

Terbit online pada laman web jurnal: <http://jurnal.iaii.or.id>

# JURNAL RESTI

(Rekayasa Sistem dan Teknologi Informasi)

Vol. 4 No. 4 (2020) 618 - 626

ISSN Media Elektronik: 2580-0760

## Implementasi Fitur Keamanan dengan *JSON Web Token* dan Fitur *Geo-tagging* pada Aplikasi Web Service *Training From Home*

Aal Hibsy<sup>1</sup>, Arief Wibowo<sup>2</sup>

<sup>1,2</sup>Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur  
<sup>1</sup>aalhibsy@gmail.com, <sup>2</sup>arief.wibowo@budiluhur.ac.id

### Abstract

*In the Covid-19 pandemic phase, some business processes were halted, adapted, and modified to deal with the policy of social restrictions. This impact is experienced by all levels of society, including athletes who are forced to do training from home (Training From Home). Performance evaluation of athletes who do exercises from home must be able to be evaluated remotely, including in terms of presence during the exercise training program. Presence is one of the benchmarks of a person's level of performance or activity in terms of accuracy and discipline in a program of activities. Attendance activities in the form of check-in must be ensured safe and accurate, especially if there is data connectivity with the webserver. This study aims to implement security features with JSON Web Token (JWT) based on the 256 Hash algorithm. The research also implements geo-tagging features to obtain accurate coordinates based on location points. Athlete attendance data obtained by the presence of these features are then synchronized via web service using the REST architecture. All stages of implementation are then tested by the Black Box method, and the results show that JSON Web Token (JWT) is able to secure the authentication and data security process, while the Geo-tagging feature is capable of sending accurate position data. Testing the functionality of the web service shows that all features work well within 44.8 ms, while the positioning accuracy of the geo-tagging feature reaches an accuracy of 90.9%.*

*Keywords: Presence, RESTful, Web Service, Geo-Tagging, JSON Web Token*

### Abstrak

Dalam fase pandemi Covid-19, beberapa proses bisnis menjadi terhenti, mengalami adaptasi dan modifikasi untuk menghadapi kebijakan pembatasan sosial. Dampak ini dialami oleh seluruh lapisan masyarakat termasuk bagi para atlet yang terpaksa harus melakukan latihan dari rumah (*Training From Home*). Penilaian kinerja atlet yang melakukan latihan dari rumah, harus dapat dievaluasi dari jarak jauh, termasuk dalam hal presensi selama melakukan program latihan olahraga. Presensi merupakan salah satu tolak ukur dari tingkat kinerja atau keaktifan seseorang dalam hal ketepatan dan kedisiplinan di suatu program kegiatan. Kegiatan presensi berupa *check-in* harus dipastikan aman dan akurat terutama bila terdapat konektivitas data dengan *web server*. Tujuan penelitian ini adalah mengimplementasi fitur keamanan dengan *JSON Web Token (JWT)* berbasis algoritma Hash 256. Penelitian juga melakukan implementasi fitur *geo-tagging* untuk mendapatkan titik lokasi berbasis koordinat yang akurat. Data presensi atlet yang didapatkan dengan adanya fitur-fitur tersebut kemudian disinkronisasi melalui *web service* menggunakan arsitektur REST. Seluruh tahap implementasi kemudian diuji dengan metode *Black Box*, dan hasil menunjukkan bahwa *JSON Web Token (JWT)* mampu mengamankan proses autentikasi dan pengamanan data, sementara fitur *Geo-tagging* mampu mengirimkan data posisi yang akurat. Pengujian fungsionalitas pada *web service* menunjukkan bahwa seluruh fitur bekerja dengan baik dalam waktu 44.8 ms, sementara ketepatan posisi dari fitur *geo-tagging* mencapai akurasi sebesar 90,9%.

Kata kunci: Presensi, *RESTful*, *Web Service*, *Geo-tagging*, *JSON Web Token*

### 1. Pendahuluan

Seiring dengan perkembangan jaman, di era modern ini teknologi digital dan internet menjadi kebutuhan yang sangat meningkat. Pemanfaatan teknologi digital dan internet diterapkan dalam berbagai aspek kehidupan. Demikian pula dalam dunia pendidikan, penggunaan

teknologi digital dapat mendorong pendidikan yang lebih berkualitas.

Salah satu indikator tingkat kedisiplinan seorang mahasiswa pada perguruan tinggi adalah tanda kehadiran atau presensi. Presensi masih merupakan salah satu aspek yang mempengaruhi penilaian dan juga menilai kinerja dari seorang mahasiswa. Presensi dapat

Diterima Redaksi : 20-05-2020 | Selesai Revisi : 09-08-2020 | Diterbitkan Online : 20-08-2020

menjadi salah satu tolak ukur dari tingkat kinerja atau keaktifan seseorang dalam hal ketepatan dan kedisiplinan waktu suatu kegiatan.

Bagi para atlet olahraga, baik yang berada di pemusatan latihan, di universitas atau di institusi lainnya, masa pandemi Covid-19 ini membuat mereka harus disiplin dalam menjalankan program latihan yang diberikan. Hal ini untuk menjamin agar para atlet tetap memiliki kesiapan fisik. Hal tersebut berlaku juga bagi para atlet-atlet skala nasional berusia muda yang saat ini melaksanakan kegiatan latihan di rumah masing-masing atau dikenal dengan istilah *Training From Home (TFH)*. Namun demikian, kegiatan latihan atlet relatif kurang maksimal dalam pengawasan oleh pelatih di masa pandemi Covid-19, terutama bagi para atlet yang tersebar di rumah masing-masing, karena di sektor perguruan tinggi menerapkan kebijakan pembelajaran daring [1].

Dalam situasi pandemi Covid-19, penerapan presensi kehadiran tentu menjadi masalah, karena adanya pembatasan sosial berskala besar, tidak memungkinkan untuk seorang mahasiswa melakukan presensi secara fisik, apalagi jika mahasiswa tersebut adalah seorang atlet yang wajib melakukan latihan olahraga secara rutin. Namun demikian, para mahasiswa atlet diminta melakukan presensi dari lokasi masing-masing, dengan pengawasan melalui aplikasi *Training From Home* yang digunakan bersama antara atlet dan pelatih serta pembina olahraga. Aplikasi ini bekerja dengan teknologi *web service* berbasis *Restful*, yang kompatibel dengan aplikasi mobile berbasis android.

Android merupakan sistem operasi untuk *platform* perangkat *mobile* yang populer. Sebanyak 80% pengguna *smartphone* di Indonesia menggunakan sistem operasi android [2]. Sebagai salah satu sistem operasi mobile yang sangat populer, Android menyediakan beragam fitur seperti *GPS* dan *API* yang dapat dimanfaatkan dalam pengembangan aplikasi [3].

*Web service* adalah sebuah perangkat lunak yang dirancang untuk mendukung interoperabilitas interaksi mesin ke mesin melalui sebuah jaringan [10]. *Web service* secara teknis memiliki mekanisme interaksi antar sistem sebagai penunjang interoperabilitas, baik berupa agregasi (pengumpulan) maupun sindikasi (penyatuan). Dalam penggunaan *Web Service*, terdapat beberapa arsitektur yang dapat digunakan diantaranya adalah *Extensible Markup Language Remote Procedure Call (XML-RPC)*, *Simple Object Access Protocol (SOAP)* dan *Representational State Transfer (REST)*. *XML-RPC* merupakan cara tradisional yang digunakan dalam pertukaran dan integrasi sistem. *Web Service* dengan arsitektur *SOAP* telah digunakan untuk mendukung proses integrasi dan *scalability* sistem. Arsitektur metode *REST* merupakan konfigurasi dengan nilai latency terbaik untuk diimplementasikan dalam proses integrasi data [4]. Dari beberapa arsitektur *web*

*service REST* memiliki *performance* yang lebih bagus dibandingkan dengan *SOAP* untuk pengujian *request* dan respon untuk *web service* [5].

Implementasi *web service* sudah banyak dilakukan pada beberapa sektor. Dalam penelitian terdahulu, digunakan *web service* dengan *REST API* yang dilengkapi *access token* pada sistem pengelola data masyarakat. Penelitian tersebut membahas tentang pembuatan sistem berbasis web yang dirancang untuk mengelola data masyarakat dengan fitur untuk monitoring akses ke dalam basis data di server, sehingga *Administrator* dapat dengan mudah memonitoring kondisi *traffic* data [6].

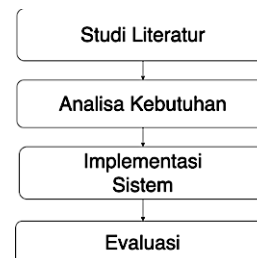
Platform *web service* yang digunakan dalam *monitoring* presensi harian kepegawaian telah dibuat pada studi lain [7], mampu menampilkan data presensi kepegawaian dalam format json menggunakan konsep *REST API*. Dengan menggunakan *REST API* maka dapat menjembatani berbagai perangkat akses informasi dari yang membutuhkan dan mempermudah perangkat lain untuk mengakses informasi.

Studi ini bertujuan untuk menerapkan fitur keamanan *Json Web Token (JWT)* pada *web service* pada mode autentikasi sistem. Penelitian ini juga mengembangkan fitur *geo-tagging* pada *web service* untuk mendukung proses presensi (*check-in*) dari berbagai lokasi. Dengan penerapan fitur *JWT* dan *geo-tagging* tersebut maka diharapkan dapat memecahkan masalah keamanan dan masalah akurasi informasi pada teknologi *web service* yang digunakan atlet selama melakukan *Training From Home*.

## 2. Metode Penelitian

Penelitian saat ini merupakan jenis penelitian eksperimental, yang mengimplementasi metode keamanan dengan *Json Web Token* dan peningkatan akurasi informasi dengan *geo-tagging* pada aplikasi *web service*.

Penelitian ini dibagi menjadi beberapa tahap. Tahap penelitian terlihat pada Gambar 1.



Gambar 1. Diagram Metode Penelitian

Tahap pertama adalah tahap studi literatur, berisi pencarian literatur yang dapat dijadikan sebagai landasan penelitian. Kedua adalah tahap analisis kebutuhan, pada tahap ini dianalisis kebutuhan berdasarkan rumusan masalah, yaitu bagaimana

penerapan fitur autentikasi sebagai solusi fitur keamanan dari *web service*, dan bagaimana fitur *geo-tagging* yang terintegrasi dengan API *Google Maps* agar mampu meningkatkan akurasi presensi mahasiswa atlet di lokasi masing-masing. Setelah fitur keamanan dan akurasi *geo-tagging* diimplementasi, maka tahap evaluasi bertujuan untuk mengetahui aspek validitas keamanan dan akurasi yang diinginkan, serta fungsionalitas sistem *web service* yang telah dikembangkan.

Pada penelitian lainnya penggunaan *REST* sangat rendah dalam segi keamanan. Mengamankan *RESTful Web Service* mencakup mengamankan data serta seluruh komunikasi untuk melindungi kerahasiaan dan integrasi data. Pada penelitian tersebut untuk mengatasi masalah keamanan, digunakan *JSON Web Token (JWT)* [8].

*REST* merupakan singkatan dari *Representational State Transfer*. Istilah *REST* atau *RESTful* pertama kali diperkenalkan oleh Roy Fielding pada disertasinya di tahun 2000. *REST* bukanlah sebuah standar protokol *web service*, melainkan hanya sebuah gaya arsitektur. Ide dasar dari arsitektur *REST* adalah bagaimana menghubungkan jalur komunikasi antar mesin atau aplikasi melalui *HTTP* sederhana [10]. Teknologi *web service* memerlukan sisi keamanan yang baik. Hal itu bertujuan untuk menjaga kerahasiaan data dari transaksi yang ada. Keamanan data tersebut sangat penting untuk mencegah pencurian data oleh pihak yang tidak bertanggung jawab, yang akan membuat hilangnya kepercayaan pelanggan dengan jasa yang diberikan. Untuk menjawab permasalahan yang terjadi, maka diperlukan proses otentikasi keamanan akses aplikasi *web service* yaitu *JSON Web Token*.

*JWT* merupakan sebuah *token* berbentuk *string* yang terdiri dari tiga bagian yaitu : *header*, *payload* dan *signature* yang digunakan untuk proses otentikasi dan pertukaran informasi. *Token* terdiri dari dua jenis : *token* pembawa dan *token* pemegang kunci. Sedangkan berdasarkan tujuan terdapat dua skema : *token* identitas dan *token* akses. Cara kerja *JWT* sama seperti *password*, ketika pengguna berhasil *login* maka server akan memberikan *token* yang disimpan di *local storage* atau *cookies browser*. *Token* digunakan untuk mengakses halaman tertentu, pengguna akan mengirim balik *token* tersebut sebagai bukti bahwa pengguna sudah berhasil *login*[4].

Penelitian terdahulu yang telah memanfaatkan fitur *geo-tagging* pada aplikasi presensi karyawan masih ditemukan batasan bahwa aplikasi tersebut hanya dapat digunakan ketika user telah terhubung pada jaringan lokal perusahaan [9].

*Geotag* atau *Geotaging* merupakan suatu proses dalam penambahan informasi posisi data dengan memanfaatkan *GPS* sehingga mendapatkan informasi *latitude* dan *longitude* dalam sebuah foto digital. Dengan memanfaatkan fitur *geo-tagging* dalam informasi sebuah foto maka letak pengambilan foto tersebut dapat

dengan mudah diketahui. Terdapat beberapa metode yang dapat dilakukan untuk melakukan *geo-tagging* pada suatu media, salah satunya adalah *Geo-coding* manual [11].

Pada dasarnya *Geo-coding* manual merupakan metode dimana informasi mengenai lokasi ditambahkan secara manual dengan menginputkan koordinat tertentu atau memilih lokasi pada saat melakukan *upload* suatu media ke dalam internet. Tingkat akurasi dari metode *geo-tagging* ini tergantung pada *tools* yang digunakan ataupun *GPS receiver* untuk mendapatkan koordinat yang akurat [11].

*Google Map API* merupakan sebuah alat atau penyedia layanan yang disediakan oleh sebuah teknologi yang bernama *Google* kepada para penggunanya agar dapat memanfaatkan *Google Map* disaat mengembangkan suatu aplikasi yang dibangun. *Google Map API* juga menyediakan berbagai fitur untuk mendapatkan data dan menambah kontak melalui berbagai jenis penyedia layanan yang dipunyai serta mengizinkan beberapa pengguna untuk membangun sistem *enterprise* di dalam *website* [12].

### 3. Hasil dan Pembahasan

Kebutuhan fungsional pada sistem Presensi mahasiswa atlet antara lain dapat melakukan *login* menggunakan email dan *password* yang sudah dibuat oleh *admin* agar presensi ini berjalan aman. Tahap selanjutnya, atlet membuat presensi yang disertai dengan pengambilan foto dan mengirim lokasi dengan fitur *geo-tagging*. Atlet juga dapat melihat daftar presensi yang telah selesai dan pelatih serta pembina adalah pihak yang berkepentingan menggunakan fitur keamanan dan akurasi untuk penilaian kinerja atlet.

Untuk mewujudkan kebutuhan fungsional tersebut maka diperlukan beberapa tahapan implementasi. Tahap pertama adalah implemetasi *JWT* pada *Web Service* *Laravel* disertai konfigurasi penambahan *JWT package*. Tahap kedua adalah implementasi fitur *geotagging* pada aplikasi *mobile* dengan *Flutter*. Pada tahap ini aspek keamanan akan dipenuhi dan diterapkan pada *web service*.

#### 3.1. Implementasi *RESTful Web Service JSON Web Token (JWT)*

Adapun beberapa tahapan yang dilakukan dalam penambahan *JWT package* pada *Laravel*, dimulai dari proses instalasi dengan *composer*.

```
Composer Command
composer require tymon/jwt-auth:dev-develop --prefer-source
```

Tahap kedua adalah mendeklarasikan *JWT Class* pada *provider* dalam *file config/app.php*.



Pada Tabel 1, merupakan token yang diberikan kepada setiap atlet yang melakukan proses *login*, dengan jangka waktu 24 jam.

Pada Gambar 5 dilakukan pengujian dengan metode *POST* pada *path* `/atlet/presensi/mystatus` dengan parameter tanggal dan *token* pada *header*. untuk melihat status presensi pada tanggal yang ditentukan. Apabila *token* valid maka akan menghasilkan respon *success* yang ditunjukkan pada gambar 5.

```

1  {
2  "status": 200,
3  "presensi_status": {
4    "nama": "Mahesa Pandu",
5    "ukm": "Basket Putraa",
6    "tanggal": "2020-05-19",
7    "materi": "Jogging",
8    "status": 1
9  },
10 "message": "Sudah Entry Presensi Hari Ini.",
11 "status_terakhir_latihan": 1,
12 "terakhir_latihan": {
13   "tanggal": "2020-05-19",
14   "materi": "Jogging",
15   "jam_in": "14:33:23",
16   "jam_out": "14:33:52"
17 },
18 "ukm": [
19   {
20     "ukm_id": 3,
21     "ukm_nama": "Basket Putraa"
22   }
23 ]
24 }

```

Gambar 4. Respon Cek Status

Respon yang dihasil pada Gambar 4 menghasilkan respon kode 200 dan memiliki ukuran 771 B dengan waktu respon 88 ms.

Pengujian selanjutnya dilakukan apabila token yang digunakan untuk request pada *web service* tidak valid maka akan menghasilkan respon seperti yang ditunjukkan pada Gambar 5. Pada Gambar 5 *web service* memberikan respon *error* : *"Unauthorized"* dengan respon kode 401, waktu respon 111 ms dan ukuran 310 B.

```

{
  "status": 401,
  "errors": [
    "Unauthorized."
  ]
}

```

Gambar 5. Respon Token Invalid

Pada Gambar 6 merupakan respon pengujian dari penerapan metode *POST* `/atlet/presensi/add` untuk menambah data presensi atlet. Dalam pengujian ini dibutuhkan parameter *body* yang terdiri dari *ukm\_id*, tanggal, denyut\_nadi, kondisi\_tubuh, jam, latitude, longitude, alamat dan foto. Respon yang dihasilkan pada Gambar 7 memiliki ukuran 342 B dengan waktu respon 36 ms.

```

{
  "status": 201,
  "success": true,
  "message": "Presensi In Sukses."
}

```

Gambar 6. Respon Metode POST Presensi

Gambar 7 menunjukkan respon dari penerapan metode *GET* `/atlet/presensi/mylist` untuk melihat daftar presensi yang telah dibuat. Pada Gambar 8 menghasilkan respon kode 200 dengan waktu 26 ms dan ukuran 1.02 KB.

```

{
  "id": 21,
  "ukm": "Basket Putera",
  "nama": "Mahesa Pandu",
  "materi": "Jogging",
  "tanggal": "2020-06-26",
  "jam_in": "00:06:02",
  "jam_out": "00:06:23",
  "lat_in": "-6.24299170",
  "long_in": "106.86527170",
  "lokasi_in": "Jl. Letjen M.T. Haryono, Rw.6, Kp. Melayu, Kecamatan Jatinegara, Kota Jakarta Timur, Daerah Khusus Ibukota Jakarta 13330, Indonesia",
  "lat_out": "-6.24299170",
  "long_out": "106.86527170",
  "lokasi_out": "Jl. Letjen M.T. Haryono, Rw.6, Kp. Melayu, Kecamatan Jatinegara, Kota Jakarta Timur, Daerah Khusus Ibukota Jakarta 13330, Indonesia",
  "foto_in": "https://api.aalhbsy.com/ta/latihan/9ec04036-5184-485b-8787-e3134cd385ee16011457453477608352.jpg",
  "foto_out": "https://api.aalhbsy.com/ta/latihan/84e79380-56cc-437d-9cde-7a41e3c43fb88983063279537893759.jpg",
  "status": 1
}

```

Gambar 7. Respon Metode GET Daftar Presensi

Pada Gambar 8 merupakan penerapan metode *GET* `/atlet/profile` untuk melihat profil atlet. Gambar 9 menghasilkan respon kode 200 dengan waktu respon 37 ms dan memiliki ukuran 430 B.

```

{
  "status": 200,
  "id": 1,
  "nama": "Mahesa Pandu",
  "nim": "1611501493",
  "telepon": "08221213131",
  "email": "akumahesaa@gmail.com",
  "ukm": [
    {
      "ukm_nama": "Basket Putera"
    }
  ]
}

```

Gambar 8. Respon Metode GET Profil

### 3.2. Implementasi fitur *Geo-tagging*

Pada tahap ini dilakukan implementasi fitur *check-in* presensi yang dilengkapi fitur *geo-tagging* berbasis *Google Maps* API. Pengembangan aplikasi Presensi *Training From Home* berbasis android dilakukan menggunakan Bahasa pemrograman *dart* dengan menggunakan *framework* *Flutter*. Selain itu juga digunakan *package geo-locator* yang berfungsi untuk mendapatkan titik koordinat saat aplikasi dijalankan dan *package geo-coder* untuk menentukan alamat pada titik koordinat yang didapat.

Untuk menggunakan *package geo-locator & geo-coder* tahap pertama adalah menginisialisasi *package geo-locator* dengan menambahkan *dependency* pada *file pubspec.yaml*. Langkah ini dilakukan agar aplikasi yang



dibangun dapat menggunakan *geo-locator* untuk mendapatkan titik koordinat lokasi saat ini.

#### Inisialisasi package

```
dependencies:
  geolocator: ^5.3.1
  geocoder: ^0.2.1
```

Dalam penggunaannya, untuk mendapatkan lokasi saat ini maka menggunakan *function geo-locator*. Pada *function geo-locator* akan menghasilkan variable *Position* berupa data *latitude* dan *longitude* seperti yang ditunjukkan pada Gambar 9.

#### Function geo-locator

```
import 'package:geolocator/geolocator.dart';

Position position =
  await Geolocator().getCurrentPosition
  (desiredAccuracy: LocationAccuracy.high);
```

Titik Koordinat Anda  
Longitude : 106.80256 Latitude : -6.2696667

Gambar 9. Hasil Function *Geo-locator*

Gambar 9 menunjukkan bahwa *function geo-locator* yang diimplementasikan telah berfungsi dengan baik, terlihat bahwa didapatkan posisi *latitude* dan *longitude* dari penerapan *function*. Setelah mendapatkan titik koordinat *latitude* dan *longitude* maka untuk mendapatkan alamat lengkap digunakan fungsi dari *geocoder* sehingga menghasilkan informasi seperti yang ditunjukkan pada Gambar 10.

#### Function geo-coder

```
import 'package:geocoder/geocoder.dart';

final coordinates =
  Coordinates(location.latitude, location.longitude);
var addresses =
  await Geocoder.local.findAddressesFromCoordinates(
    coordinates);

alamat =
  addresses.first.addressLine.toString();
```

Lokasi Anda  
Jl. Asem V No.9, RT.13/RW.5, Cipete Sel., Kec.  
Cilandak, Kota Jakarta Selatan, Daerah  
Khusus Ibukota Jakarta 12410, Indonesia

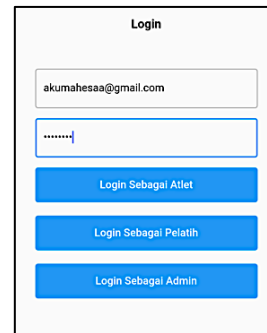
Gambar 10. Hasil Function *Geo-coder*

### 3.3. Sinkronisasi Fitur pada Aplikasi Android

Setelah fitur keamanan autentikasi dan akurasi diimplementasi pada Web Service, maka tahap selanjutnya dilakukan sinkronisasi fitur autentikasi yang membuktikan berjalannya fitur keamanan pada *web service* disertai fitur *geo-tagging* pada aplikasi *Training From Home*.

Modul pada fitur ini merupakan penerapan *web service* layanan *login* menggunakan metode *POST* dengan

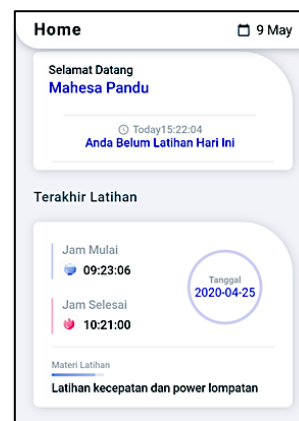
parameter *email* dan *password*. Untuk dapat mengakses aplikasi pengguna harus mengisi email dan *password* yang telah didaftarkan. Hasil penerapan *web service* ini terlihat pada Gambar 11.



Gambar 12. Implementasi Antarmuka *Login*

Terlihat pada Gambar 11, tampilan layar yang ditunjukkan merupakan halaman yang pertama kali tampil saat aplikasi dijalankan. Fitur autentikasi ini berjalan dengan baik dengan penerapan *JWT*.

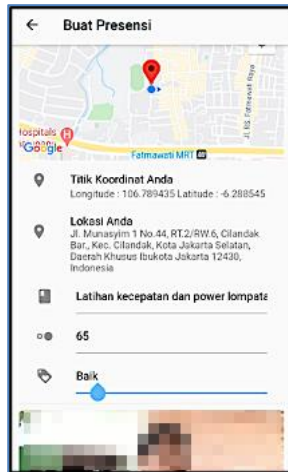
Fitur berikutnya adalah informasi status presensi latihan pada hari ini dan informasi tentang waktu terakhir kali atlet membuat presensi. Informasi yang ditampilkan terlihat seperti pada Gambar 12.



Gambar 13. Halaman Utama

Gambar 12 merupakan penerapan *web service* pada menu utama aplikasi dengan metode *POST* dengan parameter tanggal yang diambil otomatis saat aplikasi dijalankan. Mode ini berjalan dengan menerima respon berupa status presensi dan informasi kapan terakhir user membuat presensi.

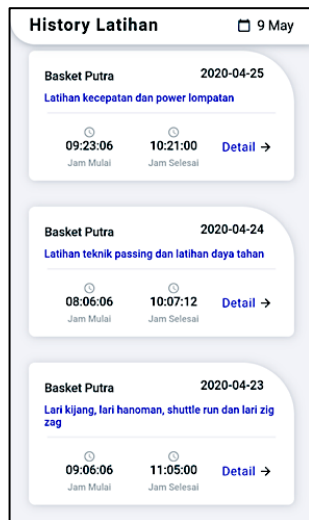
Fitur utama aplikasi yang dikembangkan adalah penandaan lokasi presensi atau *check-in*. Fitur ini merupakan implementasi fitur *geo-tagging* menggunakan *marker* yang secara otomatis menampilkan titik koordinat lokasi saat user membuat presensi. Tampilan selengkapnya terlihat pada Gambar 13.



Gambar 13. Halaman Buat Presensi

Terlihat pada Gambar 13 bahwa fitur ini berhasil menerapkan layanan *web service* metode *POST* dengan pengiriman data informasi berupa titik koordinat, lokasi, materi latihan, denyut nadi, kondisi tubuh dan foto, selanjutnya data tersebut disimpan dalam *database*.

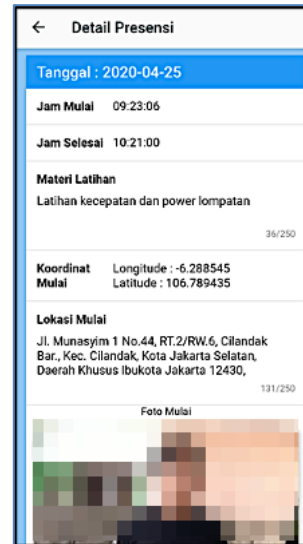
Fitur lainnya adalah informasi untuk menampilkan histori latihan sebagaimana terlihat pada Gambar 14.



Gambar 14. Halaman History Latihan

Gambar 14 menunjukkan adanya informasi daftar presensi latihan yang sudah selesai dilakukan sebelumnya. Halaman ini menerapkan layanan *web service* dengan metode *GET* sehingga mendapatkan respon data *JSON* yang berisi informasi presensi atlet secara lengkap.

Informasi yang dapat disampaikan juga adalah data detail presensi yang telah dibuat. Halaman detail presensi menampilkan informasi presensi yang telah dibuat secara lengkap meliputi tanggal, jam latihan, materi latihan, foto latihan dan alamat lokasi latihan, sebagaimana terlihat pada Gambar 15.



Gambar 15. Halaman Detail Presensi

Pada Gambar 15 terlihat implementasi *web service* dengan metode *GET* yang dilakukan pada halaman *History* Latihan. Proses selanjutnya, data tersebut dikirim pada halaman Detail Presensi berdasarkan data latihan yang dipilih oleh atlet.

### Pengujian Fitur dan Sistem

Berdasarkan hasil pengembangan aplikasi berbasis *web service* yang telah dilakukan, maka dilanjutkan proses pengujian fitur dan sistem yang telah disinkronisasi, menggunakan metode *BlackBox*. Tujuan tahap ini adalah untuk mengetahui bahwa aspek fungsionalitas aplikasi yang dibangun sudah sesuai dengan kebutuhan, yaitu aman dan akurat. Proses pengujian dilakukan dalam dua tahap, yaitu tahap pengujian keamanan dan tahap pengujian akurasi *geo-tagging*. Pengujian melibatkan sebelas atlet pada lokasi berbeda-beda. Hasil pengujian untuk fungsionalitas sistem dijelaskan pada Tabel 2.

Tabel 2. Hasil Pengujian Layanan *Web Service*

No	Layanan	Metode	Path	Waktu
1.	Login	POST	/atlet/login	37 ms
2.	Presensi Status	POST	/atlet/presensi/mystatus	88 ms
3.	Buat Presensi	POST	/atlet/presensi/add	36 ms
4.	Daftar Presensi	GET	/atlet/presensi/mylist	26 ms
5.	Profil	GET	/atlet/profile	37 ms
<b>Rata-rata</b>				44.8 ms

Tabel 3. Hasil Pengujian Fungsionalitas Aplikasi

No	Nama Fungsi	Status
1.	Aplikasi dapat berjalan pada sistem operasi android	Berhasil
2.	Aplikasi berhasil mengirim data login akun ke server	Berhasil

No	Nama Fungsi	Status
3.	Aplikasi dapat mengakses kamera dan berhasil mengambil foto	Berhasil
4.	Aplikasi berhasil mengambil data lokasi(latitude, longitude, alamat)	Berhasil
5.	Aplikasi berhasil menampilkan daftar data latihan sebelumnya	Berhasil
6.	Aplikasi berhasil menampilkan informasi detail presensi yang dipilih oleh pengguna	Berhasil
7.	Aplikasi berhasil menampilkan profil pengguna	Berhasil

Tahap pengujian terakhir adalah evaluasi fitur *geo-tagging* yang dikembangkan. Proses ini dilakukan dengan cara menguji ketepatan lokasi berbasis posisi *Longitude* dan *Latitude* yang dihasilkan pada aplikasi, dibandingkan dengan posisi yang sama dari aplikasi *Google Maps*.

Tabel 4. Evaluasi Fitur *Geo-tagging*

User	Input	Output	Status
mrizky*** ***@gmail.com	Longitude : 106.68*** Latitude : - 6.295***	Jl. Jombang Astek No *** Kec. Serpong, Tangsel	Berhasil dan akurat
angg**** @gmail.com	Longitude : 106.72*** Latitude : - 6.291***	Jl. Cirendeu Permai Ry No.**, Jakarta Selatan	Berhasil, titik lokasi kurang akurat
bitabi**** ***@gmail.com	Longitude : 106.78*** Latitude: - 6.28***	Gg. H. Nyandeng Cilandak, Jakarta Selatan	Berhasil dan akurat
descart*** ***@gmail.com	Longitude : 106.80*** Latitude : - 6.27***	Jl. Anggur 1 Blok *** Jakarta Selatan,	Berhasil dan akurat
egibuj**** *@gmail.com	Longitude : 106.70*** Latitude : - 6.22***	Jl. Perumahan Griya Kencana Blok *** Tangerang	Berhasil dan akurat
1611**** @student.* ***.ac.id	Longitude : 106.68*** Latitude : - 6.22***	Jl. KH Hasyim Ashari Blok *** Tangerang	Berhasil dan akurat
akuma*** **@gmail.com	Longitude : 106.69*** Latitude : - 6.24***	Jl. Maharta *** Tangerang Selatan	Berhasil dan akurat
mirvanfi** ***@gmail.com	Longitude : 106.80*** Latitude : - 6.26***	Jl. Saraswati No.***, Jakarta Selatan	Berhasil dan akurat
mjf**** @gmail.com	Longitude : 106.70*** Latitude : - 6.29***	Jl. Bintang Blok *** Tangerang Selatan	Berhasil dan akurat
rizkyde*** **@gmail.com	Longitude : 106.72*** Latitude : - 6.36***	Perum. Shaffa Residence, Jl. Kesadaran *** Kota Depok	Berhasil dan akurat

User	Input	Output	Status
prayud*** **@gmail.com	Longitude : 106.76*** Latitude : - 6.31***	Jl. Bea Cukai No.***, Kota Tangerang Selatan	Berhasil dan akurat

Terlihat pada Tabel 4, Berdasarkan 11 lokasi yang dijadikan posisi *check-in* atau presensi, 10 lokasi memiliki kesamaan posisi dengan titik *Longitude* dan *Latitude* yang dihasilkan pada aplikasi *Google Maps*. Satu posisi yang tidak sukses dianalisis dan ditemukan penyimpangan posisi bergeser sekitar 50 meter dari titik *check-in* presensi yang semestinya, diduga ketidakakuratan ini dikarenakan gangguan pada sinyal perangkat *mobile* yang digunakan. Dengan demikian fitur *geo-tagging* pada aplikasi yang dikembangkan memiliki akurasi sebesar  $(10/11) \times 100\%$  atau sama dengan 90,9%.

#### 4. Kesimpulan

Berdasarkan seluruh hasil implementasi *web service* berbasis android dengan keamanan berbasis JWT dan *geo-tagging* yang telah dilakukan, maka dapat diambil kesimpulan penerapan *JSON Web Token* pada *web service* mampu menghasilkan *token* sebagai bentuk autentikasi pengguna yang berhak mengakses *web service* pada aplikasi Presensi Atlet *Training From Home*. Hal ini menunjukkan bahwa aspek keamanan hak akses pada proses pertukaran data telah terjamin dan berhasil diimplementasikan dengan baik, setiap atlet yang *login* akan mendapatkan *token* masing-masing yang memiliki batas waktu selama 24 jam. Apabila email atau *password* salah maka akses *token* tidak akan diberikan, dan apabila *token* salah maka atlet tidak dapat autentikasi untuk melakukan aktifitas.

Berdasarkan proses evaluasi fitur *geo-tagging*, diketahui bahwa ketepatan posisi dari fitur *geo-tagging* yang diuji dengan metode kesesuaian dengan *Google-maps* mencapai akurasi sebesar 90,9%. Dengan demikian aplikasi *web service* berbasis *android* yang dibangun ini mampu memberikan akurasi posisi presensi yang tepat bagi mahasiswa atlet olahraga yang melakukan program kegiatan *Training From Home*.

#### Daftar Rujukan

- [1] R. Fabriza, "Dampak Virus COVID-19 Bagi Mahasiswa," *Unisbank.Ac.Id*, hal. 1–6, 2020.
- [2] B. T. Wibowo dan S. N. Anwar, "Rancangan Hazard Reporting System Dengan Metode Geolocation Tagging Pada Platform Android Di Pt. Angkasa Pura I (Persero)," hal. 978–979, 2018.
- [3] I. Salamah, L. Lindawati, dan H. F. Handayani, "Rancang Bangun Aplikasi Informasi Perkuliahan Berbasis Android," *Jurasik (Jurnal Riset Sistem Informasi dan Teknik Informatika)*, vol. 4, no. 1, hal. 1, 2019.
- [4] R. Gunawan dan A. Rahmatulloh, "JSON Web Token (JWT) untuk Authentication pada Interoperabilitas Arsitektur berbasis RESTful Web Service," *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, vol. 5, no. 1, hal. 74, 2019.



- [5] M. G. L. Putra dan M. I. A. Putera, “Analisis Perbandingan Metode Soap Dan Rest Yang Digunakan Pada Framework Flask Untuk Membangun Web Service,” *SCAN - Jurnal Teknologi Informasi dan Komunikasi*, vol. 14, no. 2, hal. 1–7, 2019.
- [6] M. I. Perkasa dan E. B. Setiawan, “Pembangunan Web Service Data Masyarakat Menggunakan REST API dengan Access Token,” *Jurnal ULTIMA Computing*, vol. 10, no. 1, hal. 19–26, 2018.
- [7] H. Sy dan Rismayani, “Monitoring Absensi Harian Kepegawaian Pada Instansi Pemerintahan Kota Makassar Berbasis,” *Seminar Nasional Informatika*, hal. 236–239, 2015.
- [8] A. Rahmatulloh, H. Sulastri, dan R. Nugroho, “Keamanan RESTful Web Service Menggunakan JSON Web Token (JWT) HMAC SHA-512,” *Jurnal Nasional Teknik Elektro dan Teknologi Informasi (JNTETI)*, vol. 7, no. 2, 2018.
- [9] A. Ari dan G. Sepriansyah, “Implementasi Geotagging Pada Aplikasi Absensi Karyawan Berbasis Android Webservice ( Studi Kasus : PT . Pupuk Sriwidjaja Palembang ),” vol. 9, no. 1, hal. 841–852, 2019.
- [10] G. Tendra dan D. Wulandari, “Implementasi Representational State Transfer Dan Geotagging Pada Aplikasi Pelaporan Kecelakaan Lalu Lintas,” vol. 4, no. 1, 2020.
- [11] U. Defitria, B. Priyambadha, dan D. S. Rusdianto, “Pembangunan Aplikasi Social Geotagging Destinasi Wisata Berbasis Android,” *Journal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 12, hal. 6610–6617, 2018.
- [12] A. B. Kamaludin dan D. Dharmayanti, “Aplikasi Monitoring Kurir Antar Jemput Hewan,” 2019.
-