



**RANCANG BANGUN SISTEM DETEKSI API
BERBASIS KAMERA DENGAN NOTIFIKASI *REAL-
TIME* MENGGUNAKAN METODE *DEEP LEARNING***

SKRIPSI

Mohamad Rafi Al-Yahya

2107411031

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
POLITEKNIK NEGERI JAKARTA**

2025



**RANCANG BANGUN SISTEM DETEKSI API
BERBASIS KAMERA DENGAN NOTIFIKASI *REAL-
TIME* MENGGUNAKAN METODE *DEEP LEARNING***

SKRIPSI

**Dibuat untuk Melengkapi Syarat-Syarat yang Diperlukan untuk Memperoleh
Diploma Empat Politeknik**

Mohamad Rafi Al-Yahya

2107411031

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
POLITEKNIK NEGERI JAKARTA**

2025



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

SURAT PERNYATAAN BEBAS PLAGIARISME

Saya yang bertanda tangan dibawah ini :

Nama : Mohamad Rafi Al-Yahya

NIM : 2107411031

Jurusan : Teknik Informatika dan Komputer / Teknik Informatika

Judul skripsi : RANCANG BANGUN SISTEM DETEKSI API BERBASIS KAMERA DENGAN NOTIFIKASI REAL-TIME MENGGUNAKAN METODE DEEP LEARNING

Menyatakan dengan sebenarnya bahwa skripsi ini benar-benar merupakan hasil karya saya sendiri, bebas dari peniruan terhadap karya dari orang lain. Kutipan pendapat dan tulisan orang lain ditunjuk sesuai dengan cara-cara penulisan karya ilmiah yang berlaku.

Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa dalam skripsi ini terkandung ciri-ciri plagiat dan bentuk-bentuk peniruan lain yang dianggap melanggar peraturan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Depok, 9 Juni 2025

Yang Membuat Pernyataan,



Mohamad Rafi Al-Yahya

NIM 2107411031



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

LEMBAR PENGESAHAN

Skripsi diajukan oleh:

Nama : Mohamad Rafi Al-Yahya
NIM : 2107411031
Program Studi : Teknik Informatika
Judul Skripsi : RANCANG BANGUN SISTEM DETEKSI API
BERBASIS KAMERA DENGAN NOTIFIKASI
REAL-TIME MENGGUNAKAN METODE DEEP
LEARNING

Telah diuji oleh tim penguji dalam Sidang Skripsi pada hari Selasa Tanggal 24,
Bulan Juni, Tahun 2025 dan dinyatakan **LULUS**.

Disahkan Oleh:

Pembimbing I : Maria Agustin, S.Kom., M.Kom.

(.....)

Penguji I : Dr. Dewi Yanti Liliana, S.Kom., M.Kom.

(.....)

Penguji II : Rizki Elisa Nalawati, S.T., M.T.

(.....)

Penguji III : Fachroni Arbi Murad, S.Kom., M.Kom.

(.....)

Mengetahui

Jurusan Teknik Informatika dan Komputer

Ketua



Dr. Anita Hidayati, S.Kom., M.Kom.

NIP. 197908032003122003



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Allah SWT atas segala rahmat, karunia, serta petunjuk-Nya, sehingga penulis dapat menyelesaikan penyusunan skripsi ini yang berjudul **“Rancang Bangun Sistem Deteksi Api Dini Berbasis Kamera dengan Notifikasi Real-Time Menggunakan Metode Deep Learning”** sebagai salah satu syarat untuk memperoleh gelar Sarjana Terapan pada Program Studi Teknik Informatika, Jurusan Teknik Informatika dan Komputer, Politeknik Negeri Jakarta.

Penyusunan skripsi ini tidak lepas dari bantuan, dukungan, bimbingan, dan doa dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Ibu Maria Agustin, selaku dosen pembimbing yang telah membantu penulis dari awal hingga akhir pengerjaan skripsi ini;
2. Kedua orang tua, yang telah memberikan dukungan selama masa perkuliahan penulis di Teknik Informatika Politeknik Negeri Jakarta, serta;
3. Seluruh teman-teman dan kerabat yang telah banyak membantu penulis dalam pengerjaan skripsi ini, baik secara langsung maupun tidak langsung.

Penulis berharap semoga Tuhan Yang Maha Esa mengaruniakan rahmat dan hidayah-Nya kepada mereka semua. Semoga skripsi ini dapat bermanfaat bagi kita semua.

Depok, 9 Juni 2025

Mohamad Rafi Al-Yahya



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

**SURAT PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK
KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Politeknik Negeri Jakarta, saya yang bertanda tangan dibawah ini:

Nama : Mohamad Rafi Al-Yahya
NIM : 2107411031
Jurusan/Program Studi : T. Informatika dan Komputer / Teknik Informatika

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Politeknik Negeri Jakarta Hak Bebas Royalti Non-Eksklusif atas karya ilmiah saya yang berjudul:

**RANCANG BANGUN SISTEM DETEKSI API BERBASIS KAMERA DENGAN
NOTIFIKASI REAL-TIME MENGGUNAKAN METODE DEEP LEARNING**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksklusif ini, Politeknik Negeri Jakarta berhak menyimpan, mengalihmediakan/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan skripsi saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik hak cipta.

Demikian pernyataan ini saya buat dengan sebenar-benarnya.

Depok, 17 Juli 2025

Yang Menyatakan,



Mohamad Rafi Al-Yahya

NIM 2107411031



Rancang Bangun Sistem Deteksi Api Berbasis Kamera Dengan Notifikasi Real-Time Menggunakan Metode Deep Learning

ABSTRAK

Kebakaran merupakan salah satu bencana yang dapat menimbulkan kerugian besar, baik dari segi materi, lingkungan, maupun keselamatan jiwa. Sistem deteksi kebakaran konvensional seperti pendeteksi asap dan suhu sering kali memiliki keterbatasan dalam mendeteksi api secara cepat karena hanya merespons setelah adanya gejala lanjutan seperti peningkatan suhu atau munculnya asap. Oleh karena itu, diperlukan sistem deteksi dini yang mampu mengenali api secara visual melalui kamera secara real-time dan memberikan notifikasi secara cepat kepada pengguna.

Penelitian ini bertujuan untuk merancang dan membangun sistem deteksi api dini berbasis kamera dengan menggunakan metode deep learning, khususnya model YOLO (You Only Look Once), yang mampu mendeteksi keberadaan api, asap, dan kondisi normal melalui citra video dari webcam. Sistem ini diintegrasikan dengan Firebase untuk pengiriman notifikasi real-time ke aplikasi Android yang telah dirancang khusus, serta menyimpan data log kejadian untuk keperluan pemantauan. Metodologi pengembangan perangkat lunak yang digunakan adalah model Waterfall yang terdiri dari tahap analisis, perancangan, implementasi, dan pengujian.

Hasil pengujian menunjukkan bahwa model mampu mendeteksi objek api dan asap dengan tingkat akurasi yang cukup baik, dan sistem secara keseluruhan dapat mengirim notifikasi ke aplikasi Android secara real-time ketika terjadi kebakaran. Dengan demikian, sistem ini diharapkan dapat membantu pengguna untuk melakukan tindakan pencegahan lebih cepat dan meminimalisasi risiko kebakaran.

Kata Kunci: Deteksi Api, Model YOLO, Notifikasi Real-Time

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



DAFTAR ISI

SURAT PERNYATAAN BEBAS PLAGIARISME	iii
LEMBAR PENGESAHAN	iv
KATA PENGANTAR.....	v
SURAT PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS	vi
ABSTRAK	vii
DAFTAR ISI.....	viii
DAFTAR TABEL	xi
DAFTAR GAMBAR.....	xii
BAB I.....	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan dan Manfaat	3
1.4.1 Tujuan	3
1.4.2 Manfaat	3
1.4.3 Sistematika Penulisan	3
BAB II	5
2.1 Landasan Teori.....	5
2.1.1 Deteksi Kebakaran	5
2.1.2 Deep Learning.....	5
2.1.3 Convolutional Neural Network (CNN).....	5
2.1.4 Algoritma YOLO (You Only Look Once).....	8
2.1.5 Python	8
2.1.6 Kotlin	9
2.1.7 SDLC (Software Development Life Cycle).....	9
2.1.8 Use Case Diagram.....	9
2.1.9 Metode Waterfall	9
2.1.10 AI Project Lifecycle	10
2.1.11 Black Box Testing.....	12
2.1.12 System Usability Scale (SUS).....	12
2.2 Penelitian Terdahulu	13
BAB III.....	15

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

3.1 Rancangan Penelitian.....	15
3.2 Tahapan Penelitian.....	15
3.2.1 Requirement Analysis	15
3.2.2 <i>Data Acquisition</i>	16
3.2.3 <i>Data Exploration</i>	16
3.2.4 <i>Modelling</i>	16
3.2.5 <i>Evaluation</i>	16
3.2.6 <i>Model Deployment</i>	16
3.2.7 <i>System Design</i>	16
3.2.8 <i>System Implementation</i>	17
3.2.9 <i>Unit Testing</i>	17
3.2.10 <i>Maintenance</i>	17
3.3 Objek Penelitian.....	18
3.4 Model yang Digunakan.....	18
3.5 Teknik Pengumpulan Data dan Analisis Data	18
BAB IV HASIL DAN PEMBAHASAN	19
4.1 Analisis Kebutuhan.....	19
4.1.1 Kebutuhan Fungsional	19
4.1.2 Kebutuhan Non-fungsional	19
4.1.3 Kebutuhan Perangkat Keras.....	20
4.1.4 Kebutuhan Perangkat Lunak.....	20
4.2 Perancangan Sistem	20
4.2.1 Admin Login.....	22
4.2.2 Deteksi api dan Notifikasi <i>Real-Time</i>	22
4.3 Implementasi Sistem.....	23
4.3.1 <i>Data Acquisition</i>	23
4.3.2 <i>Data Exploration</i>	24
4.3.3 <i>Modelling</i>	26
4.3.4 <i>Evaluation</i>	27
4.3.5 <i>Model Deployment</i>	36
4.3.6 <i>System Design</i>	37
4.3.7 <i>System Implementation</i>	38
4.4 Pengujian.....	42
4.4.1 <i>Black Box Testing</i>	43
4.4.2 System Usability Scale.....	45
4.5 <i>Maintenance</i>	48



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB V.....	49
5.1 Kesimpulan	49
5.2 Saran	49
DAFTAR PUSTAKA	50
DAFTAR RIWAYAT HIDUP PENULIS	52





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu	13
Tabel 4. 1 Pengujian <i>confidence threshold</i>	36
Tabel 4. 2 Skenario Pengujian <i>Black Box</i>	43
Tabel 4. 3 Hasil Pengujian Black Box	44
Tabel 4. 4 Kategori Penilaian	45
Tabel 4. 5 Pertanyaan SUS	46
Tabel 4. 6 Hasil Pengujian SUS (1)	46
Tabel 4. 7 Hasil Pengujian SUS (2)	47
Tabel 4. 8 Hasil Akhir Pengujian SUS	47

**POLITEKNIK
NEGERI
JAKARTA**



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR GAMBAR

Gambar 2. 1 Representasi Visual <i>Convolutional Layer</i>	6
Gambar 2. 2 Representasi visual lapisan <i>pooling</i>	7
Gambar 2. 3 Representasi visual <i>Activation Function</i>	7
Gambar 2. 4 Arsitektur YOLO	8
Gambar 2. 5 Kategori Penilaian SUS	12
Gambar 3. 1 Tahapan Penelitian	15
Gambar 4.1 <i>Use Case Diagram</i>	21
Gambar 4.2 <i>Activity Diagram Login</i>	22
Gambar 4.3 <i>Activity Diagram</i> Deteksi Api dan Notifikasi <i>Real-Time</i>	22
Gambar 4.4 Alur Proses Sistem	23
Gambar 4.5 Contoh Dataset Api	24
Gambar 4.6 Contoh Dataset Asap	24
Gambar 4. 7 <i>User Interface Label Studio</i>	25
Gambar 4. 8 Format Ekspor Dataset Label Studio	26
Gambar 4. 9 Situs Resmi Ultralytics	27
Gambar 4. 10 <i>Training Model YOLO</i>	27
Gambar 4. 11 <i>Evaluation Metrics YOLOv8</i>	28
Gambar 4. 12 <i>Confusion Matrix YOLOv8</i>	29
Gambar 4. 13 <i>Evaluation Metrics YOLOv9</i>	30
Gambar 4. 14 <i>Confusion Matrix YOLOv9</i>	31
Gambar 4. 15 <i>Evaluation Metrics YOLOv10</i>	32
Gambar 4. 16 <i>Confusion Matrix YOLOv10</i>	33
Gambar 4. 17 <i>Evaluation Metrics YOLOv11</i>	34
Gambar 4. 18 <i>Confusion Matrix YOLOv11</i>	34
Gambar 4. 19 Perbandingan Performa Model	35
Gambar 4. 20 Mockup User Interface Aplikasi Mobile	38
Gambar 4. 21 Halaman Login	39
Gambar 4. 22 Halaman Notifications	40
Gambar 4. 23 Halaman Activity Log (Umum)	41
Gambar 4. 24 Halaman Activity Log (Admin)	42



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB I PENDAHULUAN

1.1 Latar Belakang

Kebakaran merupakan salah satu bencana yang dapat menyebabkan kerugian besar, baik secara materiil, lingkungan, maupun keselamatan jiwa. Berdasarkan data dari Dinas Penanggulangan Kebakaran dan Penyelamatan Provinsi DKI Jakarta, frekuensi kebakaran di Jakarta mengalami peningkatan dari tahun 2021 hingga 2023. Penyebab kebakaran bervariasi, namun umumnya dipicu oleh keterlambatan dalam mendeteksi api sejak dini, sehingga proses pemadaman menjadi lebih sulit dan kerugian semakin besar.

Sistem deteksi kebakaran konvensional saat ini, seperti pendeteksi asap (*smoke detector*) dan pendeteksi panas (*heat detector*), memiliki keterbatasan. Sistem tersebut baru dapat bekerja setelah adanya peningkatan suhu atau keberadaan asap dalam jumlah tertentu, yang sering kali menyebabkan keterlambatan dalam merespons kebakaran. Oleh karena itu, dibutuhkan sistem deteksi kebakaran yang lebih cepat dan akurat, yang dapat melengkapi atau bahkan menggantikan sistem deteksi kebakaran konvensional.

Seiring dengan perkembangan teknologi kecerdasan buatan (*Artificial Intelligence*) dan pengolahan citra (*Computer Vision*), kini memungkinkan untuk mendeteksi api lebih awal menggunakan kamera dan metode *deep learning*. Dengan teknologi ini, sistem dapat menganalisis gambar atau video secara *real-time* untuk mendeteksi keberadaan api, tanpa harus menunggu asap atau panas meningkat. Selain itu, pengguna juga akan mendapatkan notifikasi secara langsung melalui ponsel, kapanpun dan dimanapun, selama terhubung ke jaringan internet. Hal ini tentu akan memungkinkan respons yang lebih cepat, sehingga upaya pencegahan dan mitigasi kebakaran dapat segera dilakukan sebelum api menyebar lebih luas.

Penelitian ini bertujuan untuk merancang dan membangun sistem deteksi api dini berbasis kamera menggunakan metode *deep learning*, yang mampu



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

mengidentifikasi keberadaan api secara otomatis dan memberikan notifikasi *real-time* kepada pengguna. Dengan adanya sistem ini, diharapkan dapat membantu mengurangi risiko kebakaran, mempercepat respons darurat, dan meningkatkan efektivitas sistem deteksi kebakaran modern.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan sebelumnya, maka beberapa masalah yang dapat dirumuskan pada penelitian ini adalah:

1. Bagaimana cara untuk membangun sistem deteksi api berbasis kamera dengan notifikasi *real-time* dengan menggunakan metode *deep learning*?
2. Algoritma apa yang akan digunakan untuk mendapatkan model *deep learning* dengan akurasi yang dapat diandalkan?
3. Bagaimana cara untuk memastikan keseluruhan sistem bekerja secara *real-time*, termasuk memberikan notifikasi ke perangkat pengguna?
4. Bagaimana cara untuk mengoptimalkan algoritma *deep learning* untuk mengenali pola visual dari api di berbagai kondisi lingkungan dengan faktor-faktornya (jarak api ke kamera, tingkat pencahayaan, dll)?

1.3 Batasan Masalah

Berdasarkan masalah-masalah yang telah dirumuskan sebelumnya, maka batasan masalah yang didapat yaitu:

1. Sistem ini adalah sistem deteksi api berbasis kamera dengan menganalisis citra. Sistem ini tidak mempertimbangkan faktor-faktor kebakaran lainnya seperti peningkatan suhu, keberadaan gas, dan lain sebagainya.
2. Notifikasi *real-time* yang diberikan adalah notifikasi kepada pengguna melalui perangkat *android*, dan bukan kepada pihak berwenang seperti pemadam kebakaran.
3. Sistem yang dirancang pada penelitian ini merupakan prototipe kecil dengan satu buah kamera, dan bukan mengintegrasikan beberapa kamera.
4. Dataset yang akan digunakan untuk pelatihan model *deep learning* bersumber dari gabungan dataset publik dan data tambahan yang didapat secara mandiri.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

1.4 Tujuan dan Manfaat

Berikut adalah tujuan dan manfaat dari pembuatan sistem deteksi api berbasis kamera ini.

1.4.1 Tujuan

Adapun tujuan dari penelitian ini, yaitu:

1. Merancang dan membangun sebuah sistem deteksi api berbasis kamera menggunakan metode *deep learning* yang memberikan notifikasi *real-time* ketika api muncul.
2. Mencari algoritma terbaik yang dapat digunakan untuk mendeteksi api.
3. Memberikan solusi sistem keamanan pelapis yang memungkinkan untuk mendeteksi api lebih cepat dibandingkan dengan *smoke detector* atau *heat detector*.

1.4.2 Manfaat

Adapun manfaat dari penelitian ini, yaitu:

1. Tersedianya sebuah sistem untuk digunakan dalam mendeteksi api dan asap sedini mungkin, sehingga mengurangi risiko terjadinya kebakaran dalam skala besar. Sistem ini dapat digunakan di ruangan-ruangan seperti kamar, ruang tamu, garasi, dan lain sebagainya.
2. Melengkapi sistem deteksi kebakaran yang sudah ada saat ini, sehingga api atau asap dapat dideteksi sesegera mungkin agar pengguna dapat melakukan tindakan setelah mendapatkan notifikasi pada perangkat seluler.
3. Apabila digunakan di area luar ruangan seperti perkebunan atau hutan, sistem ini berpotensi untuk turut berkontribusi menjaga ekosistem ketika akan terjadi kebakaran di area tersebut.
4. Penelitian ini berpotensi untuk membuka ruang bagi inovasi-inovasi lain di bidang ini kedepannya.

1.4.3 Sistematika Penulisan

Sistematika penulisan yang digunakan pada penelitian ini, yaitu:

1. BAB I



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Memaparkan tentang latar belakang, perumusan masalah, batasan masalah, tujuan dan manfaat, serta sistematika penulisan.

2. BAB II

Memaparkan tentang tinjauan pustaka yang relevan dan bersumber dari buku, jurnal, atau website yang dapat dipertanggungjawabkan kredibilitasnya.

3. BAB III

Memaparkan tentang metode penelitian, yang diuraikan secara rinci menjadi rancangan penelitian, tahapan penelitian, objek penelitian, model/framework/teknik yang digunakan, serta teknik pengumpulan dan analisis data.

4. BAB IV

Memaparkan tentang hasil dan pembahasan penelitian yang meliputi analisis kebutuhan, implementasi, pengujian serta hasil analisis pengujian.

5. BAB V

Memaparkan penutup dari penelitian berupa kesimpulan dan saran.

**POLITEKNIK
NEGERI
JAKARTA**



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

2.1.1 Deteksi Kebakaran

Deteksi kebakaran diperlukan segera setelah pertama kali api muncul. Saat ini, sudah ada beberapa alat yang dapat membantu proses deteksi tersebut, diantaranya adalah *smoke detector* dan *heat detector*. Kedua alat tersebut bekerja dengan cara mendeteksi adanya asap atau perubahan suhu yang terjadi di dalam ruangan. Oleh karena itu, kedua alat tersebut memiliki keterbatasan pada waktu yang diperlukan untuk mendeteksi adanya kebakaran, serta menjadi kurang efektif ketika digunakan pada area terbuka seperti perkebunan (Siregar et al., 2024).

2.1.2 Deep Learning

Deep Learning adalah kelas pembelajaran mesin yang berkinerja jauh lebih baik pada data tidak terstruktur. *Deep learning* memungkinkan model komputasi untuk mempelajari fitur secara progresif dari data di berbagai tingkatan. Popularitas *deep learning* semakin meningkat seiring dengan meningkatnya jumlah data yang tersedia serta kemajuan perangkat keras yang menyediakan komputer yang kuat (Mathew et al., 2021).

2.1.3 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan salah satu bidang di area *deep learning*. Hingga saat ini, sudah banyak aplikasi yang menggunakan CNN sebagai algoritmanya, termasuk namun tidak terbatas pada *computer vision* dan *natural language processing* (Li et al., 2021).

CNN merupakan sistem kecerdasan buatan yang terdiri dari beberapa lapisan jaringan syaraf yang dapat mengenal dan mengklasifikasikan objek dan juga melakukan segmentasi objek dalam sebuah citra.

Pada umumnya, sebuah model CNN terdiri atas 4 jenis lapisan, yaitu *Convolutional*, *Pooling*, *Activation Function*, dan *Fully-Connected*. (Taye, 2023).



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

1. Convolutional

Merupakan bagian krusial dari keseluruhan struktur CNN. Lapisan ini adalah sekelompok *filter* atau *kernel* yang diterapkan pada data masukan sebelum digunakan. Lebar, tinggi, dan bobot dari setiap lapisan *kernel* ini digunakan untuk mengekstrak karakteristik dari data masukan. Rumus perhitungan keluaran dari lapisan *convolutional* ini adalah sebagai berikut.

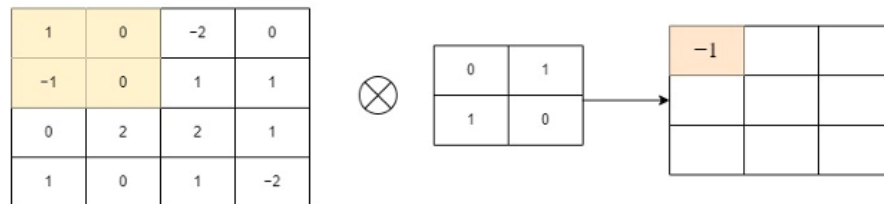
$$(K - L + 1) \quad (1)$$

Keterangan:

K = Input

L = Filter

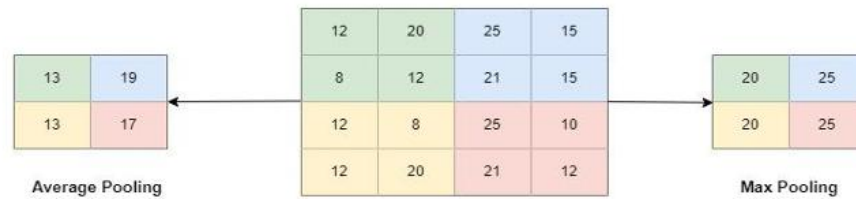
Sebagai contoh, ketika ada gambar masukan berukuran 4 x 4, dan diproses melalui *kernel* 2 x 2 dengan bobot acak tanpa *padding*, maka gambar keluaran dari lapisan *kernel* tersebut menjadi $(4 - 2 + 1) = 3 \times 3$. (Taye, 2023).



Gambar 2. 1 Representasi Visual *Convolutional Layer*

2. Pooling

Layer *pooling* atau yang biasa disebut juga dengan lapisan *down-sampling*, berperan untuk mengurangi dimensi dari peta fitur setelah melewati lapisan *convolutional*, namun tetap mendapatkan informasi-informasi yang penting. Ada 3 jenis lapisan *pooling*, yaitu max, min, dan average. Jenis *pooling* yang paling umum digunakan adalah *max pooling* dengan ukuran 2 x 2. (Taye, 2023).



Gambar 2. 2 Representasi visual lapisan *pooling*

3. *Activation Function*

Activation Function berfungsi untuk mengubah atau menghilangkan *output* yang dihasilkan dari lapisan-lapisan *convolution* dan *pooling* sebelumnya dan berperan untuk membatasi keluaran tersebut. Ada 3 jenis *activation function* yang lumrah digunakan dalam CNN, yaitu:

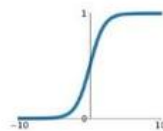
Sigmoid: hanya mengizinkan nilai *output* dengan bilangan diantara 0 dan 1, dan nilai *input* harus berupa bilangan real.

Tanh: Nilai *input* harus berupa bilangan real dan keluarannya harus bernilai 1.

ReLU: Semua nilai *input* dan *output* dikonversikan menjadi bilangan positif. Hal ini membuat waktu yang dibutuhkan untuk memproses menjadi lebih singkat dan sumber daya yang dibutuhkan lebih sedikit. (Taye, 2023).

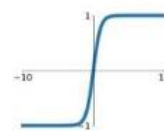
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



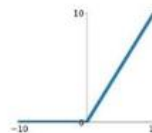
tanh

$$\tanh(x)$$



ReLU

$$\max(0, x)$$



Gambar 2. 3 Representasi visual *Activation Function*

4. *Fully-Connected*



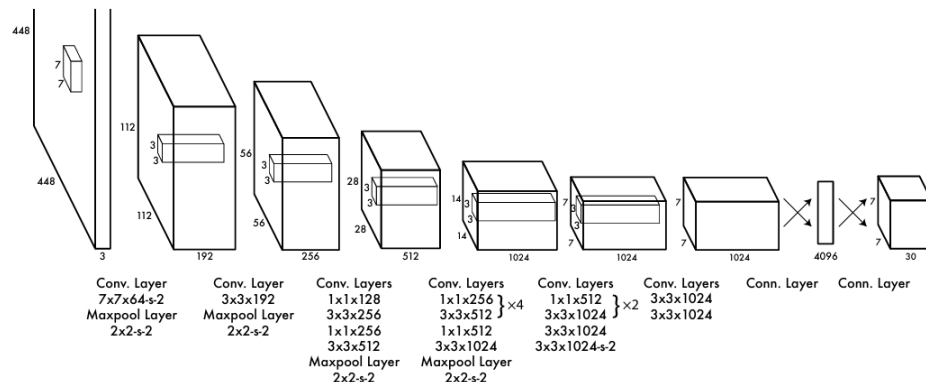
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Lapisan ini adalah lapisan tempat titik temu seluruh *neuron* keluaran dari lapisan-lapisan ekstraksi fitur sebelumnya. Agar dapat diproses oleh lapisan ini, seluruh data harus diubah menjadi satu dimensi terlebih dahulu.

2.1.4 Algoritma YOLO (You Only Look Once)

YOLO (*You Only Look Once*) merupakan sebuah algoritma *deep learning* yang menggunakan konsep *convolutional neural network* yang mempelajari pola-pola dari sebuah objek pada citra. Algoritma YOLO bersifat *pre-trained*, dan dapat bekerja secara *real-time* dengan tetap mempertahankan akurasi yang tinggi.



Gambar 2. 4 Arsitektur YOLO

Arsitektur dari algoritma YOLO sendiri memiliki 24 *convolutional layer* dan 2 *fully-connected layer*, dengan semua *input* citra akan diubah ukurannya menjadi 480 x 640. Fungsi *activation* pada YOLO menggunakan ReLu, kecuali pada *layer* terakhir yang menggunakan fungsi linear (Redmon, 2016).

2.1.5 Python

Python merupakan sebuah bahasa pemrograman *interpreted* yang memiliki beragam kegunaan, fleksibel, dan sangat mudah untuk dibaca. Bahasa pemrograman ini dapat digunakan baik untuk proyek skala kecil maupun besar. Meski memiliki fungsionalitas yang luas, bahasa pemrograman Python lebih sering dipergunakan dalam bidang *machine learning*, mengembangkan *neural network* untuk memproses data dalam skala besar. (Muis dan Muhammad, 2023).



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

2.1.6 Kotlin

Kotlin merupakan sebuah bahasa pemrograman yang pertama kali diperkenalkan oleh JetBrains pada tahun 2011. Meski pada awalnya Kotlin dikembangkan untuk pengembangan aplikasi Android, kini bahasa ini sudah mendukung untuk perangkat iOS, Web, dan aplikasi desktop. Kelebihan bahasa Kotlin ada pada keamanannya, kompatibilitasnya dengan bahasa pemrograman Java, serta dukungan perangkat lunak yang baik, sehingga membuat ini menempati posisi kedua sebagai bahasa pemrograman paling populer untuk pengembangan Android (Kothari et al., 2024).

2.1.7 SDLC (Software Development Life Cycle)

Software Development Life Cycle atau SDLC adalah sebuah siklus yang digunakan dalam pengembangan atau pembuatan perangkat lunak yang bertujuan untuk pemecahan masalah secara efisien dan efektif. SDLC berbentuk sebuah kerangka yang berisikan tahapan-tahapan pengembangan perangkat lunak agar hasil akhir dari proyek tersebut sesuai dengan ekspektasi pihak-pihak yang terlibat, baik pengguna maupun pengembang perangkat lunak itu sendiri (Fitriyanto et al., 2023).

2.1.8 Use Case Diagram

Use case adalah representasi berbentuk diagram dari fungsi-fungsi dalam suatu sistem yang dilihat dari sudut pandang pengguna. *Use case* menjelaskan apa saja yang dapat dilakukan oleh sistem beserta dengan seluruh komponennya. Dalam implementasinya, *use case* menggunakan skenario sebagai gambaran langkah-langkah yang menggambarkan interaksi antara pengguna dan sistem secara berurutan. Melalui *use case*, dapat diidentifikasi berbagai fitur atau kemampuan sistem, bagaimana pengguna berinteraksi dengannya, serta hubungan antara pengguna dengan setiap fungsi sistem yang tersedia (Setiyani, 2021).

2.1.9 Metode Waterfall

Metode *Waterfall* merupakan metode pengembangan perangkat lunak yang bersifat linear. Sesuai dengan namanya "*Waterfall*" yang berarti air terjun, metode ini mewajibkan untuk melakukan seluruh tahapan pengerjaan secara



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

berurutan. (Siregar, 2022). Tahapan-tahapan dalam metode ini meliputi *requirement, design, implementation, testing, dan maintenance*.

1. *Requirement Analysis*

Tahap pertama Dimana pengembang melakukan pengumpulan data, baik melalui wawancara, survei, atau diskusi dengan calon pengguna terkait sistem yang akan dibuat.

2. *Design*

Tahap *design* merupakan tahap dimana pengembang merangkum seluruh hasil pengumpulan data untuk kemudian dibuat desain awal arsitektur sistem yang sesuai dengan kebutuhan.

3. *Implementation*

Tahap *implementation* merupakan tahap dimana pihak pengembang mulai masuk ke pengerjaan teknis dari rancangan sistem yang telah didesain sebelumnya. Pembuatan kode program, komponen *hardware*, serta integrasi antar keduanya dilakukan pada tahapan ini.

4. *Testing*

Tahap *verification* merupakan tahapan pengujian dari sistem yang telah dibuat. Pengujian ini dilakukan agar memastikan bahwa sistem bekerja sesuai dengan kebutuhan pengguna.

5. *Maintenance*

Tahap *maintenance* seringkali disebut juga tahap pemeliharaan, karena pada tahapan ini, pada dasarnya adalah memastikan bahwa sistem sebelumnya yang telah fungsional sesuai dengan kebutuhan pengguna tetap bekerja sebagaimana mestinya kedepannya.

2.1.10 AI Project Lifecycle

AI Project Lifecycle adalah metode pengembangan sistem yang mengambil sudut pandang secara lebih spesifik ke bagian *Artificial Intelligence*. (Bandi &



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Kagitha, 2024). Pada dasarnya, *AI Project Lifecycle* memiliki beberapa tahapan dalam implementasinya, yaitu:

1. *Problem Scoping*

Pada tahapan ini, pengembang akan mengidentifikasi masalah yang akan dipecahkan melalui model yang akan dibuat, serta untuk siapa model ini dibuat.

2. *Data Acquisition*

Pada tahapan ini, pengumpulan data akan dilakukan. Pengembang dapat melakukan pengumpulan data melalui berbagai sumber, seperti repositori dataset publik, ataupun melakukan *data scraping* secara mandiri. Data yang terkumpul tadi akan membentuk sebuah *dataset* yang akan digunakan untuk pelatihan model di tahap-tahap selanjutnya.

3. *Data Exploration*

Pada tahapan ini, data-data yang telah dikumpulkan sebelumnya akan diolah agar siap digunakan untuk pelatihan model. Biasanya pada tahapan ini akan dilakukan anotasi data, augmentasi, ataupun *preprocessing* data lainnya.

4. *Modelling*

Pada tahapan ini, pengembang akan melakukan riset mengenai metode ataupun algoritma yang akan digunakan untuk model yang akan dibuat, serta kemudian akan melatih model menggunakan metode atau algoritma yang dipilih menggunakan *dataset* yang telah diolah sebelumnya.

5. *Evaluation*

Pada tahapan ini, pengembang akan mengevaluasi kinerja model yang telah dilatih menggunakan *dataset* yang ada. Parameter yang umum ditinjau pada evaluasi model diantaranya seperti *accuracy*, *precision*, *recall*, dan *F1-Score*.

6. *Deployment*

Pada tahapan ini, model yang telah siap digunakan akan diintegrasikan dengan platform dari sistem yang ingin digunakan, seperti web ataupun aplikasi.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

2.1.11 Black Box Testing

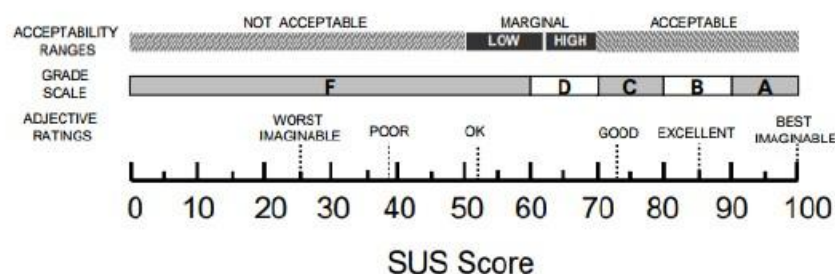
Black Box Testing adalah metode pengetesan yang berfokus pada fungsionalitas sistem sesuai dengan *requirement* awal tanpa perlu melihat kedalam baris kode program. Tujuannya adalah memastikan fitur-fitur yang ada pada sistem yang telah dibuat berjalan sebagaimana mestinya.

2.1.12 System Usability Scale (SUS)

System Usability Scale (SUS) adalah instrumen pengujian yang digunakan untuk mengukur seberapa layak sistem yang telah dibuat untuk dapat digunakan. *System Usability Scale* adalah pengujian berupa 10 pertanyaan yang diberikan kepada calon pengguna. Untuk masing-masing pertanyaan, SUS memiliki 5 jawaban, mulai dari “Sangat Tidak Setuju” dengan skor 1, hingga “Sangat Setuju” dengan skor 5. Dari 10 pertanyaan tersebut, nilai akhir masing-masing responden kemudian dihitung dengan ketentuan sebagai berikut:

1. Skor untuk pertanyaan dengan nomor ganjil (1, 3, 5, 7, 9) didapat dari bobot skor jawaban dikurang dengan 1.
2. Skor untuk pertanyaan dengan nomor genap (2, 4, 6, 8, 10) didapat dari 5 dikurang dengan bobot skor jawaban.
3. Skor akhir didapat dengan menjumlahkan hasil setiap pertanyaan dikalikan dengan 2,5.

Setelah skor untuk seluruh responden dihitung, selanjutnya adalah menentukan nilai akhir SUS dengan mencari skor rata-rata SUS, dengan menjumlahkan seluruh nilai responden dibagi dengan jumlah responden. Pengkategorian baik atau tidaknya nilai akhir SUS dapat dilihat pada Gambar 2.5.



Gambar 2. 5 Kategori Penilaian SUS



2.2 Penelitian Terdahulu

Berikut adalah beberapa penelitian terdahulu yang serupa:

Tabel 2. 1 Penelitian Terdahulu

Judul	Tahun Terbit	Algoritma	Hasil
Image-Based Outlet Fire Causing Classification Using CNN-Based Deep Learning Models (Lee et al.)	2024	YOLOv5, YOLOv6, YOLOv8, RetinaNet, SSD300, Faster-RCNN	Dari hasil membandingkan beberapa algoritma berbeda, YOLOv5 memiliki akurasi mAP@0,5 paling tinggi sebesar 89,1% dengan <i>inference time 22.5 frame per second</i> .
Fire Sensor and Surveillance Camera-Based GTCNN for Fire Detection System (Sridhar et al.)	2023	Gaussian Threshold Convolutional Neural Network (GTCNN)	Penelitian ini berhasil mendapatkan akurasi model GTCNN sebesar 98,23% akurat dalam mendeteksi keberadaan api, lebih baik 0,21% dari YOLOv5 dengan dataset yang sama.
Rancang Bangun <i>Quadcopter Drone</i> Untuk Deteksi Api Menggunakan YOLOv4 (Prasanta et al.)	2022	YOLOv4	Hasil penelitian ini menunjukkan <i>drone</i> berhasil mendeteksi api dan menghasilkan rata-rata akurasi 92%.
Deteksi api kebakaran berbasis	2022	YOLO	Hasil penelitian menunjukkan akurasi 80% di siang hari (latar terang),

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

computer vision dengan algoritma YOLO (Widharma et al.)			dan 96% di malam hari (latar gelap).
---	--	--	--------------------------------------



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB III

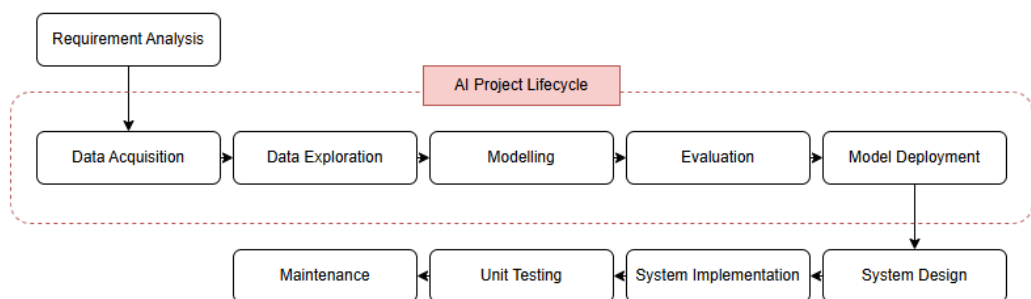
METODE PENELITIAN

3.1 Rancangan Penelitian

Penelitian ini berfokus pada pengembangan sistem deteksi api berbasis kamera menggunakan metode *deep learning* yang dapat berperan sebagai solusi sistem pendeteksi api yang dapat lebih cepat mendeteksi api ketimbang sistem yang sudah ada seperti *smoke detector* atau *heat detector*. Pendekatan yang digunakan adalah pendekatan kuantitatif, karena pengumpulan data dilakukan dengan menggunakan kuesioner yang opsi jawabannya sudah tersedia dan keluarannya adalah jumlah responden, juga salah satu fokus dari penelitian ini adalah akurasi akhir model dari produk ini.

3.2 Tahapan Penelitian

Tahapan penelitian merupakan serangkaian struktur langkah-langkah yang harus dilakukan guna mencapai tujuan penelitian yang telah ditetapkan. Secara umum, tahapan pada penelitian ini dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Tahapan Penelitian

3.2.1 Requirement Analysis

Tahap ini bertujuan untuk mendefinisikan permasalahan yang ada secara terperinci, sehingga dapat menjadi latar belakang penelitian yang akan dilakukan. Proses identifikasi masalah pada penelitian ini dilakukan dengan cara studi kasus kebakaran yang pernah terjadi, serta faktor-faktor yang menyebabkan kebakaran tersebut.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

3.2.2 Data Acquisition

Tahap ini bertujuan untuk mengumpulkan data-data yang diperlukan untuk mengembangkan model yang akan dibuat untuk memecahkan masalah yang telah diidentifikasi sebelumnya. Data didapatkan melalui repositori publik seperti *Kaggle*, serta melakukan *data scraping* secara mandiri.

3.2.3 Data Exploration

Tahapan ini bertujuan untuk mengolah data yang telah didapatkan pada tahapan sebelumnya agar menjadi *dataset* yang siap digunakan untuk pelatihan model. Pengolahan yang dimaksud biasanya mencakup anotasi dan *data cleaning*.

3.2.4 Modelling

Pada tahapan *modelling*, dilakukan pelatihan dan pengembangan model deteksi kebakaran berbasis citra, dengan *dataset* yang telah diolah sebelumnya digunakan sebagai data pelatihan model. Pada penelitian ini, akan mengomparasikan model YOLO dari berbagai versi dan mencari versi model YOLO yang terbaik untuk digunakan pada sistem akhir.

3.2.5 Evaluation

Pada tahapan ini, model yang sudah di *training* menggunakan *dataset* yang telah disiapkan akan diukur kinerjanya dengan *metric* yang ada. Model akan diukur kinerjanya berdasarkan *precision*, *recall*, *f1-score*, dan nilai mAP-nya (*mean average precision*).

3.2.6 Model Deployment

Setelah model selesai dilatih dan telah mendapatkan nilai-nilai *metrics* yang diharapkan, model sudah bisa digunakan dan akan disiapkan untuk digunakan pada sistem akhir. Merupakan tahapan terakhir dalam penelitian ini. Pada tahap ini, seluruh hasil dan dokumentasi penelitian dari awal hingga akhir, serta temuan-temuan penelitian akan dipaparkan dalam laporan akhir.

3.2.7 System Design

System Design adalah tahapan pertama dalam pengembangan aplikasi setelah model selesai dikembangkan. Pada tahapan ini, seluruh mekanisme teknis yang akan dibuat pada sistem divisualisasikan dalam bentuk diagram, seperti *activity*



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

diagram dan *use case diagram*. Proses pengembangan sistem akan mengacu kepada diagram-diagram yang sudah dibuat, dan memastikan setiap fitur yang telah dibuat pada rancangan dapat terimplementasikan.

3.2.8 System Implementation

Pada tahapan ini, dimulai pengembangan sistem dengan mengacu kepada rancangan yang telah dibuat pada langkah sebelumnya. Esensi dari tahapan ini adalah mentranslansikan rancangan yang sudah dibuat menjadi suatu kesatuan sistem yang memiliki *interface* agar dapat digunakan oleh aktor-aktor yang akan terlibat dalam sistem. Pada penelitian ini, sistem yang dibuat berupa aplikasi Android menggunakan bahasa pemrograman Kotlin. Aplikasi ini juga nantinya akan diintegrasikan dengan *Firestore Database* untuk penyimpanan data, *Firebase Cloud Messaging* untuk pengiriman notifikasi, serta model *deep learning* yang telah dikembangkan.

3.2.9 Unit Testing

Setelah seluruh sistem telah selesai dibuat dan diintegrasikan, pengujian atau *testing* perlu dilakukan guna memastikan bahwa sistem yang dibuat dapat bekerja sebagaimana mestinya. Dari sisi pengembang, *black box testing* akan dilakukan untuk memastikan seluruh fungsionalitas sistem dapat berjalan dengan baik, serta dapat mencari kemungkinan adanya *bug* pada sistem yang belum ditemukan. Selain dari sisi pengembang, sistem juga perlu diuji oleh calon pengguna dari sistem ini, dalam hal ini adalah khalayak umum. Setelah mencoba penggunaan sistem, calon pengguna nantinya akan memberi *feedback* berupa kuesioner *System Usability Scale*.

3.2.10 Maintenance

Maintenance merupakan tahapan terakhir setelah keseluruhan sistem selesai dikembangkan dan diimplementasikan. Dalam tahapan ini, dilakukan pemeliharaan dan optimasi secara berkala terhadap sistem guna memastikan fungsi-fungsi yang ada pada sistem tetap berjalan sebagaimana mestinya seiring berjalannya waktu, juga untuk meningkatkan kualitas sistem kedepannya, baik dari sisi aplikasi *mobile*, ataupun kemampuan *model* dalam melakukan klasifikasi asap dan api.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

3.3 Objek Penelitian

Objek penelitian ini adalah “Sistem Deteksi Api berbasis Kamera” yang produk akhirnya ditujukan agar dapat digunakan oleh pemilik-pemilik properti seperti rumah, apartemen, kantor, atau yang lainnya sebagai sistem keamanan pendeteksi kebakaran.

3.4 Model yang Digunakan

Model yang akan digunakan dalam skripsi ini adalah YOLO. Versi-versi dari model algoritma YOLO yang ada akan dibandingkan dengan *dataset* yang ada, dan versi YOLO yang memiliki performa keseluruhan terbaik akan digunakan di sistem produk akhir. Adapun alasan dari penggunaan model YOLO adalah karena kemampuannya untuk bekerja secara *real-time* dan dengan akurasi yang menjanjikan dan tetap mempertahankan kecepatan inferensinya.

3.5 Teknik Pengumpulan Data dan Analisis Data

Pengumpulan data pada penelitian ini menggunakan dua teknik berbeda. Yang pertama adalah melalui studi literatur di awal penelitian terkait sistem deteksi api, model CNN dan algoritma YOLO, serta sistem notifikasi *real-time*. Teknik kedua adalah mendapatkan data melalui kuesioner yang disebar ke calon pengguna, untuk mendapatkan *feedback* terkait kelayakan sistem untuk digunakan serta kemungkinan untuk merekomendasikan sistem ini kepada orang lain.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB IV HASIL DAN PEMBAHASAN

4.1 Analisis Kebutuhan

Analisis kebutuhan bertujuan untuk menentukan arah penelitian dan spesifikasi seperti apa yang dibutuhkan. Sistem deteksi api dengan menggunakan metode *deep-learning* ini dirancang untuk melakukan pendeteksian api sedini mungkin melalui citra dengan memberikan notifikasi secara *real-time* kepada penggunanya. Analisis kebutuhan ini dikelompokkan sebagai berikut:

4.1.1 Kebutuhan Fungsional

Kebutuhan fungsional mencakup fitur atau layanan yang harus dimiliki pada sistem ini agar berfungsi sebagaimana mestinya. Kebutuhan fungsional dari sistem ini, meliputi:

1. Deteksi Api menggunakan Kamera
2. Model *Deep Learning* Klasifikasi Api
3. Pengiriman Notifikasi secara *Real-Time*
4. *Activity Log* untuk menyimpan data pendeteksian api

4.1.2 Kebutuhan Non-fungsional

Kebutuhan non-fungsional adalah persyaratan yang berhubungan dengan kualitas atau performa sistem secara menyeluruh. Kebutuhan non-fungsional dari sistem ini, meliputi:

1. Performa: Sistem harus mampu mendeteksi api kurang dari 5 detik setelah citra api tertangkap oleh kamera, dan akurasi minimal dari model *deep learning* yang digunakan adalah 70%.
2. Dapat diandalkan: Sistem harus dapat mendeteksi api dalam kondisi lingkungan yang bermacam-macam, seperti Cahaya redup, berasap, dan lain-lain.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.1.3 Kebutuhan Perangkat Keras

Berikut adalah spesifikasi perangkat keras yang digunakan pada penelitian ini:

1. Kamera: Webcam dengan Resolusi 720p
2. Komputer
 - a. CPU: Intel Core i5-12450HX
 - b. GPU: Nvidia RTX 3050
3. Perangkat Seluler berbasis Android untuk menerima notifikasi.

4.1.4 Kebutuhan Perangkat Lunak

Berikut adalah perangkat lunak yang digunakan pada penelitian ini:

1. Sistem Operasi: Windows 11
2. Framework *Deep Learning*: PyTorch dan YOLO
3. Code Editor: Android Studio dan Visual Studio Code
4. Database: Firestore Database
5. API Notifikasi: Firebase Cloud Messaging

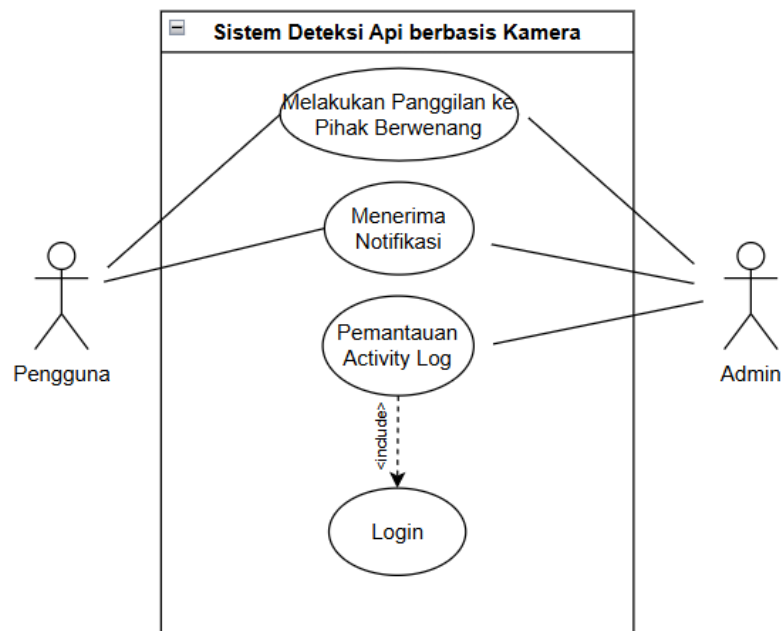
4.2 Perancangan Sistem

Sistem yang dirancang pada penelitian ini merupakan sistem pendeteksian api secara *real-time* dengan menggunakan satu buah kamera yang terpasang pada satu ruangan. Sistem ini akan memiliki dua jenis pengguna, yaitu pengguna reguler dan juga admin. Admin memiliki kemampuan yang sama dengan pengguna biasa, hanya saja admin memiliki kelebihan untuk dapat memantau *activity log*. Untuk visualisasi masing-masing pengguna dapat dilihat pada *use case* di Gambar 4.1.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4.1 Use Case Diagram

Use case diatas merupakan representasi berbentuk diagram dari fitur-fitur serta pengguna yang ada pada sistem (Setiyani, 2021). Pengguna pada umumnya dapat melakukan panggilan ke pihak berwenang, serta menerima notifikasi ketika terdeteksi adanya kebakaran. Admin dapat melakukan hal yang sama dengan pengguna pada umumnya, ditambah dengan izin untuk memantau *activity log*, yang untuk mengaksesnya harus *login* sebagai admin terlebih dahulu.

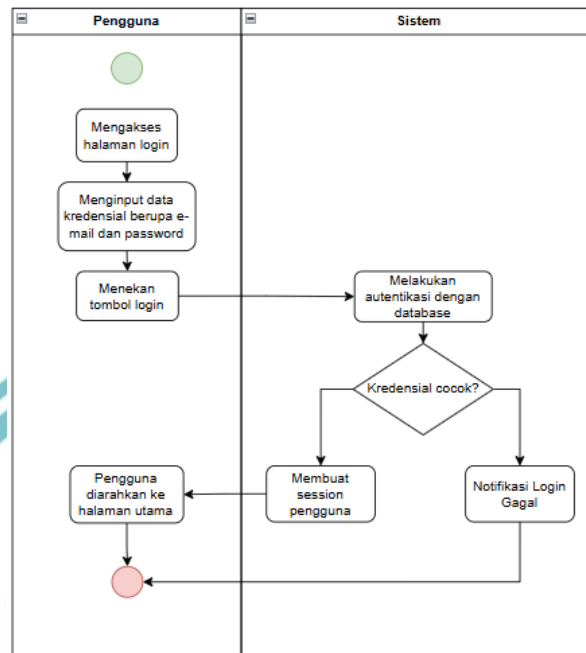
Adapun komponen arsitektur dari sistem ini meliputi:

1. Kamera, untuk mengambil gambar/video secara *real-time*.
2. Model *Deep Learning*, untuk memproses hasil gambar/video yang diambil.
3. Server, untuk menjalankan model *deep learning* agar dapat memproses data.
4. *Firestore Database*, untuk menyimpan *activity log* dan hasil analisis.
5. Aplikasi Mobile, untuk menerima notifikasi *real-time* dan memantau *activity log*.

Untuk visualisasi alur penggunaan fitur utama deteksi api serta fitur *login admin* dapat dilihat sebagai berikut.

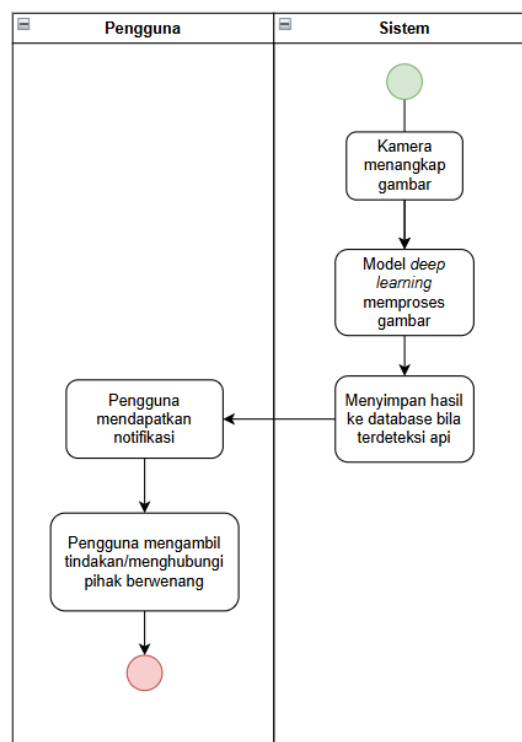


4.2.1 Admin Login



Gambar 4.2 Activity Diagram Login

4.2.2 Deteksi api dan Notifikasi *Real-Time*



Gambar 4.3 Activity Diagram Deteksi Api dan Notifikasi *Real-Time*

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

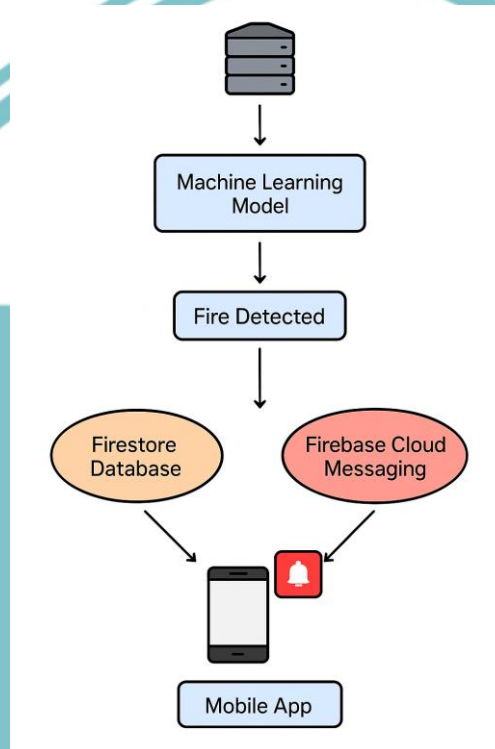


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.3 Implementasi Sistem

Implementasi dari sistem ini berpandu kepada rancangan tahapan penelitian yang telah dibuat pada poin 3.2, dengan terbagi menjadi dua bagian penting, yaitu pengembangan model *deep learning* dan pengembangan sistem aplikasi. Secara keseluruhan, alur proses cara kerja sistem ini dapat digambarkan seperti diagram yang dapat dilihat pada Gambar 4.3 berikut.



Gambar 4.4 Alur Proses Sistem

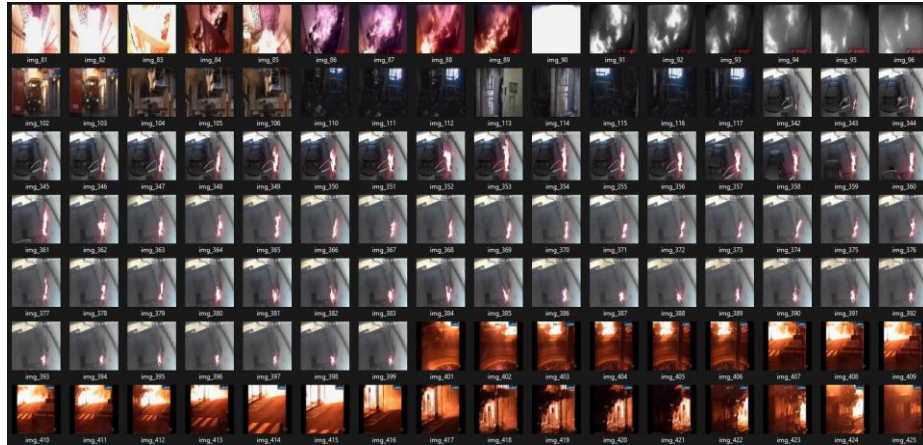
4.3.1 Data Acquisition

Sebagai persiapan untuk pelatihan model *deep learning* deteksi api, perlu adanya *dataset* sebagai bahan untuk pelatihan model. *Dataset* yang digunakan pada penelitian ini adalah data berbentuk citra api dan asap, yang sebagian datanya didapatkan dari repositori publik *Kaggle*, dan sebagian data lainnya diambil langsung oleh penulis. Contoh *dataset* yang digunakan untuk pelatihan model pada penelitian ini dapat dilihat pada Gambar 4.5 dan Gambar 4.6.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4.5 Contoh Dataset Api



Gambar 4.6 Contoh Dataset Asap

4.3.2 Data Exploration

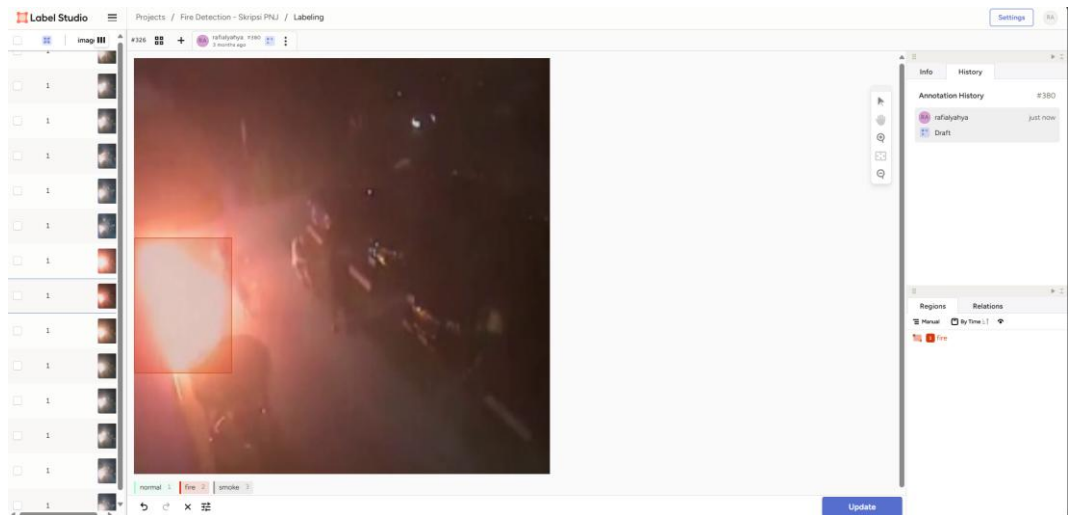
Setelah mendapatkan *dataset* yang diperlukan, tahapan selanjutnya adalah melakukan pengolahan terhadap data-data tersebut. Karena data yang sudah didapatkan masih berbentuk mentah, maka perlu dilakukan augmentasi data terlebih dahulu, agar satu data citra dapat memiliki lebih banyak varian dengan perbedaan seperti sudut kemiringan yang berbeda, ataupun citra yang di-mirror. Selain itu, dataset juga belum memiliki anotasi untuk kelas-kelas yang ingin digunakan, yaitu api dan asap. Oleh karena itu, perlu dilakukan proses *labelling* terlebih dahulu, agar saat pelatihan model dapat mengenali koordinat objek api dan asap yang ada pada citra. Untuk proses *labelling*, penulis menggunakan *Label Studio* dengan bentuk anotasi *Bounding Box*.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4. 7 User Interface Label Studio

Selain anotasi data, penulis juga melakukan *data cleaning* atau pembersihan data melalui *Label Studio*. Tujuan dilakukannya pembersihan data ini adalah untuk memastikan bahwa *dataset* yang akan digunakan bersih, dalam artian bebas dari kesalahan, dan relevan dengan permasalahan yang ingin dipecahkan melalui model yang akan dilatih, agar meningkatkan peluang model *deep learning* memiliki akurasi yang baik.

Setelah seluruh proses *labelling* dan *data cleaning* selesai dilakukan, *dataset* siap diekspor, lengkap dengan file .txt yang berisikan kelas beserta dengan koordinat anotasi pada masing-masing citra. Adapun format ekspor yang digunakan adalah format YOLO, sesuai dengan model yang akan digunakan pada penelitian ini, dapat dilihat seperti pada Gambar 4.7.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Export data



You can export dataset in one of the following formats:

JSON

☐ List of items in raw JSON format stored in one JSON file. Use to export both the data and the annotations for a dataset. It's Label Studio Common Format

JSON-MIN

☐ List of items where only "from_name", "to_name" values from the raw JSON format are exported. Use to export only the annotations for a dataset.

CSV

☐ Results are stored as comma-separated values with the column names specified by the values of the "from_name" and "to_name" fields.

TSV

☐ Results are stored in tab-separated tabular file with column names specified by "from_name" "to_name" values

COCO

☐ Popular machine learning format used by the COCO dataset for object detection and image segmentation tasks with polygons and rectangles. image segmentation object detection

Pascal VOC XML

☐ Popular XML format used for object detection and polygon image segmentation tasks. image segmentation object detection

YOLO

☒ Popular TXT format is created for each image file. Each txt file contains annotations for the corresponding image file, that is object class, object coordinates, height & width. image segmentation object detection

Gambar 4. 8 Format Ekspor Dataset Label Studio

4.3.3 Modelling

Setelah seluruh rangkaian proses *Data Exploration* selesai dilakukan, dan *dataset* sudah siap untuk digunakan, selanjutnya adalah proses utama pengembangan model. Sebelum melakukan *training* model, *dataset* yang sudah disiapkan akan dipecah menjadi dua bagian, yaitu data *train* dan data *validation*. Untuk rasio perbandingannya antar kedua bagian tersebut adalah 80% dari keseluruhan data untuk *train*, dan sisa 20% dari keseluruhan data untuk *validation*. Selain itu, pembagian data juga dilakukan secara acak, sehingga mengurangi kemungkinan data yang bersifat mirip untuk berada di salah satu bagian yang sama, sehingga dapat mengurangi bias dan model dapat melakukan generalisasi data dengan baik.

Setelah proses pembagian data selesai, selanjutnya adalah melakukan pelatihan model. Dalam penelitian ini, ada beberapa versi YOLO yang digunakan sebagai bahan komparasi, yaitu YOLOv8, YOLOv9, YOLOv10, dan YOLOv11. Untuk seluruh versi YOLO yang digunakan, jenis model yang digunakan adalah YOLO berukuran *medium* (YOLOvXm). Alasan penggunaan model YOLO berukuran *medium* ini adalah dengan mempertimbangkan rasio antara performa



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang menggunakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

model dengan kecepatan inferensi model yang paling setimpal dibanding dengan ukuran-ukuran lainnya. Ukuran model juga mempengaruhi waktu *training* yang diperlukan. Seluruh model YOLO yang digunakan didapatkan dari situs resmi *Ultralytics* sebagai penyedia model YOLO.

Model	Input Size	AP _{val}	FLOPs (G)	Latency (ms)
YOLOv10n	640	38.5	6.7	1.84
YOLOv10s	640	46.3	21.6	2.49
YOLOv10m	640	51.1	59.1	4.74
YOLOv10b	640	52.5	92.0	5.74
YOLOv10l	640	53.2	120.3	7.28
YOLOv10x	640	54.4	160.4	10.70

Gambar 4. 9 Situs Resmi Ultralytics

Setelah mengunduh file YOLO yang dibutuhkan, selanjutnya adalah melakukan pelatihan dengan menggunakan masing-masing versi YOLO. Seluruh pelatihan menggunakan parameter yang sama, yaitu *imgsz* (ukuran *input* citra) = 640, *batch* = 8, *epochs* = 50, dan *device* = 0 (penggunaan *discrete graphics card* untuk pelatihan model).

```
from ultralytics import YOLO

if __name__ == '__main__':
    model = YOLO("yolov11/yolo12m.pt")
    model.train(data = "yolov11/dataset.yaml", imgsz = 640, batch = 8, epochs = 50, device = 0, name="fire-detection")
```

Gambar 4. 10 Training Model YOLO

4.3.4 Evaluation

Setelah model selesai melalui proses pelatihan menggunakan *dataset* dan parameter yang serupa, tahapan selanjutnya adalah *evaluation*, yaitu mengevaluasi kinerja model yang dilatih dengan memperhatikan *evaluation metrics* yang ada.

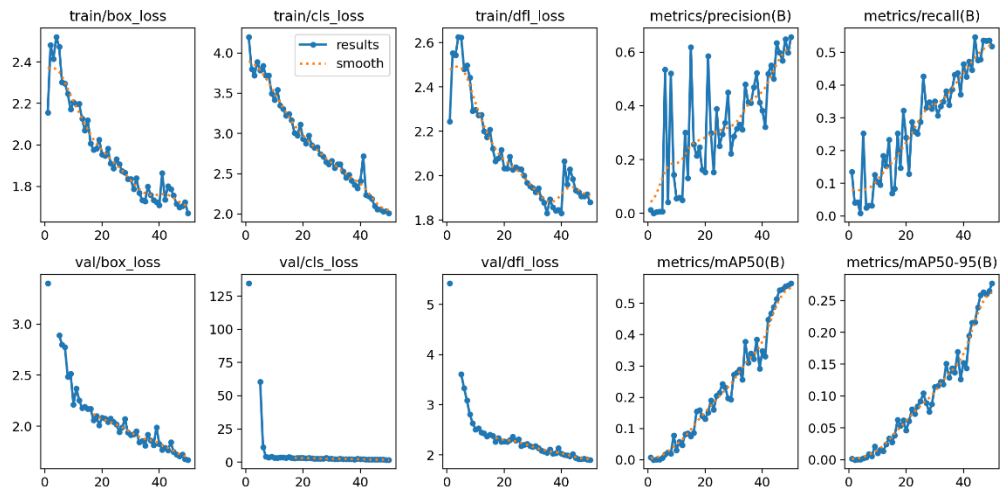
1. YOLOv8

YOLOv8 merupakan versi YOLO yang paling terdahulu yang diujikan pada penelitian ini. Dengan menggunakan parameter yang serupa, model YOLOv8 memiliki nilai *metrics* evaluasi sebagai berikut.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4.11 *Evaluation Metrics YOLOv8*

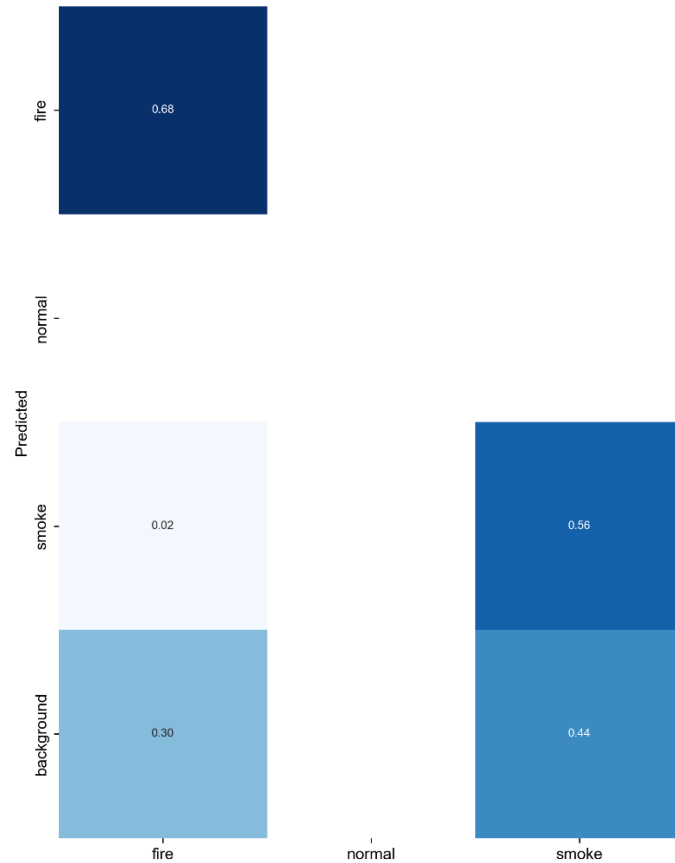
Seperti yang dapat dilihat pada Gambar 4.10, model YOLOv8m mengalami peningkatan yang cukup konsisten dari setiap iterasinya, mulai dari *loss* yang semakin mengalami penurunan hingga *epoch* ke-50, serta *recall* dan *precision* yang secara garis besar mengalami peningkatan hingga *epoch* ke-50. Meski begitu, nilai *mAP50* (*Mean Average Precision* dengan *threshold* persimpangan atas gabungan di 0,50) masih cukup rendah, di sekitar 0,58, yang sudah mulai mengalami stagnansi pada *epoch* ke-42 dan tidak mengalami peningkatan yang signifikan hingga *epoch* ke-50.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4. 12 *Confusion Matrix* YOLOv8

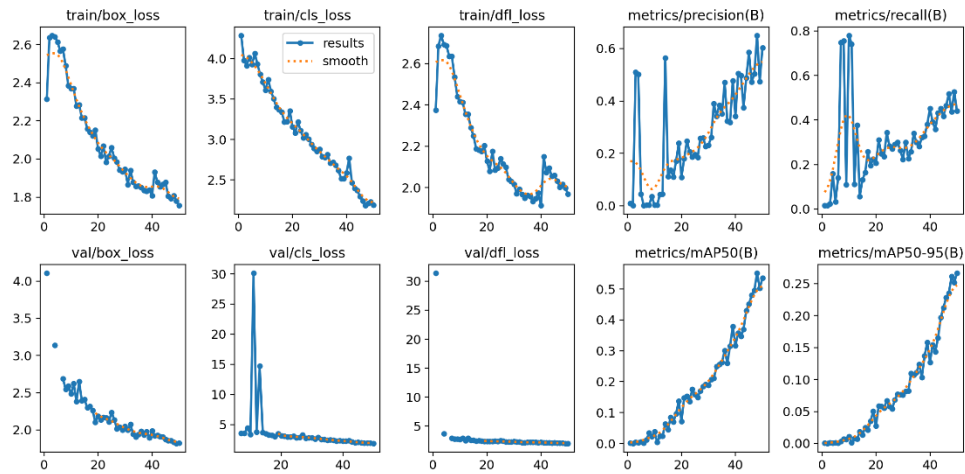
Confusion Matrix pada Gambar 4.11 merupakan representasi visual dari pendeteksian masing-masing kelas. Dapat dilihat bahwa dari 100% citra api, 68% yang terklasifikasi sebagai api (*true positive*), 2% terklasifikasi sebagai asap, dan model tidak dapat mengklasifikasikan objek pada 30% citra api lainnya (terklasifikasi sebagai *background*). Kemudian dari 100% citra asap, 56% citra terklasifikasi sebagai asap (*true positive*), dan 44% citra lainnya terklasifikasi sebagai *background*. Kesimpulannya adalah model sudah dapat membedakan ciri-ciri visual antara api dan asap, namun terkadang masih cukup kesulitan menangkap kedua objek tersebut, dibuktikan dari 30% citra api dan 44% citra asap yang masih tidak terklasifikasi dengan baik.

Pada praktiknya, model YOLOv8 sudah dapat mendeteksi kedua kelas, yaitu api dan asap meski belum konsisten, dan masih sensitif terhadap cahaya atau objek-objek lain yang menyerupai api dan asap.



2. YOLOv9

Pada penelitian ini, model YOLOv9 memiliki performa yang serupa dengan pendahulunya, bahkan mengalami sedikit penurunan. Dengan menggunakan parameter yang serupa, model YOLOv9 memiliki *metrics* evaluasi sebagai berikut.



Gambar 4. 13 *Evaluation Metrics* YOLOv9

Seperti yang dapat dilihat pada *evaluation metrics* di Gambar 4.12, YOLOv9 juga mengalami peningkatan yang konsisten di setiap iterasinya, dengan *loss* yang konsisten mengalami penurunan, serta *precision* dan *recall* yang meningkat hingga iterasi terakhir. Meski begitu, hingga *epoch* ke-50, nilai *mAP50* tertinggi hanya berada di angka 0,55, yang berarti lebih rendah dibanding dengan YOLOv8 sebelumnya.

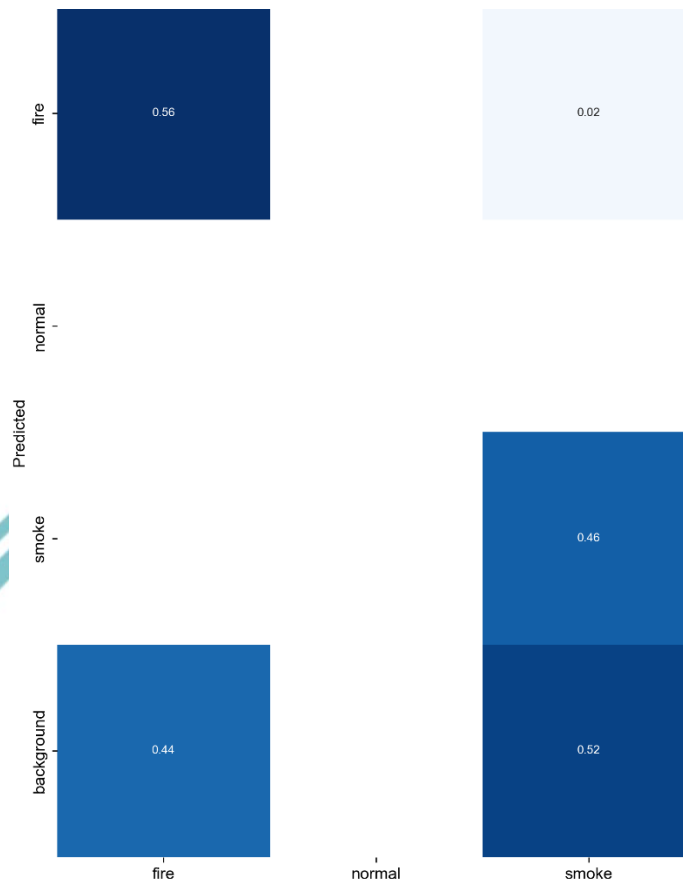
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4. 14 *Confusion Matrix* YOLOv9

Dapat dilihat pada *Confusion Matrix* di Gambar 4.13, bahwa pendeteksian masing-masing kelas, baik api maupun asap, masih kurang memuaskan. Dari 100% citra api, hanya 56% yang benar-benar terklasifikasi sebagai api (*true positive*), sementara 44% lainnya tidak terklasifikasi sebagai api maupun asap (*background*). Di sisi lain, dari 100% citra asap, hanya 46% yang benar-benar terklasifikasi sebagai asap, 2% terklasifikasi sebagai api, dan sisa 52% lainnya tidak terklasifikasi sebagai api maupun asap. Hal ini menunjukkan masalah yang serupa yang ditemukan juga pada YOLOv8, yaitu model masih kesulitan untuk menangkap citra api ataupun asap, terutama pada data citra yang objek api atau asapnya kurang jelas.

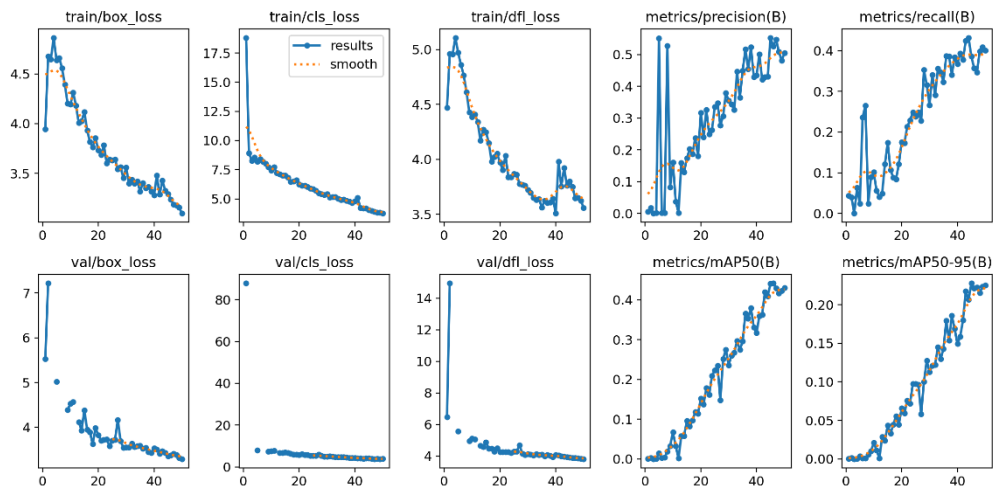
Pada praktiknya, model YOLOv9 juga masih mampu untuk mendeteksi api ataupun asap, namun performanya tidak lebih baik dari model YOLOv8 dalam kasus ini.



3. YOLOv10

Diantara seluruh model yang diujikan pada penelitian ini, YOLOv10 memiliki performa yang paling rendah dibandingkan dengan versi-versi lainnya.

Evaluation Metrics dari YOLOv10 dapat dilihat pada Gambar 4.14



Gambar 4. 15 *Evaluation Metrics* YOLOv10

Seperti yang dapat dilihat, meskipun value *metrics* yang dihasilkan model ini semakin meningkat pada setiap iterasi, hingga *epoch* ke-50 model ini masih memiliki *loss* yang cukup tinggi, ditambah dengan *precision* dan *recall* yang masih cukup rendah disekitar 0,55 dan 0,47, serta nilai mAP50 yang berada di 0,45.

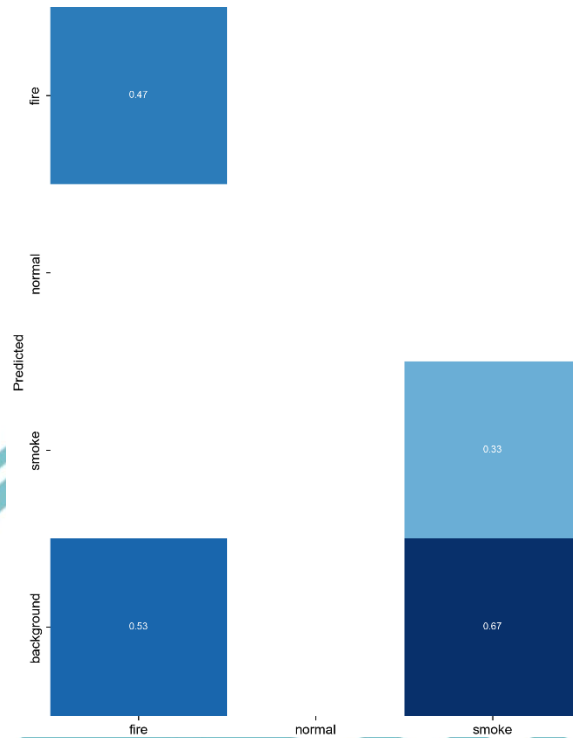
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4. 16 *Confusion Matrix* YOLOv10

Pada *confusion matrix* dari YOLOv10, dapat dilihat bahwa klasifikasi masing-masing kelas masih belum memuaskan. Dari 100% citra api, hanya 47% yang terklasifikasi sebagai api, dan sisa 53% lainnya tidak terklasifikasi sebagai api ataupun asap (*background*), dan dari 100% citra asap, hanya 33% yang terklasifikasi sebagai asap, dan sisa 67% lainnya tidak terklasifikasi sebagai asap ataupun api. Hal ini menunjukkan bahwa model belum dapat mendeteksi baik api maupun asap dengan baik pada sebuah citra, meski sudah dapat membedakan ciri-ciri visual diantara keduanya, ditandai dengan tidak adanya *false positive* ataupun *false negative* pada model ini.

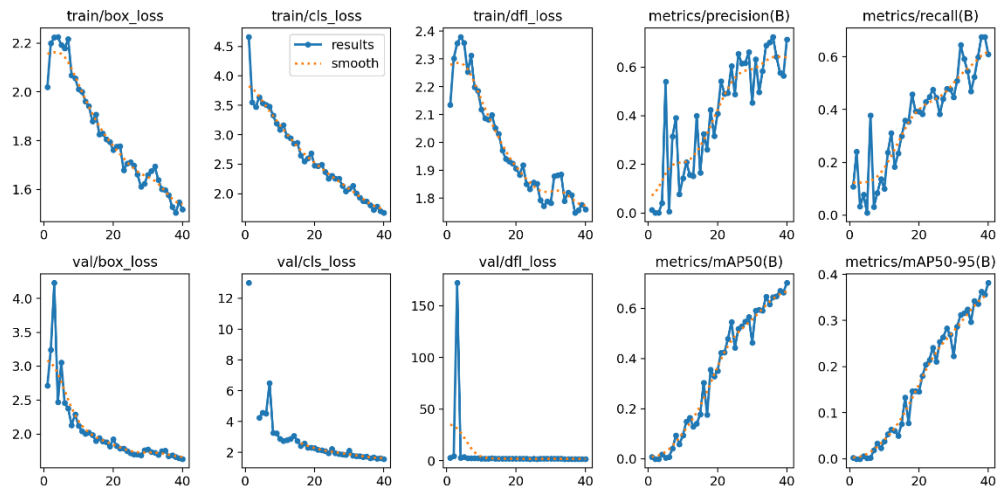
4. YOLOv11

Dari seluruh model yang diujikan pada penelitian ini, YOLOv11 memiliki performa yang paling memuaskan dibanding dengan versi-versi pendahulunya. Masih dengan parameter yang serupa, model YOLOv11 memiliki *evaluation metrics* sebagai berikut.



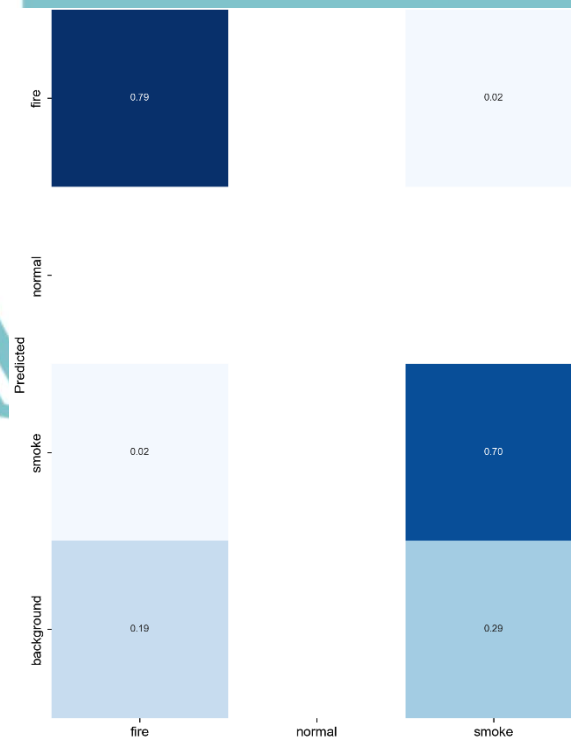
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4. 17 *Evaluation Metrics* YOLOv11

Seperti yang dapat dilihat pada Gambar 4.14, YOLOv11 juga konsisten mengalami peningkatan di setiap iterasinya, nilai *loss* yang terus menurun dan *precision* serta *recall* yang konsisten meningkat. Meski begitu, YOLOv11 memiliki nilai *mAP50* yang tertinggi dari model YOLO versi lain, yang berada di sekitar 0,70.



Gambar 4. 18 *Confusion Matrix* YOLOv11



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Dapat dilihat pada *Confusion Matrix* di Gambar 4.15, klasifikasi masing-masing kelas sudah lebih baik ketimbang YOLO versi-versi pendahulunya. Untuk kelas api, dari 100% citra, 79% terklasifikasi benar sebagai api (*true positive*), 2% terklasifikasi sebagai asap, sementara 19% sisanya tidak terklasifikasi sebagai asap maupun api (*background*). Untuk kelas asap, dari 100% citra, 70% terklasifikasi benar sebagai asap, 2% terklasifikasi sebagai api, dan sisa 28% tidak terklasifikasi sebagai asap maupun api. Hal ini menunjukkan bahwa model sudah dapat mengenali objek kedua kelas, baik asap maupun api, dan dapat membedakan diantara keduanya dengan cukup baik. Meski begitu, model ini masih memiliki kekurangan, seperti masih ada data yang tidak terklasifikasi baik di kelas api maupun asap.

Dalam praktiknya, model ini memiliki performa dan konsistensi yang paling baik diantara model-model YOLO lainnya yang diujikan pada penelitian ini. Model sudah dapat mendeteksi api atau asap secara konsisten, meski terkadang masih salah melakukan pendeteksian, seperti tidak ada objek api maupun asap, tetapi model mendeteksi adanya objek-objek tersebut.



Gambar 4. 19 Perbandingan Performa Model



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Dari keempat versi model YOLO yang diujikan, dapat dilihat bahwa versi YOLO yang memiliki performa paling baik dan paling seimbang adalah YOLOv11. Dengan nilai mAP50 yang berada di angka 0,70, *precision* di angka 0,68, dan *recall* di angka 0,66, menunjukkan bahwa YOLOv11 memiliki angka metrik evaluasi yang paling tinggi dibandingkan dengan versi-versi YOLO lainnya yang diujikan pada penelitian ini. Pada praktiknya pula, YOLOv11 ini memiliki kemampuan yang paling baik dalam melakukan klasifikasi baik api maupun asap, serta merupakan yang paling tidak sensitif terhadap cahaya.

4.3.5 Model Deployment

Setelah mengevaluasi seluruh model yang diuji, selanjutnya adalah melakukan persiapan agar model siap digunakan untuk sistem. Setelah didapati bahwa model yang memiliki performa terbaik adalah YOLOv11, maka model tersebutlah yang akan digunakan pada sistem akhir. Persiapan yang dilakukan adalah memprogram agar ketika model mendeteksi api sebanyak 70% dari total *frame* selama 5 detik berturut-turut, model akan mengirimkan *timestamp* beserta dengan tangkapan gambar ke *firestore database*, serta mengirim pesan notifikasi ke aplikasi mobile dengan perantara *Firestore Cloud Messaging*. Waktu 5 detik dan 70% *frame* digunakan sebagai *threshold* minimal pendeteksian api dan asap karena dianggap akurat dalam membedakan api bahaya yang dapat menyebabkan kebakaran dan yang tidak (Zaidi et al., 2024). Nilai *threshold confidence* juga ditambahkan untuk memastikan model mendeteksi api dan asap dengan benar dan meminimalisir sensitifitas model terhadap cahaya disekitar. Nilai *threshold* ini ditentukan melalui pengujian, dan data yang didapatkan dapat dilihat sebagai berikut.

Tabel 4. 1 Pengujian *confidence threshold*

Threshold	Performa
0,4	Api dan asap seringkali tidak dapat terdeteksi, besar kemungkinan disebabkan oleh akurasi model yang belum cukup baik.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

0,3	Api dan asap dapat terdeteksi dengan baik, sensitifitas terhadap cahaya intensitas rendah minim, namun masih sensitif terhadap cahaya intensitas tinggi.
0,2	Api dan asap dapat terdeteksi dengan baik, namun model masih cukup sensitif terhadap cahaya yang tertangkap oleh kamera

Berdasarkan hasil pengujian nilai *confidence threshold*, hasil yang paling seimbang antara kemampuan pendeteksian api dan asap dengan sensitifitas terhadap cahaya yang paling rendah adalah ketika menggunakan *threshold* dengan nilai 0,3. Sebagai tambahan untuk kenyamanan penggunaan sistem ini, model juga akan diberi waktu *cooldown* selama 1 menit agar model tidak mengirimkan notifikasi untuk api yang sama secara terus-menerus.

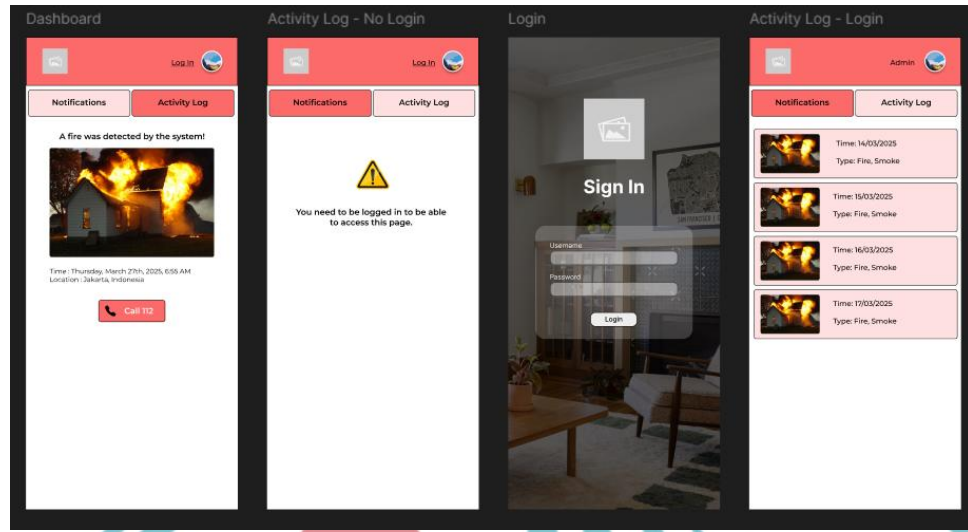
4.3.6 System Design

Tahap *System Design* merupakan tahap pertama untuk bagian pengembangan aplikasi. Untuk mempermudah proses pengembangan aplikasi *mobile*, terutama pada sektor *User Interface (UI)*, pembuatan *mockup* dilakukan sebelum proses implementasi. Tujuannya adalah agar memiliki acuan saat melakukan proses implementasi sehingga pengembangan lebih terarah. Aplikasi ini memiliki tiga jenis halaman utama, yaitu halaman *login*, halaman utama (notifikasi), serta halaman *activity log*.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4. 20 Mockup User Interface Aplikasi Mobile

Halaman *login* dirancang untuk *admin*, di desain secara minimalis dengan kolom input *e-mail* dan *password* beserta tombol *login*. Halaman *notifications* merupakan halaman utama, dan pengguna akan langsung diarahkan ke halaman ini ketika pertama kali membuka aplikasi. Halaman notifikasi ini tersusun atas *top bar* yang berisi logo aplikasi, beserta tombol untuk mengakses halaman *login*, tombol navigasi untuk berpindah halaman antara halaman notifikasi dan halaman *activity log*, notifikasi terkini dari pendeteksian kebakaran, lengkap dengan detail tanggal, waktu, dan gambar dari kejadian, serta tombol untuk melakukan panggilan darurat. Susunan untuk halaman *activity log* kurang lebih mirip dengan halaman notifikasi, dengan *top bar* dan juga tombol navigasi yang sama, yang membedakan adalah isi di bagian utama halamannya. Ketika pengguna memiliki *session admin* yang aktif, dengan kata lain pengguna telah melakukan *login* sebagai *admin*, maka halaman *activity log* akan berisikan riwayat pendeteksian kebakaran, yang terdiri atas gambar dan waktu terdeteksinya api. Jika tidak ada *session* yang aktif, dengan kata lain pengguna tidak *login* sebagai *admin*, maka halaman ini hanya akan berisikan peringatan yang menyatakan bahwa pengguna harus *login* untuk dapat mengakses halaman ini.

4.3.7 System Implementation

Setelah melakukan perancangan sistem, tahap selanjutnya adalah melakukan implementasi. Di tahap ini, seluruh rancangan yang telah dibuat akan dibuat

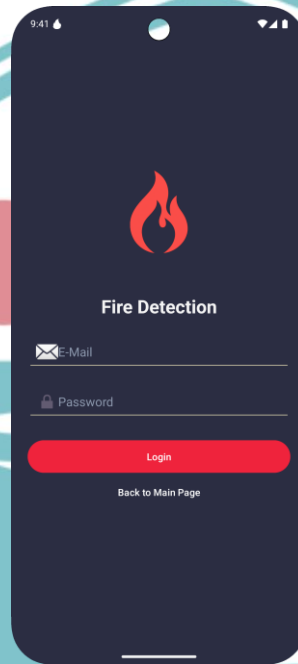


Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

menjadi sebuah sistem yang nyata dan dapat digunakan oleh pengguna. *System Implementation* meliputi dua bagian penting, yaitu aplikasi *mobile* dan model *deep learning*, yang keduanya dijumpai oleh *Firebase* dalam proses transaksi data dan pengiriman notifikasi secara *real-time*.

1. Halaman *Login*



Gambar 4. 21 Halaman Login

Halaman *login* dikhususkan bagi admin yang ingin masuk dan mengakses *activity log*, dengan memasukkan *e-mail* dan *password* yang terdaftar. Ada juga tombol *login* yang ketika ditekan dan *e-mail* serta *password* yang dimasukkan cocok, maka aplikasi akan membuat *session* dan akan masuk sebagai admin.

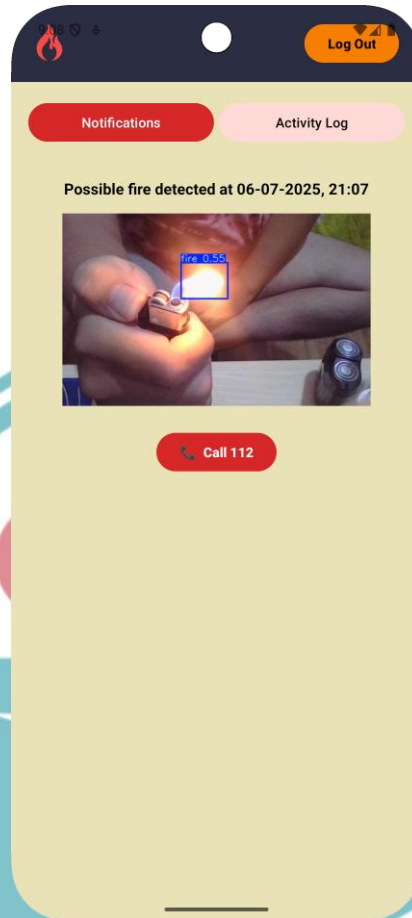
2. Halaman Utama (*Notifications*)



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4. 22 Halaman Notifications

Halaman *notifications* berisikan pendeteksian terkini, lengkap beserta tanggal dan waktu pendeteksian, serta tangkapan kamera ketika mendeteksi api. Selain itu, ada juga tombol untuk melakukan panggilan darurat, apabila pengguna membutuhkannya.

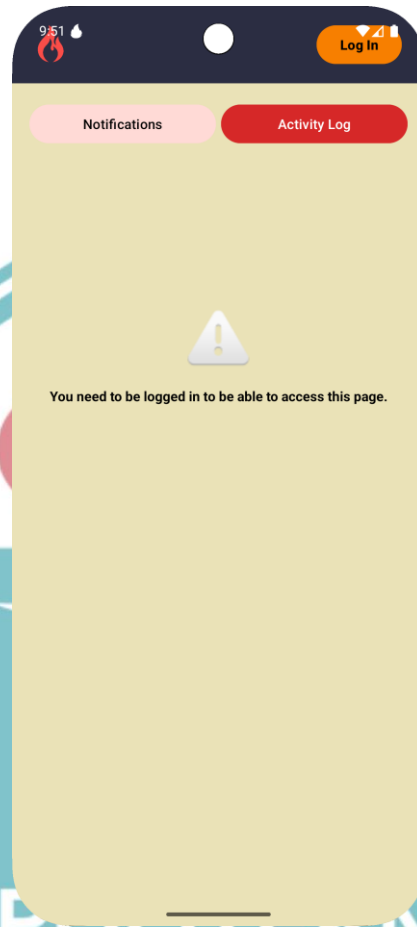


© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

3. Halaman *Activity Log* (Umum)



Gambar 4. 23 Halaman Activity Log (Umum)

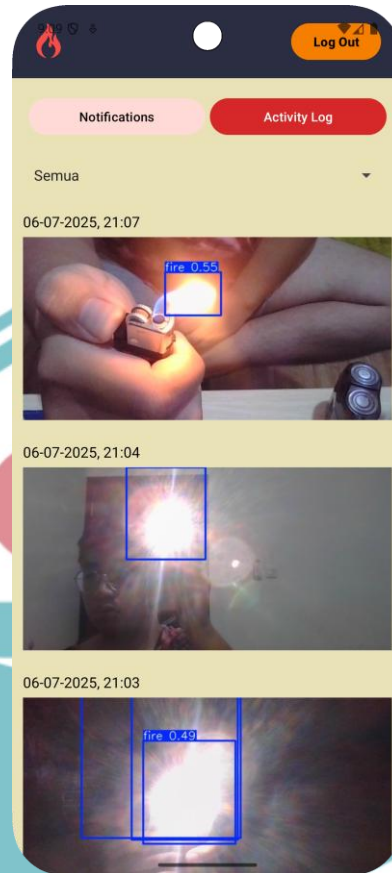
Halaman *activity log* adalah halaman yang hanya dapat diakses ketika pengguna masuk sebagai admin. Jika tidak, maka ketika ingin membuka halaman ini, hanya akan berisikan peringatan bahwa pengguna harus masuk untuk dapat mengakses halaman ini, seperti yang dapat dilihat pada Gambar 4.19.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4. Halaman *Activity Log* (Admin)



Gambar 4. 24 Halaman Activity Log (Admin)

Halaman *activity log* akan berubah menjadi daftar riwayat pendeteksian api ketika pengguna telah masuk sebagai admin. Riwayat pendeteksian juga dilengkapi dengan tanggal, waktu, serta gambar ketika api terdeteksi. Terdapat *filter* waktu untuk memudahkan pengguna mencari Riwayat pendeteksian, dan opsi yang tersedia adalah 7 hari terakhir, 30 hari terakhir, dan tampilkan semua. Detail untuk masing-masing riwayat pendeteksian juga dapat dilihat dengan menekan *item* yang diinginkan.

4.4 Pengujian

Pada penelitian ini, ada tiga instrumen pengujian yang akan digunakan, yaitu *Black Box Testing*, *System Usability Scale (SUS)* dan *Net Promoter Score (NPS)*. Tujuan dari SUS sendiri adalah untuk menentukan seberapa layak dan berguna sistem ini untuk digunakan oleh calon pengguna nantinya, dengan pengukuran secara kuantitatif yang datanya diambil melalui pengisian



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

kuesioner. Sementara untuk NPS, tujuannya adalah untuk mengukur tingkat loyalitas dan kepuasan pengguna, dilihat dari besarnya kemungkinan merekomendasikan produk yang diuji ke orang lain.

4.4.1 Black Box Testing

Black Box Testing, adalah pengujian untuk menguji fungsionalitas sistem tanpa melihat struktur kode penyusun sistem. Dengan kata lain, pengujian dilakukan dengan langsung mencoba menggunakan sistem (Shadiq et al., 2021). Adapun skenario pengujian *Black Box* pada sistem ini dapat dilihat pada Tabel 4.1.

Tabel 4. 2 Skenario Pengujian *Black Box*

Kode	Aksi	Skenario	Ekspektasi
BB-1-1	Login	Pengguna memasukkan <i>e-mail</i> dan <i>password</i> yang benar.	Menampilkan notifikasi login berhasil, membuat session, dan mengarahkan ke halaman utama serta membuka akses untuk halaman <i>activity log</i> .
BB-2-1	Logout	Pengguna menekan tombol <i>log out</i> .	Menghapus session, dan memblokir akses untuk halaman <i>activity log</i> .
BB-3-1	Pendeteksian Api dan Asap	Meletakkan api di depan kamera dalam kondisi terang	Api terdeteksi selama 5 detik secara berturut-turut, dan notifikasi



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

			terkirim ke aplikasi <i>mobile</i>
BB-3-2		Meletakkan api di depan kamera dalam kondisi gelap	Api terdeteksi selama 5 detik secara berturut-turut, dan notifikasi terkirim ke aplikasi <i>mobile</i>
BB-3-3		Tidak meletakkan api di depan kamera	Model tidak mendeteksi api dan tidak mengirim notifikasi ke aplikasi <i>mobile</i>
BB-3-4		Mengarahkan cahaya terang di depan kamera	Model tidak mendeteksi cahaya sebagai api ataupun asap

Untuk pengujiannya, apabila skenario dijalankan dan hasilnya sesuai dengan ekspektasi, maka pengujian dapat dikatakan berhasil. Sebaliknya, jika hasilnya tidak sesuai dengan ekspektasi, maka pengujian dikatakan gagal. Hasil pengujian *black box* dengan mengikuti skenario yang dibuat dapat dilihat pada Tabel 4.2.

Tabel 4. 3 Hasil Pengujian Black Box

Kode	Ekspektasi	Pengamatan	Hasil
BB-1-1	Pengguna memasukkan <i>e-mail</i> dan	Pengguna memasukkan <i>e-mail</i> dan	Sesuai



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

	<i>password</i> yang benar.	<i>password</i> yang benar.	
BB-2-1	Pengguna menekan tombol <i>log out</i> .	Pengguna menekan tombol <i>log out</i> .	Sesuai
BB-3-1	Meletakkan api di depan kamera	Meletakkan api di depan kamera dalam kondisi terang	Sesuai
BB-3-2		Meletakkan api di depan kamera dalam kondisi gelap	Sesuai
BB-3-3		Tidak meletakkan api di depan kamera	Sesuai
BB-3-4		Mengarahkan cahaya terang di depan kamera	Tidak Sesuai

4.4.2 System Usability Scale

Pengujian SUS dilakukan dengan tujuan untuk menilai kegunaan dari sistem yang dibuat secara keseluruhan. Subjek dari pengujian ini adalah khalayak umum dengan usia 21-50 tahun, yang diminta untuk menjawab 10 pertanyaan yang terdiri atas pertanyaan positif dan pertanyaan negatif yang memiliki format jawaban dengan menggunakan skala likert (Wardani et al., 2023). Adapun keterangan penilaian skala likert dapat dilihat pada Tabel 4.3.

Tabel 4. 4 Kategori Penilaian

Keterangan	Nilai
Sangat Tidak Setuju (STS)	1
Tidak Setuju (TS)	2



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Cukup (C)	3
Setuju (S)	4
Sangat Setuju (SS)	5

Daftar pertanyaan pengujian SUS dapat dilihat pada Tabel 4.4.

Tabel 4. 5 Pertanyaan SUS

Kode	Pertanyaan
SUS-1	Saya berpikir akan menggunakan sistem deteksi kebakaran ini.
SUS-2	Saya merasa sistem deteksi kebakaran ini sulit untuk digunakan.
SUS-3	Saya merasa sistem deteksi kebakaran ini mudah digunakan.
SUS-4	Saya merasa membutuhkan bantuan orang lain saat menggunakan sistem deteksi kebakaran ini.
SUS-5	Saya merasa fitur-fitur dalam sistem deteksi kebakaran ini berfungsi sebagaimana mestinya.
SUS-6	Saya merasa ada banyak hal yang tidak konsisten pada sistem ini.
SUS-7	Saya merasa orang lain akan memahami cara menggunakan sistem deteksi kebakaran ini dengan cepat.
SUS-8	Saya merasa alur atau tampilan sistem deteksi kebakaran ini membingungkan.
SUS-9	Saya merasa dapat menggunakan sistem deteksi kebakaran ini tanpa hambatan.
SUS-10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem deteksi kebakaran ini.

Adapun hasil dari pengujian SUS pada sistem ini dari 9 responden dapat dilihat pada Tabel 4.5 dan Tabel 4.6.

Tabel 4. 6 Hasil Pengujian SUS (1)

Kode	SUS-1	SUS-2	SUS-3	SUS-4	SUS-5
RES-1	5	1	5	2	5
RES-2	5	2	4	1	5



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

RES-3	5	2	5	1	4
RES-4	5	2	5	3	5
RES-5	5	1	4	1	5
RES-6	5	1	5	1	5
RES-7	5	2	4	1	4
RES-8	3	1	5	1	4
RES-9	5	1	5	1	5

Tabel 4. 7 Hasil Pengujian SUS (2)

Kode	SUS-6	SUS-7	SUS-8	SUS-9	SUS-10
RES-1	1	4	1	5	2
RES-2	1	4	1	4	1
RES-3	3	4	1	3	2
RES-4	1	4	2	5	3
RES-5	2	4	1	4	1
RES-6	1	5	1	5	1
RES-7	1	3	2	5	1
RES-8	1	5	1	4	1
RES-9	1	5	1	5	1

Berdasarkan hasil pengujian, nilai akhir SUS kemudian dihitung dengan menggunakan persamaan berikut.

$$SUS \text{ Ganjil} = \text{poin} - 1 \quad (2)$$

$$SUS \text{ Genap} = 5 - \text{poin} \quad (3)$$

$$\text{Nilai SUS} = (SUS \text{ Ganjil} + SUS \text{ Genap}) \times 2,5 \quad (4)$$

Dengan persamaan diatas, didapatkan hasil seperti pada Tabel 4.7.

Tabel 4. 8 Hasil Akhir Pengujian SUS

Kode	Nilai SUS
RES-1	92,5



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

RES-2	90
RES-3	80
RES-4	82,5
RES-5	90
RES-6	100
RES-7	85
RES-8	90
RES-9	100
Rata-rata	87,5

Rata-rata nilai SUS yang didapatkan dari 9 responden adalah 87,5, yang berdasarkan kategori penilaian SUS, nilai ini masuk kedalam kategori “*excellent*” dan termasuk ke dalam *grade* ‘B’. Hal ini menandakan bahwa meski sistem yang dikembangkan belum sempurna, sistem sudah dapat berfungsi sebagaimana mestinya.

4.5 Maintenance

Setelah seluruh sistem telah diimplementasi dan pengujian telah dilakukan, penting untuk dilakukan *maintenance* atau pemeliharaan secara berkala. Dalam sistem ini, pemeliharaan yang dilakukan adalah memastikan bahwa seluruh sistem tetap dapat berjalan sesuai fungsinya seiring berjalannya waktu, baik dari fungsi model, dari sisi *firebase*, ataupun dari sisi aplikasi *mobile*. Selain itu, dengan dilakukan pemeliharaan ini juga memungkinkan untuk menemukan *bug* yang belum ditemukan pada pengujian yang telah dilakukan. Optimalisasi sistem berupa pengembangan UI/UX ataupun kinerja model pendeteksi api juga dapat menjadi potensi pengembangan sistem untuk kedepannya.



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB V

PENUTUP

5.1 Kesimpulan

Setelah melalui serangkaian proses perancangan dan pengembangan sistem, kesimpulan yang dapat ditarik adalah sebagai berikut.

1. Sistem berhasil mendeteksi api secara real-time menggunakan model *deep learning* berbasis YOLO yang telah dilatih dengan dataset yang relevan. Model mampu mengenali objek api dengan tingkat akurasi yang cukup memenuhi standar untuk produksi berdasarkan evaluasi mAP dan *confusion matrix*.
2. Integrasi antara model deteksi dan sistem notifikasi real-time berjalan dengan baik. Saat api terdeteksi oleh kamera, sistem secara otomatis mengirimkan notifikasi ke aplikasi Android pengguna melalui *Firebase Cloud Messaging* (FCM), sehingga memungkinkan respon cepat terhadap potensi bahaya kebakaran.
3. Aplikasi *mobile* yang dikembangkan telah mendukung beberapa fitur fungsional, termasuk *login* pengguna, tampilan log aktivitas deteksi, serta tombol darurat yang dapat langsung menghubungi nomor penyelamatan kebakaran. Ini menunjang aspek keselamatan secara praktis dan efisien.

5.2 Saran

Berikut adalah saran yang dapat diberikan untuk pengembangan lebih lanjut dari skripsi ini.

1. Pengembangan multi-kelas deteksi kebakaran: Sistem dapat dikembangkan lebih lanjut untuk membedakan antara api kecil (misal lilin) dan api berbahaya, agar tidak menimbulkan *false alarm*.
2. Akurasi model dapat ditingkatkan lebih lanjut hingga 85-95%



DAFTAR PUSTAKA

- Mathew, A., Amudha, P. and Sivakumari, S., 2021. Deep learning techniques: an overview. *Advanced Machine Learning Technologies and Applications: Proceedings of AMLTA 2020*, pp.599-608.
- Redmon, J., 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- Siregar, R.H., Kurniawan, D., Nurmala, E. and Saifudin, I., 2024. Decreased Smoke Detector Performance Function on MV. Sarah S: Menurunnya Fungsi Kinerja Smoke Detectordi atas Kapal MV. Sarah S. *ATRIA: Jurnal Multidisiplin Riset Ilmiah*, 1(2), pp.23-30.
- Li, Z., Liu, F., Yang, W., Peng, S. and Zhou, J., 2021. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 33(12), pp.6999-7019.
- Muis, A. and Muhammad, F., 2023. Pelatihan Text Mining Menggunakan Bahasa Pemrograman Python. *Abdimas Langkanae*, 3(1), pp.36-46.
- Kothari, U., Choudhary, R., Shrivastava, S., Shrivastava, V., Pandey, A. and Sharma, S., 2024, November. The analysis and effect of Kotlin programming language in modern software development. In *AIP Conference Proceedings* (Vol. 3214, No. 1). AIP Publishing.
- Fitriyanto, G.A., Suarna, N. and Nurdiawan, O., 2023. Sistem Informasi Surat Masuk Dan Surat Keluar Pada Kantor Balai Besar Wilayah Sungai Menggunakan Metode Sistem Development Life Cycle. *Jurnal Informatika dan Teknologi Informasi*, 2(2), pp.215-219.
- Taye, M.M., 2023. Theoretical understanding of convolutional neural network: Concepts, architectures, applications, future directions. *Computation*, 11(3), p.52.
- Siregar, S.R. and Pristiwanto, P., 2022. Penerapan Metode Waterfall Dalam Pengembangan Sistem Informasi Masjid. *KLIK: Kajian Ilmiah Informatika dan Komputer*, 3(1), pp.26-32.
- H. -G. Lee, T. -N. Pham, V. -H. Nguyen, K. -R. Kwon, J. -H. Lee and J. -H. Huh, 2024. Image-Based Outlet Fire Causing Classification Using CNN-Based Deep Learning Models. *IEEE Access*, vol. 12, pp. 135104-135116
- P. Sridhar, S. K. Thangavel, L. Parameswaran and V. R. M. Oruganti, 2023. Fire Sensor and Surveillance Camera-Based GTCNN for Fire Detection System. *IEEE Sensors Journal*, vol. 23, no. 7, pp. 7626-7633

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumpulkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

- Widharma, I.G.S., Santiary, P.A.W., Sunaya, I.N., Darminta, I.K., Sangka, I.G.N. and Widiatmika, P.A.W., 2022. Deteksi api kebakaran berbasis computer vision dengan algoritma YOLO. *Journal of Applied Mechanical Engineering and Green Technology*, 3(2), pp.53-58.
- Bandi, A. and Kagitha, H., 2024. A case study on the generative AI project life cycle using large language models. *Proceedings of 39th International Conference*, 98, pp.189-199.
- Shadiq, J. et al. (2021) 'INFORMATION MANAGEMENT FOR EDUCATORS AND PROFESSIONALS Pengujian Aplikasi Peminjaman Kendaraan Operasional Kantor Menggunakan BlackBox Testing', *Information Management for Educators and Professionals*, 5(2), pp. 97–110.
- Wardani, I.K. et al. (2023) 'Pemanfaatan Metode Design Thinking dan Pengujian SUS untuk UI/UX Aplikasi Home Care Madiun Berbasis Android', *Journal of Computer and Information Systems Ampera*, 4(2), pp. 106–125. Available at: <https://journal-computing.org/index.php/journal-cisa/index>.
- Score, N.P., 2018. Net Promoter Score. *Work*, 10(10).
- Zaidi, S.F.A., Yang, J., Abbas, M.S., Hussain, R., Lee, D. and Park, C. (2024) *Vision-based construction safety monitoring utilizing temporal analysis to reduce false alarms*, *Buildings*, 14(6), 1878. <https://doi.org/10.3390/buildings14061878>
- Setiyani, L., 2021, November. Desain Sistem: Use Case Diagram. In *Prosiding Seminar Nasional Inovasi Dan Adopsi Teknologi (INOTEK)* (Vol. 1, No. 1, pp. 246-260).

DAFTAR RIWAYAT HIDUP PENULIS



Mohamad Rafi Al-Yahya

Lahir di Jakarta pada 22 November 2002. Anak pertama dari tiga bersaudara. Lulus dari SD Islam Amalina pada tahun 2015, SMP Islam Amalina pada tahun 2018, dan SMA Negeri 29 Jakarta pada tahun 2021. Saat ini sedang menempuh Pendidikan Sarjana Terapan Program Studi Teknik Informatika di Politeknik Negeri Jakarta. Tertarik pada bidang *machine learning* dan *mobile development*.

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

