

## BAB V

### IMPLEMENTASI DAN PENGUJIAN

#### 5.1 Implementasi Sistem

Bab ini membahas mengenai implementasi dan dari sistem presensi yang telah dibangun dan uji coba terhadap aplikasi presensi. Implementasi sistem ini merupakan tahapan selanjutnya dari analisis dan perancangan yang telah disampaikan pada tahap sebelumnya. Tujuan dilakukannya pengujian adalah untuk mengetahui nilai keakuratan sinyal gps pada perangkat android dan dapat berjalan baik pada aplikasi tersebut. Selain itu juga untuk mengetahui apakah aplikasi yang dibangun dapat berjalan sesuai dengan apa yang sudah direncanakan

##### 5.1.1 Implementasi Perangkat Keras

Perangkat keras yang digunakan pada proses pembuatan dan pengujian sistem presensi mobile ini adalah sebagai berikut :

###### A. Proses pembuatan

1. *Processor* : Intel(R) Celeron(R) CPU N3550 @ 1.10GHz (2CPUs)
2. *Memory* : 4096MB
3. *Storage* : 512 GB
4. *Video Card* : Intel(R) HD Graphics

###### B. Proses Pengujian

1. *Pricesor* : Exynos 9611 (10nm) dan Intel(R) Celeron(R) CPU N3550 @ 1.10GHz (2CPUs)
2. *Memory* : 4096MB
3. *Storage* : 64 GB
4. *Video Card* : Mali-G72 MP3 dan Intel(R) HD Graphics

##### 5.1.2 Implementasi Perangkat Lunak

Perangkat lunak yang digunakan pada saat proses pembuatan dan pengujian sistem presensi mobile adalah sebagai berikut :

A. Proses pembuatan

- 5. Sistem Operasi : Windows Pro 64-bit
- 6. *Database* : Google Drive
- 7. *Server* : Google Cloud, *Traccar server*
- 8. Bahasa Pemrograman : Appsheet
- 9. *Tools* : Google Chrome
- 10. *Text / Code Editor* : *Appsheet.com* dan *traccar.com*

B. Proses pengujian

- 1. Sistem Operasi : Android 10 dan Windows 10 home edition
- 2. *User Interface* : One UI 3.0

## 5.2 Implementasi Algoritma

Pada tahap ini akan dijelaskan tahapan implementasi algoritma pada bagian aplikasi, implementasi tersebut dibagi menjadi beberapa bagian, diantaranya :

### 5.2.1 Proses Akuisisi Data Lokasi

Tahap proses akuisisi data lokasi adalah tahap awal sebelum tahap proses perhitungan area lokasi terhadap *longitude* dan *latitude* area lokasi. Pertama-tama yaitu dengan mengambil data *longitude* dan *latitude* menggunakan GPS yang sudah tertanam pada perangkat android.

Pada gambar 5-1 merupakan *pseudo code* dalam proses pengambilan data lokasi melalui GPS yang tertera pada perangkat android. Agar dapat mengambil data lokasi berupa data *latitude* dan *longitude* diperlukan pemanggilan kelas *location manager* yang sudah terdapat pada perangkat android.

Kelas *location manager* adalah kelas yang menyediakan akses *location server*, yang dapat memberi izin aplikasi untuk memperoleh pembaharuan lokasi perangkat sesuai dengan interval waktu yang telah ditentukan. Setelah memanggil kelas *location manager* selanjutnya aplikasi akan memastikan apakah kondisi GPS\_PROVIDER pada perangkat yang tersedia. Jika pada perangkat tersebut GPS tidak tersedia akibatnya aplikasi akan memanggil kelas *alert dialog* yang dapat

memberikan peringatan untuk *user* agar menghidupkan GPS\_PROVIDER. Jika GPS\_PROVIDER sudah tersedia maka kelas *location manager* dapat melakukan proses pengambilan data lokasi yaitu berupa data *latitude* dan *longitude*.

```

Deklarasi :
LocationManager, boolean GPS_PROVIDER
    if(GPS_PROVIDER) then
        if(getlocation) then
            location = getlogttitude
            location = getlatitude
        endif
    else
        else alert dialog
Write longttitude, latitude

```

**Gambar 5-1** *Pseudo code* pengambilan *latitude* dan *longitude*

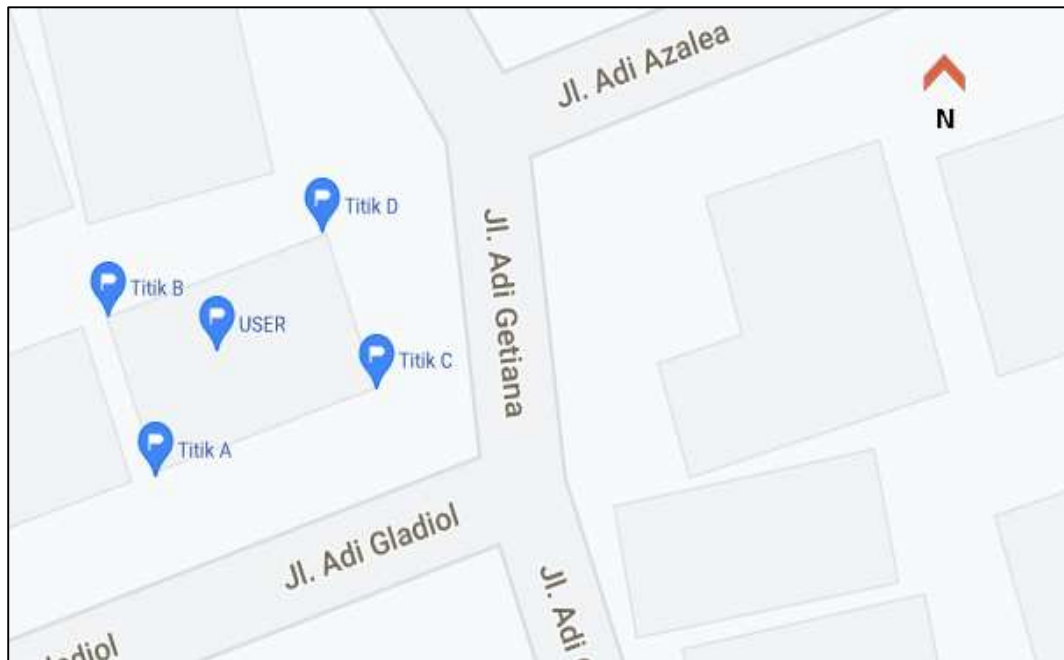
### 5.2.2 Proses Penentuan Area Lokasi

Proses perhitungan area lokasi dilakukan dengan cara mengetahui nilai dari *latitude* dan *longitude user*, setelah itu dilakukan proses AND antara nilai dari lokasi *user* dengan nilai lokasi perusahaan yang sudah dideskripsikan pada formula aplikasi.

Proses penentuan area lokasi menggunakan dua parameter yaitu parameter nilai *latitude* dan *longitude user* dan parameter nilai *latitude* dan *longitude* perusahaan, kemudian nilai dari parameter *user* akan dibandingkan dengan nilai dari parameter perusahaan yang telah ditentukan pada formula aplikasi. Perbandingan ini menggunakan nilai AND dengan *output* adalah satu nilai benar. Area lokasi perusahaan adalah nilai *latitude* dan *longitude* yang terdapat pada bagian masing-masing pojok gedung perusahaan.

Bagian masing-masing pojok gedung terdapat empat pojok, dimana pojok pertama disebut sebagai titik A, pojok kedua disebut titik B, titik ketiga disebut pojok C dan pojok keempat disebut titik C. Keempat pojok tersebut membentuk bangun datar persegi panjang apabila dilihat dari *perta/maps*. Nilai *latitude* dan *longitude* titik A yang berada pada pojok sebelah selatan barat adalah -6.965430 ,

107.69111. Sedangkan nilai titik B yang berada pada pojok sebelah utara barat dengan nilai *latitude* dan *longitude* adalah -6.965374 , 107.691096. Sedangkan nilai titik C yang berlokasi pada pojok sebelah selatan timur dengan nilai *latitude* dan *longitude* adalah -6.965399, 107.691199. Kemudian titik D yang berlokasi pada bagian pojok sebelah utara timur dengan nilai nilai *latitude* dan *longitude* adalah -6.965345, 107.691178. Agar dapat lebih dipahami area lokasi perusahaan dapat dilihat pada gambar 5-2.



**Gambar 5-2** Proses penentuan area Lokasi

Untuk perhitungan nilai lokasi dapat dilakukan dengan menggunakan nilai masukan *latitude* dan *longitude* pada perangkat yang digunakan oleh *user*. Pada kasus ini nilai masukan *latitude* dan *longitude user* adalah : -6.965386 , 107.691138

Nilai pada masing-masing pojok gedung perusahaan tersebut adalah :

Nilai Titik A(x, y) = ( -6.965430 , 107.691114 )

Nilai Titik B(x, y) = ( -6.965374 , 107.691196 )

Nilai Titik C(x, y) = ( -6.965399 , 107.691199 )

Nilai Titik D(x, y) = ( -6.965445 , 107.691178 )

Pada posisi koorninat *user*. *User* akan dianggap berada pada area lokasi jika berada didalam area dengan memenuhi kondisi :

Kondisi ke 1, garis AB: if  $Ax \leq longitude \ \&\& \ Bx \leq longitude$

Kondisi ke 2, garis BC: if  $Cx \geq longitude \ \&\& \ Dx \geq longitude$

Kondisi ke 3, garis CD: if  $Ay \leq latitude \ \&\& \ Dy \leq latitude$

Kondisi ke 4, garis DA: if  $By \geq latitude \ \&\& \ Cy \geq latitude$

Setelah semua kondisi terpenuhi dan bernilai benar maka *user* dianggap berada pada area lokasi perusahaan. *Pseudo code* pada gambar 5-3 akan memberi penjelasan lebih jalanya algoritma pendeteksi area lokasi menggunakan nilai *latitude* dan *longitude*.

```

Deklarasi :
double longitude, latitude
double Ax, Ay, Bx, By, Cx, Cy, Dx, Dy
String nama_lokasi
get longitude, latitude
if(Ax >= longitude && Bx <= longitude)
    &&(Cx >= longitude && Dx > longitude)
    &&(Ay <= latitude && Dy <= latitude)
    &&(By >= latitude && Cy > latitude)
        set nama_lokasi
endif


```

**Gambar 5-3** *Pseudo code* proses penentuan area lokasi

### 5.2.3 Proses Sinkronisasi ke Webservice

Proses sinkronisasi perangkat android *user* ke *webservice traccar* dimulai dengan mengaktifkan GPS pada perangkat *user*, lalu memasukan *server url* pada aplikasi *client/user*, kemudian mengaktifkan status *traccar* menjadi aktif.

Pada *server private traccar* set *ip address* dari perangkat *admin* agar menjadi *ip address public*. Setelah selesai maka proses *forwarding* akan dilakukan antara *server* dengan *ip address public* yang bisa diakses oleh sebuah *url* khusus yang akan langsung di-*redirect* ke *webservice traccar*. Pada gambar 5-4 merupakan proses sinkronisasi ke *webservice*.



```

C:\Users\robihayuk\Downloads\ngrok-stable-windows-amd64\ngrok.exe - ngrok http 80
ngrok by @inconshreveable (Ctrl+C to quit)

Session Status      online
Account             RObi Wahyu (Plan: Free)
Version             2.3.48
Region              United States (us)
Web Interface       http://127.0.0.1:4848
Forwarding           http://679e9e6e06d8.ngrok.io -> http://localhost:80
                    https://679e9e6e06d8.ngrok.io -> http://localhost:80

Connections:
  ttl    opn    rt1    rt5    p50    p90
   0     0     0.00  0.00  0.00  0.00

```

**Gambar 5-4** Proses sinkronisasi ke *web server*

### 5.3 Implementasi Interface

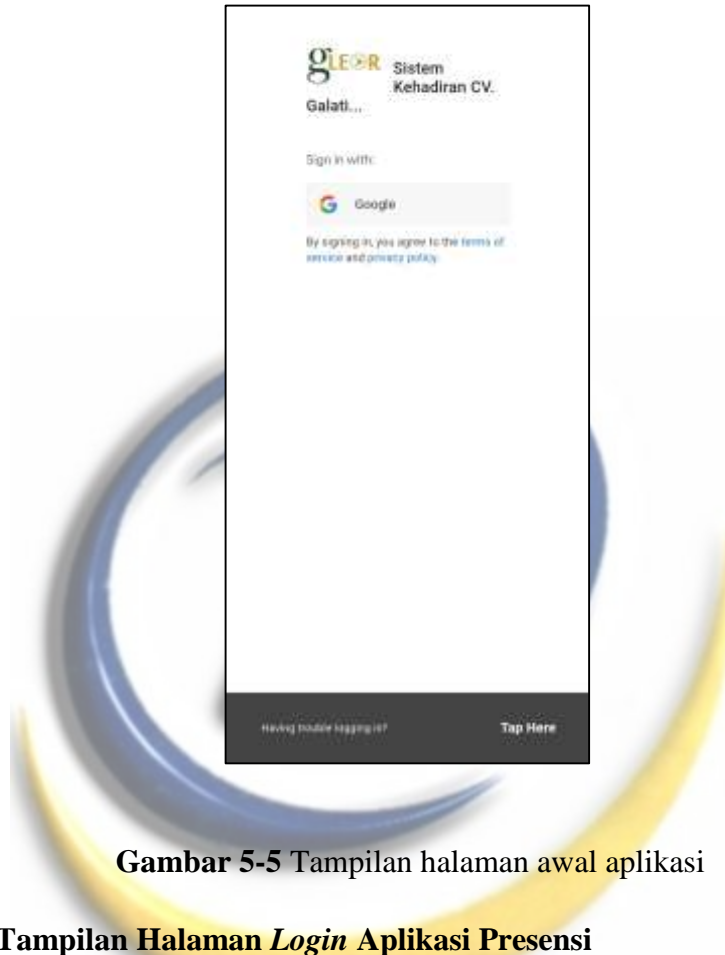
Pada tahap ini akan dijelaskan tentang implementasi *interface* dari aplikasi presensi dan *tracking user*.

#### 5.3.1 Tampilan Awal Aplikasi Sistem Presensi Android

Gambar 5-5 merupakan tampilan halaman awal aplikasi presensi. Ketika *user* membuka aplikasi presensi untuk yang pertama kalinya maka halaman inilah yang akan muncul. Apabila *user* sudah pernah membuka aplikasi presensi dan melakukan *login* menggunakan akun-nya, maka halaman ini tidak akan muncul kembali terkecuali *user* melakukan *logout* pada aplikasi presensinya.

Pada halaman ini dibagian atas terdapat logo perusahaan beserta nama dari perusahaan yang terdapat pada bagian sebelah sisi kanan dari logo perusahaan. Kemudian dibawahnya terdapat instruksi untuk melakukan *login* terlebih dahulu sebagai syarat untuk mengakses aplikasi presensi. Pada aplikasi presensi ini hanya dapat melakukan *login* menggunakan akun *google*. Pada bagian bawah halaman awal aplikasi presensi juga terdapat arahan bagi *user* yang memiliki masalah dalam proses *login* aplikasi presensi, dengan cara menekan atau memilih kata “*tap here*”

maka *user* akan diberikan penjelasan atau tatacara mengenai cara melakukan *login* aplikasi presensi.

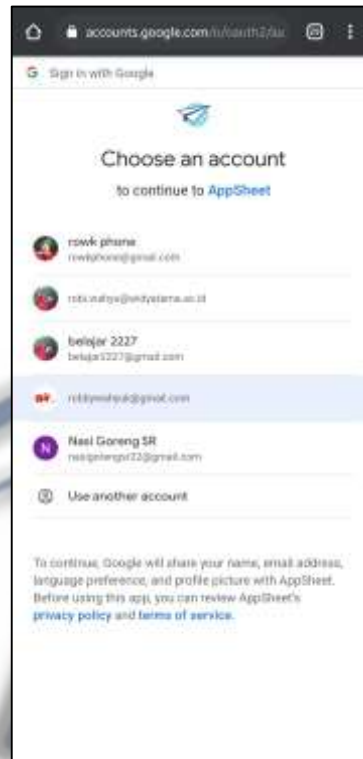


**Gambar 5-5** Tampilan halaman awal aplikasi

### **5.3.2 Tampilan Halaman *Login* Aplikasi Presensi**

Gambar 5-6 merupakan tampilan halaman *login* dari aplikasi presensi. Pada bagian atas halaman ini terdapat informasi mengenai *login* aplikasi menggunakan akun *google*. Kemudian dibawahnya terdapat informasi rekomendasi akun *google* mana yang akan digunakan sebagai akun *user*, rekomendasi akun ini akan berbeda-beda pada setiap perangkat *user*, tergantung dari berapa banyak akun *google* yang sudah ditanamkan atau dipasangkan pada perangkat *android user*. Jika tidak ada

rekomendasi maka *user* bisa menggunakan akun google yang baru yang tidak tertanam pada perangkat androidnya dengan cara memilih “*Use another account*”.



**Gambar 5-6** Tampilan halaman *login* aplikasi

### 5.3.3 Tampilan Halaman Memulai Aplikasi Presensi

Pada gambar 5-7 merupakan tampilan halaman *booting*/memulai aplikasi. Jika proses *login* telah dilakukan dan berhasil, maka halaman selanjutnya yang akan muncul adalah halaman *booting*.

Pada halaman ini terdapat logo perusahaan yang diletakan ditengah-tengah halaman aplikasi, diikuti dibawahnya terdapat informasi proses sinkronisasi aplikasi presensi. Jika *user* ingin membatalkan membuka aplikasi *user* dapat memilih atau menekan tombol *cancel* yang terdapat dibagian bawah halaman aplikasi presensi. Jika proses *booting* selesai maka aplikasi presensi akan terbuka dan dapat dikases oleh *user*.

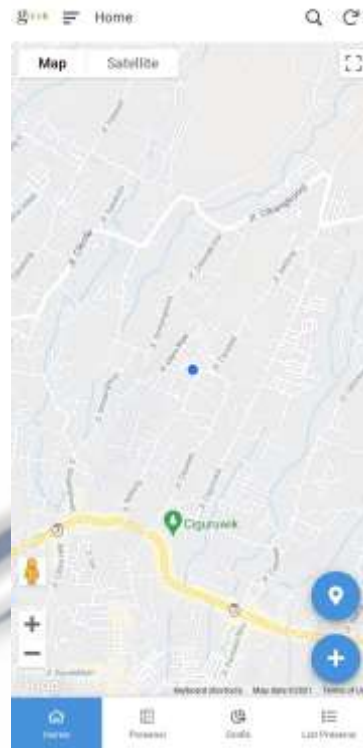


**Gambar 5-7** Tampilan halaman *booting* aplikasi

#### **5.3.4 Tampilan Halaman Depan Aplikasi Presensi**

Pada gambar 5-8 merupakan halaman utama dari aplikasi presensi. Ketika *user* sudah melakukan *login* maka setiap kali *user* membuka aplikasi presensi, maka halaman ini akan menjadi halaman utama yang selalu muncul.

Ketika *user* berada dihalaman utama, *user* dapat melihat histori lokasi *user* yang telah melakukan *input* data presensi pada peta GPS dengan cara mengklik tanda titik biru pada peta GPS akan muncul keterangan *user* disertai data presensinya, peta tersebut dapat di perbesar dan diperkecil serta dapat diubah ke mode satelit agar peta terlihat lebih realistis, pada halaman ini terdapat 2 tombol dengan logo plus dan logo penanda lokasi. Fungsi tombol dengan logo plus berfungsi sebagai jalan cepat untuk membuka halaman *input* data presensi, sedangkan tombol dengan logo lokasi berfungsi untuk menyigkronkan lokasi *user* dengan peta.



**Gambar 5-8** Tampilan halaman utama aplikasi

### 5.3.5 Tampilan Halaman *Input Data Presensi*

Pada gambar 5-9 merupakan tampilan halaman *input* data presensi, dimana halaman ini digunakan untuk melakukan presensi dengan cara memasukan data-data yang sudah ditentukan oleh perusahaan untuk melakukan presensi karyawan.

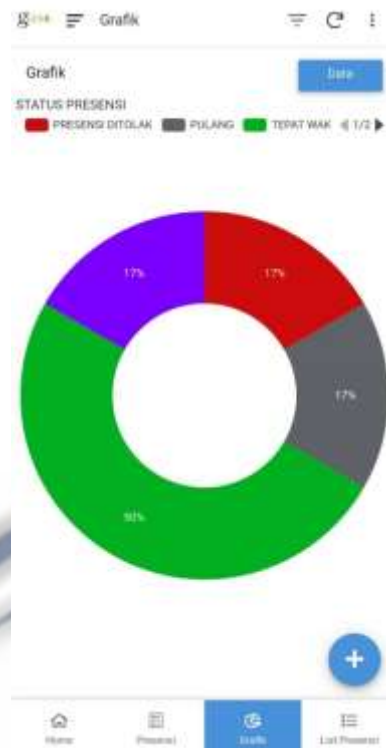
Pada halaman ini, terdapat sekumpulan *field* yang harus diisi oleh *user* agar dapat melakukan *input* data presensi. Diantaranya adalah *field* waktu dan jam yang otomatis mengikuti waktu dan jam secara *realtime* dan *user* tidak dapat mengubahnya, *field button* status karyawan, *field button* status kerja, *field text* NIP, *field* nama yang otomatis menampilkan nama karyawan sesuai dengan data nip, *field* tanda tangan digital, *field* foto selfie untuk *input* foto *selfie* melalui kamera dan tidak dapat mengambil gambar di galeri atau file *smartphone*, *field* lokasi GPS yang otomatis mengikuti lokasi *user* melalui *tracking* GPS terhadap *smartphone user*.

**Gambar 5-9** Tampilan halaman *input* data presensi

### 5.3.6 Tampilan Halaman Data Berdasarkan Grafik

Pada gambar 5-10 merupakan tampilan halaman data presensi yang dimuat dalam bentuk grafik lingkaran. Pada bagian atas halaman ini terdapat informasi mengenai perbedaan warna pada grafik, kemudian disebelah kanan atas data tombol “data” dan dibawah halaman ini terdapat grafik itu sendiri yang menunjukkan data presensi karyawan.

Didalam halaman ini terdapat dua elemen yang dihitung, yaitu tepat waktu berwarna hijau, terlambat berwarna ungu, pulang berwarna abu dan presensi ditolak berwarna merah. Selain menampilkan data secara grafik, pada halaman ini terdapat *button* atau tombol “data” yang bisa digunakan untuk menampilkan data presensi melalui tabel yang didalamnya memuat data lengkap presensi sesuai *field-field* yang ada pada halaman presensi seperti yang ada pada gambar 5-11 ini.



**Gambar 5-10** Tampilan halaman data presensi menggunakan grafik

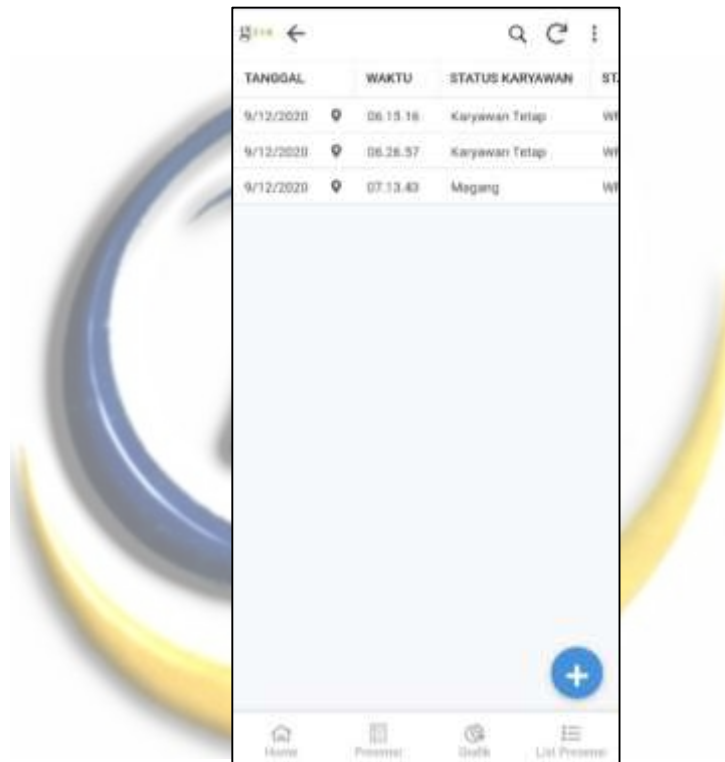
NIP	NAMA	STATUS PRESENSI
0020nwb	CEK KEMBALI POSISI ANEKA	PULANG
0020nwb	Rabi Wahyu Karmawan	TEPAT WAKTU
0020nwb	Rabi Wahyu Karmawan	PRESENSI DITOLAK
0020nwb	Rabi Wahyu Karmawan	TEPAT WAKTU
0020nwb	Rabi Wahyu Karmawan	TEPAT WAKTU
0020nwb	POSISI TIDAK VALID	TERLAMBAT

**Gambar 5-11** Tampilan halaman data presensi menggunakan tabel

### 5.3.7 Tampilan Halaman Data Berdasarkan List

Pada gambar 5-12 merupakan tampilan halaman tampilan data yang dikelompokkan berdasarkan status karyawan.

Pada halaman ini kita dapat melihat data presensi yang dikategorikan berdasarkan status karyawan, kemudian setelah dipilih akan ditampilkan data yang dikelompokkan berdasarkan status presensi, dan terakhir dikelompokkan berdasarkan tanggal.

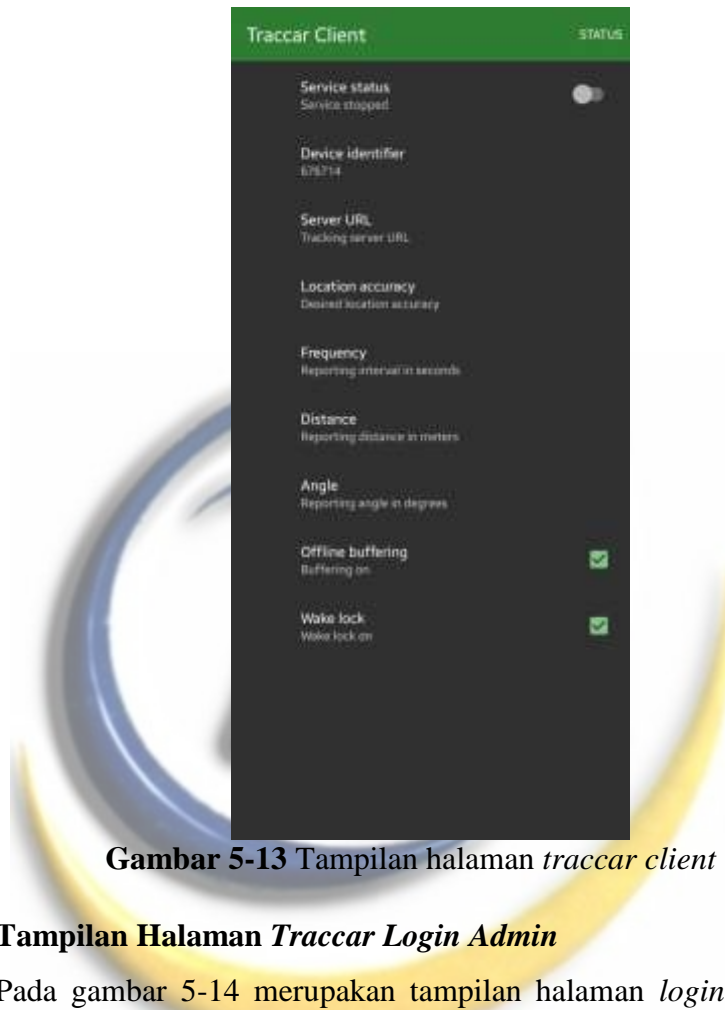


**Gambar 5-12** Tampilan halaman data presensi berdasarkan list

### 5.3.8 Tampilan Halaman Utama *Traccar Client*

Pada gambar 5-13 merupakan tampilan halaman *traccar client*. Pada halaman ini terdapat informasi mengenai data-data pengaturan aplikasi *traccar* diantaranya : *Service status*, *Device indentifier*, *Server URL*, *Location accuracy*, *Frequency*, *Distance*, *Angle*, *Offline buffering*, *Wake lock*. *Traccar* dapat berfungsi

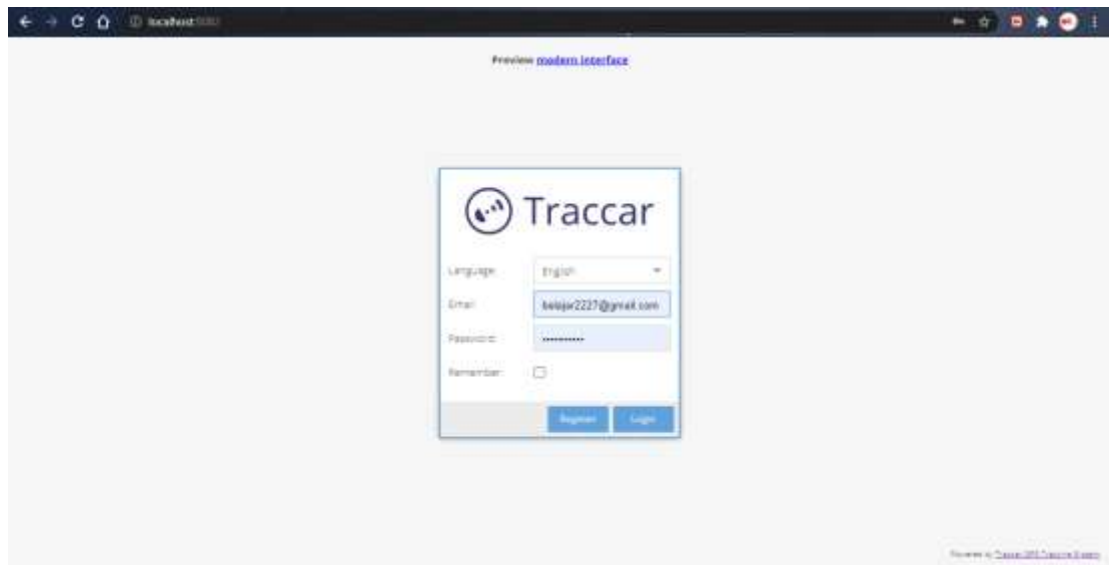
jika *service status on* maka *admin* dapat melakukan pelacakan, jika *service status off* maka pelacakan tidak dapat dilakukan.



**Gambar 5-13** Tampilan halaman *traccar client*

### 5.3.9 Tampilan Halaman *Traccar Login Admin*

Pada gambar 5-14 merupakan tampilan halaman *login admin* pada *web traccar*. Pada halaman ini terdapat formulir dibagian tengah yang berisi informasi yang harus diisi oleh *admin* untuk melakukan *login* diantaranya : *language*, *email*, *password* dan *remember me*. Kemudian dibawahnya terdapat dua tombol yaitu tombol *register* dan tombol *login*. Tombol *register* berfungsi untuk membuat akun baru dan tombol *login* berfungsi untuk melanjutkan proses *login*. Apabila proses *login* berhasil maka *admin* akan diarahkan ke halaman selanjutnya dan dapat mengakses *traccar* untuk melakukan pelacakan terhadap karyawan perusahaan.

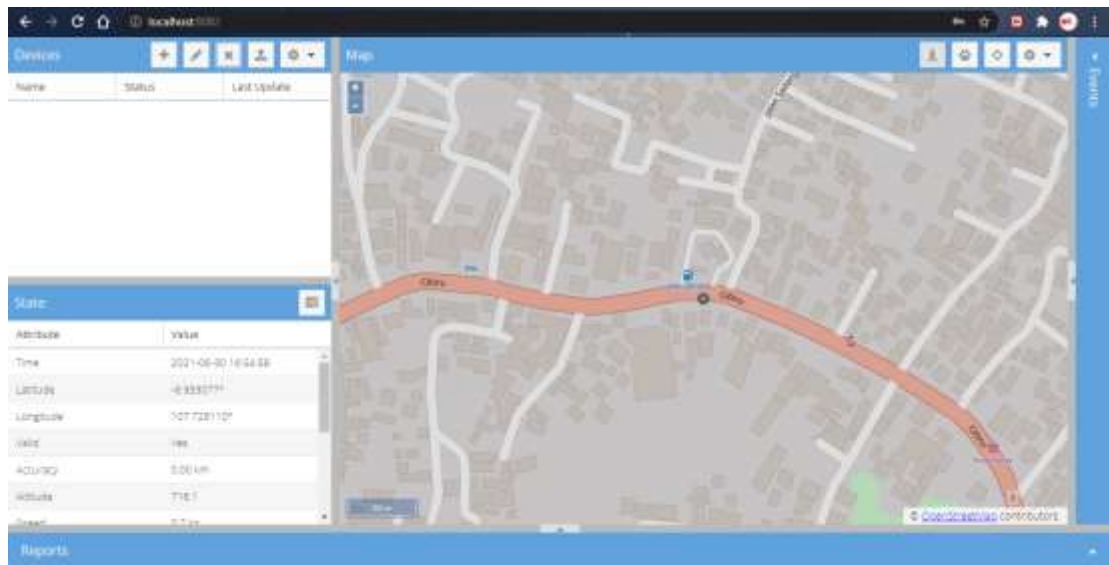


**Gambar 5-14** Tampilan halaman *login admin*

### 5.3.10 Tampilan Halaman Tracing

Pada gambar 5-15 merupakan tampilan halaman *tracking*. Pada halaman inilah proses pelacakan karyawan dilakukan. Daftar nama karyawan yang dilacak akan muncul pada tabel *view device* yang berada pada bagian atas sebelah kiri aplikasi yang memuat data *name*, status dan *last update*. Kemudian dibawah tabel *view* daftar karyawan terdapat tabel *view status* yang memberikan informasi lebih detail mengenai status karyawan yang dilacak diantaranya : *time*, *latitude*, *longitude*, *valid*, *accuracy*, *altitude*, *speed*, *battery level* dll. Kemudian pada sisi kanan tampilan halaman *tracking* terdapat peta besar yang mempunyai fungsi memberikan informasi mengenai semua lokasi karyawan yang sedang dilacak yang dapat dilihat langsung dalam peta.

Pada peta yang terdapat pada halaman *tracking* karyawan ini, tidak hanya menampilkan satu lokasi atau satu pergerakan karyawan, akan tetapi semua lokasi dan semua pergerakan karyawan yang terdaftar dapat dilihat pada peta ini secara bersamaan dan dilakukan berdasarkan interval waktu yang telah ditentukan pada formula aplikasi.



**Gambar 5-15** Tampilan halaman *tracking* karyawan

## 5.4 Pengujian Sistem

Pengujian sistem adalah tahap yang perlu dilakukan untuk menjelaskan pengoperasian perangkat lunak yang telah dibangun. Pengujian ini dibagi menjadi beberapa item yang setiap item akan diuji satu persatu.

### 5.4.1 Skenario Pengujian

Didalam rencana pengujian ini, akan dibagi menjadi beberapa item pengujian. Dari hasil pengujian yang dilakukan akan ditarik sebuah kesimpulan dan juga saran guna memperbaiki ataupun menambahkan fitur tambahan yang belum ada agar aplikasi atau sistem yang dibangun ini semakin bagus. Tujuan pengujian ini dilakukan untuk memastikan apakah aplikasi yang dibangun ini sudah berjalan dengan baik sesuai dengan yang diharapkan oleh penulis. Simulasi pengujian ini merupakan simulasi pengujian terhadap penulis, artinya pengujian dilakukan hanya secara simulasi, tidak dilakukan pengujian terhadap karyawan perusahaan dan admin perusahaan. Pengujian dilakukan terhadap *user* dan *admin* dimana penulis sendiri sebagai *user* dan *admin*. Hasil dari pengujian disajikan secara detail dan *valid*.

### 5.4.2 Pengujian

Pengujian ini menitikberatkan pada hasil keluaran dari kendali masukan. Pada kendali masukan, pengujian akan dikatakan berhasil jika keluaran sesuai dengan kendali masukan. Berdasarkan pada rencana pengujian sistem yang telah dibangun, maka dapat dilakukan pengujian sebagai berikut :

**Tabel 5-1** Rencana pengujian

Kelas Uji	Butir Uji	Identifikasi		Tingkat Pengujian	Jenis Pengujian	Jadwal Pengujian	Keterangan
		Kode SK PL	Kode PDHU PL				
Pengambilan informasi data lokasi	Mengambil data berupa nilai <i>latitude</i> dan <i>longitude</i>	KF-01	U1-01	<i>Performing Testing</i>	<i>Black Box</i>	25/08/2021 pukul 14:00	Aplikasi dapat membaca posisi <i>user</i>
Menentukan area perusahaan pada formula aplikasi	Mengambil data berupa nilai <i>latitude</i> dan <i>longitude</i> didalam are perusahaan, gedung atau bangunan	KF-02	U1-01	<i>Performing Testing</i>	<i>Black Box</i>	25/08/2021 pukul 14:00	Area perusahaan dapat dilihat di peta
Melakukan sinkronisasi ke <i>webserver traccar</i>	Melakukan sinkronisasi <i>database HI, smartphone user</i>	KF-03	U1-01	<i>Performing Testing</i>	<i>Black Box</i>	25/08/2021 pukul 14:00	Traccar client dapat tersambung dengan server traccar

	ke <i>webserver traccar</i>						
Melakukan proses presensi menggunakan aplikasi presensi	Memasukkan data presensi karyawan pada aplikasi presensi karyawan	KF-04	U1-01	<i>Performing Testing</i>		25/08/2021 pukul 14:00	Data presensi dapat disimpan
Menampilkan data lokasi karyawan pada <i>webserver traccar</i>	Menampilkan data lokasi karyawan di <i>webserver traccar</i>	KF-05	U1-01	<i>Performing Testing</i>	<i>Black Box</i>	25/08/2021 pukul 14:00	Server dapat melihat posisi <i>user</i> .

## 5.5 Hasil Pengujian

Pada tahap ini akan dijelaskan mengenai hasil dari pengujian sistem yang telah dilakukan.

### 5.5.1 Mengambil Informasi Data Lokasi

Hasil dari pengujian mengambil informasi data lokasi dapat dilihat pada tabel 5-2 yang mempunyai butir uji : mengaktifkan gps pada *smartphone user* di area lokasi perusahaan dan menyalakan fungsi akses lokasi pada aplikasi *traccar client*.

**Tabel 5-2** Hasil uji pengambilan data lokasi

Kode Uji	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria evaluasi hasil	Keluaran yang didapat	Kesimpulan
KF-01 U1-01	Mengaktifkan GPS pada <i>smartphone user</i> didalam area lokasi perusahaan	1. Pilih GPS pada Menu bar 2. Klik aktifkan	Posisi koordinat berupa <i>latitude</i> dan <i>longitude</i>	Sistem <i>android</i> mendapatkan lokasi <i>user</i> berdasarkan lokasi GPS yang aktif.	Data koordinat sama dengan posisi <i>user</i>	Sistem <i>android</i> mampu mendapatkan lokasi <i>user</i> yang telah mengaktifkan GPS pada <i>smartphone user</i>	diterima
KF-01 U1-02	Menyalakan fungsi akses lokasi pada aplikasi <i>traccar client</i>	1. Pilih aplikasi <i>traccar</i> 2. Menerima fungsi akses lokasi	Pilihan terima	Aplikasi <i>traccar client</i> dapat mengakses lokasi perangkat <i>user</i>	Aplikasi dapat mengakses lokasi <i>user</i>	Aplikasi <i>traccar client</i> dapat mengakses lokasi perangkat <i>user</i> .	diterima

### 5.5.2 Menentukan Area Perusahaan pada Formula Aplikasi

Tabel 5-3 merupakan hasil uji dari pengujian menentukan area lokasi perusahaan pada formula aplikasi, yang mempunyai butir uji mengambil data *latitude* dan *longitude* perusahaan pada *google maps* pada setiap pojok area perusahaan, memasukan data *latitude* dan *longitude* perusahaan pada formula aplikasi.

**Tabel 5-3** Hasil uji menentukan lokasi perusahaan

Kode Uji	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria evaluasi hasil	Keluaran yang didapat	Kesimpulan
KF-02 U1-01	Mengambil dan menyima n data nilai <i>latitude</i> dan <i>longitude</i> area pojok perusahaan pada formula aplikasi	3. Pilih Browser 4. Buka Google Maps 5. Tentukan koordi nat 6. Pilih dan simpan koordi nat	Memilih titik koordinat	Nilai koordinat sesuai posisi	Data berupa nilai <i>latitude</i> dan <i>longitude</i>	Nilai koodin at berupa <i>latitude</i> dan <i>longitude</i>	Diterima

### 5.5.3 Melakukan Sinkronisasi ke Webserver Traccar

Pada tabel 5-4 merupakan hasil uji sinkronisasi ke *webserver traccar*. Pada uji sinkronisasi ini terdapat butir uji diantaranya : *create ip address public*, *forwarding ip address*, memasukan *url* baru pada *traccar client*, *update location* dan *send location*.

**Tabel 5-4** Uji sinkronisasi ke *webserver*

Kode Uji	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria evaluasi hasil	Keluaran yang didapat	Kesimpulan
KF-03 U1-01	Sinkronisasi ke <i>web server</i>	7. Pilih Browser 8. Buka Google Maps 9. Tentukan koordinat 10. Pilih dan simpan koordinat	Memilih titik koordinat	Nilai koordinat sesuai posisi	Data berupa nilai latitude dan longitude	Nilai koordinat berupa latitude dan longitude	diterima

#### 5.5.4 Melakukan Proses Presensi Menggunakan Aplikasi Presensi

Pada tabel 5-5 merupakan hasil uji proses *input* data presensi menggunakan aplikasi presensi yang telah dibangun. Pada uji *input* data ini mempunyai beberapa butir uji diantaranya : tanggal dan waktu pada halaman *input* data tidak dapat diubah *user*, *user* memilih salah satu pilihan pada *field* status karyawan dan *field* status kerja pada halaman *input* data, memasukan NIP sesuai dengan NIP karyawan yang sudah di-set oleh *admin*, *field* nama dan *field* status presensi akan otomatis terisi jika NIP *user valid*, memasukan tanda tangan digital dan pada *field* lokasi GPS Aplikasi presensi menyimpan histori lokasi *user* pada saat *input* data presensi.

**Tabel 5-5** Uji *input* data presensi

Kode Uji	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria evaluasi hasil	Keluaran yang didapat	Kesimpulan
KF-04 U1-01	Tanggal dan waktu pada halaman <i>input</i> data tidak dapat diubah <i>user</i>	11. Pilih menu presensi	Memilih titik koordinat	Tanggal dan waktu saat pengisian data presensi tidak dapat diubah <i>user</i>	Tanggal dan waktu tidak dapat diubah	Tanggal dan waktu selalu <i>update</i> dan tidak dapat diubah <i>user</i> pada saat <i>input</i> data presensi	Diterima
KF-04 U1-02	<i>User</i> memilih salah satu pilihan pada <i>field</i> status karyawan dan <i>field</i> status kerja pada halaman <i>input</i> data	1. Pilih <i>field</i> status karyawan 2. Pilih <i>field</i> status kerja	Pilihan status kerja dan pilihan status kerja	<i>User</i> dapat memilih jenis status karyawan dan status karyawan	<i>Field</i> dapat terpilih salah satu	Pada saat <i>input</i> data presensi <i>field</i> status kerja karyawan dan <i>field</i> status karyawan dapat dipilih salah satu jenisnya	Diterima

KF-04 U1-03	Memasukkan NIP sesuai dengan NIP karyawan yang sudah di-set oleh <i>admin</i>	1. Pilih <i>field</i> NIP 2. Masukan NIP	Text NIP	<i>Field input</i> NIP dapat menerima NIP jika sesuai dan menolak NIP jika tidak sesuai	Dapat memasukkan NIP	NIP diterima jika sesuai dan ditolak jika tidak sesuai dengan yang sudah di-set oleh <i>admin</i>	Diterima
KF-04 U1-04	<i>Field</i> nama dan <i>field</i> status presensi akan otomatis terisi jika NIP <i>user</i> valid	1. Status nama dan status presensi akan terisi otomatis	Hasil dari text NIP	Kedua <i>field</i> ini akan memberikan informasi nama karyawan sesuai NIP dan status presensinya	Nama dan status karyawan akan terisi sesuai NIP	Dapat menampilkan informasi sesuai NIP yang dimasukkan <i>user</i> dan memberikan informasi mengenai status presensi	Diterima
KF-04 U1-05	Memasukkan tanda tangan digital	1. Pilih <i>field</i> tanda tangan digital	Tanda tangan digital	<i>Field</i> tanda tangan mampu menyimpan masukan tanda tangan digital yang disimpan dalam format <i>.jpg</i>	Tanda tangan digital sesuai dengan tanda tangan <i>user</i>	File tanda tangan tersimpan kedalam <i>database</i> aplikasi dalam format <i>.jpg</i>	Diterima

KF-04 U1-06	Pada <i>field</i> lokasi GPS Aplikasi presensi menyimpan histori lokasi <i>user</i> pada saat <i>input</i> data presensi	1. Pilih koordinat GPS	Koordinat <i>user</i>	Aplikasi diharapkan memberikan informasi lokasi <i>user</i> pada saat <i>input</i> data presensi	Lokasi dapat didapatkan aplikasi sesuai koordinat <i>user</i>	Aplikasi presensi dapat menyimpan histori lokasi pada saat <i>user</i> melakukan <i>input</i> data presensi	Diterima
----------------	--	------------------------	-----------------------	--	---	---	----------

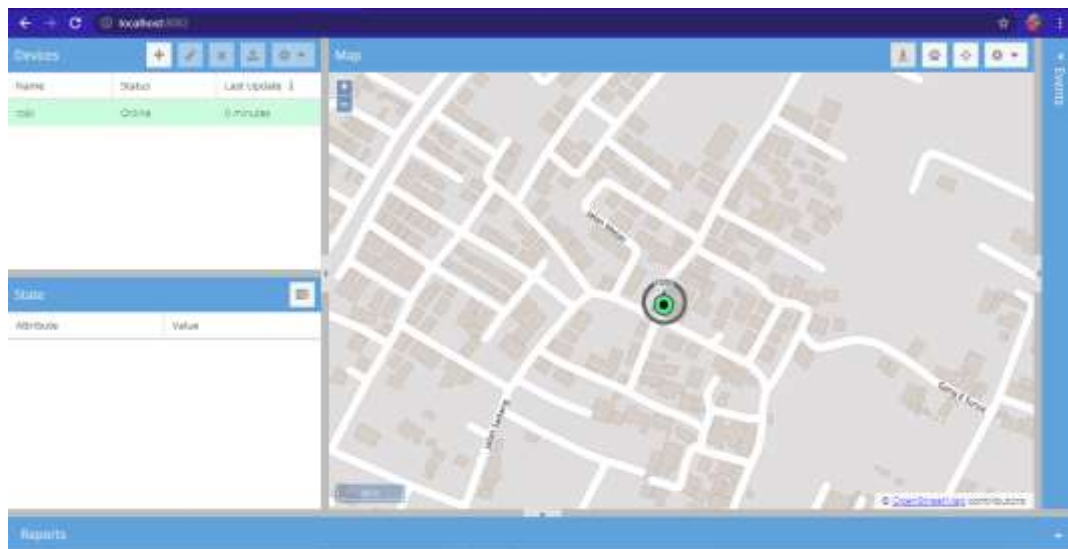
### 5.5.5 Menampilkan Data Lokasi Karyawan pada Webservice Traccar

Data lokasi karyawan tersimpan didalam *webservice online*. Pada tabel 5-6 merupakan hasil uji dari proses *tracking* terhadap salah satu *user* pada tahap menampilkan data lokasi karyawan pada *webservice traccar* dalam jangka atau interval waktu lima menit.

**Tabel 5-6** Hasil uji menampilkan informasi lokasi karyawan

Kode Uji	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria evaluasi hasil	Keluaran yang didapat	Kesimpulan
KF-05 U1-01	Menampilkan posisi <i>user</i> pada <i>server</i>	12. Buka <i>server traccar</i> 13. Login	Akun <i>admin</i>	Dapat melihat semua posisi karyawan dan melihat pergerakan yang dilakukan karyawan	Data koordinat sesuai dan valid	Data posisi dan pergerakan karyawan dapat dilihat oleh <i>admin</i> pada <i>server traccar</i>	Diterima

Gambar 5-16 merupakan tampilan tabel pada *interface web server* yang menunjukkan data lokasi karyawan yang sudah tersinkronisasi dengan *web server*. Pada halaman ini *admin* dapat melakukan manajemen data berupa menambah data, mengubah data dan menghapus karyawan dan membuat laporan mengenai kehadiran karyawan.



**Gambar 5-16** Tampilan data lokasi karyawan pada *web server*