

Implementasi Analisis Sistem *Online* Kawasan Keselamatan Operasional Penerbangan Menggunakan *Web Service* Berbasis *NodeJS* dan *Android*

Iwan Abadi^{1*}, Arief Ginanjar²

^{1,2}Program Studi Teknik Informatika, Universitas Langlangbuana, Bandung, Indonesia

Email: ¹ iwan.abadi69@gmail.com, ² arief.ginanjar@unla.ac.id

INFORMASI ARTIKEL

Histori artikel:

Naskah masuk, 11 Juli 2020

Direvisi, 29 Maret 2021

Diiterima, 23 September 2019

Kata Kunci:

KKOP

Kawasan Keselamatan Operasi Penerbangan

OLS

Obstacle Limitation Surfaces

Android Platform

NodeJS,

Web Service

ABSTRAK

Abstract- The Obstacle Limitation Surfaces (OLS) functions as a control for the height of buildings that can be built around the airport area for the community around the airport in order to realize management that meets the aspects of safety, security and service. With technological conditions that are always evolving, so the need of every level society can easily access information about correct OLS data from trusted sources real time. The combination of quantitative research methodologies and evolutionary prototyping becomes a reference for process of research, development and implementation of the OLS-Online system. The tools used to support the development process include the use of a web service to connect two or more different platforms, namely NodeJS, with this framework's speed performance in the development process and data access load speed as the reason. Then with the system-forming components designed in the form of a system architecture and database architecture, the existing KML data combined with the database architecture to display OLS data online. Then the data can be seen in an android which can be accessed online using an android phone version eight or above. The OLS-Online application is tested in terms of data accuracy and rapid server access strength.

Abstrak- Kawasan keselamatan operasional penerbangan (KKOP) berfungsi sebagai pengendali tinggi bangunan yang boleh dibangun di sekitar kawasan bandar udara bagi masyarakat sekitar bandar udara demi terwujud pengelolaan yang memenuhi aspek *safety*, *security* and *service*. Dengan kondisi teknologi yang selalu berkembang, sehingga dibutuhkan setiap lapisan masyarakat dapat dengan mudah mengakses informasi tentang KKOP yang benar dari sumber yang terpercaya secara *realtime*. Penggabungan metodologi penelitian kuantitatif serta *evolutionary prototyping* menjadi acuan proses penelitian, pembangunan serta implementasi sistem KKOP-Online ini. Adapun alat-alat yang digunakan sebagai penunjang proses pengembangan diantaranya dengan penggunaan *web service* untuk menghubungkan dua atau lebih *platform* berbeda yaitu *NodeJS* dengan alasan mengingat kinerja kecepatan *framework* ini dalam proses pengembangan serta kecepatan *load data access*. Kemudian dengan komponen pembentuk sistem yang dirancang berupa arsitektur sistem dan arsitektur basisdata, maka data KML yang ada digabungkan dengan arsitektur basisdata untuk menampilkan data KKOP secara online. Kemudian data-data tersebut dapat dilihat dalam aplikasi android yang dapat diakses secara online menggunakan handphone android versi delapan atau diatasnya. Aplikasi android KKOP-Online ini diuji dari sisi ketepatan data dan kekuatan akses server secara rapid.

Copyright © 2019 LPPM - STMIK IKMI Cirebon
This is an open access article under the CC-BY license

Penulis Korespondensi:

Iwan Abadi

Program Studi Teknik Informatika,
Universitas Langlangbuana
Jl. Karapitan No.114, Bandung, Indonesia
Email: iwan.abadi69@gmail.com

1. Pendahuluan

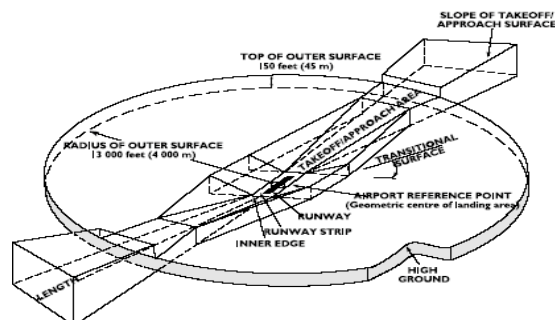
1.1 Latar Belakang

Kawasan Keselamatan Operasi Penerbangan (KKOP) adalah wilayah daratan dan/atau perairan dan ruang udara di sekitar bandar udara yang digunakan untuk kegiatan operasional penerbangan dalam rangka menjamin keselamatan penerbangan.

Tujuan penyusunan KKOP adalah untuk mengendalikan ketinggian bangunan di sekitar bandar udara demi terwujudnya pengelolaan bandar udara yang memenuhi aspek *safety, security, and services*, baik internal maupun eksternal.

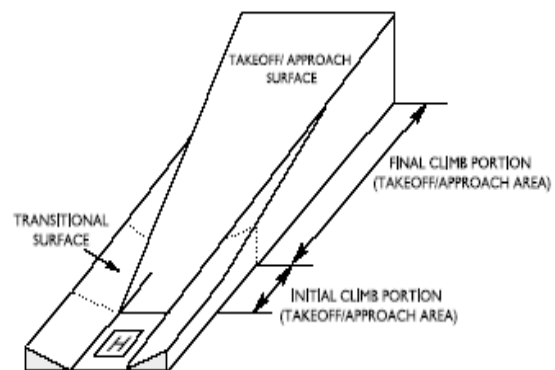
Selain itu juga dengan adanya *review* KKOP juga dapat mengendalikan pertumbuhan *obstacle* di sekitar bandar udara sehingga bandar udara dapat terus beroperasi dengan keselamatan operasional penerbangan tetap dapat terjaga. Hal tersebut dapat tercapai dengan membuat batasan permukaan *obstacle (obstacle limitation surfaces)* yang merupakan batas proyeksi suatu objek di wilayah udara.

Obstacle secara garis besar digolongkan menjadi dua jenis yaitu alami dan buatan. *Obstacle* alami merupakan *obstacle* yang ada dengan sendirinya tanpa campur tangan manusia. Ada dua jenis *obstacle* alami yaitu yang dapat tumbuh (misal: pepohonan) dan yang tidak dapat tumbuh (seperti: bukit). Sedangkan *obstacle* buatan merupakan *obstacle* yang timbul akibat campur tangan atau rekayasa manusia. Beberapa contoh *obstacle* buatan ialah gedung perkantoran, supermarket, hotel, apartemen, menara pemancar, dan lain-lain.



Gambar 1..KKOP (*Obstacle Limitation Surfaces*)

Dalam beberapa kasus terdapat suatu bandar udara pada daerah tertentu dimana *obstacle* yang ada tidak dapat dihilangkan. Untuk itu maka perlu dibuat suatu peraturan khusus mengenai prosedur operasional pesawat terbang di daerah sekitar *obstacle*. KKOP secara garis besar dapat digambarkan seperti yang terlihat pada Gambar 1 dan Gambar 2.



Gambar 2. *Take-Off dan Landing Approach Surface.*

Dari penelitian yang telah dilakukan sebelumnya yang berfokus terhadap pendekatan *Aerodrome Elevation System* dalam perhitungan pengaruh titik lokasi bangunan terhadap kawasan keselamatan operasional penerbangan [10]. Sedangkan penelitian ini dilakukan lebih fokus terhadap implementasi terhadap aplikasi *android*. Penelitian ini hanya berkonsentrasi terhadap bagaimana data kawasan KKOP dapat diimplementasikan menjadi sebuah sistem *online* yang dapat diakses melalui *internet* dan *android*, serta penulis hanya akan fokus dengan data bandar udara Husein Sastranegara Bandung. adapun spesifikasi teknis sebagai berikut:

- 1) Menggunakan sistem basis data *MariaDB* versi 10.1.19 yang diletakkan terpisah dalam Sistem Operasi *Microsoft Windows 7 Genuine 64 bit* didalam *Oracle VM Virtual Box* versi 5.2.6.
- 2) Menggunakan koneksi jaringan *local host - guest virtualbox connection*.
- 3) Menggunakan Sistem Operasi *Microsoft Windows 7 Genuine 64 bit* untuk *host environment*.

- 4) Menggunakan *Netbean 8.2* sebagai *Integrated Development Editor*.
- 5) Menggunakan *NodeJS Framework* versi 12.13.1 sebagai *Logic Backend base programming*.
- 6) Menggunakan *Android Studio* versi 3.5.1 sebagai *android development tools*.
- 7) Menggunakan *Android* versi 8.0 (Oreo) sebagai aplikasi *android*.
- 8) Menggunakan *application container Apache* versi 2.4.39 dalam *XAMP* versi 7.3.6 sebagai *server deployment application web server*.
- 9) Menggunakan *pm2 server* versi 4.1.0 sebagai *deployment nodejs application server*.
- 10) Menggunakan *file KML* yang berisi informasi geospasial yang akan melakukan *render image* diatas lapisan *google map* dalam aplikasi *android* yang dihasilkan dari aplikasi *google earth*.

1.2 Tujuan Penelitian

Adapun tujuan penelitian yang ingin dicapai dari penelitian ini adalah mencari cara bagaimana sebuah peraturan pemerintah tentang KKOP dapat diterapkan menjadi sebuah sistem online, sehingga dapat memudahkan masyarakat sekitar bandar udara atau masyarakat kalangan tertentu yang mempunyai kepentingan terhadap lingkungan bandar udara untuk mengetahui peraturan tentang batas ketinggian bangunan yang dapat dibangun disekitar bandara udara secara *online* dengan media *internet* dan *android*.

2. Metode Penulisan

2.1 Pelaksanaan Penelitian

Metodologi yang digunakan pada pembuatan laporan penelitian ini adalah dengan menggunakan metode penelitian kuantitatif dan *evolutionary prototype* serta pendekatan sistem menekankan terhadap elemen dan komponen [1],[2], adapun urutan proses dilakukan sebagai berikut:

- 1) Studi Literatur, Mempelajari sumber-sumber pustaka yang dapat dijadikan referensi. Sumber-sumber pustaka dapat berupa buku, *paper* atau halaman *web* yang membahas tentang *Android Platform*, *Web Service*, *Database MySQL* dan *NodeJS*.
- 2) Analisis, Menjelaskan bagaimana melakukan analisis terhadap arsitektur dan teknik pemrograman terhadap masing-masing teknologi serta bagaimana teknik mengkombinasikan masing-masing teknologi tersebut.
- 3) Perancangan Perangkat Lunak, Melakukan perancangan terhadap perangkat lunak dikombinasikan diantara teknologi yang akan dibangun berdasarkan hasil yang diperoleh dari analisis. Perancangan tersebut harus

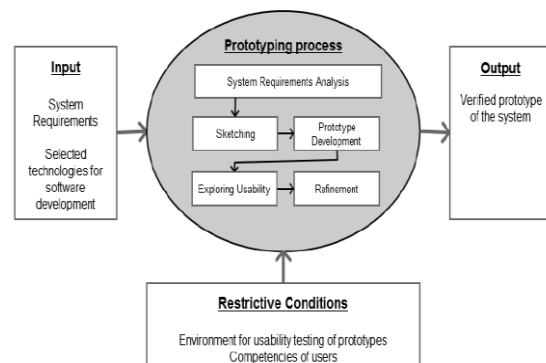
memenuhi *scenario* pengujian yang akan dilakukan.

- 4) Implementasi Perangkat Lunak, Melakukan implementasi terhadap perangkat lunak yang akan dikembangkan berdasarkan hasil yang diperoleh dari perancangan. Implementasi ini akan menghasilkan produk perangkat lunak yang berisi kombinasi teknologi seperti yang telah ditentukan dalam spesimen yang akan diuji.
- 5) Pengujian dan Evaluasi, Melakukan pengujian terhadap produk perangkat lunak yang telah dibangun dan kemudian melakukan evaluasi kinerja dari setiap *scenario* pengujian yang dilakukan.

Iteratif Tahapan Perancangan dan Implementasi Perangkat Lunak serta Pengujian dan Evaluasi, Melakukan proses pengujian terhadap spesimen perangkat lunak yang akan diteliti ketika dalam kondisi secara proses bisnis belum memenuhi keinginan peneliti maka lakukan perancangan dan implementasi ulang terhadap perangkat lunak spesimen..

2.2 Evolutionary Prototyping Model

Dengan tahapan proses yang telah dijelaskan dalam bagian pelaksanaan penelitian yang merujuk kepada tahapan *evolutionary prototyping model* maka proses tersebut dapat diilustrasikan seperti yang terlihat pada Gambar 3. Proses *evolutionary prototyping* terdiri dari empat proses utama yaitu *input*, *prototyping process*, dan *output* serta *restrictive condition*, tetapi selama pelaksanaan proses juga harus dibatasi dengan syarat; setiap fungsi yang dibangun harus memenuhi syarat yang sesuai seperti yang telah ditetapkan dalam *system requirement*, dengan kompetensi orang yang terlibat dalam *prototyping* memenuhi syarat minimum *system requirement* [3].



Gambar 3. Metodologi *Evolutionary Prototyping*[3].

Berikut merupakan tahapan yang dilakukan sesuai dengan metode yang digunakan:

- 1) Input dalam proses ini berupa *System Requirement*, berupa kebutuhan sistem yang

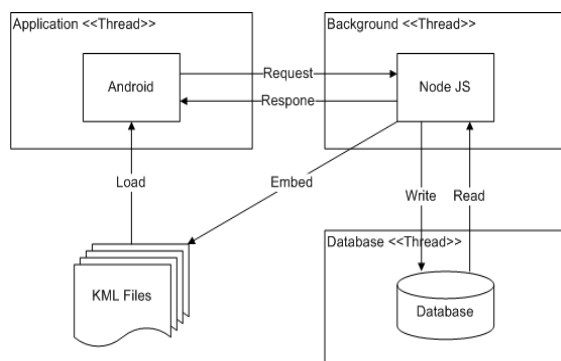
diperlukan, tujuan perangkat lunak dan gambaran komponen untuk mengembangkan perangkat lunak[4].

- 2) *Prototyping Process*, merancang perangkat lunak yang terbagi dalam lima bagian[4] sebagai berikut:
 - a) *System Requirement Analysis*, menganalisis terhadap kebutuhan sistem yang telah dibuat dalam tahapan sebelumnya.
 - b) *Sketching*, menggambarkan sistem sesuai kebutuhan sistem yang telah ditentukan.
 - c) *Prototype Development*, melakukan perancangan perangkat lunak.
 - d) *Exploring Usability*, menguji perangkat lunak yang telah dibangun.
- 3) *Refinement*, mengevaluasi dan memperbaiki kinerja dari setiap skenario pengujian yang dilakukan. *Restrictive Conditions*, menentukan batasan perangkat lunak berupa syarat yang diberlakukan terhadap semua fungsi yang terdapat didalam sistem[4].
- 4) *Output*, dalam tahapan ini dilakukan verifikasi *prototype* terhadap kebutuhan sistem yang telah ditentukan[4].

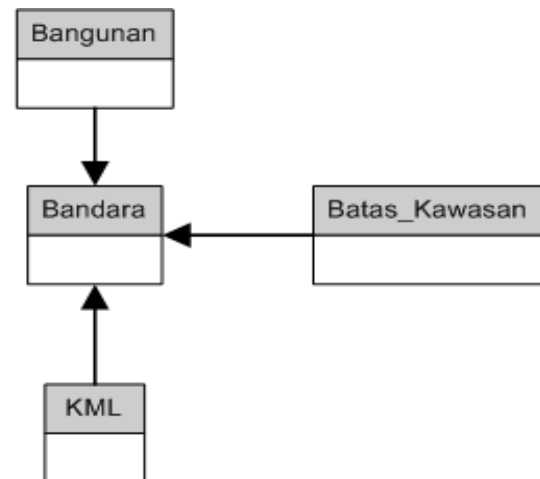
2.3 Arsitektur Spesimen

Dengan pendekatan prinsip dari *Object Oriented Programming* (OOP) serta pendekatan *Model, View dan Controller* (MVC), dimana abstraksi memungkinkan *programmer* untuk mengembangkan pemikiran yang rumit tanpa harus fokus terhadap komponen yang detail [5],[6]. Sehingga seluruh proses analisis dan pemrograman dalam proses pengembangan sistem ini dapat menggunakan konsep *Object Oriented Analysis and Design* (OOAD), OOP serta MVC.

Dengan menggabungkan arsitektur dari dua bahasa pemrograman *Android* dan *NodeJS* hingga menjadi sebuah sistem yang saling melengkapi, diharapkan dengan pengujian yang dilakukan dapat ditemukan titik optimasi yang perlu dilakukan dalam sistem *online* tersebut [7],[8],[9].



Gambar 4. Arsitektur Sistem *Online* KKOP.



Gambar 5. Arsitektur Basisdata Sistem *Online* KKOP.

```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2" xmlns:gx="http://www.google.com/kml/ext/2.2" xmlns:atom="http://www.opengis.net/kml/2.2" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:xi="http://www.w3.org/2001/XMLSchema-instance">
<Document id="000000" xmlns:atom="http://www.opengis.net/kml/2.2" http://schemas.opengis.net/kml/2.2.0/ogkml22.xsd
http://www.google.com/kml/ext/2.2 http://code.google.com/apis/kml/schema/kml22gx.xsd">
<name>BDO AR</name>
<snippet>
<style id="PolyStyle0">
<labelStyle>
<color>00000000</color>
<scale>0</scale>
<labelStyle>
<color>40404040</color>
<width>0.1</width>
</labelStyle>
<LineStyle>
<color>40000040</color>
</LineStyle>
</style>
<style id="PolyStyle00">
<labelStyle>
<color>00000000</color>
<scale>0</scale>
<labelStyle>
<color>40404040</color>
<width>0.1</width>
</labelStyle>
<LineStyle>
<color>40000040</color>
</LineStyle>
</style>
</Document>
```

Gambar 6. Tampilan *source code file* KML pembentuk tampilan bentuk KKOP diatas tampilan *google map*.

3. Proses Penelitian

3.1 Konfigurasi Sistem

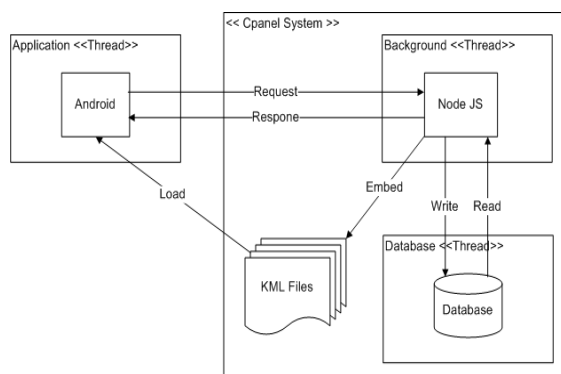
Didalam pengujian sistem yang telah dilakukan terdapat aspek teknis yang sudah diterapkan yaitu; pengujian yang dilakukan melibatkan beberapa lapisan yaitu, lapisan *database* menggunakan *MariaDB*, lapisan *application container background* dalam hal ini penulis menggunakan *NodeJS*, kemudian lapisan *application logic programming* dalam hal ini penulis menggunakan *android*. Konfigurasi *logic* tersebut dapat terlihat pada Gambar 4 dengan implementasi konfigurasi seperti yang terlihat pada Tabel 1.

Tabel 1. Implementasi Konfigurasi Di Setiap Lapisan

No	Lapisan	Teknologi	Konfigurasi
1	Database	MariaDB	Default
2	Background Application Container	PM2	Default

No	Lapisan	Teknologi	Konfigurasi
3	Application Logic Programming	Android	Disesuaikan dengan device android terbaru

Kemudian pengujian sistem online yang disimpan dalam server hosting menggunakan PM2 *NodeJS Container* serta ditambah konfigurasi tambahan berupa nama domain serta redirect port number ke *subdomain* tertentu menggunakan *cpanel*, selanjutnya aplikasi dijalankan langsung menggunakan aplikasi *android* dengan ilustrasi alur proses dapat terlihat pada Gambar 7.



Gambar 7. Arsitektur alur pengujian sistem online KKOP menggunakan *NodeJS* dan *Android*.

Sistem online KKOP menggambarkan wilayah tertentu dikawasan bandar udara yang terkena peraturan batas ketinggian bangunan yang berbeda-beda disetiap kawasan tertentu, untuk hal tersebut maka penulis membutuhkan koordinat tengah seluruh bandar udara yang termasuk dalam sistem online, adapun data koordinat tersebut dapat dilihat pada Tabel 2.

Tabel 2. Koordinat Tengah Setiap Bandara KKOP.

No	Kode	Latitude Tengah	Longitude Tengah
1	CGK	-6.13021	106.6539
2	KNO	3.633825	98.87511
3	PGK	-2.15719	106.1389
4	BDO	-6.90384	107.5802
5	PNK	-0.14673	109.4043
6	DJB	-1.6358	103.6431
7	PLM	-2.89667	104.6992
8	PKU	0.464462	101.4482
9	BTJ	5.517966	95.41731
10	HLP	-6.26445	106.8959

Data dari Tabel 2 digunakan sebagai titik acuan untuk *loading file* KML menjadi sebuah gambar KKOP yang berlapis diatas google map. Kemudian data selanjutnya adalah data batas kawasan, dimana setiap bandar udara yang terdaftar sebagai bagian dari sistem online KKOP akan membutuhkan data batas kawasan ini sebagai data penunjang dari *file* KML, dimana seluruh variabel perhitungan analisis batas ketinggian bangunan menurut pembagian area KKOP terdapat didalam tabel tersebut, adapun nama kolom dari tabel tersebut dapat dilihat pada Tabel 3.

Tabel 3. Nama Kolom Data Batas Kawasan.

No	Nama Kolom	Fungsi
1	id_bandara	Identitas Bandar Udara
2	kode	Identitas Kawasan KKOP per Bandara
3	uraian	Menerangkan Kawasan KKOP
4	kemiringan	Derajat kemiringan sebuah kawasan KKOP
5	lan_pacu	Identitas Landas Pacu (jika dalam satu bandar udara mempunyai lebih dari satu landas pacu)
6	panjang1	Panjang ujung landas pacu terhadap titik tengah landas pacu
7	panjang2	Panjang ujung landas pacu terhadap titik tengah landas pacu
8	elevasi1	Ketinggian batas kawasan terhadap ujung landas pacu
9	elevasi2	Ketinggian batas kawasan terhadap ujung landas pacu
10	ref_msl	Ketinggian batas kawasan menurut koordinat
11	elv_msl1	Ketinggian batas kawasan menurut survey
12	elv_msl2	Ketinggian batas kawasan menurut survey

Berikut disampaikan juga contoh data kawasan yang ada dari data asli yang besar dengan jumlah 12 kolom dan 623 baris, penulis akan mengambil beberapa kolom dan baris saja sebagai perwakilan data dari batas kawasan sebagaimana terlihat pada Tabel 4.

Tabel 4. Data Sampel Batas Kawasan KKOP.

No	Kode	Kemiringan	Landas Pacu	Elevasi 1
1	APLL07L101	0	07L1	46

No	Kode	Kemiringan	Landas Pacu	Elevasi 1
2	APLL07L102	0	07L1	46
3	APLL07L103	0.05	07L1	46
4	APLL07L104	0.05	07L1	46
5	APLL07L105	0.025	07L1	131
6	APLL07L106	0	07L1	151
7	APLL07L201	0	07L2	46
8	APLL07L202	0	07L2	46
9	APLL07L203	0	07L2	46
10	APLL07L204	0.05	07L2	46
11	APLL07L205	0.05	07L2	46
12	APLL07L206	0.05	07L2	46

Terdapat rumus yang akan digunakan untuk melakukan perhitungan batas ketinggian tergantung dari posisi koordinat terhadap batas kawasan dari sebuah KKOP kawasan tersebut secara umum adalah; permukaan utama, permukaan transisi, ancangan pendaratan dan lepas landas, kemungkinan bahaya kecelakaan, horizontal dalam, permukaan kerucut dan horizontal luar [10]. Dengan area batas kawasan tersebut penulis mencoba pendekatan menggunakan salah satu rumus, jika kondisi titik koordinat tegak terhadap ujung permukaan utama maka dapat menggunakan pendekatan konstanta sebagai berikut:

$$\begin{aligned}
 A &= 0.0516229999999998 \text{ (Konstanta * X)} \\
 B &= 0.05136799999999965 \text{ (Konstanta * Y)} \\
 C &= 4.91712148341899 \text{ (Konstanta - XY)} \\
 P &= 0.07282585773347510 \text{ (Konstanta / XY)} \\
 Ex &= 111194.926644559 \text{ m (Konstanta Jarak)}
 \end{aligned}$$

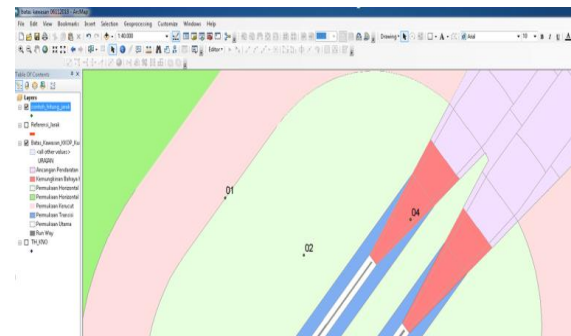
Kemudian dari konstanta tersebut penulis mencoba mengilustrasikan rumus (1):

$$D = \frac{A \cdot X - B \cdot Y - C}{P} * Ex \quad (1)$$

Sehingga dari rumus nomor (1) kita dapat mencoba mencari contoh kasus dengan menggunakan aplikasi ArcMap maka dapat kita buat simulasi titik koordinat seperti terlihat pada Gambar 8.

Dari Gambar 8 dapat diketahui titik ke 1 yang berada diantara perbatasan area berwarna biru muda yaitu kawasan horizontal dalam dan area berwarna pink yaitu kawasan kerucut dengan titik koordinat Degree Minute Second (DMS) yaitu 3° 39' 3.919" N

dan 98° 51' 52.179" E atau jika dikonversikan ke satuan desimal akan terlihat pada rumus (2) dan (3).



Gambar 8. Membuat simulasi titik koordinat dalam lingkungan bandar udara menggunakan ArcMAP.

Jika dikonversikan ke satuan desimal menjadi 3.651089 latitude dan 98.864494 longitude seperti yang terlihat pada rumus (2) dengan hasil perhitungan yang terlihat pada rumus (3).

$$D = \frac{A \cdot 3.651089 - B \cdot 98.864494 - C}{P} * Ex \quad (2)$$

$$D = 4769 \text{ m} \quad (3)$$

Kemudian diketahui juga bahwa titik berada di kawasan permukaan kerucut dengan sudut kemiringan 5% dan jarak antara titik dengan tepi permukaan horizontal dalam sejauh 4000 meter dengan batas ketinggian di koordinat titik tersebut adalah 50.271 meter maka dapat kita hitung menggunakan rumus (4) dengan hasil seperti yang terlihat pada rumus (5).

$$BT = 0.05 \cdot (4769 - 4000) + 50.271 \quad (4)$$

$$BT = 88.721 \text{ meter} \quad (5)$$

Jadi dapat kita simpulkan batas ketinggian maksimal pada koordinat 3.651089 latitude dan 98.864494 longitude adalah 88.721 meter dari permukaan tanah.

3.2 Hasil Penelitian

Selama pelaksanaan analisis, implementasi, pengembangan dan pengujian sistem online KKOP penulis menemukan beberapa kondisi sebagai berikut:

- 1) File KML sering mengalami kesulitan loading diatas google map, sehingga penulis menemukan solusi melakukan perubahan nama file kemudian di load ulang.
- 2) Ketika jaringan internet mengalami kendala sering kali gambar KKOP tidak loading sempurna, hal tersebut dapat diperbaiki dengan melakukan loading ulang.



Gambar 9. Gambar KKOP diatas wilayah Bandar Udara Husein Sastranegara menggunakan *google map*.

Pada Gambar 9 terlihat peta KKOP yang berada di wilayah bandar udara Husein Sastranegara kota Bandung, dalam *user interface* aplikasi *android* yang dikembangkan akan terlihat tampilan peta kota Bandung yang sudah dilengkapi dengan beberapa fitur lain dari aplikasi seperti yang terlihat pada Gambar 10. Fitur utama dari aplikasi KKOP ini adalah tombol simbol plus berwarna biru yang berfungsi sebagai *trigger* untuk menghitung batas maksimal ketinggian yang diperbolehkan untuk membangun sebuah bangunan dititik tersebut menurut peraturan KKOP.

Selain dari fitur utama didalam aplikasi tersebut juga terdapat beberapa fitur lain diantaranya; hitung jarak antara titik, pilih bandar udara, *my location*, serta perubahan jenis *map* ke foto satelit. Kemudian disisi kanan atas dari gambar terdapat empat fitur sekunder yaitu fitur bantuan, fitur data *obstacle*, fitur data *summary* dan fitur *hide and show* data analisis bangunan.

Dari Gambar 9 dan Gambar 10 dapat terlihat bahwa pemanfaatan teknologi *google map* digabungkan dengan hasil gambar dari teknologi *ArcGIS* berupa layer gambar KKOP yang dihasilkan dari *file KML* dapat diimplementasikan dibawah *platform Android* untuk menghasilkan *user interface* seperti yang terlihat dari gambar-gambar tersebut dari pengguna *device android*. Namun hasil tampilan tersebut perlu juga didukung dengan pengolahan data-data batas kawasan seperti yang terlihat pada tabel 3 dan tabel 4 menggunakan teknologi *backend NodeJS* yang mengirim data *output* seperti yang terlihat pada Gambar 6 kemudian data tersebut diolah supaya menjadi tampilan seperti Gambar 9 dan Gambar 10.



Gambar 10. Gambar KKOP diatas wilayah Bandar Udara Husein Sastranegara menggunakan *google map* dilengkapi dengan fitur lainnya.

Dari pengujian yang telah dilakukan berdasarkan *scenario* pengujian yang telah ditetapkan, dan ditemukan kondisi bahwa skema arsitektur sistem *online* KKOP seperti yang terlihat pada Gambar 4 dapat diimplementasikan dengan baik, begitu pula dengan perpaduan teknologi *NodeJS* dengan *file KML* hasil dari data *render ArcGIS* yang dikirim ke *platform Android* hingga berhasil menampilkan sebuah sistem *online* KKOP seperti yang terlihat pada Gambar 9 dan Gambar 10 untuk Bandar Udara Husein Sastranegara kota Bandung.

4. Kesimpulan

Dengan penelitian yang telah dilakukan tentang implementasi analisis sistem *online* kawasan keselamatan operasional penerbangan menggunakan *web service* berbasis *nodeJS* dan *android* dengan fokus terhadap wilayah KKOP Bandar Udara Husein Sastranegara kota Bandung yang dapat diketahui yaitu;

- 1) Peraturan Menteri Perhubungan tentang KKOP Bandar Udara Husein Sastranegara di kota Bandung dapat diimplementasikan dan dibangun sebagai sistem online.
- 2) Perpaduan teknologi *NodeJS*, *file* KML dan *Android* berhasil diimplementasikan sebagai sistem *online* KKOP.

Ucapan Terima kasih

Ucapan terima kasih kepada PT. Angkasa Pura 2 yang telah memberikan kesempatan kepada kami untuk melakukan penelitian terhadap data-data Bandar Udara di Indonesia.

Daftar Pustaka

- [1] Sugiyono., 2012, Metode Penelitian Kuantitatif Kualitatif dan R&D., Alfabeta, Bandung.
- [2] Roger, S. Pressman, Ph.D., 2015, Software engineering : a practitioner's approach, Eight Edition : McGraw Hill Education.
- [3] Nacheva, Radka, 2017, Prototyping Approach in User Interface Development, Conference on Innovative Teaching Methods, University of Economics Varna, Bulgaria.
- [4] Ginanjar, Arief., Sari, WP., Rahmawati, Hani., Dwipriyoko, E., 2019., Metodologi RUP Terhadap Pengolahan Data Nilai Siswa Berbasis Android dan NodeJS., Jurnal TIARSIE, Universitas Langlangbuana., Bandung.
- [5] Kalelkar, Medha., Churi, Prathamesh., Kalelkar, Deepa., 2014, Implementation of Model-View-Controller Architecture Pattern for Business Intelligence Architecture., International Journal of Computer Application., New York, USA.
- [6] Hall, Marry., 2016, Basic Object-Oriented Programming in Java., Core Servlet.
- [7] Ginanjar, Arief and Hendayun, M., 2019, Spring Framework Reliability Investigation Against Database Bridging Layer Using Java Platform, Procedia Computer Science, The Fifth Information Systems International Conference, 23-24 July 2019, Indonesia.
- [8] Ilman, Benie dan Ginanjar, Arief., 2019, Rancang Bangun Web Service JSON Menggunakan Kombinasi Spring dan MyBatis Framework dalam Lingkungan Java Platform, Jurnal Teknologika., STT Wastukencana, Purwakarta, Indonesia.
- [9] Safaat, Nazruddin., 2014., Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android., Penerbit Informatika, Bandung.
- [10] Setiawan, Awan., Yulianto, Erwin., Abadi, Iwan., Ginanjar, Arief., 2020., Obstacles Analysis Digitalization of Flight Area With Aerodrome Elevation System Approach. *SciTech Framework*, 2(1), 1-11.