**USULAN TUGAS AKHIR**

**PERANGKAT LUNAK PEMBACA METERAN AIR PDAM MENGGUNAKAN METODE YOLO**



**Oleh:**

**Dep` Niel Sinaga 2125250058**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS ILMU KOMPUTER DAN REKAYASA  
UNIVERSITAS MULTI DATA PALEMBANG  
PALEMBANG  
2025**

# HALAMAN PENGESAHAN

**USULAN TUGAS AKHIR**

**PERANGKAT LUNAK PEMBACA METERAN AIR PDAM MENGGUNAKAN METODE YOLO**

Diajukan oleh:

Dep` Niel Sinaga 2125250058

|  |  |  |
| --- | --- | --- |
| Palembang, ---- 2025 | | |
| Pengusul | | |
|  |  |  |
|  |  |  |
|  | Dep` Niel Sinaga |  |
|  | NPM 2125250004 |  |
|  |  |  |
|  | Menyetujui, |  |
|  | Dosen Pembimbing Utama |  |
|  |  |  |
|  |  |  |
|  | Tinaliah, M.Kom. |  |
|  | NIK 111070 |  |
|  | | |
| Mengetahui, | | |
| Kaprodi Informatika | | |
|  |  |  |
|  |  |  |
| Derry Alamsyah, S.Si, M.Kom, M.Pd | | |
|  | NIK 111069 |  |

# DAFTAR ISI

[HALAMAN PENGESAHAN 2](#_Toc177493753)

[DAFTAR ISI 3](#_Toc177493754)

[DAFTAR TABEL 4](#_Toc177493755)

[DAFTAR GAMBAR 5](#_Toc177493756)

[BAB I PENDAHULUAN 6](#_Toc177493757)

[1.1 Latar Belakang 6](#_Toc177493758)

[1.2 Rumusan Masalah 8](#_Toc177493759)

[1.3 Analisis terhadap Batasan 8](#_Toc177493760)

[1.3.1 Analisis terhadap Aspek Ekonomis 8](#_Toc177493761)

[BAB II LANDASAN TEORI 9](#_Toc177493762)

[2.1 Alternatif Solusi 9](#_Toc177493763)

[2.1.1 *Decision Tree* 9](#_Toc177493764)

[BAB III METODELOGI 11](#_Toc177493765)

[BAB IV PERANCANGAN 12](#_Toc177493766)

[DAFTAR PUSTAKA 13](#_Toc177493767)

# DAFTAR TABEL

# DAFTAR GAMBAR

# PENDAHULUAN

## Latar Belakang

Perkembangan teknologi informasi dan komunikasi, khususnya dalam bidang *Artificial Intelligence* (AI) dan *Computer Vision*, telah mendorong berbagai inovasi yang mendukung transformasi digital di banyak sektor, termasuk sektor pelayanan publik. Salah satu sektor yang sangat potensial untuk ditingkatkan melalui pemanfaatan teknologi ini adalah pengelolaan sumber daya air bersih oleh Perusahaan Daerah Air Minum (PDAM). Sebagai Badan Usaha Milik Daerah (BUMD), PDAM memiliki tanggung jawab penting dalam menyediakan air minum yang bersih dan layak bagi masyarakat, serta memastikan pencatatan pemakaian air yang akurat sebagai dasar penagihan pelanggan.

Namun, dalam implementasinya di lapangan, PDAM masih menghadapi sejumlah permasalahan yang signifikan. Proses pencatatan konsumsi air pelanggan masih dilakukan secara konvensional, yaitu dengan cara petugas lapangan mengambil foto meteran air dan kemudian mencatat angka yang tertera secara manual ke dalam sistem. Prosedur ini rawan terhadap kesalahan pencatatan akibat faktor manusia (*human error*), serta memerlukan waktu dan tenaga yang cukup besar (Carvalho dkk., 2023). Selain itu, tantangan teknis seperti meteran yang tertimbun, terhalang objek, atau berada di posisi yang sulit dijangkau turut menyulitkan petugas dalam melakukan pembacaan visual secara akurat (Sa’adat & Widiyanto, 2024).

Masalah ini semakin kompleks mengingat skala operasional PDAM yang luas, dengan jumlah pelanggan yang mencapai ribuan hingga jutaan. Dalam kondisi seperti ini, kesalahan pencatatan meskipun kecil dapat berdampak besar secara akumulatif. Banyak pelanggan yang menyampaikan keluhan terkait ketidaksesuaian antara angka pemakaian air dan tagihan yang diterima, yang pada akhirnya menurunkan tingkat kepercayaan masyarakat terhadap layanan PDAM (Imran dkk., 2023). Hal ini menunjukkan perlunya upaya untuk meningkatkan akurasi, transparansi, dan akuntabilitas dalam proses pencatatan penggunaan air (Febriawati dkk., 2021).

Digitalisasi proses kerja di sektor pelayanan publik menjadi kebutuhan yang tidak dapat ditunda, seiring dengan meningkatnya tuntutan efisiensi, transparansi, dan kecepatan layanan dari masyarakat (Danuri dkk., 2019). Dalam konteks PDAM, digitalisasi tidak hanya mencakup sistem pembayaran atau pengaduan pelanggan, tetapi juga proses teknis seperti pencatatan konsumsi air yang selama ini dilakukan secara manual. Ketika proses ini dilakukan oleh ribuan petugas di lapangan dengan cara yang tidak terstandarisasi, maka risiko kesalahan input, kehilangan data, hingga manipulasi informasi menjadi sangat tinggi. Oleh karena itu, penggunaan teknologi berbasis citra dan *deep learning* seperti YOLO menjadi solusi yang sangat relevan dalam menjawab tantangan otomasi di era transformasi digital, terutama di sektor-sektor vital seperti pengelolaan air bersih.

YOLO merupakan salah satu algoritma *real-time object detection* yang telah terbukti efektif dalam berbagai aplikasi pengenalan objek berbasis citra digital, termasuk pengenalan plat nomor kendaraan, pendeteksian benda pada sistem keamanan, serta pembacaan teks visual. Keunggulan YOLO terletak pada kemampuannya mendeteksi dan mengklasifikasi objek dalam satu tahap (*single-stage detector*) (Liu dkk., 2016), sehingga prosesnya lebih cepat dibandingkan metode lain seperti R-CNN yang bersifat dua tahap (*two-stage detector*) (Girshick, 2015). Kecepatan dan ketepatan inilah yang membuat YOLO sangat sesuai untuk digunakan dalam sistem pembacaan angka meteran air, di mana ketepatan angka sangat penting dan prosesnya harus efisien untuk bisa diterapkan secara luas oleh PDAM yang memiliki jumlah pelanggan besar dan lokasi pencatatan yang tersebar.

Untuk menyelesaikan permasalahan tersebut, penelitian ini diarahkan untuk merancang dan mengembangkan sebuah perangkat lunak berbasis metode YOLO yang mampu mengenali angka pada meteran air pelanggan secara otomatis dari gambar. Solusi yang diharapkan ini adalah dapat menjawab tantangan efisiensi, akurasi, dan transparansi dalam pencatatan penggunaan air oleh PDAM, sekaligus meningkatkan kualitas layanan kepada masyarakat secara menyeluruh.

## Rumusan Masalah

Pencatatan angka pada meteran air oleh PDAM masih dilakukan secara konvensional dan tidak diotomisasi sehingga menyebabkan potensi kesalahan dan keterlambatan dalam proses pencatatan angka meteran. Oleh karena itu, dibutuhkan sebuah sistem yang mampu mencatat angka meteran secara otomatis dan memudahkan petugas dalam melakukan pemantauan.

## Analisis terhadap Batasan

Analisis terhadap batasan dalam usulan tugas akhir ini dilakukan dengan melakukan wawancara terhadap 1 perusahaan umum daerah terkait denga hubungan permasalahan yang diambil. Wawancara dilakukan dengan Bapak Baijurti selaku pengawas meteran di Perumda Tirta Musi Palembang. Dengan adanya wawancara ini peneliti dapat menetapkan batasan aspek ekonomis, manufakturabilitas, dan sustainbilitas dalam pengembangan perangkat lunak untuk menyelesaikan permasalahan.

### Analisis terhadap Aspek Ekonomis

Berdasarkan hasil wawancara yang dilakukan dengan Perusahaan Umum Daerah Tirta Musi Palembang yang berlokasi di Jl. Jend. Sudirman No. Kel, 20 Ilir D. IV, Kec. Ilir Tim. I, Kota Palembang, Sumatera Selatan 30128 yang merupakan bergerak dalam bidang penyedia meteran air. Setelah menjelaskan perangkat lunak yang dikembangkan, diajukan penawaran untuk harga paket perangkat lunak didapatkan hasil dari perusahan yang ditunjukkan Tabel 1.1

**Tabel 1.1 Analisis Aspek Ekonomis dalam Penawaran perangkat Lunak**

|  |  |
| --- | --- |
| Aspek | Perumda Tirta Musi  Palembang KM 4 |
| Berdasarkan paket yang ditawarkan, paket mana yang akan dipilih Bapak? | Paket Normal |

Berdasarkan penawaran aplikasi yang telah dilakukan, Perumda Tirta Musi Palembang memilih paket normal yang bernilai Rp 15.000.000,00.

### Analisis terhadap Aspek Manufakturabilitas

Pada tahapan ini dilakukan analisis terhadap manufakturabilitas yang dimana pada tahapan ini dilakukan dengan wawancara pada 1 perusahaan. Berikut merupakan hasil dari analisis dari aspek manufakturabilitas. Hasil dari analisis aspek manufakturabilitas dapat ditunjukkan pada Tabel 1.2.

**Tabel 1.2 Analisis Aspek Manufakturabilitas Sudut Pandang Pengguna**

|  |  |
| --- | --- |
| Aspek | Perumda Tirta Musi  Palembang KM 4 |
| Waktu yang dibutuhkan untuk pengembangan perangkat lunak 20 minggu | OK |

### Analisis terhadap Aspek Sustainbilitas

Pada tahapan ini dilakukan analisis terhadap aspek sustainbilitas. Pada tahapan ini dilakukan dengan wawancara 1 perusahaan. Berikut merupakan hasil aspek sustainbilitas menurut narasumber yang telah diwawancarai. Hasil dari analisis aspek sustainbilitas dapat ditunjukkan pada Tabel 1.3.

**Tabel 1.3 Analisis Aspek Sustainbilitas dalam Sudut Pandang Pengguna**

|  |  |
| --- | --- |
| **Aspek** | **Perumda Tirta Musi**  **Palembang KM 4** |
| Proses pembacaan angka meteran air memakan waktu 1-2 detik | OK |
| Penyimpanan data pembacaan meteran air dilakukan dalam jangka waktu per hari | OK |

## Analisis terhadap Karakter Solusi

Dari hasil wawancara dengan perusahaan, didapat permasalahan yuang dapat dianalisis terhadap karakter solusi perangkat lunak yang akan dikembangkan. Permasalahan akan diberikan solusi yang ditunjukkan pada Tabel 1.4.

**Tabel 1.4 Analisis Karakter Solusi**

|  |  |
| --- | --- |
| Masalah | Solusi |
| Petugas menginput angka secara konvensional sehingga dapat terjadi kesalahan dalam menginput. | Perangkat lunak dapat membaca angka meteran secara cepat dan akurat |

## Pemilihan Solusi

## Tujuan

Tujuan dari penelitian ini adalah sebagai berikut:

Mengimplementasikan metode YOLO dalam perangkat lunak mendeteksi dan membaca angka pada meteran air PDAM secara cepat dan akurat.

Mengembangkan perangkat lunak yang dapat merekap laporan data pembacaan pada meteran air PDAM.

# LANDASAN TEORI

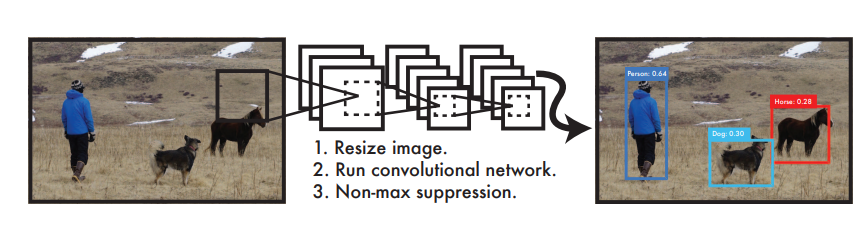
## Alternatif Solusi

### *You Only Look Once (YOLO)*

You Only Look Once (YOLO) merupakan metode yang dapat digunakan dalam proses pengembangan perangkat lunak. YOLO merupakan algoritma yang sudah sangat sering digunakan oleh para pengembang untuk mendeteksi objek. Algoritma ini menggunakan artsitetektur Convulutional Neural Network (CNN) untuk sebagai dasar dari fitur ekstrasi dari gambar yang di input untuk dilakukan dektesi pada objek. Algoritma ini termasuk memiliki kecepatan yang tinggi dalam melakukan deteksi pada objek karena hanya perlu memasukkan gambar kedalam jaringan untuk mendapatkan hasil deteksi (Peiyuan Jiang).

YOLO memiliki kemampuan generalisasi yang kuat karena YOLO dapat mempalajari fitur umum (generic feature) dan dapat di terapkan dalam pelatihan data di bidang lain. Algoritma ini dirancang untuk mempelajari fitur-fitur visual mendasar yang dapat di terapkan dalam berbagai bidang. Dengan mempelajari pola dan karakteristik sebuah objek secara umum, dalam lapisan awal algoritma ini dapat mendeteksi fitur secara sederhana seperti tepi dan tekstur. Lalu untuk lapisan dalam dapat mengidentifikasi struktur yang kompleks seperti bentuk dan konteks objek. Pelatihan yang dilakukan dalam dataset yang beragam dapat membangun generalisasi yang kuat bagi algoritma YOLO.

YOLO bekerja dengan membagi citra input menjadi grid sel yang berukuran seragam, yang dimana setiap sel bertanggung jawab untuk memprediksi bounding box dan probabilitas kelas objek yang berada yang nantinya akan diprediksi. Dengan membagi gambar menjadi beberapa bagian sel/grid dengan variasi ukuran tergantung pada ukuran citra yang di input, bounding box yang telah diprediksi dan probabilitas kelas untuk tiap sel.



**Gambar 2. 1 Cara Kerja Algoritma YOLO**

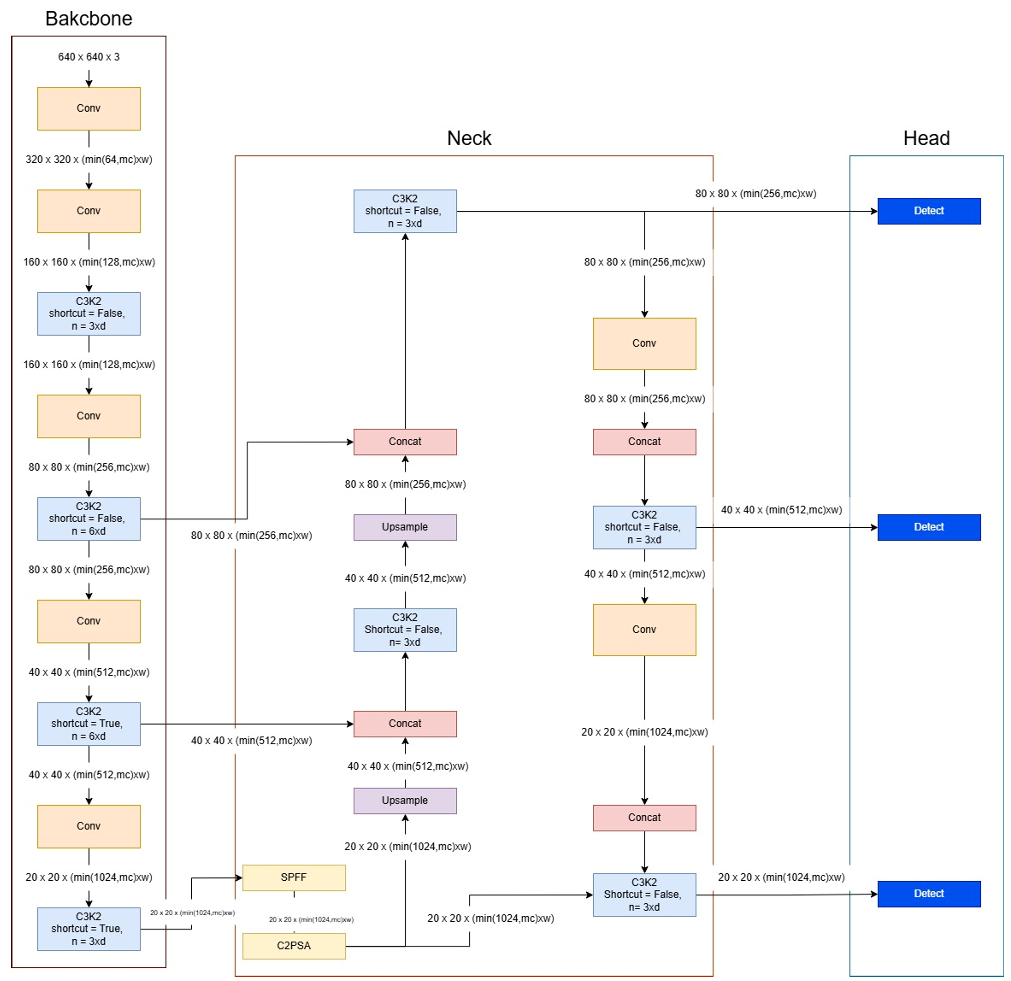
Setiap grid memiliki tanggung jawab menentukan apakah objek berada pada area grid (Gupta dkk., 2021). Deteksi objek akan dilakukan apabila objek yang ditentukan berada dalam grid. Setelah itu, Algoritma YOLO menghasilkan koordinat bounding box dari objek dengan menunjukkan nilai confidence kelas yang terbesar. Koordinat bounding box dalam vector, seperti y = [pc, bx, by, bh, bw, c1, c2] yang dimana pc menunjukkan probabilitas dari grid yang memiliki objek, bx dan by menunjukkan koordinat bounding box, bh & bw menunjukkan lebar dan tinggi dari bounding box, dan c1 & c2 menujukkan probabilitas kelas objek yang dideteksi objek yang dideteksi apabila terdapat 2 kelas.

Setelah bounding box, Non-Max Suppression (NMS) diimplementasikan untuk menyaring hasil prediksi yang memiliki probabilitas hasil yang rendah dengan menetapkan confidence threshold yang dihasilkan model. NMS mempertahankan bounding box dengan confidence tertinggi dengan menghilangkan prediksi lain yang tumpang tindih. Tingkat tumpang tindih dihitung menggunakan Intersection over Union (IoU), yaitu resio antara are tumpang tindih dan total area gabungan dari dua bounding box. Jika diantara dua bounding box memiliki IoU yang tinggi dan salah satunya memiliki confidence yang lebih rendah, maka bounding box dengan nilai confidence yang lebih rendah akan dihapus (). Hal ini membantu mengurangi duplikasi deteksi objek yang sama, sehingga meningkatkan akurasi deteksi akhir.

Saat YOLO 10 dipublikasikan, NMS tidak lagi digunakan kembali, sehingga diganti dengan menggunakan dua label assignments YOLOv10 menggunakan strategi one-to-many memungkinkan model menghasilkan beberapa prediksi setiap ground truth unutk meningkatkan recall dan akurasi, sedangan one-to-one memastikan presisi dengan memilih prediksi terbaik unuk setiap objek (). Pendekatan ini dapat menghilangkan kebutuhan Non-Max Suppression (NMS) () dengan tetap mempertahankan kecepatan dan akurasi dalam deteksi objek. Dengan demikian, YOLO dapat mengenali objek dalam lingkungan yang kompleks.

YOLO memiliki keunggulan kecepatan dalam deteksi objek pada citra dibandingkan dengan algoritma lainnya. Pada model YOLOv10, yaitu YOLO 10 Nano, sistem dapat melakukan deteksi objek dengan kecepatan 1.84 ms. Versi terbaru dari YOLO adalah YOLOv11 yang merupakan model YOLO dengan kecepatan tertinggi saat ini. Dan varian dari YOLO 11 Nano memiliki kecepatan hingga 1.55 ms jika dibanding dengan versi sebelumnya (). Pelatihan model YOLOv11 mendapati 25% lebih cepat dari YOLOv10 dengan lebih sedikit menggunakan parameter dan perunajam arsitektur.

YOLOv11 memiliki 3 bagian utama yang dapat dilihat pada Gambar 2.3 yang terdiri dari backbone, neck dan head. Backbone, neck dan head didalam arsitektur YOLO bekerja sama dengan backbone mengekstrak fitur dari gambar input, neck menggabungkan fitur multi-skala, dan head menghasilkan prediksi akhir berupa bounding box dan label kelas objek (). Backbone terdiri dari jaringan konvolusi yang berfungsi sebagai ekstrasi fitur untuk menghasilkan feature map. Neck memanfaatkan teknik seperti upsampling dan concatenation yang memungkinkan mendapatkan informasi dari objek pada skala yang berbeda, sehingga deteksi objek dapat lebih presisi terhadap perbedaan ukuran.



**Gambar 2. 2 Arsitektur YOLO 11**

Penelitian yang dilakukan oleh mengusulkan YOLOv8 sebagai solusi inovatif untuk deteksi angka pada meteran PDAM, memanfaatkan CNN untuk akurasi tinggi dan adaptasi terhadap variasi lingkungan seperti lumpur atau perubahan cahaya. Sebagai alternatif, metode Template Matching mencapai akurasi 85% tetapi kurang fleksibel dalam kondisi dinamis, sementara YOLOv4 yang terintegrasi dengan IoT menunjukkan akurasi 95,7% meski terbatas pada deteksi angka tunggal. YOLOv5s juga diuji dengan akurasi 85%, namun masih memerlukan optimasi untuk menyaingi performa YOLOv8 atau YOLOv4.

YOLOv8 dianggap paling menjanjikan karena menggabungkan kecepatan real-time dan ketahanan dalam skenario kompleks, meskipun belum diuji secara khusus pada dataset meteran PDAM. Metode lain seperti Template Matching bergantung pada kualitas template, dan YOLOv4/YOLOv5s memiliki keterbatasan dalam deteksi multi-digit atau lingkungan ekstrem. Studi-studi ini menyoroti perlunya pendekatan yang lebih robust untuk mengatasi tantangan seperti kotoran, cahaya dinamis, atau angka multi-digit.

Penelitian ini bertujuan memanfaatkan YOLOv8 untuk mengatasi keterbatasan metode sebelumnya dengan fokus pada ketahanan terhadap kondisi ekstrem dan deteksi multi-digit. Melalui augmentasi data yang merepresentasikan skenario nyata, diharapkan sistem ini dapat mencapai akurasi lebih tinggi dan lebih fleksibel dibanding Template Matching atau YOLOv5s, sekaligus memperluas aplikasi YOLOv8 dalam konteks meteran PDAM yang belum sepenuhnya dieksplorasi.

### Single Shot Detector (SSD)

Single shot detector (SSD) merupaka algoritma yang sudah sering digunakan dalam mendeteksi objek karena keseimbangan dan akurasi yang di dapatkan. Algoritma ini hanya bekerja hanya satu kali proses, yang dimana klasifikasi dan prediksi dari bounding box dilakukan secara bersama. SSD menggunakan arsitektur Convolutional Neural network yang memiliki feature map. Feature map inilah yang berfungsi sebagai ground truth box untuk memprediksi nilai confidence dari kelas objek yang akan di deteksi. Didalam feature map, SSD memprediksi bounding box (kotak pembatas) dengan menggunakann beberapa default boxes yang memiliki berbagai aspek rasion dan ukuran (). Default boxes akan memprediksi seberapa besar nilai confidence objek pada citra. SSD memungkinkan deterksi beberapa objek dalam satu gambar. Berbeda hal dengan algoritma lainnya seperti Faster R-CNN, SSD memerlukan dua langkah yakni satu untuk menghasilkan saran wilayah dan satunya lagi untuk mendeteksi objek.

SSD melakukan deteksi dalam satu langkah yang dimana membuat algoritma ini membuatnya lebih cepat (). Salah satu backbone yang digunakan dalam arsitektur SSD adalah DenseNet yang tedapat pada Gambar 2.4. DF-SSd merupakan salah satu model SSD yang menggunakan arsitektur DenseNet-S-32-1, yang dimana S menunjukkan stem block, 32 menunjukkan growth rate dalam dense blocks, dan 1 menunjukkan compression factor = 1 di transition layers (). Dengan menggunakan DenseNet, SSD dapat menggunaka feature map dengan efisien untuk meningkatkan akurasi secara signifika. Selain itu bagian neck seperti MobileNet, sering digunakan dengan berbagai macam backbone yang dimana memiliki tingkat keseimbangan dalam proses komputasi dan akurasi yang didapat. Arsitektur MobileNet memiliki Depthwise separable convolutions, squeeze-and-excitation module, dan linear bottlenecks. Sedangkan, pada bagian head digunakan pada lapisan konvolusi yang dengan fungsi aktivasi seperti softmax untuk memprediksi bounding box dari objek.

Dalam SSD juga menggunakan non-maximum suppresion (NMS) untuk mengurangi jumlah prediksi bounding box untuk setiap kelas yang berlebihan. NMS memiliki cara kerja dengan menyaring kotak yang memilik confidence score tinggi dan menghapus kotak lain yang overlap dengan bounding box yang lebih rendah. Proses ini memungkinkan SSD untuk memberikan bounding box yang lebih tepat pada objek yang terdeteksi. Selain itu, SSD juga menggunakan multiscale feature maps yang memungkinkan deteksi objek pada berbagai ukuran secara bersamaan (Karbouja dkk., 2024). Dengan mengoptimalkan kombinasi antara kecepatan dan akurasi, SSD mampu memberikan solusi yang efisien dalam deteksi objek dalam aplikasi dunia nyata. SSD menawarkan performa dalam akurasi dan kecepatan dalam deteksi objek, sehingga cocok digunakan secara real-time. Pada pengujian dengan menggunakan dataset PASCAL VOC 2007, model SSD DF-SSD300 dengan jumlah growth rate sebesar 16 mampu memiliki mean average precision (mAP) sebesar 64,3% (Zhai dkk., 2020). Dengan kemampuannya yang cepat, SSD mampu digunakan untuk mengenali objek sesuai dengan kelasnya sesuai dengan confidence score yang diperoleh. Model DF-SSD300 mampu menangkap citra sebanyak 46 hingga 59 FPS (Sanchez dkk., 2020), sehingga mampu melakukan tracking terhadap objek yang begerak. SSD dapat mengenali objek pada berbagai skala dengan memanfaatkan peta fitur dari setiap lapisan, namun masih memiliki lapisan yang dangkal dalam mendapatkan informasi, sehingga model kesulitan dalam mengenali objek yang kecil (Musalhi dkk., 2024).

# METODELOGI

## Identifikasi Masalah

Perusahaan Daerah Air Minum (PDAM) memiliki tanggung jawab yang besar dalam penyediaan air bersih kepada pelanggan yang berkaitan. Pencatatan penggunaan air bersih saat ini umumnya masih dilakukan secara konvensional, yang dimana petugas lapangan harus mendatangi setiap rumah pelanggan untuk mencatat keseluruhan penggunaan air yang tertera pada angka meteran air. Cara yang dilakukan yakni mengambil gambar meteran air pada saat kunjungan ke pelanggan lalu mencatat penggunaan air yang telah di gunakan oleh pelanggan. Masalah ini menyebabkan data penggunaan air seringkali terkambat masuk kedalam sistem dan dapat menimbulkan ketidaktepatkan dalam penagihan jumlah kepada pelanggan. Karena itu dibutuhkan suatu sistem yang dapat membantu pencatatan meteran air secara otomatis, cepat dan akurat agar proses pelayanan dapat menjadi lebih efisien.

## Analisis Solusi

Dari masalah yang telah diidentifikasi, dipilih solusi yang terbaik berdasarkan performa yang diberikan dalam aspek ekonomis, sustainbilitas dan manufakturabilitas. Serta berdasarkan penelitian terdahulu yang telah menggunakan solusi yang dipilih, yakni dengan algoritma You Only Look Once (YOLO) yang akan digunakan YOLO versi 11 Nano yang memiliki kemampuan dalam kecepatan yang baik dalam mendeteksi objek dibanding versi YOLO sebelumnya.

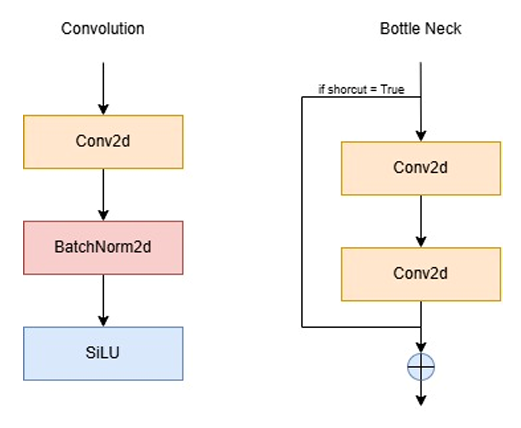
## Pengumpulan Data

Pada tahap ini, pengumpulan data dilakukan secara konvensional oleh petugas lapangan PDAM yang bertugas mengambil gambar meteran air di lokasi pelanggan. Data yang dikumpulkan berupa gambar digital meteran air yang diambil dengan menggunakan handphone petugas sesuai dengan kondisi nyata di lapangan, seperti perbedaan pencahayaan siang dan malam, sudut pengambilan gambar, serta berbagai kondisi meteran dalam bentuk bersih dan kotor.

Sebelum pengambilan data ketua tim pengawas meteran menyiapkan panduan teknis yang berisi kriteria gambar yang dibutuhkan dan menyiapkan penamaan file gambar yang telah diambil sesuai dengan nomor pelanggan yang di kunjungi.

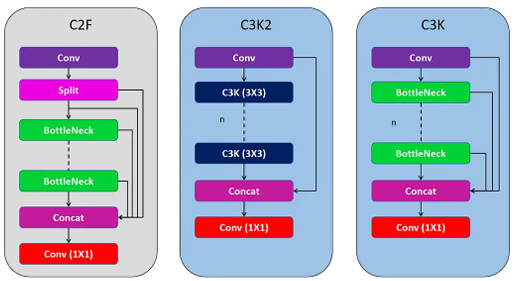
## Pengembangan Model

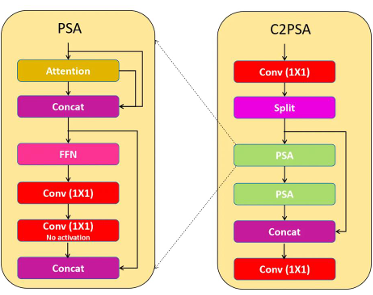
Pengembangan mode menggunakan metode You Only Look Once (YOLO) versi 11 Nano. YOLO menggunakan metode pembelajaran mesin yang diawasi (supervised learning) untuk melatih model prediksi (). YOLO menerapkan arsitektur jaringan saraf tiruan, seperti convolution layer dengan fungsi aktivasi sigmoid linear unit (SiLU), batch normalization, dan bottle neck layer untuk mengurangi proses komputasi model yang dapat dilihat pada Gambar 3.1. Model YOLO 11 Nano dibuat melalui transfer learning dari pretrained weight yang di sediakan dari library Ultralytics.



**Gambar 3.1 Convolution dan Bottle Neck Layer**

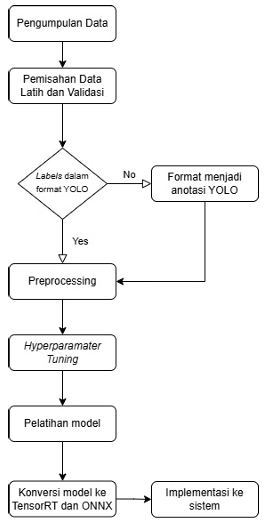
YOLOv11n masih menggunakan lapisan dari konvolusi fitur ekstrasi dari citra input dnegan aktivasi SiLU. Model YOLOv11 menggunakan blok C3k2 menggantikan blok C2f dari pendahulunya merupakan implementasi Bottleneck Cross Stage Partial (CSP) yang lebih efisien, cepat dan meningkatkan performa model yang dapat dilihat pada Gambar 3.2. Penambahan blok dari Cross Stage Partial with Spatial Attention (C2PSA) setelah blok Spatial Pyramid Pooling-Fast (SPPF) yang memungkinkan model dapat lebih fokus pada era penting dari citra yang ditunjukkan pada Gambar 3.3 (). Lapisan konvolusi akhir dan lapisan deteksi masih digunakan dengan penggunaan batch normalization dan fungsi aktivasi SiLU.



**Gambar 3.2 Perbandingan Blok C2f dan C3K2**

**Gambar 3.3 Blok C2SPA**

Pengembangan model akan melalui tahapan yang dapat dilihat pada Gambar 3.4. Tahapan awal yang dilakukan dengan pengumpulan data yang kemudian akan dilakukan pembagian data menjadi data latih dan data validasi.



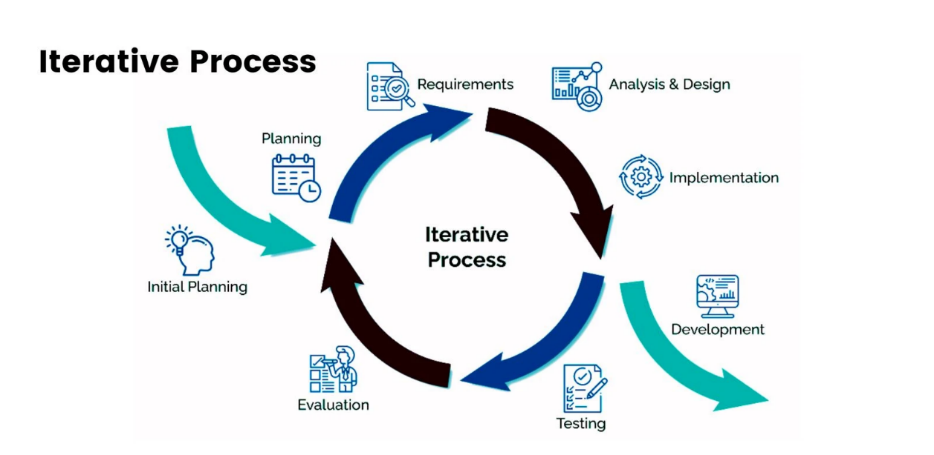
**Gambar 3.4 Tahapan Pengembangan Model**

Beberapa data telah memiliki labels yang berupa anotasi dari bounding box dalam format YOLO, yaitu <class\_id> <x\_center> <y\_center> <width> <height>. Ada beberapa data yang memiliki anotasi tidak memiliki format YOLO, sehingga akan dilakukan pengubahan labels dengan mengubah anotasi menjadi format YOLO. Kemudian, semua data akan dilakukan preprocessing berupa resize pada citra gambar yang dapat di terima oleh YOLO. Lalu dilanjutkan dengan hyperparameter tuning untuk membantu proses training model. Proses pelatihan YOLO dimulai dengan mengambil batch gambar dari dataset dan mengekstrasi fitur melalui lapisan backbone dengan jaringan konvolusi dengan bertujuan mengubah data gambar menjadi peta fitur multi-skala. Setelah itu, neck akan memproses fitur tersebut dalam berbagai skala, memungkinkan deteksi objek dalam berbagai ukuran. Pada head, model membuat prediksi berupa koordinat bounding box, kelas objek, dan confidence score untuk setiap objek yang terdeteksi. Selama training data, prediksi model dibandingka dengan anotasi ground truth melalui loss function yang menghitung tigas jenis loss, yaitu localization loss untuk posisi bounding box, classification loss untuk kelas objek, dan confidence loss untuk skor keyakinan deteksi. Beberapa jenis optimizer seperti SGD atau AdamW dapat digunakan untuk memperbarui bobot model dengan menurunkan nilai loss secara bertahap, sehingga model dapat belajar lebih akurat setiap kenaikan epoch. Proses ini akan berlanjut sampai mode mencapai konvergensi untuk menghasilkan nilai dari suatu model yang optimal dalam mendeteksi objek sesuai dengan data latih yang diberikan.

Setelah training selesai, model akan dikonversi menjadi format TensorRT dan Open Neural Network Exchange (ONNX). Pada awalnya model akan memiliki format Pytorch dan untuk menyesuaikan dengan perangkat lunak, makan format akan diubah menjadi TensorRT karena memanfaatkan Graphic Processing Unit (GPU) NVIDIA sebagai pengoptimal inferensi yang memiliki performa tinggi dan runtime yang dapat digunakan untuk melakukan inferensi dengan presisi yang lebih rendah. Apabila komputer tidak memiliki GPU NVIDIA, maka format yang akan digunakan adalah ONNX karena mendukung berbagai perangkat kerja deep learning yang dapat digunakan pada berbagai jenis platform. Model akan dievaluasi dan dibandingkan dengan setiap data untuk menentukan data melatih model dengan konvergensi yang paling baik melalui parameter, seperti mean average precision (mAP). Model akan diuji dengan data real-time yang akan diterapkan melalui upload foto gambar kedalam desktop nantinya.

## Pengembangan Perangkat Lunak

Pada tahap pengembangan, perangkat lunak dikembangkan dengan System Development Life Cycle (SDLC), yakni model iteratif. Model iteratif merupakan salah satu tahapan pengembangan yang prosesnya dilakukan secara bertahap dan penjadwalan ulang pada saat memperbaiki bagian-bagian dari rancangan sistem. Tujuan dari model iteratif ini untuk meningkatkan pengembangan melalui beberapan tahapan pengulangan kecil sehingga dapat mengerjakannya secara fleksibel tanpa adanya keharusan penyelesaian satu tahap secara utuh. Tahapan model iteratif dapat dilihat pada Gambar 3.1 yang terdiri dari perencanaan, analisis dan desain, implementasi, pengujian dan evaluasi.



**Gambar 3. 5 SDLC Model Iteratif** (Imran Hibba, 2022)

Perencanaan

Pada tahap ini, perencanaan mengenai masalah yang dihadapi dengan menentukan tujuan dari perangkat lunak dalam menyelesaikan permasalahan. Dalam model iteratif, tahapan ini dapat dilakukan secara berulang untuk mendapatkan kebutuhan perangkat lunak secara maksimal.

Analisis dan desain

Pada tahap ini, dilakukan analisis terhadap rancangan sistem perangkat lunak. Desain dapat diimplementasikan dengan membuat rancangan antarmuka perangkat lunak berdasarkan kebutuhan perangkat lunak.

Implementasi

Pada tahap ini, perangkat lunak mulai dikembangkan berdasarkan rancangan sistem dan desain yang telah dikerjakan. Perangkat lunak dikembangkan dengan kode pemrograman sesuai kebutuhan perangkat lunak. Database mulai diimplementasikan untuk mendukung perangkat lunak sebagai tempat penyimpanan.

Pengujian

Pada tahap ini, dilakukan pengujian terhadap perangkat lunak yang telah dikembangkan untuk mengukur performa dari perangkat lunak yang dibuat. Tahapan ini juga bertujuan untuk menemukan kekuragnan atau kesalahan dalam perangkat lunak. Pengujian perangkat lunak dilakukan dnegan user acceptance testing untuk mendapat feedback atau umpan balik dari pengguna mengenai perangkat lunak. Setelah dilakukan pengujian, hasilnya akan menjadi dasar dari peningkatan perangkat lunak agar lebih optimal.

Evaluasi

Dalam tahapan ini, dilakukan evaluasi dari hasil iterasi yang telah dilakukan. Evaluasi yang dilakukan adalah dapatkah perangkat lunak telah memenuhi kebutuhan dari user, setelah itu akan di *deploy* kepada pengguna.

## Pengujian

Pada tahap pengujian, performa dari model yang telah dikembangkan akan diukur dari segi proses hyperparameter tuning. Proses tuning dilihat pada tabel 3.1 yang mencakup penyesuaian parameter seperti learning rate, momentum, batch size, optimizer, dan lainnya untuk menghasilkan metrik evaluasi. Metrik evaluasi yang dihasilkan berupa precision, recall, f1 score, mean average precision at IoU 50 (mAP50), dan mean average precision at IoU 50-95 (mAP50-95). Metrik evaluasi ini bertujuan memastikan bahwa model dapat mendeteksi dan menghitung objek dengan akurat dan konsisten dalam kondisi lingkungan pengujian.

**Tabel 3. 1Hyperparameter Tuning Model**

|  |  |
| --- | --- |
| Parameter | Nilai |
| Initial learning rate | 0,01, 0.001 |
| Final learning rate | 0.1, 0,01 |
| Optimizer | SGD, AdamW |
| Epoch | 100, 200, 300 |

Precision merupakan metrik untuk mengukur seberapa akurat model dalam menghasilkan prediksi yang benar. Precision dihitung dengan membagi jumlah True Positives (TP) dengan jumlah True Positive dan False Positives (FP) seperti yang ditunjukkan pada persamaan (1) sedangkan recall merupakan metrik untuk mengukur seberapa baik model dalam menangkap seluruh contoh positif yang benar. Recall dihititung dengan membagi jumlah True Positives dengan jumlah True Positives dan False Negatives (FN) seperti yang ditunjukkan pada persamaan (2).

|  |  |
| --- | --- |
|  | (1) |
|  | (2) |

F1 score merupakan metrik yang mewakili rata-rata harmonis dari precision dan recall, F1 score yang tingg menunjukkan kinerja model yang baik. Perhitungan F1 score ditunjukkan pada persamaan (3).

|  |  |
| --- | --- |
|  | (3) |

Perhitungan mAP (mean average precision) pada persamaan (4) didapat dari average precision (AP) pada persamaan (3) yang menggunakan precision dan recall. AP dihitung berdasarkan kurva precision-recall dengan mempertimbangkan berbagai ambang batas untuk menilai kinerja model dari deteksi objek.

## Hasil

Output yang dihasilkan dari penelitian ini berupa perangkat lunak berbasis desktop yang dimana dapat mendeteksi angka dalam foto meteran yang diambil dan menampilkan data pembacaan pada meteran yang telah di deteksi.

# PERANCANGAN

## Spesifikasi Solusi

Spesifikasi solusi dari perangkat lunak pembaca angka meteran air yang dikembangkan saat ini, yaitu:

Perangkat lunak terdiri dari aplikasi desktop.

Perangkat lunak desktop dapat mendeteksi angka meteran pdam.

Perangkat lunak terhubung dengan PostgreSQL sebagai database.

Perangkat lunak desktop dapat merekap laporan pembacaan meteran air pdam.

Perangkat lunak desktop memiliki model YOLO 11 Nano yang dibuat melalui library Ultralytics dan kerangka kerja deep learning Pytorch.

## Rencana Pengujian

Rencana pengujian untuk mengukur spesifikasi yang telah diajukan akan dilakukan terhadap perangkat lunak dan model yang dikembangkan dengan penguji, yaitu pengguna dan expert. Perangkat lunak diuji dengan menggunakan metode blackbox testing yang dapat dilihat pada Tabel 4.1.

**Tabel 4. 1 Black Box Testing Perangkat Lunak**

|  |  |
| --- | --- |
| No | Pengujian |
| 1. | Pengguna login dengan memasukkan username dan password. |
| 2. | Pengguna lupa pasword mendapatkan verifikasi email. |
| 3. | Pengguna memasukkan foto kedalam menu dan memulai proses deteksi angka yang berada didalam meteran. |
| 4. | Sistem membaca angka meteran dan mengekstrak angka. |
| 5. | Sistem merekap data laporan pembacaan meteran |

Rencana pengujian kinerja (performance testing) perangkat lunak dengan model YOLO juga dilkaukan untuk menentukan seberapa baik model YOLO dalam berbagai kondisi untuk mendeteksi dan menghitung jumlah pengunjung. Rencana pengujian kinerja perangkat dengan model YOLO dapat dilihat pada Tabel 4.2.

**Tabel 4. 2 Rencana Pengujian Kinerja Model**

|  |  |  |
| --- | --- | --- |
| No. | Kondisi | Hasil Yang Diharapkan |
| 1. | Foto meteran dalam kondisi bersih | Dapat terdeteksi |
| 2. | Foto meteran dalam keadaan buram | Dapat terdeteksi |
| 3. | Foto meteran dalam keadaan kotor | Dapat terdeteksi |
| 4. | Foto meteran dalam kondisi pencahayaan yang baik | Dapat terdeteksi |
| 5. | Foto meteran dalam kondisi pencahayaan yang kurang | Dapat terdeteksi |
| 6. | Foto meteran dalam kondisi pantulan pencahayaan yang berlebihan | Dapat terdeteksi |
| 7. | Foto meteran dalam kondisi tertimbun sebagian | Dapat terdeteksi |
| 8. | Foto meteran dalam kondisi tertimbun sepenuhnya | Tidak dapat terdeteksi |

## Perancangan Sistem

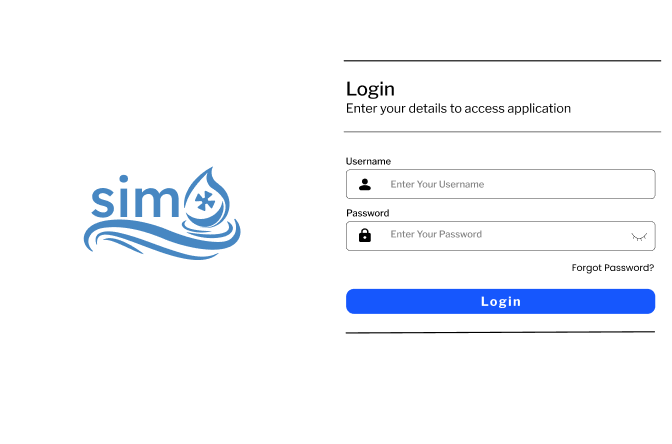
Perancangan sistem dalam pengembanga perangkat lunak pembaca angka meteran air pdam menggunakan metode YOLOv11. Perancangan didasarkan dengan metode SDLC yang diterapkan yakni, Iteratif. Proses pengembangan perangkat lunak antara lain:

Perencanaan

Pada tahap ini, dilakukan wawancara dengan perusahaan atau instansi yang terkait dengan permasalahan yang di angkat tugas akhri ini untuk memvalidasinya. Perusahaan tersebut merupakan Perusahaan Umum Daerah Tirta Musi Palembang KM 4. Selain itu, dilakukan pengumpulan dataset yang dimana dataset yang di ambil oleh petugas pengawas meteran mengambil gambar meteran untuk melakukan pencatatan penagihan. Dengan model yang dipilih yakni YOLO 11 Nano akan diuji performanya dengan dataset yang telah dikumpulkan, lalu akan di cari hasil terbaiknya. Setelah itu, model akan dilakukan hyperparameter tuning menggunakan dataset dengan performa yang terbaik.

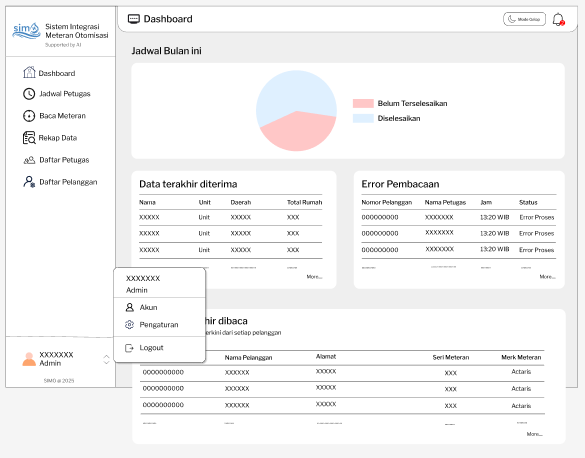
Analisis dan Desain

Pada tahap ini, dilakukan analisis terhadap kebutuhan perangkat lunak yang akan dikembangkan berdasarkan hasil wawancara dnegan perusahaaan. Rancangan antarmuka dari aplikasi desktop di desain dalam bentuk wireframe dnegan menggunakan Figma. Perangkat lunak akan dibuat berdasarkan kebutuhan pengguna untuk mengatasi permasalahan yang diangkat. Rancangan antarmuka pada aplikasi desktop memiliki tampilan login yang dapat dilihat pada Gambar 4.1.



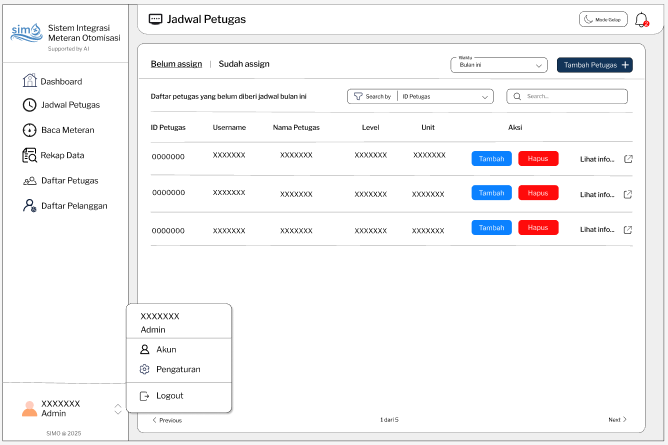
**Gambar 4. 1 Rancangan Antarmuka Login Aplikasi Desktop**

Pada rancangan antarmuka aplikasi desktop, terdapat 6 menu utama yakni dashboard, jadwal petugas, baca meteran, rekap data, daftar petugas dan daftar pelanggan. Halaman Dashboard berisi informasi keterangan total pembacaan meteran yang telah di lakukan dan sebaliknya yang dapat dilihat pada Gambar 4.2



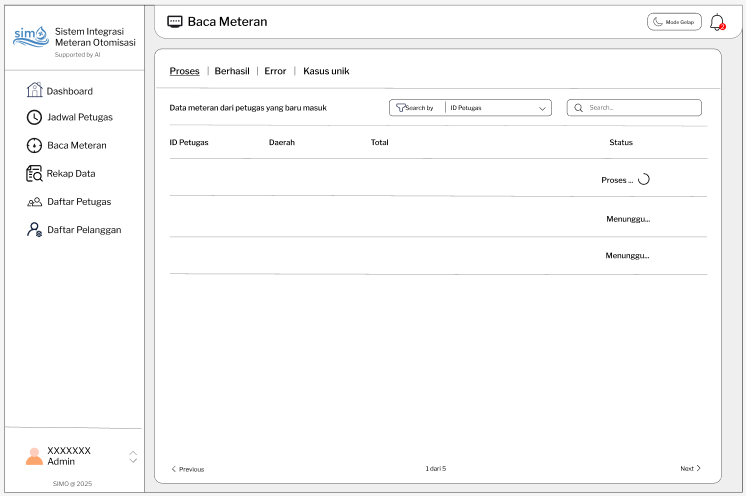
**Gambar 4.2 Rancangan Antarmuka Halaman Dashboard**

Halaman jadwal petugas berisikan daftar petugas yang terdaftar dalam pengawas meteran dan jadwal petugas yang akan membaca meteran yang diberikan oleh kepala pengawas meteran yang dapat dilihat pada Gambar 4.3.

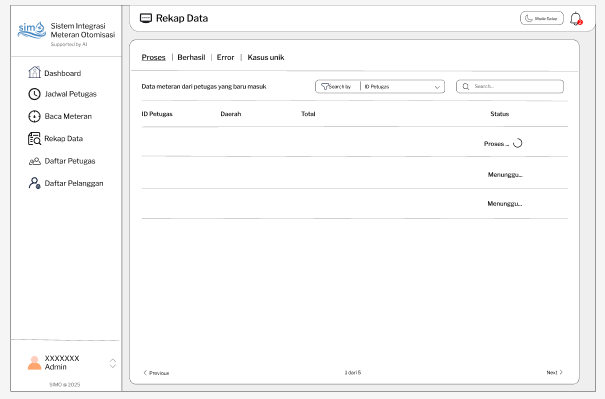


**Gambar 4. 3 Rancangan Antarmuka Halaman Jadwal Petugas**

Halaman baca meteran berisikan informasi keterangan proses pembacaan meteran yang telah di proses oleh kepala pengawas meteran yang dapat dilihat pada Gambar 4.4.

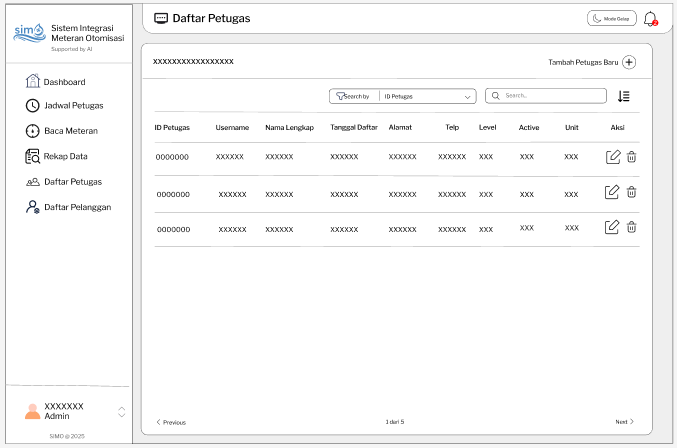


**Gambar 4. 4 Rancangan Antarmuka Halaman Baca Meteran**

Halaman rekap data berisi rekapan data keseluruhan pembacaan data meteran yang telah dibaca angkanya yang dapat dilihat pada Gambar 4.5.

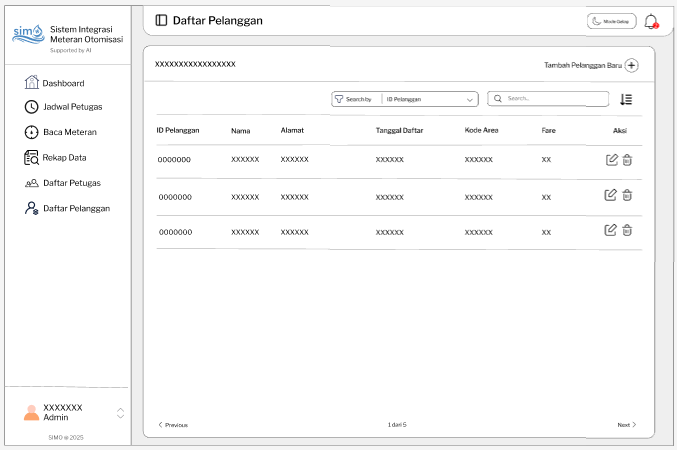
**Gambar 4. 5 Rancangan Antarmuka Halaman Rekap Data**

Halaman daftar petugas berisikan pendaftaran petugas yang akan di daftarkan dan menampilkan seluruh petugas yang telah terdaftar dalam pengawas meteran yang dapat dilihat pada Gambar 4.6.



**Gambar 4. 6 Rancangan Antarmuka Halaman Daftar Petugas**

Halaman daftar pelanggan berisi semua pelanggan yang telah terdaftar dalam pemasangan meteran air yang dapat dilihat pada Gambar 4.7.



**Gambar 4. 7 Rancangan Antarmuka Halaman Daftar Pelanggan**

3. Implementasi

Pada tahap ini, model akan dikembangkan dengan menggunakan library Utralytics versi 8.126. yang menyediakan model dan Pytorch versi 2 sebagai kerangka kerja deep learning dan pengembangan model menggunakan Python versi 3.11.1. Model dilatih dengan 2 dataset yang berbeda.

4. Pengujian

Pada tahap ini, pengujian dilakukan dengan menguji perangkat lunak yang telah terintegrasi dengan model YOLO dalam deteksi angka meteran air untuk mengetahui terjadinya error atau bug dalam perangkat lunak. Pengujian dilakukan dengan menggunakan YOLO dengan varian YOLO 11 Nano dengan dataset meteran air yang merupakan dataset dengan hasil terbaik dibanding dataset lainnya.

5. Evaluasi

Pada tahap ini akan dilakukan evaluasi apakah perangkat lunak telah memenuhi kebutuhan dari pengguna dan menyelesaikan permasalahan yang diangkat dalam tugas akhir ini.

## Verifikasi Perancangan

Verifikasi perancangan dilakukan dnegan membuat daftar semua atribut pada metode yang digunakan kemudian akan diverifikasi menggunakan tanda centang pada atribut yang ditunjukkan pada Tabel 4.4.

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Item** | **Verifikasi** | **Keterangan** |
| 1. | YOLOv11 |  | Model YOLOv11 yang digunakan adalah yolo 11n.pt karena memiliki komputasi yang paling rendah, sehinga memiliki kecepatan proses yang lebih baik yang cocok digunakan secara realtime |
| 2. | Epoch |  | Jumlah epoch yang digunakan dalam penentuan dataset, yaitu 100. Pada proses hyperparameter tuning, jumlah epoch yang digunakan adalah 50, 75, 100 |
| 3. | Image size |  | Ukuran citra yang digunakan pada penentuan dataset dan proses hyperparameter tuning menggunakan citra 640x640 pixels |
| 4. | Batch size |  | Jumah batch size yang digunakan adalah 16 |
| 5. | Optimizer |  | Optimizer yang digunakan pada pnentuan dataset adlah Stohastic Gradient Descent (SGD) pada proses hyperparameter tuning menggunakan SGD dan Adaptive Moment Estimation Weight Decay (AdamW). |
| 6. | Initial Learning Rate |  | Nilai initial learning rate pada penentuan dataset adalah 0.01, sedangkan pada proses hyperparameter tuning adalah 0.01 dan 0.001. |
| 7. | Final Learning Rate |  | Nilai final learning rate pada penentuan dataset adalah 0.01, sedangkan pada proses hyperparameter tuning adalah 0.1 dan 0.01. |
| 8. | Momentum |  | Pada optimizer AdamW, momentum merupakan nilai Beta1. Nilai momentum yang digunakan adalah 0.937. |
| 9. | Dropout |  | Nilai dropout yang digunakan dalam penentuan dataset adalah 0, sedangkan pada proses hyperparameter tuning adalah 0 dan 0.2. |
| 10. | Mask Ratio |  | Jumlah mask ratio yang digunakan pada penentuan dataset adalah 4, sedangkan pada proses hyperparameter tuning adalah 3 dan 4. |
| 11. | Cosaine learning |  | Nilai cosaine learning yang digunakan bernilai True. |
| 12. | Confidence threshold |  | Nilai confidence thershold yang digunakan pada penentuan dataset adalah 0, sedangkan pada proses hyperparameter tuning adalah 0 dan 0.25 |
| 13. | Intersection over Union (IoU) |  | Nilai IoU yang digunakan pada penentuan dataset adalah 0.7, sedangkan pada proses hyperparamater tuning adalah 0.6 dan 0.7. |
| 14. | Warmup epochs |  | Nilai warmup epochs yang digunakan adalah 10. |
| 15. | Mosaic |  | Nilai mosaic yang digunakan adalah 1.0. |
| 16. | Close mosaic |  | Nilai close mosaic yang digunakan adalah 30. |

## Jadwal Pelaksanaan

Jadwal pelaksanaan tugas akhir dimulai dari persiapan proposal dan wawancara hingga penyelesaian laporan tugas akhir yang dapat dilihat pada Tabel 4.5.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No. | Kegiatan | Bulan/Minggu | | | | | | | | | | | | | | | | | | | | | | | |
| Feb 2025 | | | | Mar 2025 | | | | Apr 2025 | | | | Mei 2025 | | | | Juni 2025 | | | | Juli 2025 | | | |
| 1. | Persiapan Proposal dan Survei |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2. | Penyusunan Proposal |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3. | Perancangan Sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4. | Perancangan Model |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5. | Implementasi Antar Muka |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6. | Pengujian Perangkat Lunak |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7. | Penyusunan Laporan Akhir |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# DAFTAR PUSTAKA