

# Pengembangan Aplikasi *Desktop* dalam Sistem Penyimpanan Otomatis Berbasis Robot dan Visi Komputer

## *Development of a Desktop Application for an Automated Storage System Based on Robotics and Computer Vision*

<sup>1</sup>Ahmad Sahro\*) dan <sup>1</sup>Eko Mursito Budi

<sup>1</sup>Fakultas Teknologi Industri, Institut Teknologi Bandung, Bandung 40132, Indonesia

\*) *corresponding email: ahmadsahro@gmail.com*

### Abstrak

Proses penyimpanan dan pengelolaan barang di sektor industri yang masih dilakukan secara konvensional memiliki risiko kerentanan yang tinggi, seperti kesalahan entri data, ketidakteraturan penempatan barang, serta efisiensi waktu rendah akibat ketergantungan pada intervensi manusia. Fenomena ini menuntut adanya digitalisasi melalui sistem otomasi yang presisi dan terintegrasi. Penelitian ini bertujuan untuk merancang, mengimplementasikan, dan menguji aplikasi *desktop* sebagai unit kendali pusat pada sistem penyimpanan otomatis. Keterbaruan penelitian ini terletak pada pengembangan arsitektur aplikasi yang berfungsi sebagai pusat supervisi. Sistem ini tidak hanya mengelola basis data, tetapi juga mengoordinasikan alur kerja penyimpanan secara *real-time*, komunikasi industri dua arah berbasis protokol FINS, serta melakukan validasi posisi melalui deteksi visual. Metode penelitian yang diterapkan adalah *engineering research* dengan pendekatan *research by project*. Tahapan penelitian disusun secara sistematis meliputi identifikasi kebutuhan, perancangan arsitektur, implementasi kode program, integrasi perangkat keras, hingga pengujian fungsionalitas menyeluruh. Aplikasi dikembangkan menggunakan JavaFX yang diintegrasikan dengan PLC Omron dan modul visi komputer berbasis algoritma YOLOv8. Hasil pengujian menunjukkan bahwa aplikasi mampu menjalankan fungsi supervisi administratif, sinkronisasi komunikasi data dengan PLC, dan integrasi modul visi komputer dengan tingkat stabilitas yang optimal. Penggunaan sistem terbukti mampu meminimalkan kesalahan manusia dan meningkatkan akurasi penempatan barang melalui mekanisme umpan balik visual dan pencatatan data terpusat.

*Kata Kunci* : aplikasi desktop, penyimpanan otomatis, robotika, visi komputer, PLC, sistem supervise

### Abstract

Conventional storage and management processes in industrial sector are highly vulnerable to risks such as data entry errors, disorganized item placement, and time inefficiency due to human intervention. This phenomenon necessitates digitalization through precise and integrated automation. This research aims to design, implement, and test a application as the central control unit. The novelty of this study is the development of an application architecture that functions as a centralized supervision. This system not only manages the database but also coordinates storage workflows in real time, facilitates two-way industrial communication with FINS protocol, and performs position validation via visual detection. The research method applied is *engineering research*, using a *research-by-project* approach. The research stages are systematically, encompassing requirements identification, architecture design, program implementation, hardware integration, and comprehensive functionality testing. The application was developed using JavaFX, integrated with an Omron PLC and a computer vision module based on the YOLOv8 algorithm. Test results demonstrate that the application can perform administrative supervision functions, synchronize data communication with the PLC, and integrate the computer vision module with optimal stability. The implementation of this system is proven to minimize human error and improve the accuracy of goods placement through visual feedback mechanisms and centralized data.

*Keyword* : desktop application, automated storage, robotics, computer vision, PLC, supervisory system

Makalah diterima 08 Februari 2026– makalah direvisi 22 April 2026 – disetujui 24 April 2026

Karya ini adalah naskah akses terbuka dengan lisensi [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/).



## 1 Pendahuluan

Proses simpan dan ambil barang dalam penyimpanan yang dilakukan secara manual berpotensi menimbulkan kesalahan pencatatan, pencampuran barang, serta ketergantungan tinggi terhadap intervensi manusia. Perkembangan teknologi penyimpanan otomatis telah banyak diteliti untuk meningkatkan kinerja proses seperti pada penelitian PLC Based Automated Storage [1] dan Automatic Storage Retrieval System [2]. Penelitian telah membahas pemanfaatan sistem informasi dan otomasi untuk meningkatkan akurasi dan manajemen data. Namun, sebagian besar sistem yang dikembangkan masih berfokus pada pengelolaan data

secara perangkat lunak atau maksimal hanya integrasi dengan sistem robotik (otomasi), hal ini memungkinkan terjadi suatu gap error yang muncul secara bertahap dan dapat terakumulasi. Penelitian ini mengusulkan sistem penyimpanan barang otomatis yang terintegrasi, di mana aplikasi desktop berperan sebagai pusat supervisi sistem [3],[4]. Aplikasi ini dirancang untuk mengelola proses penyimpanan barang, mengoordinasikan operasi robot melalui komunikasi industri [5],[6], serta melakukan validasi keberadaan dan posisi barang menggunakan sistem visi komputer [7] [8],[9] untuk mengoreksi data error kumulatif yang terjadi.

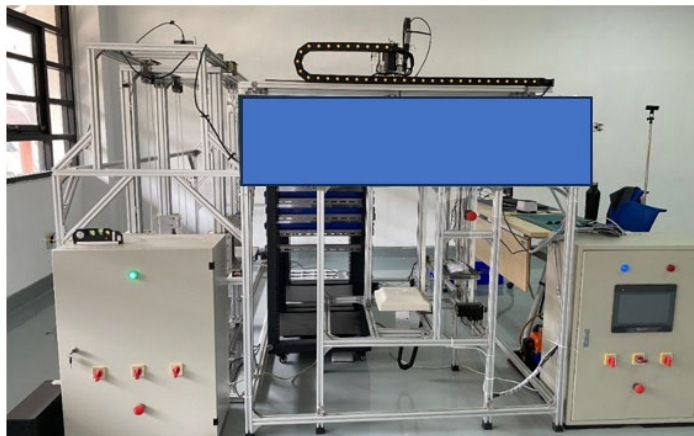
Fokus utama penelitian ini terletak pada perancangan dan implementasi aplikasi desktop sebagai sistem supervisi terintegrasi yang menyatukan sistem penyimpanan barang, komunikasi industri berbasis protokol FINS, dan validasi berbasis visi komputer dalam satu sistem PLC [10],[11].

## 2 Metode Penelitian

Penelitian ini menggunakan pendekatan *engineering research* yang berfokus pada perancangan dan pengembangan sistem untuk menyelesaikan permasalahan nyata. Metode penelitian disusun agar proses perancangan, implementasi, dan pengujian aplikasi [12],[13],[14] dapat dilakukan secara terstruktur dan sistematis. Metode yang diterapkan mencakup jenis pendekatan penelitian, penyusunan tahapan penelitian, perancangan arsitektur sistem, perancangan aplikasi, perancangan basis data serta integrasi sistem.

### 2.1 Jenis Pendekatan Penelitian

Penelitian ini termasuk dalam jenis *engineering research* yang berfokus pada perancangan dan pengembangan solusi berbasis teknologi untuk menyelesaikan permasalahan.



Gambar 1. Prototipe yang dibangun

Pendekatan yang digunakan adalah *research by project*, di mana penelitian dilakukan melalui perancangan, implementasi, dan evaluasi langsung terhadap sistem. Gambar 1 merupakan prototipe sistem yang dikembangkan. Pendekatan ini dipilih karena tujuan utama penelitian adalah menghasilkan sebuah aplikasi *desktop* yang berfungsi sebagai sistem utama dalam proses penyimpanan berbasis robot dan visi komputer.

### 2.2 Tahapan Penelitian

Pengembangan sistem ini menerapkan metode Scrum [15], yang bersifat iteratif dan inkremental. Metode Scrum memungkinkan pengembangan aplikasi dilakukan secara bertahap, fleksibel, dan berkelanjutan. Gambar 2 adalah diagram Scrum yang akan digunakan dalam penelitian ini.



Gambar 2. Diagram Scrum yang digunakan dalam penelitian

**Tahap Analisis:** mengidentifikasi kebutuhan fungsional dan nonfungsional sistem, termasuk pemetaan proses penyimpanan barang, kebutuhan aplikasi *desktop*, serta kebutuhan integrasi sistem.

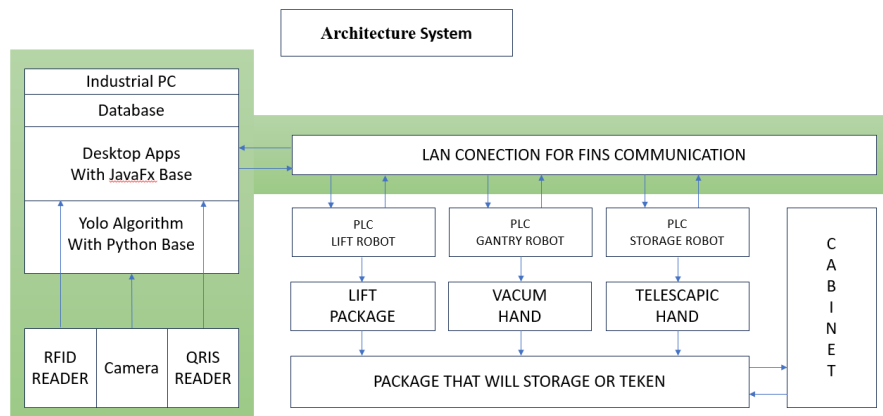
**Tahap Implementasi:** proses pengembangan aplikasi *desktop* berdasarkan hasil analisis dan perancangan.

**Tahap Pengujian:** memastikan bahwa setiap fungsi yang dikembangkan berjalan sesuai dengan kebutuhan. Pengujian meliputi pengujian fungsional aplikasi *desktop* dan pengujian integrasi antar subsistem.

**Tahap Evaluasi:** menilai kinerja sistem untuk evaluasi, perbaikan dan pengembangan berkelanjutan. pada siklus Scrum berikutnya hingga aplikasi dinyatakan stabil dan sesuai dengan tujuan Penelitian.

### 2.3 Arsitektur Sistem yang Dikembangkan

Arsitektur sistem pada penelitian dirancang untuk memadukan sistem pencatatan dan pengelolaan data berbasis aplikasi *desktop* dengan eksekusi proses fisik di dunia nyata yang dilakukan oleh sistem robotik penyimpanan. Aplikasi *desktop* berperan mengelola data, alur proses, dan supervisi, sedangkan sistem robotik [16],[17] berfungsi mengeksekusi perintah dan instruksi yang diberikan aplikasi.



Gambar 3. Arsitektur sistem yang dikembangkan

Gambar 3 menunjukkan bahwa sistem dibagi menjadi dua bagian, yaitu sisi perangkat lunak (*background* hijau) yang berpusat pada *Industrial PC*, dan perangkat keras yang terdiri dari sistem robotik penyimpanan dan PLC sebagai pengendali utama. Pada penelitian ini fokus pembahasan adalah arsitektur perangkat lunak dengan difokuskan pada fitur visi komputer. Pembahasan mengenai perancangan mekanikal robot telah dibahas secara khusus oleh Salman Satria dalam skripsinya [18].

### 2.4 Perancangan Aplikasi *Desktop*

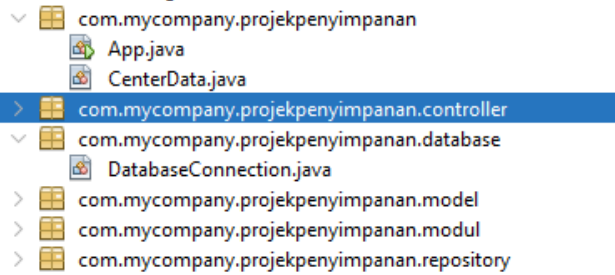
Gambar 4 merupakan struktur perancangan aplikasi *desktop* dilakukan, dengan pendekatan *modular* [19] memastikan sistem mudah dikembangkan, dipelihara, dan diperluas. Pembagian modul dalam aplikasi bertujuan agar alur kerja sistem penyimpanan barang lebih terstruktur, mulai dari proses autentikasi pengguna, identifikasi barang, komunikasi dengan sistem robotik, hingga pengelolaan dan pencatatan data.



Gambar 4. Diagram *controller* sebagai pusat *activity* and *logic*

Dengan struktur ini, aplikasi dapat berfungsi sebagai pusat kendali dan supervisi yang mengoordinasikan interaksi antara pengguna, perangkat identifikasi, sistem visi komputer, PLC, serta basis data. Setiap modul dirancang agar saling terintegrasi namun tetap memiliki batasan tanggung jawab yang jelas. Dengan *controller* berperan sebagai pusat kendali sistem yang akan menjadi tumpuan kerja. Modul Antarmuka untuk tampilan,

Modul *Auth* menangani autentikasi pengguna dan pengelolaan alur kerja, Modul Komunikasi *FINS* untuk komunikasi, integrasi sistem visi komputer, identifikasi barang menggunakan RFID [20],[21],[22] dan barcode [23],[24], serta pengelolaan data dan *log* sistem. Pendekatan ini juga mendukung prinsip *scalability*, di mana penambahan fitur atau perangkat baru dapat dilakukan melalui penambahan atau pengembangan modul terkait tanpa harus mengubah struktur aplikasi secara keseluruhan.

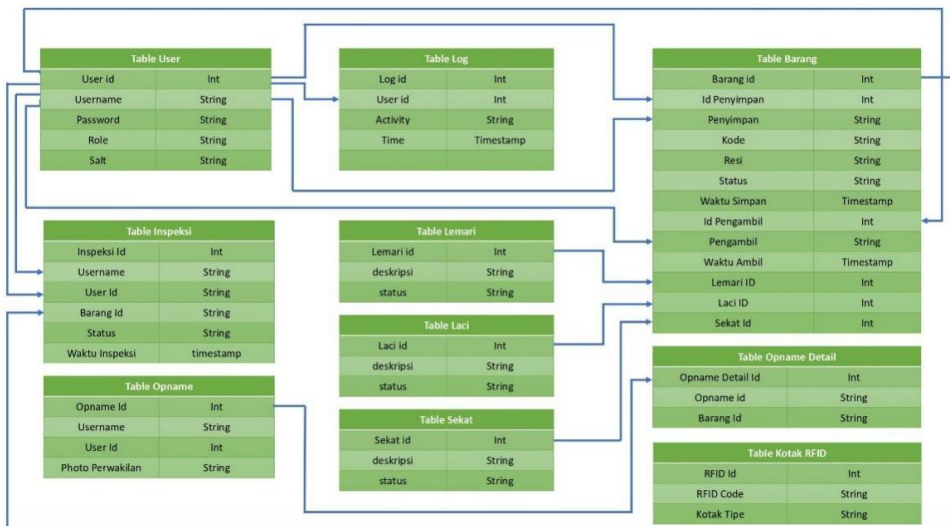


Gambar 5. Struktur folder yang digunakan agar *scalable*

Gambar 5 menunjukkan pembagian struktur folder untuk menjaga keteraturan dan kemudahan pemeliharaan, aplikasi *desktop* ini dirancang dengan struktur perangkat lunak yang terorganisir. Struktur aplikasi dibagi ke dalam beberapa lapisan utama, yaitu *Model*, *Repository*, *Controller*, *View*, serta modul *Third Party*. *Model* berfungsi merepresentasikan data dan entitas sistem, *Repository* bertanggung jawab terhadap pengelolaan akses basis data, *Controller* mengelola logika aplikasi dan alur proses, sedangkan *View* menangani tampilan antarmuka pengguna. Pemisahan struktur tersebut membantu memastikan bahwa setiap bagian aplikasi memiliki tanggung jawab yang jelas dan tidak saling bergantung secara langsung. Pendekatan ini mendukung pengembangan aplikasi yang lebih terstruktur, mudah dipelihara, dan siap untuk pengembangan lanjutan.

## 2.5 Perancangan Basis Data

Dalam arsitektur sistem, *database MySQL* [25] berperan sebagai pusat penyimpanan data digital yang menjembatani proses operasional antara aplikasi dan sistem fisik robotik. Setiap aktivitas simpan dan ambil barang yang dijalankan oleh sistem dicatat dalam *database* sebagai data transaksi. Pencatatan ini mencakup informasi identitas barang, waktu transaksi, jenis tindakan, serta status eksekusi sistem.



Gambar 6. Rancangan relasi antar *table*

Pada Gambar 6 merupakan desain relasi *table* pada *Database*. *Database* juga digunakan untuk menyimpan data pemetaan lokasi penyimpanan, di mana setiap barang yang disimpan memiliki relasi dengan posisi fisik tertentu di dalam sistem lemari penyimpanan bersekat.

## 2.6 Pembagian *Role*

Penelitian ini mengembangkan sistem *multi role*, hal ini dilakukan agar aplikasi memiliki sistem terpadu dan *real live industry*:

- **Role Penyimpan** = memiliki halaman untuk proses menyimpan dan mengambil barang yang meliputi halaman simpan, halaman simpan banyak, dan halaman ambil

- **Role Supervisor** = memiliki halaman untuk proses supervisi yang meliputi daftar barang dilengkapi dengan detail barang dan fitur inspeksi, daftar riwayat yang dilengkapi fitur detail riwayat barang, dan halaman opname yang dilengkapi detail data opname
- **Role Teknisi** = memiliki halaman untuk proses pemeliharaan dan pengembangan sstem yang meliputi daftar user yang digunakan untuk mengelola akses, halaman daftar lemari, daftar laci, dan daftar sekat untuk mengelola jumlah penyimpanan, dan halaman Daftar RFID untuk kebutuhan mendaftarkan jumlah kotak

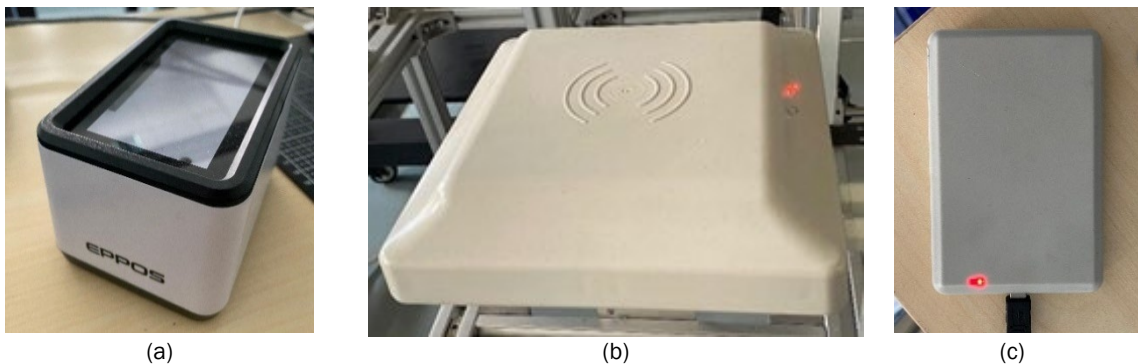
## 2.7 Integrasi Sistem

Integrasi sistem dilakukan untuk memastikan seluruh komponen pendukung dapat berfungsi secara terpadu dalam mendukung operasional sistem penyimpanan otomatis. Aplikasi *desktop* berperan sebagai pusat pengendali dan supervisi yang mengoordinasikan pertukaran data dan perintah antara modul-modul pendukung serta subsistem lainnya.

Daftar list modul pendukung meliputi :

- Modul Identifikasi Barang (RFID dan *Barcode*)

Modul ini bertanggung jawab dalam proses pengenalan dan validasi identitas barang serta pembaca data untuk disimpan. Modul mengintegrasikan perangkat *barcode scanner* dan RFID reader sebagai sumber data identifikasi utama dalam proses simpan dan ambil barang.

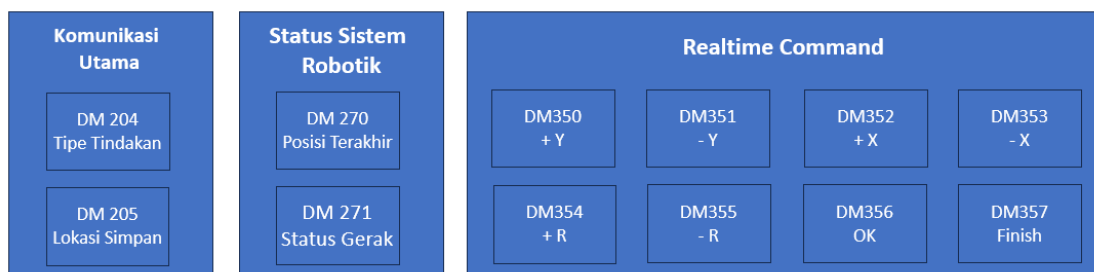


Gambar 7. (a) EPPOS Scanner, (b), RFID UHF IP, (c) RFID UHF USB

Gambar 7 a menunjukkan barcode scanner sementara b dan c menunjukkan modul RFID yang digunakan, modul tergabung dalam Sistem identifikasi yang mengintegrasikan teknologi pemindaian untuk menjamin akurasi serta keamanan data transaksi. *Barcode scanner* berfungsi membaca kode unik pada label barang kemudian digunakan oleh aplikasi sebagai *key* dalam pencatatan dan pencarian data. Secara paralel, RFID reader digunakan untuk memvalidasi identitas kotak penyimpanan melalui *tag* unik.

- Modul Komunikasi *FINS*

Protokol *FINS* (Factory Interface Network Service) digunakan sebagai media komunikasi antara aplikasi dengan PLC Omron yang mengendalikan sistem robotik. Dibandingkan metode *middleware* atau *fieldbus*, *FINS* menawarkan keterhubungan langsung dengan struktur memori PLC tanpa memerlukan lapisan konversi data tambahan.



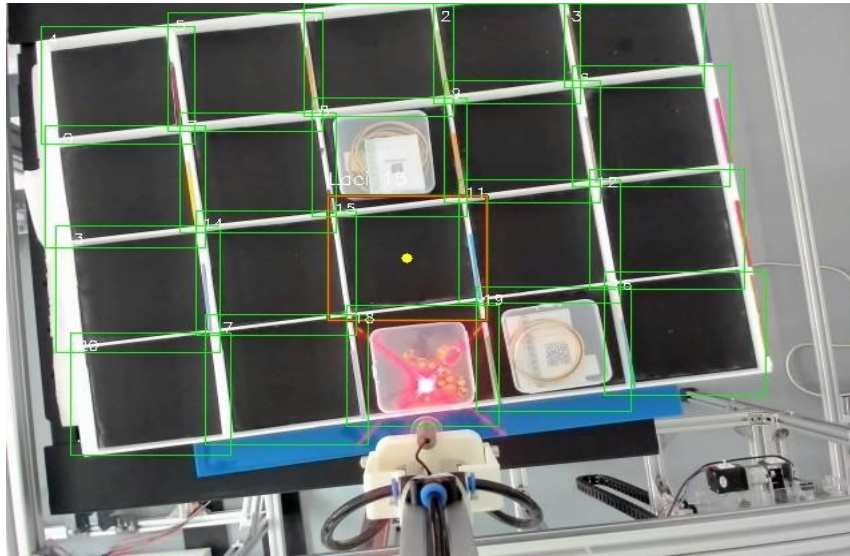
Gambar 8. Pin DM yang digunakan

Gambar 8 menunjukkan implementasi *FINS* dari sisi PLC, *area Data Memory* (DM) pada PLC dipilih sebagai antarmuka utama pertukaran data. Pemilihan *area DM* didasarkan pada fleksibilitas akses, sehingga memudahkan proses pengembangan aplikasi supervisi dan mengurangi kompleksitas integrasi. Untuk menjaga keteraturan komunikasi dan mempermudah pengelolaan data, area DM dibagi menjadi

beberapa kelompok data berdasarkan fungsinya. Kelompok pertama adalah DM Data Komunikasi Utama, yang digunakan untuk menerima perintah utama dari aplikasi *desktop*. Kelompok kedua DM Data Status Sistem Robotik digunakan untuk memonitor sistem robotik. Dan yang ketiga DM *Realtime Command* digunakan untuk mengendalikan robot secara manual.

- Modul Visi Komputer

Meskipun pergerakan utama robot dikendalikan oleh PLC berdasarkan *sequence* yang telah ditentukan, terdapat keterbatasan akurasi pemosisian apabila sepenuhnya hanya mengandalkan sistem mekanik terkontrol. Perbedaan toleransi mekanik, variasi posisi kotak, serta potensi pergeseran kecil pada laci penyimpanan menyebabkan dibutuhkan sistem verifikasi tambahan.



Gambar 9. Process YOLO dalam mendeteksi sekat kosong

Gambar 9 menunjukkan implementasi YOLO (*You Only Look Once*) yang merupakan salah satu algoritma *deep learning* untuk deteksi objek yang bekerja dengan pendekatan *single-stage detector*. Algoritma YOLO dipilih karena karakteristiknya sesuai dengan kebutuhan sistem pemosisian *real-time*. YOLO memiliki waktu inferensi yang cepat dibandingkan dengan metode deteksi lain seperti R-CNN, Fast R-CNN, atau Faster R-CNN.

### 3 Hasil dan Diskusi

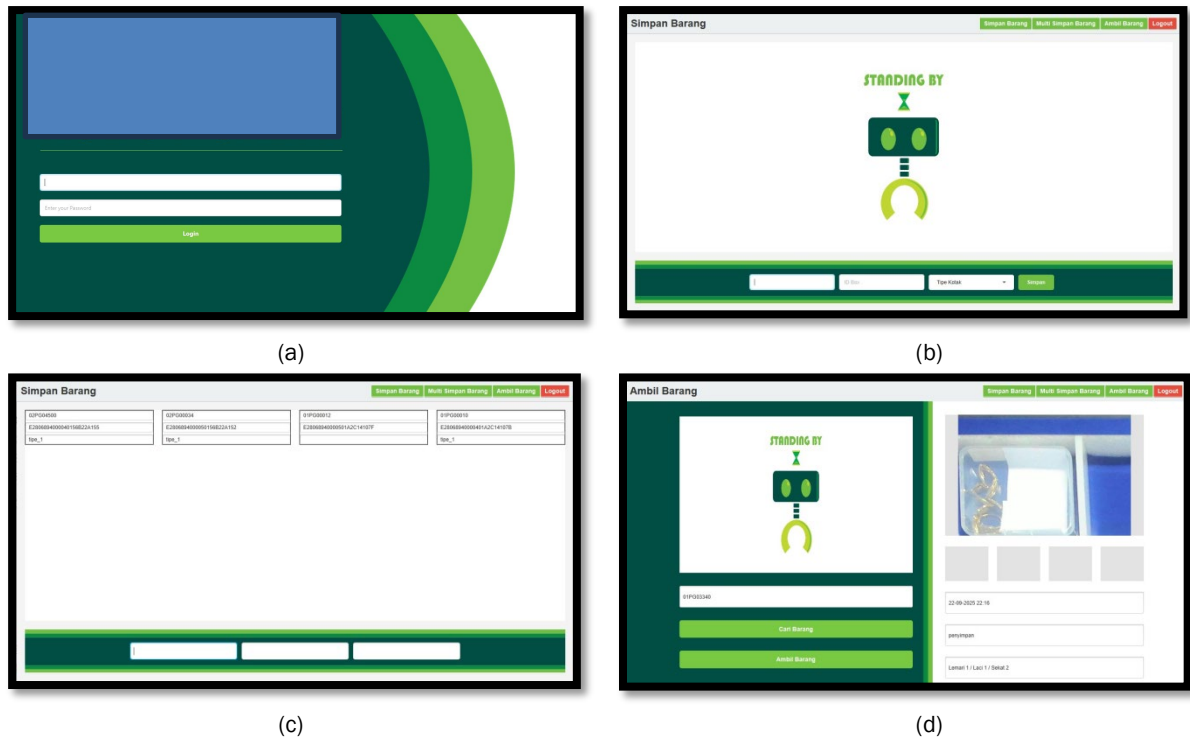
Bagian ini menyajikan hasil implementasi sistem serta pembahasan mengenai fungsi dan kinerja aplikasi *desktop* yang dikembangkan. Hasil penelitian difokuskan pada evaluasi aplikasi sebagai keluaran utama penelitian, termasuk implementasi antarmuka, fungsi setiap halaman, serta integrasi modul dan evaluasi pelatihan sistem visi komputer YOLO.

#### 3.1 Fungsi dan Antar Muka Aplikasi

Dalam sistem yang dikembangkan, aplikasi *desktop* digunakan oleh beberapa aktor dengan peran dan tanggung jawab yang berbeda, yaitu penyimpanan, supervisor, dan teknisi. Masing-masing aktor memiliki kebutuhan informasi, tingkat akses, serta bentuk interaksi yang berbeda terhadap sistem. Perancangan antarmuka aplikasi pada penelitian ini difokuskan untuk mendukung proses operasional utama, pengawasan sistem, serta kegiatan konfigurasi dan pemeliharaan. Setiap tampilan dirancang agar mampu menyajikan informasi yang relevan bagi pengguna tanpa menimbulkan kompleksitas yang tidak diperlukan.

- Antarmuka aplikasi untuk penyimpanan

Antarmuka aplikasi dirancang sebagai antarmuka operasional dalam proses penyimpanan barang. Memungkinkan pengguna berinteraksi dengan sistem robotik dan perangkat identifikasi secara jelas, responsif, dan tanpa kesalahan.



Gambar 10. Empat halaman milik penyimpan : (a) halaman *Login* , (b) halaman *Simpan* , (c) halaman *Simpan Banyak* (d) halaman *Ambil*

**Gambar 10 (a) Halaman *Login*** : berfungsi sebagai gerbang awal sistem untuk memastikan bahwa hanya pengguna yang berwenang yang dapat mengakses aplikasi. Untuk pengguna dengan peran penyimpan, sistem secara otomatis mengarahkan ke halaman *Simpan* tanpa perlu memilih menu tambahan. Mekanisme ini dirancang untuk mempercepat alur kerja dan mengurangi langkah yang tidak diperlukan.

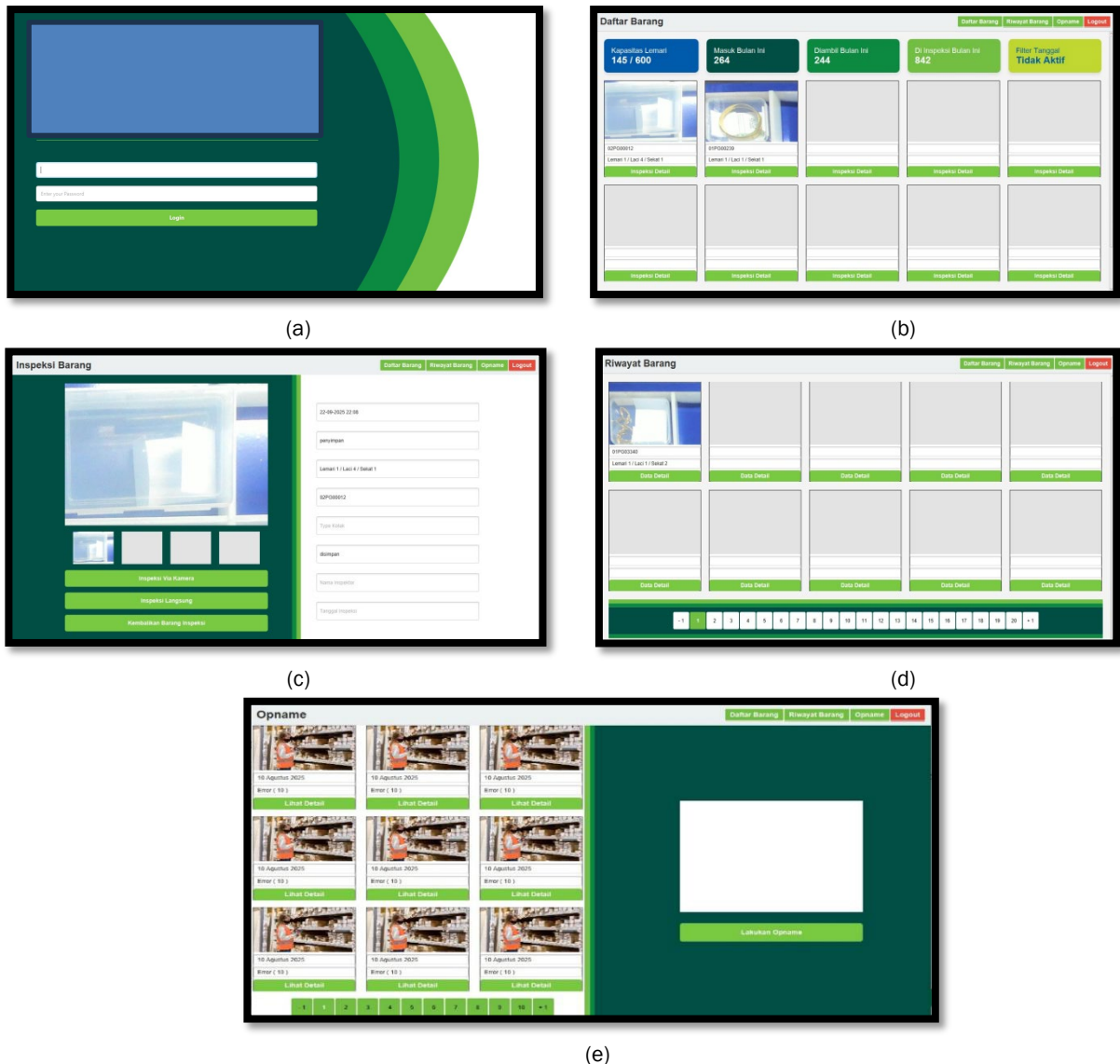
**Gambar 10 (b) Halaman *Simpan*** : dirancang untuk menampilkan status sistem dan robot secara *real-time*. Ketika proses penyimpanan dijalankan, aplikasi akan menampilkan tahapan proses yang sedang berlangsung, seperti pergerakan robot, status eksekusi PLC, serta hasil pembacaan perangkat identifikasi.

**Gambar 10 (c) Halaman *Multi Simpan*** : digunakan untuk mendukung proses penyimpanan banyak barang secara berurutan tanpa harus kembali ke halaman *simpan* untuk setiap *item*. Halaman ini memungkinkan penyimpan untuk melakukan pemindaian barang dan kotak penyimpanan secara berulang, sementara aplikasi secara otomatis mengatur urutan proses penyimpanan dan alokasi sekat yang tersedia.

**Gambar 10 (d) Halaman *Ambil*** : digunakan oleh penyimpan untuk melakukan proses pengambilan barang dari sistem penyimpanan otomatis. Pada halaman ini, penyimpan dapat memasukkan data barang yang akan diambil berdasarkan informasi pada *barcode* barang. Setelah data ditemukan, aplikasi akan mengirimkan perintah ke sistem robotik untuk mengambil barang dari lokasi penyimpanan yang sesuai. Selama proses berlangsung, halaman *ambil* menampilkan status pergerakan robot dan konfirmasi tahapan proses secara *real-time*, sehingga penyimpan dapat memastikan bahwa proses pengambilan berjalan.

- Antarmuka aplikasi untuk *supervisor*

Antarmuka aplikasi untuk *supervisor* dirancang sebagai antarmuka pengawasan dan verifikasi terhadap seluruh aktivitas penyimpanan dan pengambilan barang. Berbeda dengan penyimpan yang berfokus pada operasi langsung, *supervisor* berperan dalam memastikan kesesuaian data, kondisi fisik barang, serta akurasi sistem secara keseluruhan. Oleh karena itu, antarmuka ini menitikberatkan pada kelengkapan informasi, kemudahan penelusuran data, serta dukungan proses inspeksi.



Gambar 11. Lima halaman milik Penyimpan : (a) halaman *Login* , (b) halaman *Daftar Barang* , (c) halaman *Inspeksi Barang* (d) halaman *Riwayat Barang* , (e) halaman *Opname*.

**Gambar 11(a) Halaman *Login*** : sama dengan halaman *login* yang telah dibahas sebelumnya. Namun ketika memasukkan identitas *authentication supervisor* akan masuk ke halaman *supervisor*.

**Gambar 11(b) Halaman *Daftar Barang*** : Melalui antarmuka ini, *supervisor* dapat melihat semua daftar barang yang masih disimpan di sistem penyimpanan juga *user* dapat memilih barang yang ingin di inspeksi.

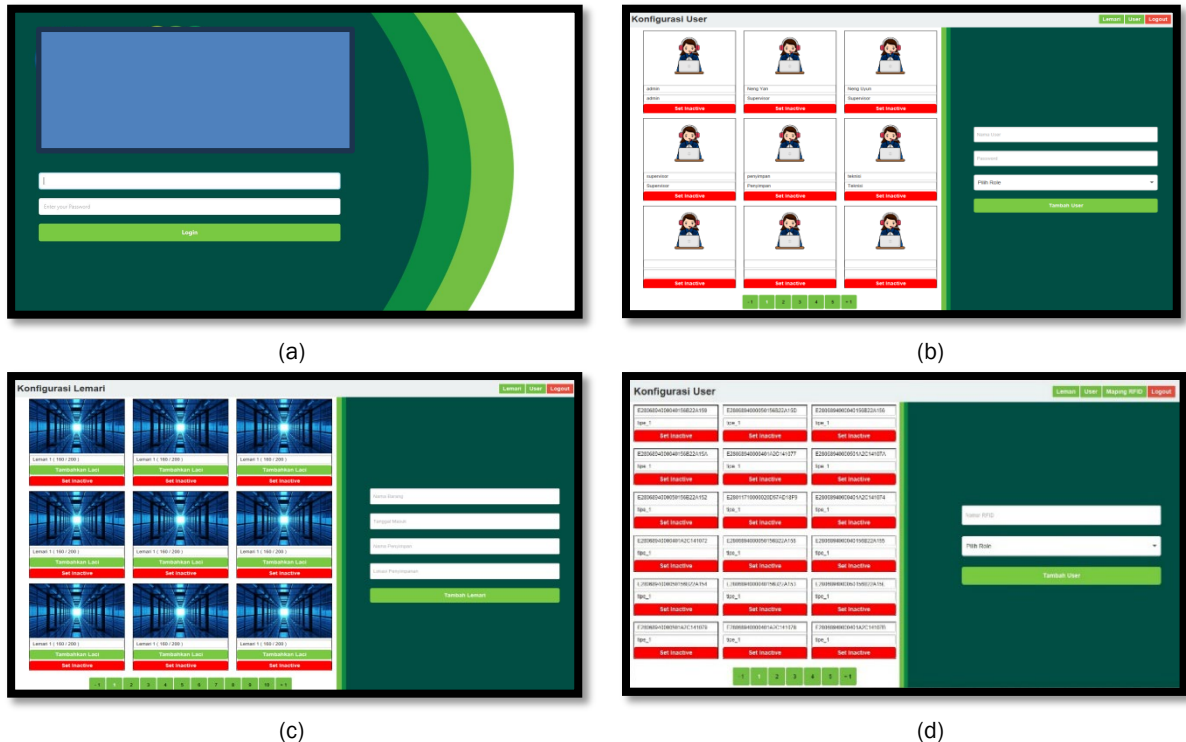
**Gambar 11(c) Halaman *Inspeksi Barang*** : Halaman inspeksi barang digunakan oleh *supervisor* untuk melakukan pemeriksaan kondisi barang yang tersimpan. Pada halaman ini, *supervisor* dapat melihat data detail barang beserta gambar terakhir yang tersimpan di dalam sistem.

**Gambar 11 (d) Halaman *Riwayat Barang*** : Halaman riwayat barang berfungsi untuk menampilkan catatan aktivitas setiap barang yang tersimpan di dalam sistem. Riwayat ini mencakup proses penyimpanan, pengambilan, inspeksi, serta pengembalian barang beserta informasi waktu dan pengguna yang terlibat.

**Gambar 11(e). Halaman *Stock Opname*** : Halaman *stock opname* digunakan untuk melakukan pengecekan menyeluruh terhadap kondisi seluruh barang yang tersimpan di dalam sistem dengan membandingkan data barang yang tercatat di *database* dengan kondisi aktual yang diperoleh melalui visi kamera.

- Antarmuka Aplikasi Untuk teknisi

Antarmuka aplikasi untuk teknisi dirancang sebagai antarmuka administratif dan teknis yang digunakan untuk melakukan pengaturan, konfigurasi, serta pemeliharaan sistem. Berbeda dengan *supervisor*, teknisi memiliki peran dalam memastikan sistem dapat beradaptasi terhadap perubahan konfigurasi fisik maupun kebutuhan operasional tanpa memerlukan perubahan pada kode program.



Gambar 12. empat Halaman milik Teknisi : (a) halaman *Login* , (b) halaman *Daftar User* , (c) halaman *Daftar Lemari* , (d) halaman *Daftar RFID*.

**Gambar 12(a). Halaman *Login*** : Halaman *login* ini sama dengan halaman *login* yang telah dibahas sebelumnya. Namun ketika memasukkan identitas *authentication* teknisi akan masuk ke halaman teknisi.

**Gambar 12(b). Halaman *Konfigurasi User*** : Halaman konfigurasi *user* digunakan oleh teknisi untuk mengelola akun pengguna yang terdaftar di dalam sistem. Pada halaman ini, teknisi dapat menambahkan pengguna baru, mengubah data pengguna yang sudah ada, serta mengatur peran atau hak akses sesuai dengan kebutuhan operasional. Peran pengguna dibedakan menjadi penyimpan, *supervisor*, dan teknisi.

**Gambar 12(c). Halaman *Konfigurasi Lemari & Laci*** : Halaman konfigurasi lemari digunakan untuk mengelola struktur fisik penyimpanan di dalam sistem. Pada halaman ini, teknisi dapat menambahkan, mengubah, atau menghapus data lemari, laci, serta jumlah sekat yang tersedia tanpa perlu melakukan pemrograman ulang pada aplikasi. Setiap konfigurasi yang dilakukan akan langsung disinkronkan dengan *database* sistem.

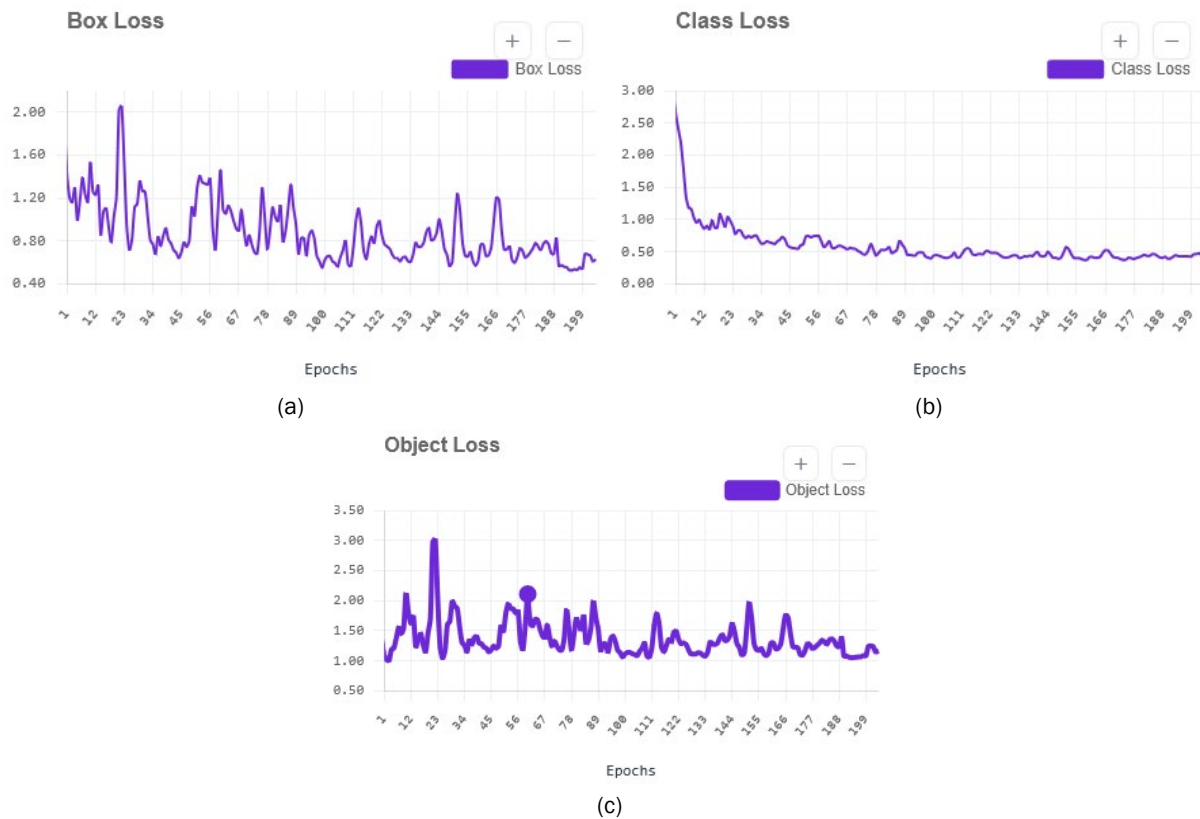
**Gambar 12(d). Halaman *Register RFID*** : Halaman *register RFID* digunakan untuk mendaftarkan *tag* RFID yang akan digunakan sebagai identitas kotak penyimpanan di dalam sistem. Setiap kotak penyimpanan memiliki satu RFID unik yang disimpan di *database* bersamaan dengan informasi tipe kotak dan keterkaitannya dengan struktur lemari dan laci.

### 3.2 Sistem Visi Komputer

Subbab ini membahas hasil pelatihan, evaluasi, serta penerapan sistem visi komputer yang digunakan dalam penelitian. *Model* deteksi objek dikembangkan menggunakan algoritma YOLOv8 dengan *Dataset* sebanyak 120 citra yang dikumpulkan secara langsung pada lingkungan prototipe sistem pada rentang waktu pukul 07.00 hingga 18.00 untuk merepresentasikan variasi kondisi pencahayaan. *Dataset* kemudian diproses menggunakan *platform* Roboflow untuk anotasi, manajemen *dataset*, serta pembagian data pelatihan dan pengujian dan proporsi pembagian data ditetapkan sebesar 70% untuk pelatihan dan 30% untuk pengujian.

Pada bagian berikutnya akan dipaparkan hasil pelatihan *model*, meliputi perkembangan nilai *loss*, evaluasi metrik akurasi, analisis kesalahan deteksi, serta penerapan *model* dalam sistem robotik yang terintegrasi.

### 3.2.1 Performa Model



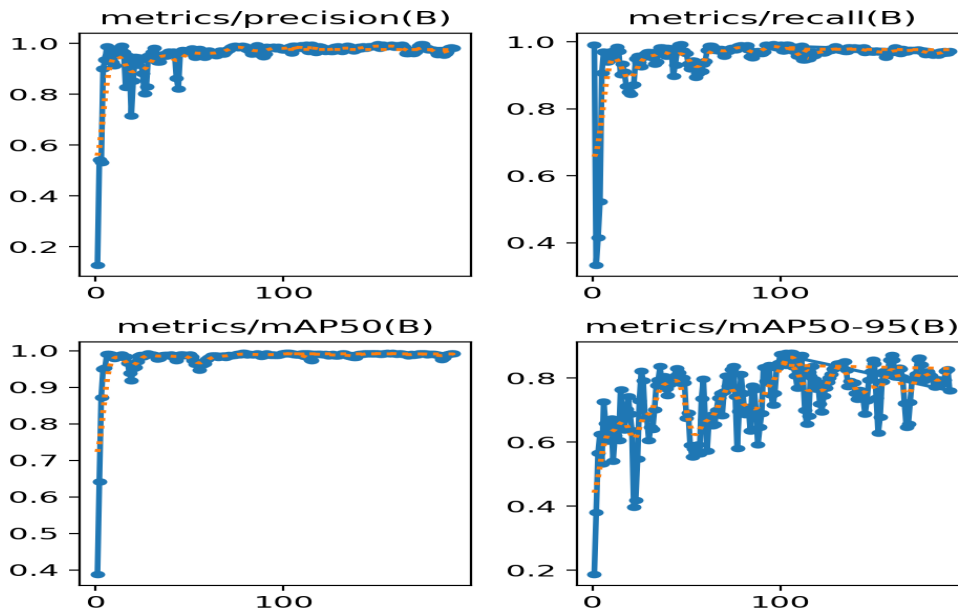
Gambar 13. Performa *model* meliputi (a) *Box Loss*, (b) *Class Loss* dan (c) *Object Loss*

Gambar 13 (a), (b) dan (c) menunjukkan proses pelatihan *model* dilakukan hingga mencapai 200 *epoch* untuk memastikan kematangan bobot *model*. Evaluasi konvergensi dilakukan dengan mengamati grafik *Box Loss* pada Gambar 13 (a), *Class Loss* pada Gambar 13 (b), dan *Object Loss* pada Gambar 13 (c) menunjukkan tren penurunan yang signifikan pada fase awal pelatihan. Pada parameter *Box Loss*, nilai awal yang berfluktuasi di atas 0.8 mulai menunjukkan stabilitas setelah melewati *epoch* ke-70 dan akhirnya konvergen pada kisaran 0.51 hingga 0.55 di fase akhir. Pola serupa terlihat pada *Class Loss*, di mana nilai yang awalnya berada di atas 3.0 menurun drastis dalam 70 *epoch* pertama dan melandai secara konsisten di angka  $\pm 0.4$ , yang mengindikasikan bahwa *model* telah mampu melakukan klasifikasi objek dengan tingkat kepercayaan yang tinggi.

Meskipun terdapat anomali berupa penurunan performa sesaat (*dip*) pada rentang *epoch* 60 hingga 70, *model* menunjukkan kemampuan pemulihan (*recovery*) yang cepat dan kembali stabil hingga akhir sesi pelatihan. *Object Loss* juga terpantau stabil di kisaran 0.87 tanpa adanya lonjakan yang mengindikasikan ketidakstabilan deteksi. Secara keseluruhan, tidak ditemukan indikasi *overfitting* yang mencolok karena kurva *loss* tetap berada pada tren mendatar (*plateau*) dan tidak menunjukkan kenaikan kembali.

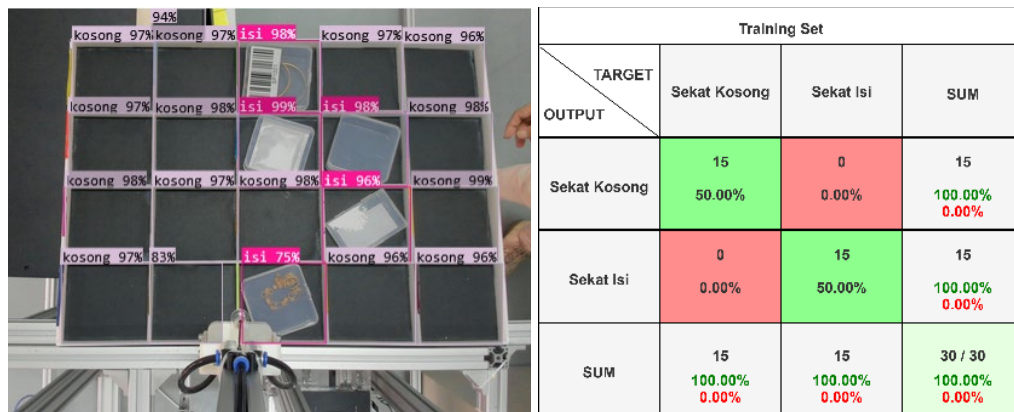
### 3.2.2 Metrik Validation (Precision, Recall, dan mAP)

Gambar 14 menunjukkan hasil evaluasi performa *model* dilakukan dengan menganalisis metrik *Precision*, *Recall*, dan *mean Average Precision* (mAP). Berdasarkan hasil pengujian, *model* menunjukkan performa yang sangat impresif dengan nilai *Precision* sebesar 98,2% dan *Recall* sebesar 97,5%. Hal ini mengindikasikan bahwa *model* tidak hanya akurat dalam memprediksi kelas objek secara tepat, tetapi juga sangat sensitif dalam mendeteksi seluruh objek yang ada pada *dataset* tanpa banyak yang terlewat. Nilai mAP@50 mencapai 99,1%, yang menunjukkan tingkat akurasi rata-rata yang sangat tinggi pada ambang batas *Intersection over Union* (IoU) 0,5. Meskipun grafik mAP50-95 menunjukkan fluktuasi yang lebih dinamis karena standar evaluasi yang lebih ketat, nilai akhirnya tetap stabil di kisaran yang tinggi, yang membuktikan ketahanan *model* terhadap berbagai variasi posisi dan ukuran objek.



Gambar 14. Metrik Precision, Recall, dan mean Average Precision (mAP)

### 3.2.3 Analisis Confusion Matrix



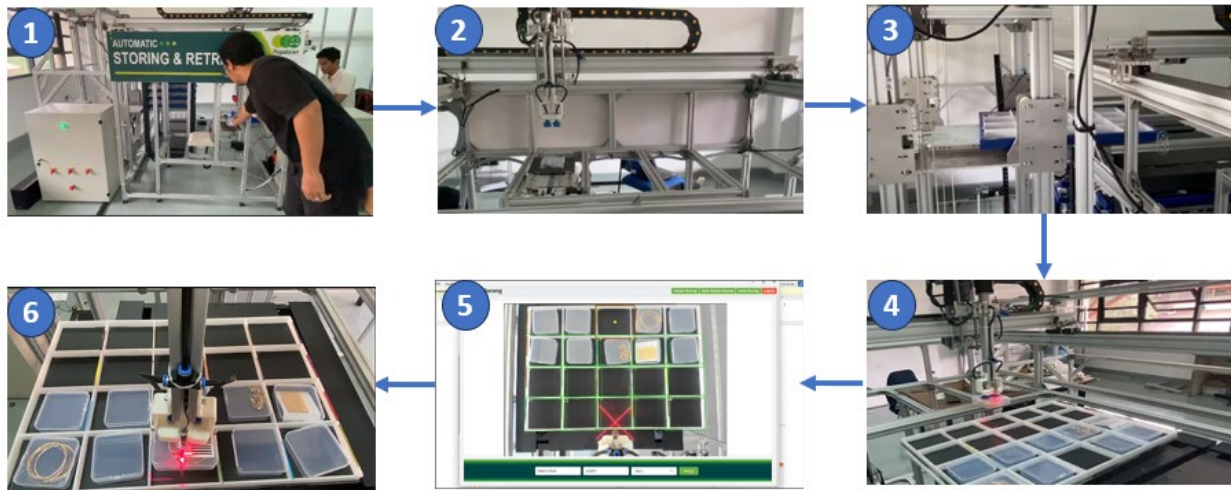
Gambar 15. Testing dan Confusion Matrix.

Gambar 15 menunjukkan Evaluasi lebih lanjut dilakukan menggunakan *Confusion Matrix* untuk melihat performa klasifikasi *model* terhadap kelas "Sekat Kosong" dan "Sekat Isi". Berdasarkan tabel hasil pengujian, *model* mencapai tingkat akurasi sebesar 100% pada sampel yang diuji. Dari total 15 data aktual untuk kategori "Sekat Kosong", *model* berhasil memprediksi seluruhnya dengan benar (15 *True Negative*) tanpa ada kesalahan klasifikasi menjadi "Sekat Isi". Demikian pula pada kategori "Sekat Isi", sebanyak 15 data aktual diprediksi secara akurat (15 *True Positive*) oleh *model*.

Hasil ini menunjukkan bahwa tidak terdapat nilai *False Positive* maupun *False Negative* dalam pengujian tersebut, yang berarti *model* memiliki kemampuan diskriminasi yang sangat baik dalam membedakan antara kompartemen yang terisi objek dengan kompartemen yang kosong. Nilai presisi dan *recall* yang mencapai 100% pada tabel ini memperkuat temuan pada grafik pelatihan sebelumnya bahwa *model* YOLOv8 yang dilatih telah konvergen secara optimal dan sangat reliabel untuk digunakan dalam tugas deteksi sekat ini.

### 3.2.4 Penerapan dalam Sistem

Gambar 14 menunjukkan alur proses penyimpanan barang mulai 1 user memasukkan paket, Robot 1 dan Robot 2 memposisikan paket pada area transisi, sementara Robot 3 menyiapkan laci penyimpanan sesuai koordinat yang telah ditentukan (meliputi proses 2, 3, dan 4 *sequence*). Sinkronisasi memastikan kesiapan mekanik sebelum proses identifikasi visual. Kamera dipasang secara tetap pada posisi tetap diatas bidang kerja sehingga sudut pandang dan skala citra konsisten pada setiap siklus. Desain statis ini meningkatkan *repeatability* dan mengurangi kesalahan.



Gambar 14. Alur proses integrasi penggunaan modul robot dan visi komputer

Pada titik pengambilan citra (*sequence 5*), Robot 2 berhenti untuk melakukan akuisisi gambar. Model YOLOv8 memproses citra untuk mendeteksi *slot target* sesuai data pada *database*, kemudian menentukan titik pusat objek sebagai representasi posisi aktual. Koordinat piksel hasil deteksi ditransformasikan ke koordinat dunia menggunakan parameter kalibrasi (*sequence 6*). Selisih antara posisi aktual dan posisi referensi dihitung sebagai nilai koreksi terhadap sumbu pergerakan robot. Data koreksi ini dikirimkan ke pengendali *robot* untuk melakukan penyesuaian akhir sebelum paket dipindahkan ke dalam laci. Integrasi antara desain mekanik yang stabil, sistem visi terkalibrasi, dan kendali berbasis koreksi menghasilkan proses penyimpanan yang presisi, adaptif, dan konsisten dalam implementasi praktis.

## 4 Kesimpulan

Penelitian ini berhasil mengembangkan sistem penyimpanan otomatis berbasis robot yang terintegrasi dengan aplikasi *desktop* dan sistem visi komputer dalam satu arsitektur berbasis *database*. Perancangan antarmuka berbasis peran, yaitu penyimpan, supervisor, dan teknisi, mampu mendukung proses operasional, pengawasan, serta konfigurasi sistem secara terstruktur dan efisien. Dari sisi kecerdasan visual, model deteksi berbasis YOLOv8 menunjukkan performa yang sangat baik dengan nilai *Precision* 98,2%, *Recall* 97,5%, dan *mAP@50* sebesar 99,1%, serta akurasi 100% pada pengujian *Confusion Matrix*. Implementasi koreksi koordinat berbasis umpan balik visual memungkinkan robot melakukan penyesuaian posisi secara presisi terhadap deviasi mekanik dan variasi lingkungan. Dengan demikian, sistem yang dikembangkan tidak hanya unggul dalam evaluasi kuantitatif, tetapi juga terbukti stabil, adaptif, dan reliabel dalam penerapan nyata.

## Ucapan Terima kasih

Penelitian ini dapat terlaksana berkat dukungan P.T. Pegadaian Indonesia.

## Referensi

- [1] D.-T. Doan, Q.-T. Nguyen, and T.-T. Pham, "Building a model of automatic storage system Using PLC FX5UJ," *International Journal of Advances in Engineering and Management (IJAEM)*, vol. 5, no. 12, pp. 212-219, Dec. 2023. [Online]. Available : <https://ijaem.net/Plc%20FX5UJ.pdf>
- [2] Y. Z. KOÇ, "Automatic Storage and Retrieval System (AS/RS) design for manufacturing system," *International Journal of Computational and Experimental Science and Engineering (IJCESEN)*, vol. 9, no. 2, hlm. 102-104, 2023. <https://doi.org/10.22399/ijcesen.1260956>
- [3] R. Dharmawati, "Industrial Automation— PLC, SCADA, IoT-based monitoring, smart factories, and cyber-physical production systems," *Elimensi Journal of Electrical Engineering*, vol. 2, no. 2, 2024. <https://doi.org/10.54209/elimensi.v2i02.418>
- [4] H. Abbas, S. Shaheen, and M. Amin, "Simple, flexible, and interoperable SCADA system based on agent technology," *Intelligent Control and Automation*, vol. 6, no. 3, pp. 184-199, Agt. 2015, doi: <https://doi.org/10.4236/ica.2015.63018>
- [5] K. Kostenko, "Real-time communication between Robot PLC and PC over Ethernet-based Protocols," *Innopolis University*, pp. 1-6, 2021. <https://doi.org/10.48550/arXiv.1902.01924>

- [6] J. Seong, R. Ranjan, J. Kye, S. Lee, and S. Lee, "Enhancing industrial communication with ethernet/internet protocol: a study and analysis of real-time cooperative robot communication and automation via Transmission Control Protocol/Internet Protocol," *Sensors MDPI*, vol. 23, no. 8580, pp. 1-24, Okt. 2023. <https://doi.org/10.3390/s23208580>.
- [7] J. Giner, D. Katic, K. Kovacs, R. Glawar, and W. Sihh, "A computer vision based approach to reduce system downtimes in an automated high-rack logistics warehouse," *Procedia CIRP*, vol. 118, hlm. 1078-1083, 2023. <https://doi.org/10.1016/j.procir.2023.06.185>
- [8] V. B. Ayoola, G. Osam-Nunoo, C. Umeaku, and B. O. Awotiwon, "IoT-driven smart warehouses with computer vision for enhancing inventory accuracy and reducing discrepancies in automated systems," *IRE Journals*, vol. 8, no. 5, pp. 1-15, Nov. 2024. [Online]. Available : <https://www.researchgate.net/386110610>
- [9] C. Chen, J. Liu, H. Yin, and B. Huang, "A vision-based method for detecting the position of stacked goods in automated storage and retrieval systems," *Sensors MDPI*, vol. 25, no. 8, pp. 2623, Apr. 2025. <https://doi.org/10.3390/s25082623>
- [10] K. A. Gupta, N. Armani, T. C. Manjunath, and H. V. Manjunath, "Design and implementation of PLC based industrial application prototypes," *Indian Journal of Science and Technology*, vol. 10, no. 35, pp. 1-8, Sep. 2017. <https://doi.org/10.17485/ijst/2017/v10i35/118962>
- [11] G. Mareeswaran B., K. Kumar U., K. A., and G. P. Ram, "PLC based automated storage and retrieval system with a robotic end effector," *International Journal of Engineering Trends and Technology (IJETT)*, vol. 67, no. 7, pp. 14-18, Jul. 2019. <https://doi.org/10.14445/22315381/IJETT-V67I7P203>
- [12] I. F. Muharramsyah, S. Mu'Arif, A. Dwijotomo, Kamarudin, and Ridwan, "Smart Rack Inventory berbasis RFID untuk manajerial gudang," *Journal of Applied Electrical Engineering*, vol. 8, no. 2, pp. 96-101, Des. 2024. <https://doi.org/10.30871/jaee.v8i2.8562>
- [13] S. Alyahya, Q. Wang, and N. Bennett, "Application and integration of an RFID-enabled warehousing management system - a feasibility study," *Journal of Industrial Information Integration*, vol. 4, pp. 10-17, Des. 2016. <https://doi.org/10.1016/j.jii.2016.08.001>
- [14] W. Larutama, A. Suyatno, A. P. R. Lubis, and N. H. Camellia, "Information system design in warehouse inventory control," *Journal of Logistics and Supply Chain*, vol. 3, no. 1, pp. 35-44, 2023. <https://doi.org/10.17509/jlsc.v3i1.62069>
- [15] J. Noll, M. A. Razzak, J. M. Bass, and S. Beecham, "A study of the Scrum Master's role," *International Conference on Product-Focused Software Process Improvement*, pp. 307-323 Oktober 2017, [https://doi.org/10.1007/978-3-319-69926-4\\_22](https://doi.org/10.1007/978-3-319-69926-4_22)
- [16] H. Liu et al., "Deep-learning-based accurate identification of warehouse goods for robot picking operations," *Sustainability MDPI*, vol. 14, no. 13, pp. 7781, Jun. 2022. <https://doi.org/10.3390/su14137781>
- [17] Q. Pan, "Design and analysis of an autonomous warehouse robot system with 6-DOF manipulator," *Proceedings of the 2023 International Conference on Machine Learning and Automation*, hlm. 1-10, 2023. <https://doi.org/10.54254/2755-2721/34/20230310>
- [18] M. S. S. Agung, "Pengembangan sistem kontrol industri untuk robot simpan ambil otomatis" Tugas Akhir, Program Studi Teknik Fisika, Fakultas Teknologi Industri, Institut Teknologi Bandung, Bandung, 2025. [Online]. Available : <https://digilib.itb.ac.id/robot-simpan-ambil-otomatis>
- [19] A. R. Guntakandla, "Modular architecture: A scalable and efficient system design approach for enterprise applications," *World Journal of Advanced Research and Reviews*, vol. 26, no. 1, pp. 3114-3126, 2025. <https://doi.org/10.30574/wjarr.2025.26.1.1340>
- [20] R. P. Soesanto, Y. Nurdiansyah, and A. Kurniawati, "Streamlining warehouse process: A UHF RFID-enabled smart warehouse system design," *International Journal of Applied Technology Research*, vol. 5, no. 2, pp. 107-119, Jun. 2024. <https://doi.org/10.35313/ijat.v5i2.143>
- [21] R. Bunker and A. Elsherbeni, "A modular integrated RFID system for inventory control applications," *Electronics MDPI*, vol. 6, no. 1, pp. 1-13, Jan. 2017. <https://doi.org/10.3390/electronics6010006>
- [22] G. R. T. White, G. Gardiner, G. Prabhakar, and A. A. Razak, "A comparison of barcoding and RFID technologies in practice," *Journal of Information, Information Technology, and Organizations*, vol. 2, pp. 119-132, 2007. <https://doi.org/10.28945/142>
- [23] A. Tiwari and S. S. Roy, "Significant usage of RFID and barcode in logistic management," *Specialusis Ugdymas / Special Education*, vol. 1, no. 43, pp. 913-921, 2022. [Online]. Available : <https://www.researchgate.net/367049053>
- [24] C. Deepali, P. Monika, and S. Dharmendra, "Integrating barcode technology into warehouse management systems for enhanced efficiency and inventory accuracy," *Journal of Computer Science Research (JoCoSiR)*, vol. 2, no. 1, pp. 1-13, Jan. 2024. <https://doi.org/10.65126/jocosir.v2i1.34>
- [25] U. K. Siregar et al., "Pengembangan database Management system menggunakan My SQL," *SAINTEK: Jurnal Sains, Teknologi & Komputer*, vol. 1, no. 1, pp. 8-12, Jan. 2024. <https://doi.org/10.56495/saintek.v1i1.450>