jika tidak ada maka data akan otomatis tertulis tidak ada transaksi) yang akan direcord oleh sistem dengan metode HTTP menggunakan *Rest API*. Data tersebut selanjutnya akan disimpan pada *database* lokal yang ada di komputer kantor. Selanjutnya ketika *sales* selesai bertransaksi, *sales* bisa melakukan *checkout* (selesai) dan melanjutkan perjalanan ke tempat yang sudah dipilih sebelumnya. *Track record data* mulai dari apakah *sales* melakukan kunjungan, transaksi saat ada di toko, hingga waktu saat melakukan transaksi disimpan di *database* MySQL. Apabila kantor ingin melihat data transaksi yang telah dilakukan oleh *sales*, data dapat dilihat di *software* lokal kantor.

### 3. HASIL DAN PEMBAHASAN

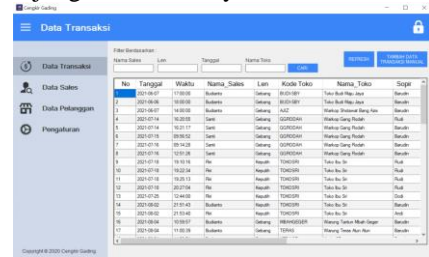
#### 3.1 Hasil Perancangan Software



Gambar 3.1 Tampilan Aplikasi Android

Gambar 3.1 merupakan tampilan aplikasi android yang telah dibuat untuk sistem ini. Aplikasi android ini dioperasikan oleh *sales* saat melakukan transaksi di toko. Setelah *sales* berhasil melakukan *login*, maka *user/sales* akan masuk pada tampilan *Home* aplikasi android. Terdapat 3 menu yang ada pada *taskbar* di bawah tampilan. Menu yang pertama adalah menu untuk *Home* yang akan menampilkan halaman utama pada aplikasi. Menu yang kedua adalah *scan QR-Code* untuk

melakukan transaksi ketika *sales* berada di toko tujuannya. Sedangkan menu yang ketiga adalah menu untuk melihat *list* toko yang akan dikunjungi *sales* nantinya.



Gambar 3.21 Tampilan Cengkir Gading App

Gambar 3.2 merupakan tampilan default dari Cengkir Gading App. Software ini dioperasikan oleh admin yang ada di kantor untuk *monitoring sales*. Tampilan halaman ini akan muncul apabila admin telah melakukan *login* dengan *username* dan *password* yang benar. Halaman ini menampilkan detail data transaksi selama *sales* beroperasi. Selain menampilkan detail data transaksi, pada Cengkir Gading App ini juga terdapat berbagai fitur yang mendukung seperti, dapat menampilkan lokasi terkini *sales* yang dipilih, *export* data transaksi ke dalam *Microsoft Excel*, dan lain sebagainya.

#### 3.2 Hasil Percobaan

Percobaan yang dilakukan adalah perhitungan metode *Haversine Formula*. Pada perhitungan ini dilakukan 2 percobaan, yang pertama percobaan perhitungan secara manual menggunakan *Microsoft Excel*, percobaan kedua dilakukan dengan cara pemrograman menggunakan Android Studio. Pada percobaan pertama dan kedua digunakan rumus yang sama. Dari dua percobaan tersebut akan dicocokkan dengan jarak yang ada *Google Maps* dengan koordinat yang sama. Dari perbandingan tersebut nantinya akan terlihat seberapa besar error dari masing – masing percobaan yang telah dilakukan. Untuk perhitungan *Haversine Formula* menggunakan rumus sebagai berikut : Haversine formula [4]

:

$$a = \sin^2(\Delta\phi/2) + \cos \phi_1 \cdot \cos \phi_2 \cdot \sin^2(\Delta\lambda/2) \quad (1)$$

$$c = 2 \cdot \text{atan2}(\sqrt{a}, \sqrt{1-a}) \quad (2)$$

$$d = R \cdot c \quad (3)$$

Catatan : sudut tersebut harus dalam radian untuk fungsi trigonometri.

$$1^\circ \times \pi/180 = 0.01745 \text{ rad} \quad (4)$$

$$\phi = \text{latitude} \quad (5)$$

$$\lambda = \text{longitude} \quad (6)$$

$$R = \text{radius bumi} \quad (7)$$

$$\pi = \text{pi (3.14)} \quad (8)$$

Tabel 3.13 Tabel Perbandingan *error*

	Toko	Total Err. Man	Total Err. Pro
Keputih	Bu Nini	1.90%	1.90%
	Pakni	2.05%	2.05%
	Camelia	0.86%	0.85%
	Trenstar	-2.51%	-2.52%
	Mars	0.01%	0.01%
Gebang	Mbah Geger	0.45%	0.45%
	Teras	0.26%	0.26%
	Kedai 37	0.27%	0.27%
	Bu Prapti	0.22%	0.22%
	Rm. Sarwono	0.22%	0.22%
Rungkut	Toko Sasmito	0.24%	0.24%
	Abimart	0.14%	0.14%
	Warung Mia	0.17%	0.17%
	Warkop Upik	0.27%	0.26%
Gubeng	Warung Cipto	-0.03%	-0.03%
	Wr. Bubur	-0.03%	-0.03%
	Wr. Bu Icus	-0.03%	-0.03%
	Kedai Maple	-0.01%	-0.02%
Mulyorejo	Toko Pojok	0.06%	0.05%
	Rm. Yauk	0.13%	0.13%
	Depot Tejo	0.19%	0.18%
	Wr. Bu Yadi	0.20%	0.19%
Mulyosari	Mie Ulama	0.11%	0.11%
	Bakso Wngr	0.11%	0.11%
	Alifa	-0.08%	-0.08%
	Wr. Bambu	-0.38%	-0.38%

	Rata Rata	0.183380 35%	0.181736 18%
--	-----------	-----------------	-----------------

Dari percobaan di atas terdapat setidaknya 26 toko dengan 19 toko terdapat beberapa *checkpoint* pada tiap perjalanan menuju ke toko. Dikarenakan perhitungan menggunakan metode *Haversine Formula* adalah dengan menarik garis lurus pada 2 koordinat yang berbeda, sedangkan jika 2 koordinat tersebut dimasukkan ke dalam *Google Maps* untuk menghitung jarak dari kedua koordinat tersebut, *Google Maps* tidak bisa langsung menarik garis lurus seperti pada metode *Haversine Formula*. Sehingga, pada perjalanan dari titik *user* (kantor) ke toko akhirnya dipecah menjadi beberapa *checkpoint*. Pada akhirnya dari total 26 toko yang dijadikan koordinat tujuan untuk percobaan, berkembang menjadi 377 titik percobaan. Dengan demikian rata – rata *error* yang dihasilkan pada 377 percobaan ada pada **Tabel 3.1**.

Tabel 3.2 Tabel Pengujian Data Transaksi

No	Detail Data Transaksi ke -	KET	
		Android	VB
1	Data ke - 62	Terkirim	Terkirim
2	Data ke - 63	Terkirim	Terkirim
3	Data ke - 64	Terkirim	Terkirim
4	Data ke - 65	Terkirim	Terkirim

Pada **Tabel 3.2** merupakan beberapa data pengujian metode HTTP untuk *GET* dan *POST*. Terdapat 65 data transaksi yang pernah dilakukan dimana dari keseluruhan data tersebut tidak ada yang tidak terkirim ke *client*. Data transaksi ini *disubmit* oleh *sales* menggunakan metode *POST* kemudian data masuk ke *server* dan dapat diakses oleh *client* menggunakan metode *GET*, sehingga data transaksi dapat dilihat oleh *client*.

Tabel 3.3 Tabel Pengujian Notifikasi

No	Nama Sales	Jarak	Notifikasi
1	Budianto	350	Muncul
2	Sandi	230	Muncul
3	Riri	270	Muncul
4	Rara	198	Muncul
5	Surya	210	Muncul
6	Santi	195	Muncul

**Tabel 3.3** merupakan pengujian untuk notifikasi. Metode HTTP yang digunakan untuk sistem ini adalah *PUT*. Jenis metode ini

mampu melakukan *replace* data pada *server* dimana data yang *direplace* pada sistem ini adalah data lokasi *sales* yakni berupa *latitude* dan *longitude sales*. Dengan jenis metode ini, data koordinat *sales* akan *update* tiap detiknya, sehingga apabila *sales* terlalu jauh dari area kerja, maka notifikasi akan muncul. Jarak yang terdapat pada **Tabel 3.3** merupakan jarak *sales* dari titik tengah area kerjanya. Dari 6 kali percobaan yang dilakukan, tidak ada satupun *sales* yang tidak mendapat notifikasi ketika *sales* keluar dari area kerjanya.

Tabel 3.4 *Report Data Transaksi*

Tanggal	09/04/2021				
Waktu	10:33:57	10:36:30	10:46:06	10:51:06	10:51:49
Sales	Riri				
Len	Keputih				
Toko	Warkop Pak Ni	Toko Bu Nini	Trendstar	Toko Camelia	Mars Mart
Aqua	1	1	1	1	1
Ades	1	1	1	1	1
Cleo	1	0	2	0	0
Club	0	0	0	0	0
Pristine	0	0	0	0	0
Harga	12.000	10.000	21.000	15.000	10.000
Awal_Aqua	5	4	3	2	1
Akhir_Aqua	4	3	2	1	0
Awal_Ades	5	4	3	2	1
Akhir_Ades	4	3	2	1	0
Awal_Cleo	3	2	2	0	0

Akhir_Cleo	2	2	0	0	0
------------	---	---	---	---	---

Pada **Tabel 3.4** merupakan cuplikan dari tabel data yang digunakan untuk kepentingan administrasi perusahaan. Dari **Tabel 3.4** dapat dilihat bahwa stok awal barang memiliki jumlah yang berbeda – beda tiap toko. Hal ini terjadi karena, ketika *sales* melakukan transaksi pada toko pertama jumlah stok awal barang berasal dari jumlah yang telah dimasukkan oleh admin, kemudian ketika *sales* melakukan transaksi di toko kedua (toko selanjutnya), maka jumlah stok awal barang yang digunakan adalah jumlah sisa stok dari toko pertama (toko sebelumnya) dimana jumlah atau angka tersebut didapat dari jumlah stok awal barang yang dimasukkan oleh admin dikurangi dengan jumlah barang yang ditransaksikan. Misalnya, pada saat *sales* berangkat, jumlah stok awal Aqua adalah 5, ketika tiba di Warkop Pak Ni, barang yang ditransaksikan sebanyak 1 karton, maka sisa stok setelah transaksi pada Warkop Pak Ni adalah 4. Nilai 4 ini akan dijadikan sebagai jumlah stok awal pada toko selanjutnya. Dapat dilihat pada **Tabel 3.4**, setelah dari Warkop Pak Ni, *sales* melanjutkan kembali transaksinya di Toko Sembako Bu Nini, jumlah stok awal barang pada saat akan transaksi di Toko Sembako Bu Nini adalah 4, dimana jumlah tersebut diambil dari sisa stok barang saat di toko sebelumnya. Alur jumlah stok barang ini akan berhenti apabila nilai sisa stok barang adalah 0, yang artinya barang yang dibawa oleh *sales* sudah habis atau selesai ditransaksikan.

## 4. PENUTUP

### 4.1 Kesimpulan

Berdasarkan hasil yang telah didapatkan pada analisa dan pengujian sistem yang telah dibuat, maka dapat disimpulkan bahwa:

- 1) Rancangan sistem yang telah dibuat menggunakan metode HTTP berjalan dengan baik. Hal ini dapat dibuktikan dengan rincian data hasil percobaan, salah satu contohnya pada aplikasi Android dimana terdapat data transaksi yang muncul pada halaman home sesaat setelah *sales* melakukan transaksi.
- 2) Metode *Haversine Formula* pada penentuan jarak dan area sales meliputi:
  - a) Metode *Haversine Formula* relatif andal untuk penentuan jarak dari 2

titik koordinat yang berbeda. Hal ini terbukti dengan kecilnya nilai error yang dihasilkan pada 377 titik percobaan yang telah dilakukan yaitu sebesar 0.18% pada masing – masing metode percobaan.

- b) Hasil dari metode percobaan manual dan percobaan pada program relatif sama. Dari hasil percobaan yang telah dilakukan, nilai yang dihasilkan dengan perhitungan manual dan program tidak berbeda jauh, hal ini dapat dilihat pada Tabel 4.4 dimana error yang dihasilkan hampir sama.
- 3) Dalam pengujian sistem ini, terdapat beberapa *output* yang dihasilkan yaitu tampilan data transaksi sales pada *user* (*sales*) dan kantor, *login* dan *checkin* yang mengharuskan *sales* berada di tempat yang bersangkutan, serta notifikasi ke perangkat Android apabila *sales* keluar area kerja. *Output – output* tersebut berjalan dengan baik saat pengujian dilakukan.

#### 4.2 Saran

Berdasarkan hasil dan kesimpulan Tugas Akhir ini, terdapat beberapa saran untuk melanjutkan dan memperbaiki kekurangan yang ada pada Tugas Akhir ini, yaitu:

- 1) Metode pengolahan data apabila menggunakan metode MQTT akan lebih cepat dan lebih optimal.
- 2) Penggunaan sistem client server dengan supporting cloud akan lebih meningkatkan kinerja dengan jangkauan dan jumlah titik yang lebih luas.
- 3) Menggunakan provider dengan kualitas internet yang baik dan stabil akan lebih meningkatkan kinerja *output* dari sistem ini utamanya dalam implementasi metode *Haversine Formula* dan metode HTTP.

#### 5. DAFTAR PUSTAKA

- [1] Christian, C. D. 2018. *Evaluasi Internal Control dan Desain Pengendalian* : Jurnal Ilmiah Mahasiswa Universitas Surabaya, Volume 7
- [2] de Vries, R. E. and van Gelder, J. L. 2015. *Explaining workplace delinquency: The role of Honesty–Humility, ethical culture, and employee surveillance* : Elsevier B.V.

[3] Kpmg. 2013. *KPMG Integrity Survey* : KPMG

[4] Vennes, C. 2002. *Calculate distance, bearing and more between Latitude/Longitude points* : MIT Open Source