

# Implementasi *Framework* Laravel dan *Database* Dalam Pengembangan *Website* Prediksi Kebakaran Hutan di Indonesia

1<sup>st</sup> Aldi Fauzan

Program Studi S1 Teknik Komputer  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia

aldifauzaan@student.telkomuniversity.ac.id

2<sup>nd</sup> Casi Setianingsih

Program Studi S1 Teknik Komputer  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia

setiacasie@telkomuniversity.ac.id

3<sup>rd</sup> Meta Kallista

Program Studi S1 Teknik Komputer  
Fakultas Teknik Elektro  
Universitas Telkom  
Bandung, Indonesia

metakallista@telkomuniversity.ac.id

**Abstrak** — Kebakaran hutan dan lahan merupakan salah satu bencana alam yang menyebabkan berbagai dampak negatif pada habitat manusia dan ekosistem hutan. Hutan memiliki berbagai peranan penting dalam menyumbangkan oksigen bagi keberlangsungan hidup serta berfungsi sebagai paru-paru dunia. Setiap tahunnya, Indonesia mengalami berbagai kejadian kebakaran hutan dan lahan yang dapat berdampak di berbagai aspek. Oleh karena itu, dibutuhkan tindakan pencegahan yang dapat meminimalisir dampak kebakaran hutan. Pencegahan dapat dilakukan dengan mengimplementasikan algoritma *machine learning* untuk melakukan proses prediksi kebakaran dan pemanfaatan *Geographical Information System* dalam melakukan pemetaan berdasarkan perhitungan FWI. *Framework* yang digunakan dalam pengembangan *website* prediksi kebakaran hutan adalah *Laravel* dan *PostgreSQL* sebagai database pengembangan sistem. Pengujian dilakukan dengan dua metode yaitu Alpha Testing dan Beta Testing.

**Kata kunci**— *Geographical Information System, Fire Weather Index, Laravel, PostgreSQL.*

## I. PENDAHULUAN

Indonesia merupakan salah satu negara dengan hutan tropis terbesar kedelapan di dunia dan disebut sebagai salah satu paru-paru dunia yang menyumbangkan oksigen untuk keberlangsungan hidup [1]. Namun, hutan yang ada semakin menurun diakibatkan oleh kejadian kebakaran hutan. Kebakaran hutan telah menjadi ancaman besar di seluruh dunia, menyebabkan banyak dampak negatif pada habitat manusia dan ekosistem hutan [2]. Kebakaran hutan dan lahan disebabkan oleh dua faktor utama, yaitu manusia dan alam.

Faktor oleh manusia biasanya bertujuan untuk alih fungsi lahan untuk perkebunan maupun industri, sedangkan untuk faktor alam biasanya disebabkan oleh perubahan iklim yang ekstrim. Kebakaran hutan telah mengakibatkan kerugian dari berbagai aspek seperti kesehatan, ekonomi, dan lingkungan. Berdasarkan data dari Kementerian Lingkungan Hidup dan Kehutanan, tercatat luas kebakaran hutan dan lahan di Indonesia sepanjang Januari hingga Oktober 2023 yakni sekitar 994.313 hektare [3]. Selama periode antara 2019 dan Oktober 2023, kebakaran hutan dan lahan di Indonesia totalnya sudah mencapai 3.504.274 hektare. Kasus

kebakaran hutan merupakan bencana tahunan yang sering terjadi di Indonesia. Oleh karena itu, dibutuhkan tindakan pencegahan kebakaran hutan agar dapat meminimalisir dampak yang ada.

Upaya pencegahan yang dapat dilakukan dalam meminimalisir risiko kebakaran hutan adalah dengan melakukan prediksi kebakaran secara akurat. Setelah itu, dilakukan perhitungan *Fire Weather Index* (FWI) yang dapat menilai potensi kebakaran hutan berdasarkan hasil prediksi dari keempat parameter yaitu suhu, kelembaban, curah hujan, dan kecepatan angin. FWI adalah komponen dari sistem peringatan kebakaran hutan yang digunakan untuk mengukur resiko kebakaran berdasarkan kondisi cuaca.

Sistem yang dikembangkan dalam memprediksi kebakaran hutan akan memanfaatkan algoritma *machine learning* untuk melakukan prediksi berbagai parameter dan pengembangan *Geographical Information System* (GIS).

## II. KAJIAN TEORI

Pengembangan *Website* sistem prediksi kebakaran hutan dan lahan di Indonesia dengan memanfaatkan berbagai komponen seperti *Fire Weather Index, Geographical Information System, Laravel, dan PostgreSQL*. Berbagai fitur yang dapat digunakan dalam *website* ini seperti admin dapat melakukan input data dan melakukan proses prediksi kebakaran hutan. *PostgreSQL* digunakan sebagai platform *database* dalam melakukan penyimpanan berbagai data seperti provinsi, kabupaten, *user*, data aktual, dan data hasil prediksi dari parameter. Selain itu, digunakan GIS dalam melakukan proses visualisasi pemetaan FWI dari hasil prediksi kebakaran hutan agar dapat memberikan informasi pencegahan yang mudah dipahami oleh *user*.

### A. *Fire Weather Index*

*Fire Weather Index* (FWI) merupakan indeks atau sistem yang digunakan dalam menilai potensi kebakaran hutan. Indeks ini dipengaruhi oleh beberapa parameter seperti temperature, kelembaban, kecepatan angin, dan curah hujan untuk menilai risiko kebakaran hutan dan lahan. FWI terdiri dari beberapa komponen penyusun utama seperti *Fine Fuel Moisture Code* (FFMC), *Duff Moisture Code* (DMC), *Drought Code* (DC), *Initial Spread Index* (ISI), *Buildup Index* (BUI), dan *Fire Weather Index* (FWI).

Tabel 1 Kategori FWI

No	Kategori	FWI
1.	Rendah	< 1
2.	Sedang	1 – 6
3.	Tinggi	6 – 13
4.	Sangat Tinggi	> 13

### B. Geographical Information System

*Geographical Information System* (GIS) merupakan sistem informasi yang dapat digunakan untuk menangkap, menganalisis, mengelola, dan menyajikan beragam jenis data geografis atau spasial [4]. Pemetaan GIS digunakan dalam memetakan daerah-daerah yang rentan terhadap kebakaran berdasarkan data historis. Pemanfaatan GIS dapat mempermudah visualisasi data risiko kebakaran dalam bentuk peta yang intuitif dan mudah dipahami oleh pengguna, membantu dalam pengambilan keputusan yang cepat dan tepat dalam upaya pencegahan kebakaran hutan.

### C. Laravel

*Laravel* merupakan sebuah *framework* dari bahasa pemrograman PHP yang memiliki banyak fitur dalam membantu developer dalam membangun sebuah aplikasi berbasis *website* [5]. *Laravel* menerapkan konsep *Model-View-Controller* (MVC) dalam sebuah pola desain arsitektur yang digunakan dalam pengembangan *website*. Konsep MVC ini memisahkan sistem menjadi tiga komponen yaitu *Model*, *View*, dan *Controller*. Komponen *Model* menangani logika data dan berinteraksi dengan *database*, mengambil dan menyimpan informasi sesuai kebutuhan. Komponen *View* bertanggung jawab untuk menyajikan data kepada pengguna, merender elemen *interface* pengguna berdasarkan data yang diterima. *Controller* bertindak sebagai perantara, memproses permintaan pengguna, berinteraksi dengan *Model* untuk mengambil atau memperbarui data, dan menentukan *View* mana yang akan ditampilkan.

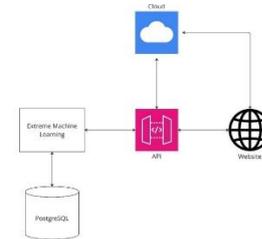
### D. PostgreSQL

*PostgreSQL* merupakan sistem manajemen basis data open-source yang sangat baik dan fleksibel. *PostgreSQL* merupakan *database* yang dikategorikan sebagai *Relational Database Management System* (RDBMS). RDBMS merupakan sistem manajemen database yang memungkinkan untuk menyimpan, mengelola, dan mengatur data berbasis tabel yang terkait satu sama lain melalui *primary key* dan *foreign key*. Pemilihan *PostgreSQL* dikarenakan berbagai faktor seperti efisiensi dalam *database* ukuran yang besar terutama data yang akan dikelola juga memiliki skala yang besar, sehingga dibutuhkan kecepatan dalam pengambilan *query* dari *database* yang digunakan. Percobaan pengambilan *query* dari kedua *database* telah dilakukan sebanyak 50 ribu hingga 1 juta *record* dengan hasil *response time PostgreSQL* memiliki hasil yang lebih baik dibandingkan *MySQL* dengan perbandingan selisih rata-rata 0.44 s antara kedua *database* [6]. Selain itu, terdapat penelitian yang dilakukan juga terhadap *response time* dengan menggunakan beberapa *query* seperti *SELECT*, *INSERT*, *UPDATE*, *DELETE*, *sum()*, dan *count()* dengan membandingkan antara 5 *limit* data yaitu 250, 2500, 5000, 1500, dan 20000 menghasilkan bahwa *response time* dari *PostgreSQL* lebih baik dibandingkan menggunakan *MySQL* [7].

## III. METODE

Berikut ini merupakan berbagai rancangan detail solusi yang terpilih dari berbagai arsitektur seperti arsitektur sistem, flowchart, data flow diagram, *entity relationship diagram* dan lainnya.

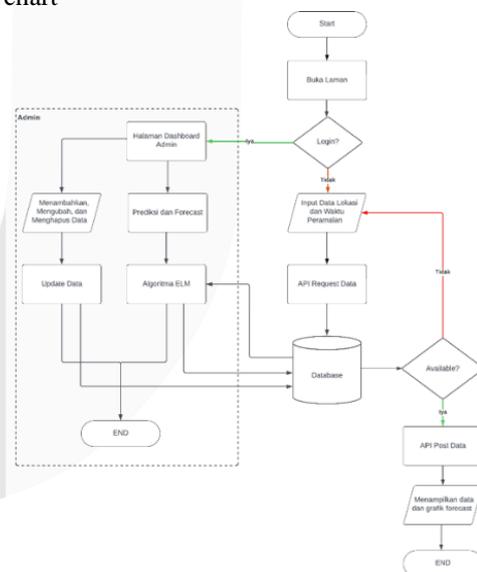
### A. Arsitektur Sistem



Gambar 1 Arsitektur Sistem

Berdasarkan rancangan arsitektur sistem yang telah divisualisasikan diatas, data yang dilakukan preprocessing akan disimpan dalam *Database Management System* (DBMS) yang akan menggunakan *PostgreSQL*. Data yang disimpan pada *database* merupakan data-data utama seperti lokasi, suhu, kelembaban, curah hujan, dan kecepatan angin. Setelahnya, akan dilakukan pengambilan *query* dari *database* dan akan dilakukan proses perhitungan prediksi dan pengambilan keputusan menggunakan algoritma yang akan digunakan. Hasil dari perhitungan prediksi akan disampaikan melalui *response API* yang di *deploy* pada *cloud*. Setelah melakukan *deploy* di *cloud* maka hasil *back-end* akan ditampilkan melalui *front-end* atau tampilan dari *website*. *Front-end* juga akan melakukan permintaan data sesuai area dan waktu yang ditentukan.

### B. Flowchart

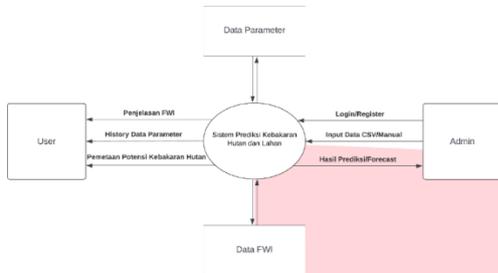


Gambar 2 Flowchart

Gambar di atas menunjukkan alur kerja dari website prediksi kebakaran hutan dan lahan di Indonesia. Proses dimulai dengan *user* membuka halaman *website*. Sistem kemudian memeriksa apakah *user* sudah login. Jika sudah login, *user* akan diarahkan ke Halaman *Dashboard Admin* di mana mereka dapat menambahkan, mengubah, dan menghapus data, melakukan prediksi dan *forecast*, serta memperbarui data yang ada. Prediksi dilakukan menggunakan algoritma *Extreme Learning Machine* berdasarkan data yang ada.

Jika *user* belum login, *user* akan diarahkan untuk melihat hasil prediksi dan *forecast* dengan memasukkan data lokasi dan waktu yang diinput. Setelah data dimasukkan, sistem akan mengirimkan *requests data* ke *database* melalui API *back-end*. Jika data tersedia, akan mengirim hasilnya kembali ke *front-end*, di mana data dan grafik prediksi akan ditampilkan kepada *user*. Jika data tidak tersedia, proses berakhir dan *user* harus mengulangi input data.

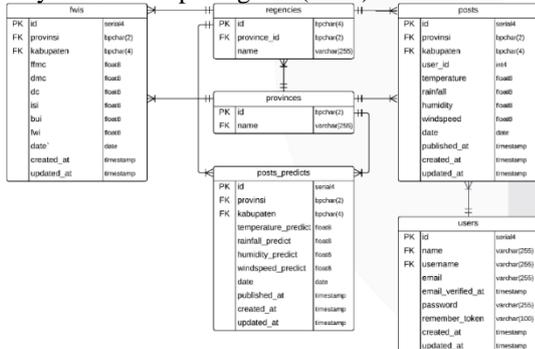
### C. Data Flow Diagram (DFD)



Gambar 3 Data Flow Diagram Level 0

Gambar di atas menunjukkan DFD dari sistem prediksi kebakaran hutan dan lahan. *User* dapat melihat penjelasan tentang *Fire Weather Index (FWI)*, mengakses data historis parameter, dan melihat peta potensi kebakaran hutan berdasarkan hasil prediksi sistem. *Admin* dapat *login* atau mendaftar, memasukkan data baru melalui *import file CSV* atau *input manual*, serta mengakses hasil prediksi atau *forecast* yang dihasilkan oleh sistem. Sistem prediksi ini memproses data dari dua sumber utama, yaitu data parameter dan data FWI, untuk menghasilkan prediksi kebakaran hutan yang akurat. Data parameter dan data FWI diterima oleh sistem untuk diproses lebih lanjut dalam menghasilkan *output* yang dapat diakses oleh *user* dan *admin*.

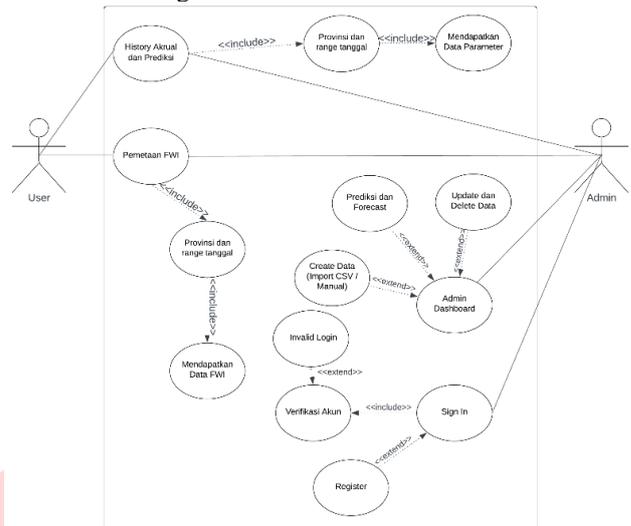
### D. Entity Relationship Diagram (ERD)



Gambar 4 Entity Relationship Diagram

Gambar di atas menunjukkan ERD untuk sistem prediksi kebakaran hutan dan lahan. Diagram ini mencakup tabel *fwis*, *regencies*, *provinces*, *posts*, *users*, dan *posts\_predicts*. Tabel *fwis* menyimpan data indeks cuaca kebakaran dan terhubung dengan *provinces* dan *regencies*. Tabel *regencies* dan *provinces* menyimpan nama dan ID kabupaten serta provinsi. Tabel *posts* mencatat data cuaca seperti suhu, curah hujan, kelembaban, dan kecepatan angin, serta ID pengguna. Tabel *users* menyimpan informasi pengguna seperti nama, *username*, dan *email*. Tabel *posts\_predicts* menyimpan prediksi data cuaca.

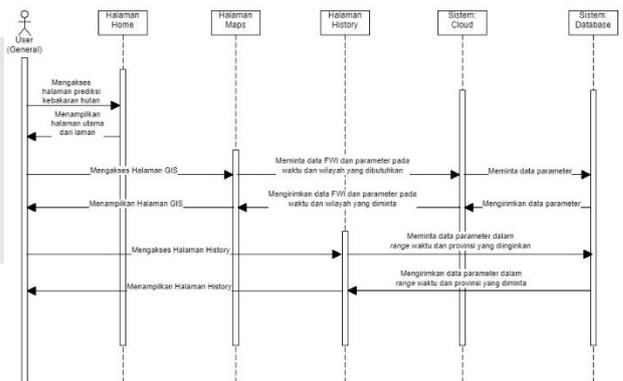
### E. Use Case Diagram



Gambar 5 Use Case Diagram

Berdasarkan gambar diatas terdapat dua aktor yaitu *user* dan *admin* yang akan mengakses dari *website*. *User* memiliki dapat interaksi dengan sistem, seperti melihat Pemetaan FWI dan *History Aktual dan Prediksi*. Dalam melakukan *History Aktual dan Prediksi*, *user* harus memasukkan provinsi dan *range tanggal*, dan sistem akan mendapatkan data parameter yang diperlukan untuk prediksi. Sedangkan pada pemetaan FWI, *user* juga harus memasukkan provinsi dan *range tanggal* untuk mendapatkan data FWI. Sementara itu, *admin* memiliki berbagai tugas yang lebih kompleks. *Admin* dapat melakukan *Sign In* untuk masuk ke sistem. Selain itu, *admin* juga memiliki akses untuk melakukan *Register* akun baru. Setelah masuk ke dalam sistem, *admin* dapat mengakses *Admin Dashboard*, yang memiliki beberapa ekstensi yaitu *Prediksi dan Forecast*, serta *Update dan Delete Data*. *Admin* juga memiliki kemampuan untuk *Create Data* dengan mengimpor data menggunakan *format CSV* atau secara manual.

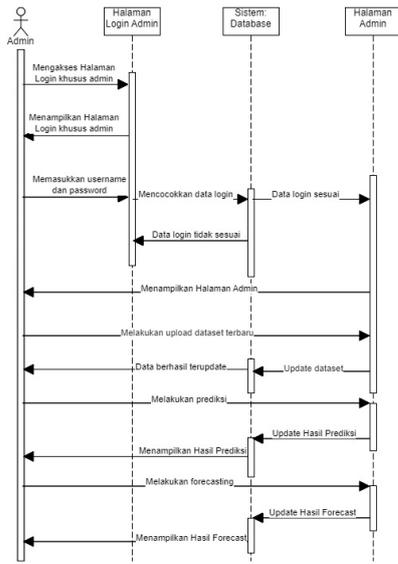
### F. Sequence Diagram



Gambar 6 Sequence Diagram User

Berdasarkan gambar diatas, objek yang dimaksud adalah setiap halaman, *cloud server*, dan *database* yang akan saling terhubung kepada *user* yang akan melakukan interaksi kepada objek tersebut. Ketiga objek tersebut akan saling berhubungan berdasarkan urutan kejadian dengan *output* yang diinginkan. *User* akan berinteraksi dengan halaman yang akan dipilih, maka akan berkomunikasi dengan *cloud server* untuk mengakses data dalam *database*. Hubungan ini

akan membentuk interaksi yang diarahkan untuk mendapatkan respon yang dibutuhkan dari sistem.



Gambar 7 Sequence Diagram Admin

Diagram diatas menjelaskan alur interaksi dari *admin* antar setiap halaman dan menghubungkan ke dalam sistem *database*. Alur ini menjelaskan bahwa *admin* akan mengkases halaman *login* terlebih dahulu dan akan memasukkan *username* dan *password*. Kedua data tersebut akan dicocokkan dengan *database* yang akan memverifikasi data tersebut sesuai dengan *database*. Jika sudah terverifikasi maka akan masuk ke dalam halaman *admin* dan melakukan *upload dataset* terbaru ke dalam *database*.

#### IV. HASIL DAN PEMBAHASAN

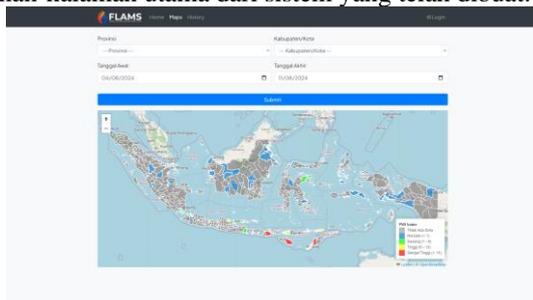
##### A. Hasil Implementasi Sistem

Hasil pengembangan sistem prediksi kebakaran hutan berbasis *website* menggunakan *framework Laravel* dan database *PostgreSQL* telah berhasil dikembangkan. Sistem ini dirancang untuk memprediksi kebakaran hutan berdasarkan parameter-parameter cuaca seperti suhu, kelembaban, curah hujan, dan kecepatan angin.

Tabel 2 Spesifikasi Database

Spesifikasi	Keterangan
Region	asia-southeast2 (Jakarta)
vCPU	1 vCPU
Memory	3.75 GB
Storage	10 GB

Tabel di atas merupakan spesifikasi dari *database PostgreSQL* yang sudah dihosting menggunakan *Google Cloud*. *Database* ini digunakan untuk konfigurasi koneksi dengan aplikasi *Laravel*. *PostgreSQL* ini dihosting di region *asia-southeast2* (Jakarta). Berikut ini adalah tampilan halaman-halaman utama dari sistem yang telah dibuat:

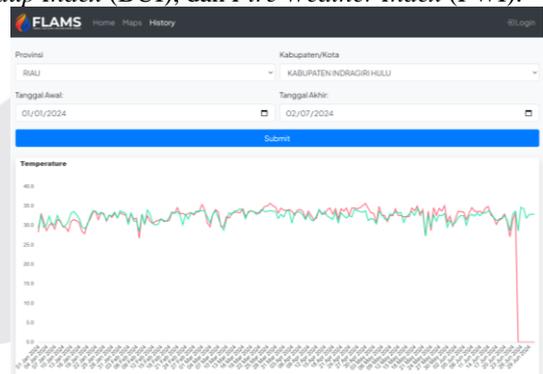


Gambar 8 Halaman Maps Indonesia



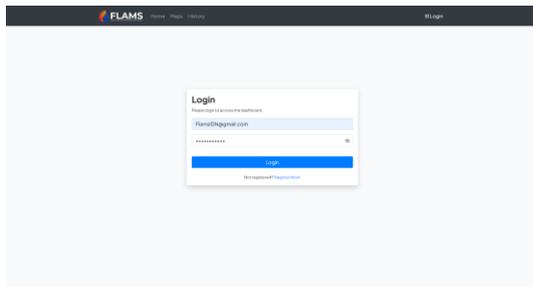
Gambar 9 Grafik Komponen FWI

Gambar 8 menunjukkan tampilan peta Indonesia dengan overlay potensi kebakaran hutan berdasarkan nilai FWI. Peta ini menggunakan *Geographical Information System (GIS)* untuk memetakan daerah-daerah yang memiliki potensi kebakaran berdasarkan data historis dan prediksi cuaca. Sedangkan pada Gambar 9 merupakan tampilan grafik komponen-komponen dari *Fire Weather Index (FWI)*, yaitu *Fine Fuel Moisture Code (FFMC)*, *Duff Moisture Code (DMC)*, *Drought Code (DC)*, *Initial Spread Index (ISI)*, *Buildup Index (BUI)*, dan *Fire Weather Index (FWI)*.



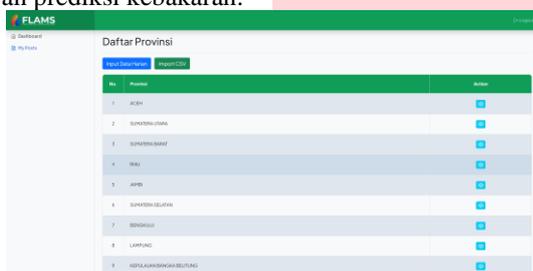
Gambar 10 Halaman History

Gambar 10 memungkinkan pengguna memilih provinsi, kabupaten/kota, dan rentang tanggal untuk melihat data historis dan prediksi suhu. Grafik ini menampilkan parameter data aktual dan parameter yang diprediksi menggunakan algoritma *Extreme Learning Machine*, memungkinkan analisis keakuratan prediksi dan pola cuaca yang mempengaruhi risiko kebakaran hutan.



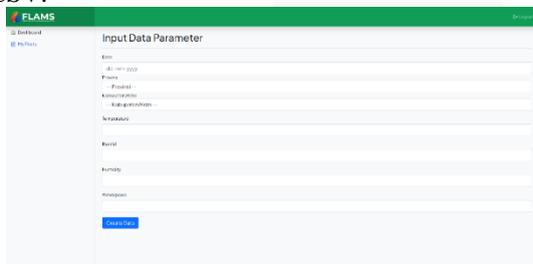
Gambar 11 Halaman Login

Gambar 11 menunjukkan halaman *login* yang digunakan untuk mengakses fitur-fitur *admin* seperti input data, *dashboard admin*, dan proses melakukan prediksi dan *forecast*. Pengguna harus memasukkan *email* dan kata sandi mereka untuk masuk ke sistem. Setelah berhasil *login*, pengguna dengan hak akses *admin* dapat mengelola data yang digunakan oleh sistem, memperbarui data, dan melihat laporan prediksi kebakaran.

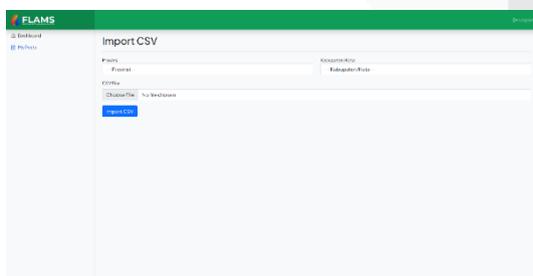


Gambar 12 Halaman Daftar Provinsi

Gambar 12 menunjukkan halaman Daftar Provinsi yang dapat diakses setelah *login*. Halaman ini memungkinkan *admin* untuk mengelola data provinsi, termasuk menambah, memperbarui, dan menghapus data. Terdapat tombol untuk menginput data secara manual atau mengimpor data melalui file CSV.



Gambar 13 Halaman Input Data



Gambar 14 Halaman Import CSV

Gambar 13 dan 14 menunjukkan halaman Input Data Parameter dan Import CSV yang memungkinkan *admin* untuk melakukan input data suatu wilayah. Halaman Input Data Parameter memungkinkan *admin* memasukkan detail seperti provinsi, kabupaten/kota, tanggal, suhu, curah hujan, kelembaban, dan kecepatan angin secara manual. Sementara itu, halaman Import CSV memudahkan *admin* *input file* CSV

yang berisi data parameter cuaca dalam jumlah besar, dengan memilih provinsi dan kabupaten/kota.



Gambar 15 Halaman Detail Data

Gambar 15 menunjukkan halaman Detail Data yang menampilkan data cuaca untuk Jawa Barat, Kabupaten Bogor. *Admin* dapat melihat data parameter seperti suhu, kelembaban, curah hujan, dan kecepatan angin, serta hasil prediksi dan *forecast* untuk setiap tanggal. Halaman ini juga menyediakan tombol untuk mengedit dan menghapus data, serta tombol untuk menginput data secara manual dan mengimpor data melalui file CSV. Selain itu, terdapat *button* prediksi untuk menghasilkan prediksi berdasarkan data yang ada, dengan menjalankan proses prediksi yang dijalankan dengan algoritma *Extreme Learning Machine*. Tombol *forecast* memberikan peramalan cuaca untuk tujuh hari ke depan, memungkinkan *admin* untuk memantau potensi risiko kebakaran hutan.

## B. Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa sistem prediksi kebakaran hutan dapat berfungsi dengan baik dan dapat memberikan hasil yang akurat. Pengujian sistem ini dilakukan dengan dua metode yaitu *Alpha Testing* dan *Beta Testing*.

*Alpha Testing* adalah pengujian awal yang dilakukan oleh pengembang website dengan tujuan untuk mengidentifikasi dan memperbaiki bug atau kesalahan sebelum *website* diuji oleh pengguna eksternal. *Beta Testing* adalah pengujian yang melibatkan *user* untuk mendapatkan *feedback* terkait perspektif *user* dan mengidentifikasi suatu permasalahan pada *user*. Tahapan ini melibatkan pemilihan kelompok *user*, memberikan akses kepada *user* ke *website* serta instruksi skenario pengujian, pengumpulan *feedback* berupa laporan bug, dan lainnya.

### 1. Alpha Testing

Pengujian alpha testing ini menggunakan jenis *blackbox testing*. Hasil Pengujian alpha testing akan menguji tingkat keberhasilan aplikasi yang dapat berjalan. Berdasarkan hasil dari Alpha Testing yang sudah dilakukan pada setiap halaman yang sudah dibuat, hasil yang diperoleh sudah berjalan dengan sesuai harapan. Berikut ini hasil perhitungan dari Alpha Testing:

$$\text{Alpha Testing} = \frac{\text{Total Berhasil}}{\text{Total Pengujian}} \times 100\% \quad (1)$$

$$\text{Alpha Testing} = \frac{90}{90} \times 100\% = 100\% \quad (2)$$

Berdasarkan hasil pengujian *Alpha Testing*, hasil pengujian fungsi pada setiap aplikasi dapat berjalan sesuai dengan skenario.

## 2. Beta Testing

*Beta Testing* dilakukan dengan menyebarkan kuesioner melalui Google Form kepada sekelompok pengguna yang *user* dengan kriteria pengisian kuesioner. Kriteria pengguna yang diikutsertakan dalam pengujian ini adalah masyarakat berusia di atas 18 tahun. Selain itu, pengguna diutamakan peserta yang berdomisili di daerah yang rentan terhadap kejadian kebakaran hutan dan lahan. Hal ini bertujuan untuk mendapatkan *feedback* yang relevan dari *user* yang kemungkinan besar akan menggunakan aplikasi. Jawaban responden akan diukur menggunakan Skala *Likert*. Skala *Likert* adalah alat ukur yang digunakan dalam mengukur pendapat seseorang terhadap suatu pertanyaan. Skala ini umumnya digunakan dalam riset berupa *survey* [8]. Berikut ini adalah tabel yang menjelaskan keterangan dari Skala *Likert*:

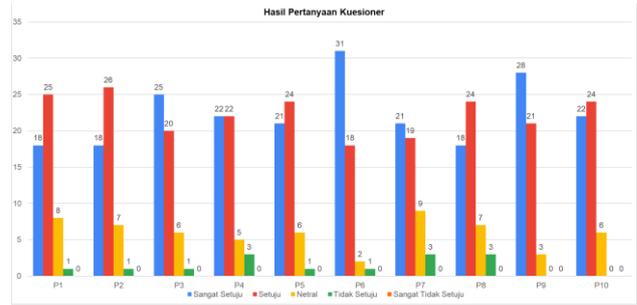
Tabel 3 Skala Likert

Nilai	Keterangan
1	Sangat Tidak Setuju
2	Tidak Setuju
3	Netral
4	Setuju
5	Sangat Setuju

Pengujian *Beta Testing website* terdiri dari 10 pertanyaan yang diajukan kepada *user*. Berikut ini adalah pertanyaan yang akan diisi oleh responden:

Tabel 4 Pertanyaan Beta Testing

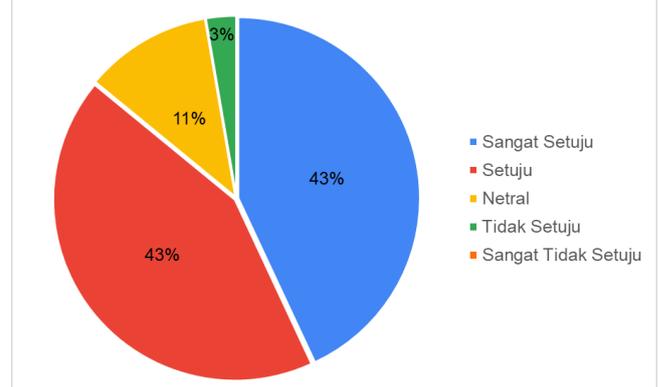
No.	Pertanyaan	Nilai				
		1	2	3	4	5
1.	Apakah desain <i>User Interface</i> dari Website Sistem Prediksi Kebakaran Hutan dan Lahan ini dapat dikatakan menarik?					
2.	Apakah website memiliki antar muka yang mudah digunakan ( <i>User Friendly</i> )?					
3.	Apakah fitur pada setiap halaman mudah digunakan dan sesuai dengan harapan ketika dibuka?					
4.	Apakah panduan pengguna ( <i>User Guide</i> ) sangat membantu dalam menjalankan website?					
5.	Apakah website dapat memberikan informasi yang cukup jelas dan mudah dipahami?					
6.	Apakah teks dalam website ini dapat dibaca dengan jelas dan mudah dipahami?					
7.	Apakah website dapat diakses dengan lancar?					
8.	Apakah <i>user</i> dapat dengan mudah melakukan <i>predict</i> dan <i>forecast</i> ?					
9.	Apakah Anda dapat mudah memahami tentang sistem prediksi Kebakaran Hutan?					
10.	Apakah Anda puas dengan kinerja dengan <i>website</i> ini ketika menggunakan berbagai fitur?					



Gambar 16 Visualisasi Jawaban Responden

Pelaksanaan pengujian *beta testing* dilakukan secara daring pada rentang tanggal 26 Juni hingga 7 Juli 2024 dengan menyebarkan kuesioner. Hasil dari 52 responden menunjukkan lima wilayah dengan jumlah responden terbanyak, yaitu Jawa Tengah sebesar 17.3%, Bengkulu sebesar 23.1%, Riau sebesar 17.3%, Jawa Timur sebesar 7.7%, dan Kepulauan Riau dan Sumatera Barat sebesar 5.8%. Responden lainnya berasal dari berbagai wilayah lainnya. Pembagian usia responden adalah sebagai berikut: 21-25 tahun sebanyak 59.6%, 17-20 tahun sebanyak 13.5%, 26-30 dan di atas 35 tahun masing-masing sebanyak 9.6%, serta kelompok usia 30-35 masing-masing sebanyak 7.7%.

Total Keseluruhan Jawaban dari 10 Pertanyaan



Gambar 17 Total Keseluruhan Jawaban dari 10 Pertanyaan

Grafik diatas menampilkan jumlah total jawaban dari 10 pertanyaan berdasarkan skala *Likert* dengan *range* nilai 1 – 5. Berdasarkan total keseluruhan jawaban dari 10 pertanyaan, terdapat sekitar 43% responden memilih sangat setuju dari keseluruhan pertanyaan dan sekitar 43% responden memilih setuju dari keseluruhan pertanyaan.

### a. Pengujian Usability

*Usability Testing* merupakan teknik yang digunakan untuk mengevaluasi sistem atau produk dalam proses pengujian langsung pada pengguna [9]. Berikut ini hasil pengujian *usability testing* berdasarkan setiap pertanyaan yang sudah diajukan kepada *user*:

Tabel 5 Hasil Usability Testing

No.	Pertanyaan	Nilai	Keterangan
1.	P1	83.07%	Sangat Setuju
2.	P2	83.46%	Sangat Setuju
3.	P3	86.53%	Sangat Setuju
4.	P4	84.23%	Sangat Setuju
5.	P5	85%	Sangat Setuju

6.	P6	90.38%	Sangat Setuju
7.	P7	82.30%	Sangat Setuju
8.	P8	81.92%	Sangat Setuju
9.	P9	89.61%	Sangat Setuju
10.	P10	86.15%	Sangat Setuju
<b>Rata-rata</b>		<b>85.265%</b>	<b>Sangat Setuju</b>

Setelah dilakukan pengujian *beta testing* dengan menyebarkan *survey* sebanyak 52 responden, dapat disimpulkan responden pengisian *survey* Sangat Setuju dengan rata-rata kepuasan responden sebesar 85.265% dari keseluruhan pertanyaan

#### b. Pengujian Validitas

Pengujian Validitas dilakukan dengan menggunakan metode korelasi. Jika nilai  $r_{xy}$  lebih besar dari  $r_{tabel}$ , maka pertanyaan dianggap valid dikarenakan menunjukkan hubungan signifikan dengan skor total.

Tabel 6 Hasil Pengujian Validitas

Pertanyaan	$r_{xy}$	$r_{tabel}$	Status
P1	0.7702	0.271	Valid
P2	0.7340	0.271	Valid
P3	0.7536	0.271	Valid
P4	0.7923	0.271	Valid
P5	0.7974	0.271	Valid
P6	0.5587	0.271	Valid
P7	0.7357	0.271	Valid
P8	0.8461	0.271	Valid
P9	0.6727	0.271	Valid
P10	0.8596	0.271	Valid

Pemilihan nilai  $r_{tabel}$  sebesar 0.271 didasarkan pada jumlah responden sebanyak 52 orang dengan tingkat signifikansi yang digunakan adalah  $\alpha = 0.05$  (5%). Koefisien korelasi Pearson ( $r_{xy}$ ) akan dibandingkan nilai tabel distribusi ( $r_{tabel}$ ). Semua pertanyaan dari P1 hingga P10 memiliki  $r_{xy}$  yang lebih besar dari  $r_{tabel}$  sebesar 0.271, sehingga semua pertanyaan valid

#### c. Pengujian Reliabilitas

Tabel 7 Hasil Pengujian Reliabilitas

Jumlah Variansi Butir	Total Variansi	$r_{11}$	Reliabilitas
5.665	31.18	0.909	Sangat Tinggi

Nilai Koefisien *Cronbach Alpha* ( $r_{11}$ ) yang diperoleh sebesar 0.909 menunjukkan tingkat reliabilitas yang sangat tinggi. Hal ini membuktikan bahwa jawaban kuesioner yang digunakan dalam pengujian dapat diandalkan.

## V. KESIMPULAN

*Alpha Testing* dilakukan untuk memastikan bahwa sistem berfungsi sesuai dengan spesifikasi yang telah ditetapkan. Hasil *Alpha Testing* menunjukkan bahwa semua fitur dapat berjalan dengan baik melalui skenario *Blackbox Testing*. *Beta Testing* yang melibatkan 52 responden menunjukkan hasil yang sangat positif, dengan rata-rata kepuasan responden mencapai 85.265%. Validitas pertanyaan dalam kuesioner terbukti dengan metode korelasi *Pearson*, dan tingkat reliabilitas yang sangat tinggi dengan koefisien *Cronbach*

*Alpha* sebesar 0.909 menunjukkan bahwa jawaban responden dapat diandalkan.

## REFERENSI

- [1] L. Dita Shafitri dan Y. Prasetyo, "Analisis Deforestasi Hutan di Provinsi Riau dengan Metode Polarimetri dalam Pengindraan Jauh," *Jurnal Geodesi Undip Januari*, vol. 7, no. 1, hlm. 212–222, Jan 2018, doi: <https://doi.org/10.14710/jgundip.2017.19330>.
- [2] U. Dampage, L. Bandaranayake, R. Wanasinghe, K. Kottahachchi, dan B. Jayasanka, "Forest fire detection system using wireless sensor networks and machine learning," *Sci Rep*, vol. 12, no. 1, hlm. 46, Jan 2022, doi: [10.1038/s41598-021-03882-9](https://doi.org/10.1038/s41598-021-03882-9).
- [3] N. F. Hidayah, "Luas Kebakaran Hutan dan Lahan Indonesia Turun Pada 2022," GoodStats. Diakses: 20 Oktober 2023. [Daring]. Tersedia pada: <https://data.goodstats.id/statistic/Fitrinurhdyh/luas-kebakaran-hutan-dan-lahan-indonesia-turun-pada-2022-h5Qrs>
- [4] M. Rifandi, N. Yona, S. Munti, dan B. Setiawan, "Web-Based Programming of Geographic Information System Programming Of Broken Road Mapping In Kampar Regency," *Journal of Engineering Science and Technology Management*, vol. 1, no. 1, hlm. 28–34, Jul 2021, doi: <https://doi.org/10.31004/jestm.v1i1.13>.
- [5] T. Bin Tahir, Muh. Rais, dan Moch. Apriyadi HS, "Aplikasi Point OF Sales Menggunakan Framework Laravel," *JIKO (Jurnal Informatika dan Komputer)*, vol. 2, no. 2, hlm. 55–59, Okt 2019, doi: [10.33387/jiko.v2i2.1313](https://doi.org/10.33387/jiko.v2i2.1313).
- [6] A. D. Praba dan M. Safitri, "Studi Perbandingan Performansi Antara MySQL dan PostgreSQL," *Jurnal Khatulistiwa Informatika*, vol. 8, no. 2, Des 2020, doi: [10.31294/jki.v8i2.8851](https://doi.org/10.31294/jki.v8i2.8851).
- [7] Y. Y. Putra, O. Purwaningrum, dan R. H. Winata, "Perbandingan Performa Respon Waktu Kueri MySQL, PostgreSQL, dan MongoDB," *Jurnal Sistem Informasi dan Bisnis Cerdas*, vol. 15, no. 1, hlm. 39–48, Mar 2022, doi: [10.33005/sibc.v15i1.2749](https://doi.org/10.33005/sibc.v15i1.2749).
- [8] V. H. Pranatawijaya, W. Widiatry, R. Priskila, dan P. B. A. A. Putra, "Penerapan Skala Likert dan Skala Dikotomi Pada Kuesioner Online," *Jurnal Sains dan Informatika*, vol. 5, no. 2, hlm. 128–137, Des 2019, doi: [10.34128/jsi.v5i2.185](https://doi.org/10.34128/jsi.v5i2.185).
- [9] S. W. Ningrum, I. Akrunanda, dan A. R. Perdanakusuma, "Evaluasi dan Perbaikan Usability Aplikasi Mobile Ojesy Menggunakan Metode Usability Testing dan Use Questionnaire," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 3, no. 5, hlm. 4825–4834, Mei 2019, [Daring]. Tersedia pada: <http://j-ptiik.ub.ac.id>