



**MAGANG - EF234722**

## **Pengembangan Dashboard untuk Mendukung Revolusi Industri 4.0 PT Kayaba Indonesia**

**PT Kayaba Indonesia**

Blok II No. 4, Jl. Jawa Kawasan Industri MM No. 2100, Jatiwangi,  
Kec. Cikarang Barat, Kab. Bekasi, Jawa Barat 17520

Periode: 4 Agustus 2023 - 31 Desember 2023

### **Oleh:**

Azzura Mahendra Putra Malinus

5025201211

### **Pembimbing Jurusan**

Moch. Nafkhan Alzamzami, S.T., M.T.

### **Pembimbing Lapangan**

Jajang Khairil Azhar, S.Kom.

DEPARTEMEN TEKNIK INFORMATIKA  
Fakultas Teknologi Elektro dan Informatika Cerdas  
Institut Teknologi Sepuluh Nopember  
Surabaya 2023



**MAGANG - EF234722**

**Pengembangan Dashboard untuk Mendukung Revolusi Industri 4.0 PT Kayaba Indonesia**

PT Kayaba Indonesia

Blok II No. 4, Jl. Jawa Kawasan Industri MM No. 2100,  
Jatiwangi, Kec. Cikarang Barat, Kab. Bekasi, Jawa Barat 17520  
Periode: 4 Agustus 2023 - 31 Desember 2023

Oleh:

Azzura Mahendra Putra Malinus

5025201211

**Pembimbing Jurusan**

Moch. Nafkhan Alzamzami, S.T., M.T.

**Pembimbing Lapangan**

Jajang Khairil Azhar, S.Kom.

DEPARTEMEN TEKNIK INFORMATIKA

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya 2023

*[Halaman ini sengaja dikosongkan]*

## DAFTAR ISI

<b>DAFTAR ISI</b>	<b>iv</b>
<b>DAFTAR GAMBAR</b>	<b>ix</b>
<b>DAFTAR TABEL</b>	<b>xii</b>
<b>DAFTAR KODE SUMBER</b>	<b>xiii</b>
<b>LEMBAR PENGESAHAN</b>	<b>xv</b>
<b>KATA PENGANTAR</b>	<b>xix</b>
<b>BAB I PENDAHULUAN</b>	<b>1</b>
<b>1.1. Latar Belakang</b>	<b>1</b>
<b>1.2. Tujuan</b>	<b>2</b>
<b>1.3. Manfaat</b>	<b>2</b>
<b>1.4. Rumusan Masalah</b>	<b>2</b>
<b>1.5. Lokasi dan Waktu Magang/Kerja Praktik</b>	<b>3</b>
<b>1.6. Metodologi Magang/Kerja Praktik</b>	<b>3</b>
<b>1.6.1. Perumusan Masalah</b>	<b>3</b>
<b>1.6.2. Studi Literatur</b>	<b>3</b>
<b>1.6.3. Analisis dan Perancangan Sistem</b>	<b>4</b>
<b>1.6.4. Implementasi Sistem</b>	<b>4</b>
<b>1.6.5. Pengujian dan Evaluasi</b>	<b>4</b>
<b>1.6.6. Kesimpulan dan Saran</b>	<b>4</b>
<b>1.7. Sistematika Laporan</b>	<b>4</b>
<b>1.7.1. Bab I Pendahuluan</b>	<b>4</b>
<b>1.7.2. Bab II Profil Perusahaan</b>	<b>4</b>

1.7.3. Bab III Tinjauan Pustaka	5
1.7.4. Bab IV Analisis dan Perancangan Infrastruktur Sistem	5
1.7.5. Bab V Implementasi Sistem	5
1.7.6. Bab VI Pengujian dan Evaluasi	5
1.7.7. Bab VII Kesimpulan dan Saran	5
<b>BAB II PROFIL PERUSAHAAN</b>	<b>7</b>
2.1. Profil PT Kayaba Indonesia	7
2.2. Lokasi	8
<b>BAB III TINJAUAN PUSTAKA</b>	<b>10</b>
3.1. Pengembangan Web	10
3.2. Python	10
3.3. PHP	11
3.4. Database	11
3.5. Flask	11
3.6. TensorFlow Object Detection API	12
3.7. OpenVINO	13
3.8. TrOCR	13
3.9. Automatic Plate Number Recognition	14
3.10. Damping Force	15
3.11. Production Planning and Control	15
<b>BAB IV ANALISIS DAN PERANCANGAN SISTEM</b>	<b>18</b>
4.1. Analisis Sistem	18

4.1.1. Dashboard Automatic Plate Number Recognition	18
4.1.2. Perbaikan Dashboard Damping Force	21
4.1.3. Dashboard PPC Planning Assy	26
4.2. Perancangan Sistem	30
4.2.1. Dashboard Automatic Plate Number Recognition	30
4.2.2. Perbaikan Dashboard Damping Force	33
4.2.3. Dashboard PPC Planning Assy	35
<b>BAB V IMPLEMENTASI SISTEM</b>	<b>39</b>
5.1. Implementasi Dashboard APNR	39
5.1.1. Implementasi Model Deteksi Objek	39
5.1.2. Implementasi Model OCR	45
5.1.3. Implementasi Dashboard	51
5.2. Implementasi Perbaikan Dashboard Damping Force	59
5.2.1. Dashboard Damping Force sebelum Perbaikan	59
5.2.2. Implementasi Perbaikan Backend	60
5.2.3. Implementasi Perbaikan Frontend	63
5.3. Implementasi Dashboard PPC Planning Assy	65
5.3.1. Struktur Kode	65
5.3.2. Implementasi Halaman Login	66
5.3.3. Implementasi Halaman Planning	67
5.3.4. Implementasi Halaman Historical	68

<b>BAB VI PENGUJIAN DAN EVALUASI</b>	<b>71</b>
<b>6.1. Tujuan Pengujian</b>	<b>71</b>
<b>6.2. Kriteria Pengujian</b>	<b>71</b>
<b>6.3. Skenario Pengujian</b>	<b>71</b>
<b>6.4. Evaluasi Pengujian</b>	<b>72</b>
<b>BAB VII KESIMPULAN DAN SARAN</b>	<b>76</b>
<b>7.1. Kesimpulan</b>	<b>76</b>
<b>7.2. Saran</b>	<b>76</b>
<b>DAFTAR PUSTAKA</b>	<b>78</b>
<b>BIODATA PENULIS I</b>	<b>81</b>

*[Halaman ini sengaja dikosongkan]*

## DAFTAR GAMBAR

<b>Gambar 2. 1</b>	Logo PT Kayaba Indonesia	7
<b>Gambar 4. 1</b>	Use Case Diagram Dashboard APNR	19
<b>Gambar 4. 2</b>	Activity Diagram Use Case F1-001	20
<b>Gambar 4. 3</b>	Activity Diagram Use Case F1-002	20
<b>Gambar 4. 4</b>	Activity Diagram Use Case F1-003	21
<b>Gambar 4. 5</b>	Use Case Diagram Dashboard Damping Force	23
<b>Gambar 4. 6</b>	Activity Diagram Use Case F2-001	24
<b>Gambar 4. 7</b>	Activity Diagram Use Case F2-002	24
<b>Gambar 4. 8</b>	Activity Diagram Use Case F2-003	25
<b>Gambar 4. 9</b>	Activity Diagram Use Case F2-004	25
<b>Gambar 4. 10</b>	Use Case Diagram Dashboard PPC Planning Assy	27
<b>Gambar 4. 11</b>	Activity Diagram Use Case F3-001	28
<b>Gambar 4. 12</b>	Activity Diagram Use Case F3-002	29
<b>Gambar 4. 13</b>	Activity Diagram Use Case F3-003	29
<b>Gambar 4. 14</b>	Activity Diagram Use Case F3-004	30
<b>Gambar 4. 15</b>	Skema Sistem Dashboard APNR	31
<b>Gambar 4. 16</b>	CDM Basis Data Dashboard APNR	32
<b>Gambar 4. 17</b>	CDM Basis Data Dashboard Damping Force	35
<b>Gambar 4. 18</b>	CDM Basis Data Dashboard PPC Planning Assy	37
<b>Gambar 5. 1</b>	Sampel Dataset Model Deteksi Objek	40
<b>Gambar 5. 2</b>	Pengantorian Gambar	40
<b>Gambar 5. 3</b>	Contoh File Label dalam Format PascalVOC	40
<b>Gambar 5. 4</b>	Struktur Folder untuk Proses Fine-Tuning Model Deteksi Objek	41
<b>Gambar 5. 5</b>	Kelas yang akan Dideteksi Model	42
<b>Gambar 5. 6</b>	Model Deteksi Objek Plat Nomor	45
<b>Gambar 5. 7</b>	Sampel Dataset Model OCR	46

<b>Gambar 5. 8</b>	Label Dataset untuk Fine Tuning Model OCR	46
<b>Gambar 5. 9</b>	Model OCR Plat Nomor	51
<b>Gambar 5. 10</b>	Struktur Kode Dashboard APNR	52
<b>Gambar 5. 11</b>	Halaman Utama Dashboard APNR	59
<b>Gambar 5. 12</b>	Dashboard Damping Force sebelum Perbaikan	60
<b>Gambar 5. 13</b>	Struktur Kode Dashboard Damping Force sebelum Perbaikan	60
<b>Gambar 5. 14</b>	Struktur Kode Dashboard Damping Force setelah Perbaikan	61
<b>Gambar 5. 15</b>	Menu Utama Dashboard Damping Force	63
<b>Gambar 5. 16</b>	Halaman Visualisasi Grafik Garis Dashboard Damping Force	63
<b>Gambar 5. 17</b>	Halaman Tabel Dashboard Damping Force	64
<b>Gambar 5. 18</b>	Menu Cetak Laporan Dashboard Damping Force	64
<b>Gambar 5. 19</b>	Contoh Laporan Dashboard Damping Force	64
<b>Gambar 5. 20</b>	Struktur Kode Dashboard PPC Planning Assy	66
<b>Gambar 5. 21</b>	Halaman Login Dashboard PPC Planning Assy	67
<b>Gambar 5. 22</b>	Halaman Planning Dashboard PPC Planning Assy	68
<b>Gambar 5. 23</b>	Halaman Historical Dashboard PPC Planning Assy	69

*[Halaman ini sengaja dikosongkan]*

## DAFTAR TABEL

<b>Tabel 4. 1</b> Kebutuhan Fungsional Dashboard APNR	18
<b>Tabel 4. 2</b> Kebutuhan Nonfungsional Dashboard APNR	19
<b>Tabel 4. 3</b> Kebutuhan Fungsional Dashboard Damping Force	22
<b>Tabel 4. 4</b> Kebutuhan Nonfungsional Dashboard Damping Force	22
<b>Tabel 4. 5</b> Kebutuhan Fungsional Dashboard PPC Planning Assy	26
<b>Tabel 4. 6</b> Kebutuhan Nonfungsional Dashboard PPC Planning Assy	26
<b>Tabel 6. 1</b> Hasil Evaluasi Pengujian Dashboard APNR	72
<b>Tabel 6. 2</b> Hasil Evaluasi Pengujian Dashboard Damping Force	73
<b>Tabel 6. 3</b> Hasil Evaluasi Pengujian Dashboard PPC Planning Assy	74

## DAFTAR KODE SUMBER

<b>Kode Sumber 5. 1</b> Proses <i>Fine Tuning</i> dan Ekspor Model Deteksi Objek	44
<b>Kode Sumber 5. 2</b> Proses Konversi ke OpenVINO	45
<b>Kode Sumber 5. 3</b> Proses Fine Tuning Model OCR	48
<b>Kode Sumber 5. 4</b> Pseudocode app.py	54
<b>Kode Sumber 5. 5</b> Pseudocode plate_detection.py	57
<b>Kode Sumber 5. 6</b> Pseudocode plate_extraction.py	59
<b>Kode Sumber 5. 7</b> DDL Basis Data APNR	59
<b>Kode Sumber 5. 8</b> Pseudocode index.php Dashboard Damping Force untuk Menampilkan Menu	63

*[Halaman ini sengaja dikosongkan]*

**LEMBAR PENGESAHAN  
KERJA PRAKTIK**

**Pengembangan Dashboard untuk Mendukung Revolusi  
Industri 4.0 PT Kayaba Indonesia**

Oleh:

Azzura Mahendra Putra Malinus

5025201211

Disetujui oleh Pembimbing Kerja Praktik:

1. Moch. Nafkhan Alzamzami,  
S.T., M.T.  
NIP. 1999202311033



(Pembimbing Departemen)

2. Jajang Khairil Azhar, S.Kom.



(Pembimbing Lapangan)

*[Halaman ini sengaja dikosongkan]*

# **Pengembangan Dashboard untuk Mendukung Revolusi Industri 4.0 PT Kayaba Indonesia**

Nama Mahasiswa : Azzura Mahendra Putra Malinus  
NRP : 5025201211  
Departemen : Teknik Informatika FTEIC-ITS  
Pembimbing Departemen : Moch. Nafkhan Alzamzami, S.T.,  
M.T.  
Pembimbing Lapangan : Jajang Khairil Azhar, S.Kom.

## **ABSTRAK**

*PT Kayaba Indonesia adalah sebuah perusahaan industri manufaktur yang memproduksi shock absorber, front fork, dan oil cushion unit untuk kendaraan bermotor. Untuk bersaing di era Industry 4.0 ini, PT Kayaba Indonesia telah melakukan berbagai upaya digitalisasi ke berbagai bisnis prosesnya melalui berbagai proyek-proyek. Proyek-proyek tersebut ditangani oleh Section Industry 4.0 Departemen Management Information System (MIS). Pada magang/kerja praktik ini, penulis diberikan kesempatan untuk mengerjakan beberapa proyek-proyek yang dimiliki oleh Section Industry 4.0 di Departemen MIS PT Kayaba Indonesia.*

*Pada magang/kerja praktik ini, penulis diberikan sebanyak tiga proyek untuk dikerjakan. Proyek-proyek tersebut adalah Dashboard Automatic Plate Number Recognition, perbaikan Dashboard Damping Force, dan Dashboard PPC Planning Assy. Proyek-proyek tersebut merupakan sebuah aplikasi web serta dikembangkan menggunakan bahasa Python dan PHP. Aplikasi web tersebut nantinya dijalankan pada server perusahaan sehingga dapat digunakan oleh karyawan PT Kayaba Indonesia.*

***Kata Kunci : Web Development, Industry 4.0, PHP, Python***

*[Halaman ini sengaja dikosongkan]*

## **KATA PENGANTAR**

Puji syukur kehadirat Allah SWT, karena berkat limpahan rahmat dan karunia-Nya, penulis dapat menyelesaikan kewajiban penulis sebagai mahasiswa Departemen Teknik Informatika ITS, yaitu magang/kerja praktik yang berjudul “Pengembangan Dashboard untuk Mendukung Revolusi Industri 4.0 PT Kayaba Indonesia”.

Penulis ingin menyampaikan ucapan terima kasih kepada semua pihak yang memberi bantuan dan dukungan dalam penyusunan laporan ini:

1. Kedua orang tua penulis.
2. Bapak Jajang Khairil Azhar, S.Kom. selaku pembimbing lapangan selama magang/kerja praktik berlangsung.
3. Bapak Agus Mujtahid selaku Kepala Departemen MIS.
4. Karyawan Departemen MIS.
5. Teman-teman magang di Departemen MIS.
6. Bapak Syamsul Hidayat selaku Plant Director PT Kayaba Indonesia.
7. Bapak Moch. Nafkhan Alzamzami, S.T., M.T. selaku pembimbing magang/kerja praktik.
8. Bapak Abdul Munif, S.Kom., M.Sc. selaku koordinator magang.
9. Bapak Ir. Ary Mazharuddin Shiddiqi, S.Kom., M.Comp.Sc., Ph.D., IPM. selaku koordinator magang/kerja praktik.
10. Teman-teman ITS.
11. Orang tua teman-teman ITS.

Dalam laporan ini, dijelaskan pekerjaan yang dilakukan penulis selama melaksanakan magang/kerja praktik di PT Kayaba Indonesia. Penjelasan tersebut meliputi profil perusahaan, proyek yang diberikan kepada penulis, dan manfaat yang diperoleh penulis selama melaksanakan magang/kerja praktik.

Terdapat banyak kekurangan dalam pelaksanaan magang/kerja praktik dan penyusunan laporan magang/kerja praktik oleh penulis. Akan tetapi, laporan ini diharapkan dapat memberikan manfaat kepada pembaca untuk menambah wawasan atau menjadikannya sebagai sumber referensi.

Surabaya, 30 Januari 2024  
Azzura Mahendra Putra Malinus

*[Halaman ini sengaja dikosongkan]*

# BAB I PENDAHULUAN

## 1.1. Latar Belakang

Banyak perusahaan manufaktur di era saat ini telah mengintegrasikan teknologi digital ke dalam bisnis proses mereka. Upaya pemanfaatan teknologi inovatif ke berbagai aspek dalam proses industri, seperti pemanufakturan, desain, dan suplai, adalah ide dasar dari konsep Industry 4.0 (Hamdi, 2019). Teknologi yang digunakan untuk bertransisi antara lain Internet of Things, Artificial Intelligence, Simulation System, Cloud Computing, dll. Dengan mengadopsi Industry 4.0, perusahaan manufaktur mampu melakukan berbagai hal yang dapat merevolusi bisnis proses perusahaan, seperti penangkapan gangguan secara *real-time*, pemahaman perubahan dan tren kebutuhan pelanggan, dan pengambilan keputusan yang cepat di lingkungan yang dinamis.

Revolusi Industry 4.0 juga dilakukan di PT Kayaba Indonesia. PT Kayaba Indonesia telah memanfaatkan berbagai teknologi informasi dan komunikasi untuk mengembangkan bisnis prosesnya menjadi lebih efektif dan efisien, seperti penggunaan robot, sistem monitoring produksi, dan sistem Enterprise Resource Planning (ERP). Upaya transisi ke Industry 4.0 ini akan terus dilakukan secara bertahap oleh Departemen Management Information System (MIS), khususnya *Section* Industry 4.0. Dengan ini, PT Kayaba Indonesia dapat berkembang menjadi lebih baik dan dapat bersaing dengan perusahaan lain di era digital saat ini.

Pada laporan ini, akan dijelaskan proyek-proyek yang dikerjakan penulis selama melaksanakan magang/kerja praktik di PT Kayaba Indonesia, khususnya di Departemen MIS *Section* Industry 4.0. Proyek-proyek

yang dikerjakan oleh penulis antara lain Dashboard Automatic Plate Number Recognition, perbaikan Dashboard Damping Force, dan Dashboard PPC Planning Assy. Selain itu, juga dijelaskan terkait hal-hal lain yang meliputi profil perusahaan dan hal yang diperoleh penulis setelah melaksanakan magang/kerja praktik. Laporan ini diharapkan dapat memberikan gambaran yang jelas terkait hal-hal yang telah disebutkan sebelumnya.

## **1.2. Tujuan**

Tujuan magang/kerja praktik ini adalah menyelesaikan kewajiban mata kuliah magang/kerja praktik sebesar 4 sks dan membantu PT Kayaba Indonesia untuk mengembangkan dashboard yang berkaitan dengan pengembangan Industry 4.0 di sana.

## **1.3. Manfaat**

Bagi penulis, manfaat dari pelaksanaan magang/kerja praktik ini adalah penulis memperoleh pengalaman bekerja langsung di sebuah perusahaan untuk menerapkan ilmu yang telah diperoleh selama berkuliah di Teknik Informatika ITS. Bagi PT Kayaba Indonesia, hasil dari magang/kerja praktik penulis bermanfaat untuk mendigitalisasi sebagian kecil proses bisnis di sana yang berkaitan dengan upaya pengembangan Industry 4.0.

## **1.4. Rumusan Masalah**

Rumusan masalah dari magang/kerja praktik ini adalah sebagai berikut:

1. Bagaimana implementasi Dashboard Automatic Plate Number Recognition untuk mempermudah proses pencatatan nomor polisi kendaraan bermotor yang masuk dan keluar PT Kayaba Indonesia?

2. Bagaimana implementasi yang diterapkan untuk memperbaiki Dashboard Damping Force yang telah dikembangkan sebelumnya?
3. Bagaimana implementasi Dashboard PPC Planning Assy untuk proses penambahan plan *assembly*?

## **1.5. Lokasi dan Waktu Magang/Kerja Praktik**

Magang/kerja praktik dilaksanakan secara luring pada waktu dan tempat sebagai berikut:

Lokasi	: PT Kayaba Indonesia
Waktu	: 4 Agustus 2023 - 31 Desember 2023
Hari kerja	: Senin - Jumat
Jam kerja	: 07.30 – 16.00, 07.30 – 16.30 (Jumat)

## **1.6. Metodologi Magang/Kerja Praktik**

Metodologi dalam pembuatan buku magang/kerja praktik ini meliputi:

### **1.6.1. Perumusan Masalah**

Pada tahap ini, penulis berdiskusi dengan pembimbing lapangan, Bapak Jajang Khairil Azhar, untuk membahas mengenai gambaran dari permasalahan yang akan diselesaikan dengan proyek-proyek yang akan dikerjakan.

### **1.6.2. Studi Literatur**

Setelah memahami mengenai gambaran setiap permasalahan, dilakukan studi literatur untuk mengetahui berbagai hal terkait perancangan dan implementasi untuk setiap proyek. Hal tersebut meliputi pemahaman mengenai bisnis proses yang berkaitan, teknologi yang akan digunakan, dan contoh solusi yang sudah pernah diterapkan.

### **1.6.3. Analisis dan Perancangan Sistem**

Berdasarkan studi literatur yang telah dilakukan, dianalisis kebutuhan-kebutuhan yang ada pada setiap permasalahan serta ditentukan rancangan yang paling sesuai untuk pengimplementasian setiap proyek.

### **1.6.4. Implementasi Sistem**

Pada tahap ini, akan dilakukan pengimplementasian setiap proyek. Setiap proyek tersebut akan diimplementasikan berdasarkan rancangan yang telah ditentukan sebelumnya.

### **1.6.5. Pengujian dan Evaluasi**

Untuk setiap proyek yang telah selesai, dilakukan pengujian dan evaluasi yang bertujuan untuk mengetahui kesesuaian implementasi dengan kebutuhan yang telah didefinisikan. Jika terdapat kekurangan, implementasi yang telah dilakukan akan diperbaiki secara iteratif hingga siap untuk digunakan pada lingkungan produksi.

### **1.6.6. Kesimpulan dan Saran**

Pada tahap ini, dijelaskan kesimpulan dan saran terkait dengan magang/kerja praktik yang telah dilakukan.

## **1.7. Sistematika Laporan**

### **1.7.1. Bab I Pendahuluan**

Bab ini berisi tentang latar belakang, tujuan, manfaat, rumusan masalah, lokasi dan waktu magang/kerja praktik, metodologi, dan sistematika laporan.

### **1.7.2. Bab II Profil Perusahaan**

Bab ini berisi tentang gambaran umum PT Kayaba Indonesia yang terdiri dari profil dan lokasi perusahaan.

### **1.7.3. Bab III Tinjauan Pustaka**

Bab ini berisi tentang dasar teori dari teknologi yang digunakan dalam menyelesaikan proyek-proyek magang/kerja praktik.

### **1.7.4. Bab IV Analisis dan Perancangan Infrastruktur Sistem**

Bab ini berisi tentang tahap analisis dan perancangan sistem dari proyek-proyek magang/kerja praktik.

### **1.7.5. Bab V Implementasi Sistem**

Bab ini berisi tentang penjelasan tahap-tahap yang dilakukan untuk proses implementasi proyek.

### **1.7.6. Bab VI Pengujian dan Evaluasi**

Bab ini berisi tentang hasil uji coba dan evaluasi dari proyek yang telah dikembangkan selama pelaksanaan magang/kerja praktik.

### **1.7.7. Bab VII Kesimpulan dan Saran**

Bab ini berisi tentang kesimpulan dan saran yang diperoleh dari proses pelaksanaan magang/kerja praktik.

*[Halaman ini sengaja dikosongkan]*

## **BAB II**

### **PROFIL PERUSAHAAN**

#### **2.1. Profil PT Kayaba Indonesia**

PT Kayaba Indonesia adalah sebuah perusahaan yang berfokus pada sektor industri manufaktur. Produk yang dihasilkan oleh PT Kayaba Indonesia adalah *shock absorber*, *front fork*, dan *oil cushion unit*. Perusahaan ini berdiri pertama kali di Indonesia pada tanggal 25 Februari 1976 dengan nama PT Kayaba Jepang yang didirikan di Kawasan Industri Pulogadung, Jawa Timur. Kemudian, setelah menjalin kerjasama dengan PT Astra Otoparts Tbk, nama perusahaan diubah menjadi PT Kayaba Indonesia.



*Our Precision, Your Advantage*

**Gambar 2. 1** Logo PT Kayaba Indonesia

Sebagai sebuah perusahaan, PT Kayaba Indonesia memiliki visi dan misi. Visi PT Kayaba Indonesia adalah “World Class Shock Absorber Manufacturer, Partner of Choice in Indonesia with Engineering Competence”. Misi PT Kayaba Indonesia adalah “Providing Technology and Product that Make People Lives Safe and Comfortable”. Dengan visi dan misi tersebut, PT Kayaba Indonesia berusaha untuk berkembang menjadi lebih baik demi memenuhi permintaan dan kebutuhan akan produk-produk yang diproduksinya.

## **2.2. Lokasi**

Blok II No. 4, Jl. Jawa Kawasan Industri MM No. 2100, Jatiwangi, Kec. Cikarang Barat, Kab. Bekasi, Jawa Barat 17520

*[Halaman ini sengaja dikosongkan]*

## **BAB III**

### **TINJAUAN PUSTAKA**

#### **3.1. Pengembangan Web**

Pengembangan web merupakan sebuah pekerjaan mengembangkan sebuah website yang dapat di-*host* secara daring maupun luring (Margaret Rouse, 2020). Dalam melakukan pengembangan web, terdapat beberapa proses yang dilakukan, antara lain desain web, pengembangan konten web, pembuatan skrip sisi klien atau server, serta keamanan jaringan. Peran pengembang web pada umumnya dibagi menjadi dua, yaitu pengembang *frontend* dan *backend*. Pengembang *frontend* berfokus pada pengembangan antarmuka sebuah website. Secara umum, pengembang *frontend* menggunakan teknologi seperti Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), dan JavaScript. Di sisi lain, pengembang *backend* berfokus pada pengembangan kode di balik layar, seperti masalah basis data. Secara umum, pengembang *backend* menggunakan teknologi seperti PHP, Java, Python, dan Ruby.

#### **3.2. Python**

Python adalah sebuah bahasa pemrograman komputer yang umum digunakan untuk mengembangkan website atau perangkat lunak, mengotomasi pekerjaan, serta menganalisis data (Coursera Staff, 2023). Python merupakan bahasa pemrograman yang serbaguna dan mudah untuk dipelajari bagi pemula. Python memiliki sintaks sederhana serta mirip dengan bahasa manusia. Python juga memiliki banyak pengguna yang berkontribusi untuk membuat berbagai modul yang berguna untuk memperluas kegunaan dari Python.

### 3.3. PHP

PHP adalah sebuah bahasa pemrograman di sisi server yang umum digunakan pada pengembangan web (Yash Agarwal, 2023). PHP dikenal mudah dipelajari untuk pemula yang ingin mempelajari pengembangan web. Pengguna PHP berjumlah sangat banyak sehingga orang yang ingin belajar mudah untuk mendapatkan bantuan. PHP juga memiliki banyak sekali ekstensi dan pustaka yang merupakan hasil kontribusi dari berbagai orang. Karena PHP mendukung banyak platform dan basis data, PHP dinilai serbaguna dan fleksibel. Keuntungan lain dari PHP antara lain memiliki fitur keamanan yang kuat, performa tinggi, dan mudah di-*debug*. Akan tetapi, seiring dengan berkembangnya sebuah proyek, PHP kurang baik untuk digunakan karena kode PHP yang dikembangkan akan semakin sulit untuk dipelihara.

### 3.4. Database

Database merupakan kumpulan data terstruktur dan terorganisasi secara digital (Oracle, t.t.). Sebuah database pada umumnya berada pada sebuah *database management system* (DBMS). Database memiliki beberapa varian, antara lain database relasional, database berbasis objek, database terdistribusi, database NoSQL, database graf, dll. Pada database relasional, terdapat sebuah bahasa pemrograman yang bernama Structured Query Language (SQL). SQL digunakan pada database relasional untuk melakukan kontrol akses, kueri, perubahan, dan pendefinisian data.

### 3.5. Flask

Flask merupakan sebuah *microframework* untuk mengembangkan aplikasi web (Matthew Deery, 2023).

Flask didesain untuk dapat digunakan dengan mudah dan sederhana. Flask menggunakan Python sebagai bahasa utamanya. Salah satu kekurangan Flask sebagai *microframework* adalah kurangnya ekstensi yang dapat digunakan. Penambahan ekstensi yang cukup banyak dapat memperlambat kinerja dari aplikasi Flask.

### 3.6. TensorFlow Object Detection API

Sebuah teknik visi komputer bernama *object detection* merupakan sebuah teknik untuk mendeteksi, mencari lokasi, dan menjejak sebuah objek pada sebuah gambar atau video (Great Learning Team, 2022). Teknik *object detection* mengenali kelas dari sebuah objek dan memberikan koordinat lokasinya pada sebuah gambar berupa kotak pembatas. Cara kerja *object detection* adalah dengan membagi sebuah gambar menjadi beberapa bagian persegi dan melakukan ekstraksi fitur pada setiap persegi untuk menemukan sebuah objek. Hasil yang bertumpukan akan digabung menjadi satu kotak pembatas dengan algoritma Non-Maximum Suppression (NMS).

TensorFlow dan Keras merupakan pustaka yang bersifat sumber terbuka yang digunakan untuk komputasi numerik dan *machine learning* dengan skala besar. TensorFlow memiliki sebuah kerangka kerja yang dapat digunakan untuk mengembangkan model *object detection*, yaitu TensorFlow Object Detection API. TensorFlow Object Detection API memiliki kumpulan model yang sudah terlatih dengan beberapa dataset yang dimanakan Model Zoo. Contoh model tersebut antara lain SSD MobileNet, SSD Resnet, dan Faster RCNN. Model-model tersebut dapat dikonfigurasi dan dilatih pada dataset lain agar dapat mengenali objek-objek spesifik. API ini dapat diakses di tautan <https://github.com/tensorflow/models>.

### 3.7. OpenVINO

OpenVINO atau Open Visual Inference and Neural Network Optimization adalah sebuah alat untuk *deep learning* yang dikembangkan oleh Intel (Vidushi Meel, t.t.). OpenVINO dapat menaikkan performa ketika menjalankan sebuah model *deep learning* dengan memanfaatkan produk-produk Intel. Hal yang dapat dilakukan dengan OpenVINO antara lain menjalankan model visi komputer pada berbagai perangkat keras serta mengimpor dan mengoptimasi model dari berbagai *framework deep learning*. Meskipun bagus digunakan untuk menjalankan model, OpenVINO tidak bisa digunakan untuk melatih model.

Jika dilihat secara umum, OpenVINO melakukan dua hal kepada sebuah model yang sudah dilatih, yaitu memasukkannya ke *Model Optimizer* lalu hasil yang diperoleh dimasukkan ke *Inference Engine*. Pada *Model Optimizer*, sebuah model *deep learning* dioptimasi dan dikonversi menjadi *Intermediate Representation*, yang berbentuk sebuah file XML dan file .bin. Setelah itu, kedua file tersebut dan data input diberikan ke *Inference Engine* untuk diperiksa kesesuaiannya berdasarkan *framework* yang digunakan untuk pelatihan model dan perangkat keras yang digunakan. Setelah melalui dua hal tersebut, diperoleh model yang telah dioptimasi yang dapat di-*deploy* ke aplikasi.

### 3.8. TrOCR

Optical Character Recognition (OCR) merupakan teknik untuk mengubah gambar tulisan menjadi karakter mesin (Tejpal Kumawat, 2023). Secara umum, terdapat dua tahap dalam proses OCR, yaitu deteksi teks dan pengenalan teks. Pendeteksian teks akan mencari lokasi dari kemungkinan karakter pada sebuah gambar dan

pengenalan teks akan mengubah karakter yang dideteksi menjadi karakter mesin. TrOCR adalah sebuah OCR yang berbasis transformer. Transformer menggunakan paradigma *encoder-decoder* yang bebas konvolusi dan dapat memperoleh akurasi yang terbaik tanpa adanya *preprocessing* dan *postprocessing* yang rumit.

### **3.9. Automatic Plate Number Recognition**

Automatic Plate Number Recognition atau Pengenalan Plat Nomor Otomatis merupakan sebuah sistem visi komputer untuk membaca nomor polisi pada sebuah gambar kendaraan (Nico Klingler, 2024). Sistem ini dapat diterapkan untuk gambar atau video yang diambil menggunakan satu atau lebih kamera. Sebuah sistem APNR menggunakan optical character recognition (OCR) dan beberapa teknik pemrosesan gambar untuk mendapatkan sebuah nomor polisi. Seiring dengan berkembangnya *machine learning* dan *deep learning*, sistem APNR dapat berkembang dengan lebih cepat dalam hal performa sehingga sistem ini banyak digunakan untuk berbagai keperluan. Contoh penggunaan sistem APNR di kehidupan sehari-hari adalah untuk sistem manajemen parkir, sistem pembayaran tarif jalan tol, dan sistem manajemen lalu lintas kendaraan.

Secara umum, sistem APNR memiliki empat tahapan proses. Tahapan pertama adalah memperoleh input dari kamera yang terintegrasi dengan sistem APNR. Tahapan selanjutnya adalah mendeteksi dan memotong bagian plat nomor pada input. Untuk mendapatkan koordinat pemotongan pada input, dapat digunakan *machine learning* atau metode visi komputer, misalkan *object detection* dan *template matching*. Setelah input dipotong dengan koordinat yang diperoleh, hasil potongan dinormalisasi. Selanjutnya, input diberikan ke optical

character recognition untuk diekstrak teks nomor polisinya. Setelah diperoleh hasil ekstraksi, data dapat disimpan ke dalam database atau digunakan untuk berbagai keperluan lain.

### **3.10. Damping Force**

*Damping force* atau gaya peredam adalah sebuah gaya yang memperlambat sebuah gerakan (Unacademy, t.t.). *Damping force* bekerja pada sebuah gerakan yang berasal dari sebuah getaran atau sistem yang berosilasi. Gaya ini bertujuan untuk mengurangi frekuensi dari osilasi atau mencegah terjadinya osilasi. Efek redaman dari *damping force* muncul karena terjadinya disipasi energi. *Damping force* yang umum diamati adalah *damping force* mekanis, *damping force* kekentalan, *damping force* elektrik, dan *damping force* elektromagnetis.

### **3.11. Production Planning and Control**

*Production planning and control* atau perencanaan dan pengendalian produksi merupakan sebuah kegiatan yang dilakukan di perusahaan manufaktur untuk merencanakan dan mengendalikan siklus bahan baku pada sebuah sistem produksi agar tercapainya efisiensi dalam pemanfaatan waktu, penggunaan biaya produksi, dan pemenuhan permintaan pasar (Aufa Atila, 2021). Pada tahap perencanaan produksi, dirumuskan berbagai langkah-langkah yang akan diterapkan dalam proses produksi di masa yang akan datang, misalkan hal yang berkaitan dengan waktu dan jumlah. Langkah-langkah ini dibuat berdasarkan informasi masa lalu serta asumsi-asumsi sehingga perencanaan yang dirumuskan belum tentu sesuai dengan pelaksanaannya. Pada tahap pengendalian produksi, dilakukan pemantauan terkait

dengan penyimpangan terhadap rencana produksi pada saat proses produksi. Tindakan penyesuaian akan dilakukan jika ditemukan penyimpangan yang cukup signifikan dan hasil dari penyesuaian akan digunakan sebagai pedoman untuk merumuskan rencana-rencana produksi di waktu yang akan datang.

*[Halaman ini sengaja dikosongkan]*

## BAB IV ANALISIS DAN PERANCANGAN SISTEM

### 4.1. Analisis Sistem

Berikut adalah penjelasan mengenai definisi umum, analisis kebutuhan, *use case diagram*, dan *activity diagram* untuk setiap proyek dashboard.

#### 4.1.1. Dashboard Automatic Plate Number Recognition

Dashboard Automatic Plate Number Recognition adalah sebuah sistem informasi berbasis web yang berfungsi untuk memonitor kendaraan bermotor yang keluar dan masuk PT Kayaba Indonesia. Aplikasi ini akan mencatat nomor polisi dari setiap kendaraan bermotor sehingga dapat diperoleh data yang dapat digunakan untuk berbagai keperluan.

Pada Tabel 4.1 dan 4.2, dijelaskan kebutuhan fungsional dan nonfungsional dari aplikasi.

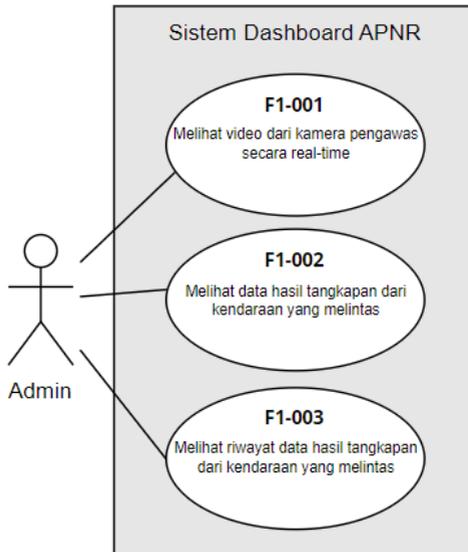
**Tabel 4. 1** Kebutuhan Fungsional Dashboard APNR

Kode	Deskripsi
F1-001	Admin dapat melihat video dari kamera pengawas secara <i>real-time</i>
F1-002	Admin dapat melihat data hasil tangkapan dari kendaraan yang melintas
F1-003	Admin dapat melihat riwayat data hasil tangkapan dari kendaraan yang melintas

**Tabel 4. 2** Kebutuhan Nonfungsional Dashboard APNR

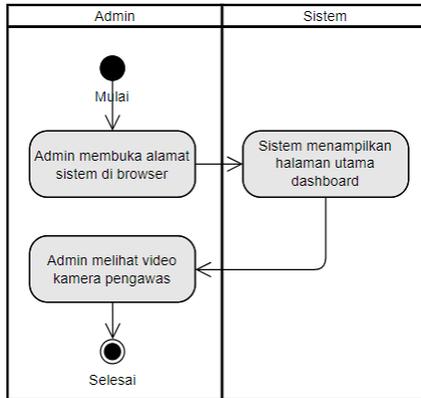
Kode	Deskripsi
NF1-001	Sistem dapat digunakan pada komputer dengan spesifikasi menengah ke bawah
NF1-002	Sistem memiliki antarmuka yang mudah dipahami
NF1-003	Sistem mendukung semua jenis browser

Pada Gambar 4.1, dijelaskan *use case diagram* dari aplikasi.

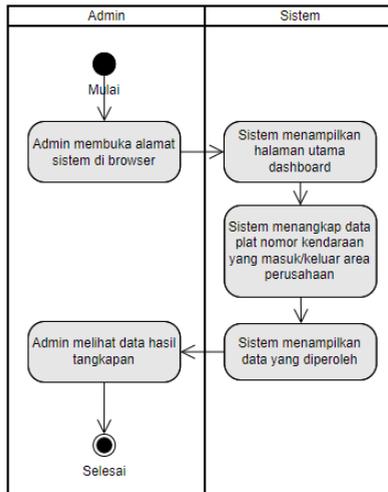


**Gambar 4. 1** Use Case Diagram Dashboard APNR

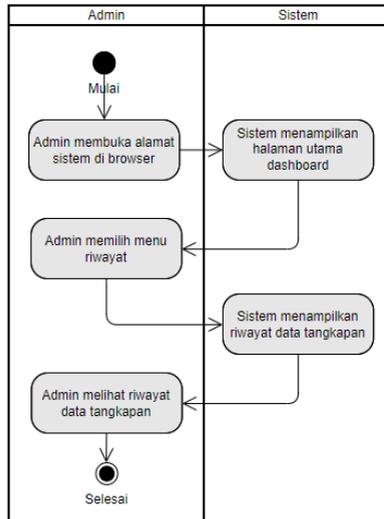
Pada Gambar 4.2, Gambar 4.3, dan Gambar 4.4, dijelaskan *activity diagram* untuk setiap *use case* dari aplikasi.



**Gambar 4. 2** Activity Diagram Use Case F1-001



**Gambar 4. 3** Activity Diagram Use Case F1-002



**Gambar 4. 4** Activity Diagram Use Case F1-003

#### 4.1.2. Perbaikan Dashboard Damping Force

Dashboard Damping Force digunakan untuk memvisualisasikan data terkait *damping force* saat proses produksi *shock absorber*. Data-data tersebut diambil pada rentang waktu tertentu dari beberapa *production line shock absorber*. Pada proyek ini, penulis melanjutkan pengerjaan proyek yang sebelumnya telah dikerjakan oleh peserta magang lain di departemen MIS PT Kayaba Indonesia. Pada saat proyek diberikan kepada penulis, proyek ini memiliki beberapa kekurangan, misalnya dashboard yang masih hanya dapat menampilkan data pada sebuah *production line shock absorber*, yaitu *line Shock Absorber 3*, adanya *bug* pada tampilan *front end*, dan kesalahan logika pada *backend*. Oleh karena itu, dashboard ini perlu diperbaiki sedemikian rupa sehingga dapat berfungsi sesuai dengan yang dibutuhkan.

Pada Tabel 4.3 dan 4.4, dijelaskan kebutuhan fungsional dan nonfungsional dari aplikasi.

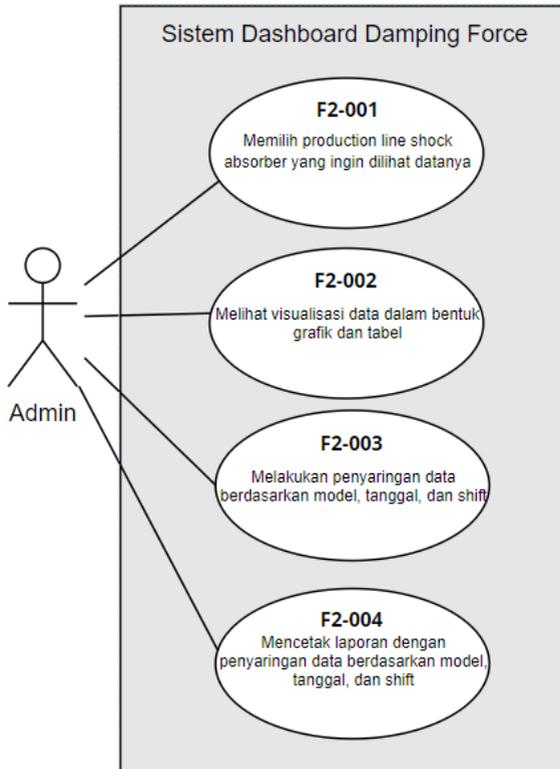
**Tabel 4. 3** Kebutuhan Fungsional Dashboard Damping Force

Kode	Deskripsi
F2-001	Admin dapat memilih <i>production line shock absorber</i> yang ingin dilihat datanya
F2-002	Admin dapat melihat visualisasi data dalam bentuk grafik dan tabel
F2-003	Admin dapat melakukan penyaringan data berdasarkan model, tanggal, dan shift
F2-004	Admin dapat mencetak laporan dengan penyaringan data berdasarkan model, tanggal, dan shift

**Tabel 4. 4** Kebutuhan Nonfungsional Dashboard Damping Force

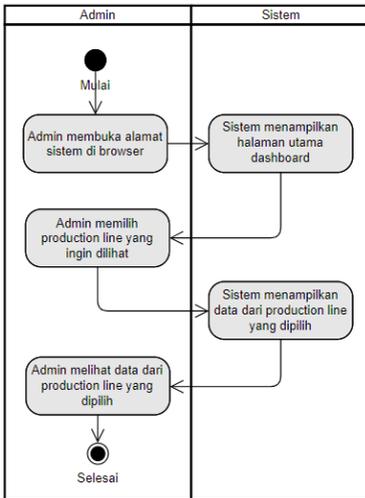
Kode	Deskripsi
NF2-001	Sistem dapat digunakan pada komputer dengan spesifikasi menengah ke bawah
NF2-002	Sistem memiliki antarmuka yang mudah dipahami
NF2-003	Sistem mendukung semua jenis browser

Pada Gambar 4.5, dijelaskan *use case diagram* dari aplikasi.

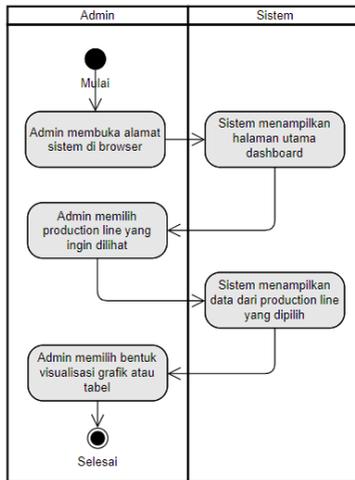


**Gambar 4. 5** Use Case Diagram Dashboard Damping Force

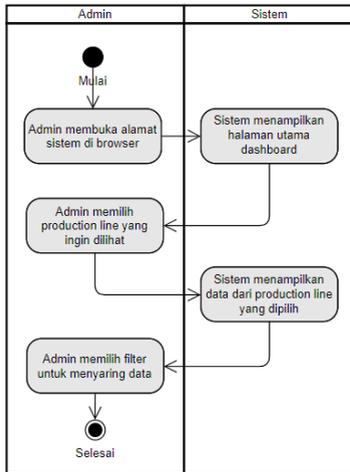
Pada Gambar 4.6, Gambar 4.7, Gambar 4.8, dan Gambar 4.9, dijelaskan *activity diagram* untuk setiap *use case* dari aplikasi.



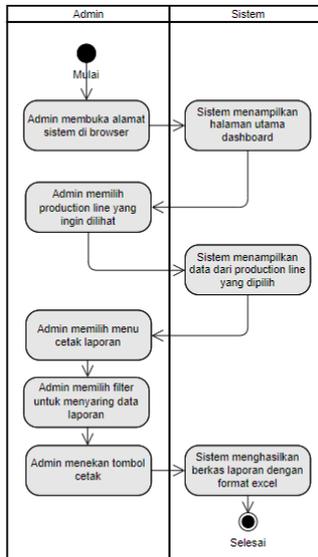
**Gambar 4. 6** Activity Diagram Use Case F2-001



**Gambar 4. 7** Activity Diagram Use Case F2-002



**Gambar 4. 8** Activity Diagram Use Case F2-003



**Gambar 4. 9** Activity Diagram Use Case F2-004

#### 4.1.3. Dashboard PPC Planning Assy

Dashboard PPC (Production Planning and Control) Planning Assy digunakan untuk membuat plan-plan *assembly* di sebuah *assembly line*. Tiap dari plan tersebut memiliki beberapa data yang harus didefinisikan, yaitu *assembly line*, shift, tanggal, *part number*, *model name*, *lot size*, dan *planning*. Riwayat plan-plan yang telah dibuat juga dapat dilihat dan dimonitor perkembangannya.

Pada Tabel 4.5 dan 4.6, dijelaskan kebutuhan fungsional dan nonfungsional dari aplikasi.

**Tabel 4. 5** Kebutuhan Fungsional Dashboard PPC Planning Assy

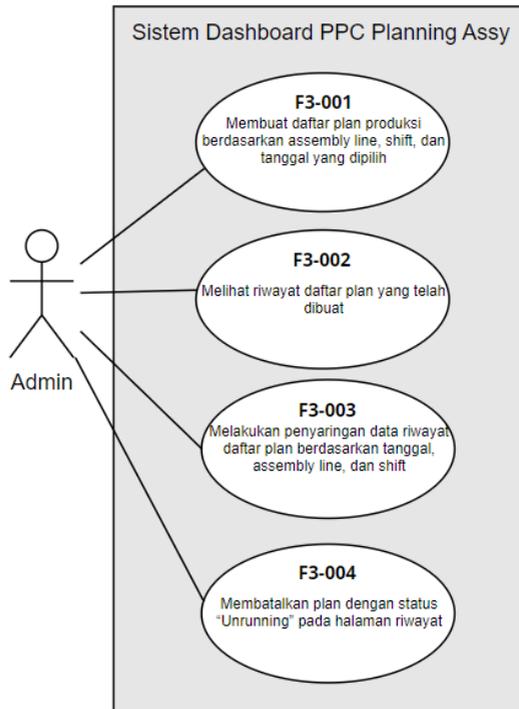
Kode	Deskripsi
F3-001	Admin dapat membuat daftar plan produksi berdasarkan <i>assembly line</i> , shift, dan tanggal yang dipilih
F3-002	Admin dapat melihat riwayat daftar plan yang telah dibuat
F3-003	Admin dapat melakukan penyaringan data riwayat daftar plan berdasarkan tanggal, <i>assembly line</i> , dan shift
F3-004	Admin dapat membatalkan plan dengan status “Unrunning” pada halaman riwayat

**Tabel 4. 6** Kebutuhan Nonfungsional Dashboard PPC Planning Assy

Kode	Deskripsi
NF3-001	Sistem dapat digunakan pada komputer dengan spesifikasi menengah ke bawah
NF3-002	Sistem memiliki antarmuka yang mudah dipahami

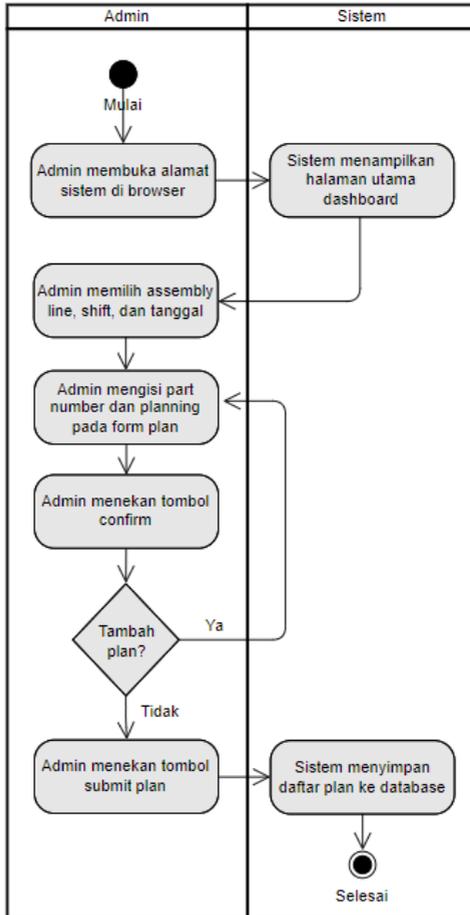
NF3-003	Sistem mendukung semua jenis browser
NF3-004	Sistem dapat melakukan <i>refresh</i> halaman riwayat secara otomatis jika terdapat data yang diperbarui

Pada Gambar 4.10, dijelaskan *use case diagram* dari aplikasi.

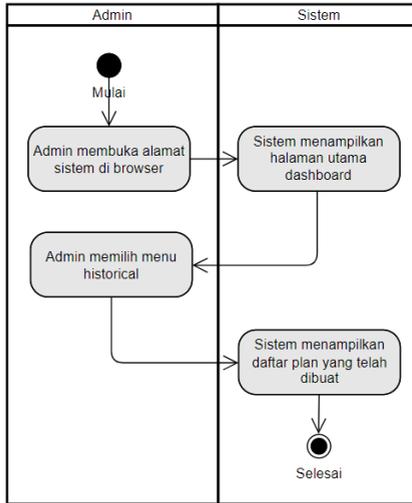


**Gambar 4. 10** Use Case Diagram Dashboard PPC Planning Assy

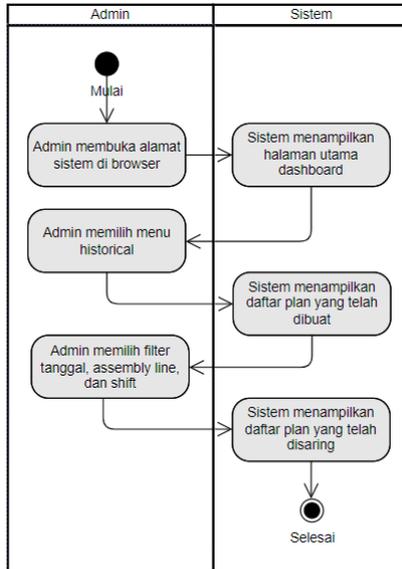
Pada Gambar 4.11, Gambar 4.12, Gambar 4.13, dan Gambar 4.14, dijelaskan *activity diagram* untuk setiap *use case* dari aplikasi.



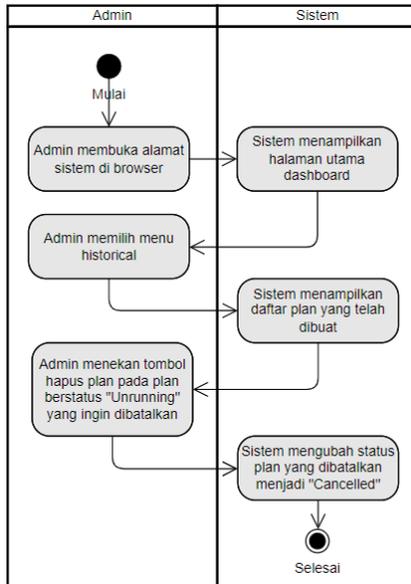
**Gambar 4. 11** Activity Diagram Use Case F3-001



**Gambar 4. 12** Activity Diagram Use Case F3-002



**Gambar 4. 13** Activity Diagram Use Case F3-003



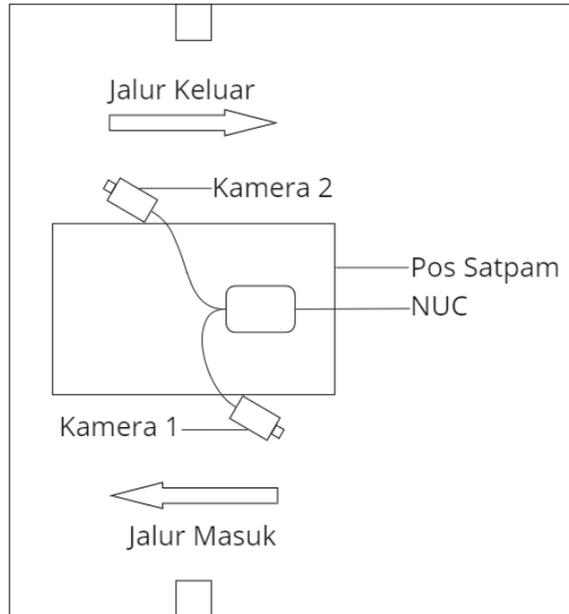
**Gambar 4. 14** Activity Diagram Use Case F3-004

## 4.2. Perancangan Sistem

Berdasarkan penjelasan analisis sistem sebelumnya, dibentuk rancangan untuk setiap proyek yang akan dikerjakan. Berikut adalah penjelasan mengenai perancangan untuk setiap proyek dashboard.

### 4.2.1. Dashboard Automatic Plate Number Recognition

Sistem Dashboard Automatic Plate Number Recognition akan dijalankan pada sebuah komputer kecil Intel NUC yang diletakkan di pos satpam perusahaan dan akan tersambung dengan dua kamera CCTV yang mengarah ke pintu masuk dan pintu keluar perusahaan. Gambar 4.15 menjelaskan skema dari sistem tersebut.



**Gambar 4. 15** Skema Sistem Dashboard APNR

Aplikasi dashboard APNR akan diimplementasikan menggunakan beberapa alat. Untuk bagian *frontend*, akan digunakan HTML, CSS, Javascript, Bootstrap, dan JQuery. Untuk bagian *backend*, digunakan *framework* Flask sebagai server web yang menggunakan bahasa Python dan Sqlite3 sebagai basis data. Dashboard ini akan memiliki dua halaman, yaitu halaman utama yang terdiri dari tangkapan kamera CCTV dan data hasil ekstraksi serta halaman riwayat ekstraksi yang menampilkan data-data plat nomor yang telah diekstraksi sebelumnya.

Pada bagian basis data, dengan nama `apnr_log`, akan dibentuk dua tabel. Kedua tabel tersebut adalah tabel log untuk menyimpan data-data tangkapan plat nomor dan tabel karyawan untuk menyimpan data karyawan. Pada

Gambar 4.16, dijelaskan CDM dari basis data yang terdiri dari tabel-tabel tersebut.

log			
id_log	Integer	M	PI
nomor_kendaraan	Varchar(10)	M	
waktu_masuk	DateTime	M	
waktu_keluar	DateTime	M	
pemilik	Varchar(20)	M	
status_kendaraan	Varchar(20)	M	

karyawan			
id_karyawan	Integer	M	PI
nama	Varchar(20)	M	
plat_nomor_kenda	Varchar(8)	M	

**Gambar 4. 16** CDM Basis Data Dashboard APNR

Selain itu, bagian *backend* juga menggunakan alat-alat yang berkaitan dengan visi komputer, seperti TensorFlow, PyTorch, OpenCV, Numpy, dan OpenVINO, untuk menerapkan fungsi utama dari APNR, yaitu mengekstraksi teks plat nomor pada kendaraan bermotor dari tangkapan kamera CCTV. Proses ekstraksi tersebut dibagi menjadi dua tahap, yaitu deteksi plat nomor dan ekstraksi plat nomor. Untuk dua tahap tersebut, diperlukan dua buah model *deep learning*, yaitu model deteksi objek dan model OCR. Model deteksi objek menghasilkan koordinat letak objek plat nomor pada sebuah gambar input. Dengan koordinat tersebut, gambar input dapat dipotong sehingga hanya fokus pada gambar plat nomor saja. Setelah gambar dipotong, gambar tersebut dijadikan sebagai input untuk model OCR dan model OCR akan mengekstrak teks nomor polisi dari gambar input.

Model deteksi objek akan dikembangkan menggunakan TensorFlow Object Detection API. Dengan API ini, dapat dibuat sebuah model deteksi objek yang dapat mendeteksi objek apa pun yang diinginkan dengan melakukan *fine tuning* pada sebuah model terlatih yang ada di TensorFlow Model Zoo terhadap dataset objek yang ingin dideteksi. Dataset yang akan digunakan untuk melakukan *fine tuning* model deteksi objek berupa sebuah gambar kendaraan bermotor dengan plat nomor Indonesia

sebagai input dan berkas XML sebagai label yang berisi koordinat letak objek plat nomor dalam format PascalVOC pada gambar input. Untuk memperoleh file XML, digunakan `labelImg` untuk menganotasi setiap gambar dataset secara manual. Jumlah gambar yang akan digunakan sebagai dataset adalah 1000 gambar dengan pembagian 900 gambar untuk *training* dan 100 gambar untuk *testing*. Jumlah berkas XML yang digunakan sebagai label mengikuti jumlah gambar yang digunakan sebagai dataset. Model terlatih yang akan digunakan sebagai model dasar adalah SSD MobileNet V2 karena model ini dinilai cukup ringan untuk dijalankan. Di samping itu, model OCR akan dikembangkan menggunakan TrOCR dari HuggingFace dengan cara melakukan *fine tuning* pada model TrOCR tersebut terhadap *font* plat nomor Indonesia. Dataset yang akan digunakan untuk melakukan *fine tuning* model TrOCR berupa gambar plat nomor Indonesia sebagai input dan teks nomor polisi dari gambar input sebagai label yang disimpan dalam sebuah berkas `.txt`. Jumlah gambar yang akan digunakan sebagai dataset untuk proses *training* adalah 915 gambar. Berkas `.txt` yang berisi label untuk proses *training* memiliki jumlah baris data sebanyak jumlah gambar yang digunakan sebagai dataset.

#### **4.2.2. Perbaikan Dashboard Damping Force**

Beberapa perubahan dilakukan pada Dashboard Damping Force. Aplikasi ini akan diberikan menu utama untuk mengakses visualisasi data seluruh *line shock absorber*. Tampilan *frontend* dashboard yang kurang responsif serta desain dashboard juga diperbaiki menjadi lebih baik. Selain itu, logika *backend* pada aplikasi juga diperbaiki agar dapat menampilkan visualisasi data pada seluruh *line shock absorber*.

Dashboard Damping Force yang baru akan memiliki tiga halaman, yaitu menu utama untuk mengakses semua *line shock absorber*, visualisasi data berbentuk grafik garis, dan visualisasi data berbentuk tabel. Halaman visualisasi data akan memiliki filter-filter, seperti jenis model, tanggal, dan shift untuk menyaring data. Selain itu, akan terdapat tombol “Cetak Laporan” untuk mencetak data-data yang ditentukan berdasarkan filter-filter yang dipilih lalu disimpan dalam format excel. Dashboard dikembangkan menggunakan teknologi yang sesuai dengan server yang dimiliki oleh PT Kayaba Indonesia. Untuk *backend*, digunakan bahasa pemrograman PHP dan basis data MariaDB yang terintegrasi dalam XAMPP. Untuk *frontend*, digunakan HTML, CSS, JavaScript, Bootstrap, JQuery, Chart.js, PHPExcel, Bootstrap Select, dan Datatable.

Pada bagian basis data, dengan nama *dampingforce*, hanya akan dibentuk sebuah tabel. Tabel tersebut adalah tabel *data2* yang menyimpan data-data *damping force* dari seluruh *production line shock absorber* yang ada. Pada Gambar 4.17, dijelaskan CDM dari basis data yang terdiri dari tabel tersebut.

```
dampingforce data2
id : int(11)
model : varchar(255)
tanggal : date
time : time
temp : varchar(50)
zero : varchar(50)
# status : int(11)
# rebound1 : int(11)
# compression1 : int(11)
# pat1 : float
# rebound2 : int(11)
# compression2 : int(11)
# pat2 : float
file : varchar(50)
u_date : datetime
speed1 : varchar(50)
speed2 : varchar(50)
rebMax1 : varchar(50)
rebMin1 : varchar(50)
compMax1 : varchar(50)
compMin1 : varchar(50)
rebMax2 : varchar(50)
rebMin2 : varchar(50)
compMax2 : varchar(50)
compMin2 : varchar(50)
zeroMin : varchar(50)
zeroMax : varchar(50)
rebPat : varchar(50)
compPat : varchar(50)
id_f : varchar(50)
line : varchar(50)
deskripsi : varchar(50)
```

**Gambar 4. 17** CDM Basis Data Dashboard Damping Force

### 4.2.3. Dashboard PPC Planning Assy

Dashboard PPC Planning Assy memiliki tiga halaman, yaitu halaman login, halaman planning, dan halaman historical. Halaman login digunakan untuk autentikasi pengguna dashboard, halaman planning digunakan untuk penambahan plan produksi oleh admin

dan halaman historical digunakan untuk melihat daftar plan yang telah ditambahkan oleh admin. Halaman planning memiliki beberapa elemen, seperti input untuk data *line assy*, shift, dan tanggal. Selain itu, terdapat form untuk menambahkan plan produksi dengan beberapa input, seperti *part number*, *model name*, *lot size*, dan *planning*. Halaman historical memiliki beberapa elemen, yaitu filter untuk menyaring data yang terdiri dari tanggal, *line assy*, dan shift, serta tabel riwayat plan yang dimunculkan sesuai filter yang dipilih.

Dashboard akan dikembangkan menggunakan teknologi yang sesuai dengan server yang dimiliki oleh PT Kayaba Indonesia. Untuk *backend*, digunakan bahasa pemrograman PHP dan basis data MariaDB yang terintegrasi dalam XAMPP. Untuk *frontend*, digunakan HTML, CSS, JavaScript, Bootstrap, JQuery, dan Bootstrap Select. Selain itu, dashboard akan dikembangkan dengan arsitektur Model-View-Controller (MVC) untuk memudahkan dalam pemeliharaan kode sumber atau penambahan fitur dashboard.

Pada bagian basis data, dengan nama model\_ff, akan dibentuk tujuh tabel. Tabel-tabel tersebut adalah *delv\_plan*, *master\_model\_ff*, *master\_model\_ff\_copy*, *pos\_it*, *pos\_ot*, *pos\_abc*, *tables\_update*, dan *users*. Tabel yang akan digunakan untuk saat ini adalah tabel *delv\_plan*, *master\_model\_ff*, *tables\_update*, dan *users* saja dan tabel yang lain akan digunakan di masa mendatang. Tabel *delv\_plan* digunakan untuk menyimpan plan, tabel *master\_model\_ff* digunakan untuk menyimpan informasi terkait model yang ada, tabel *tables\_update* merupakan sebuah view yang akan mencatat waktu perubahan dari tabel-tabel lain, dan tabel *users* digunakan untuk menyimpan data pengguna. Pada Gambar 4.18, dijelaskan CDM dari basis data yang terdiri dari tabel-tabel tersebut.



**Gambar 4. 18** CDM Basis Data Dashboard PPC Planning Assy

*[Halaman ini sengaja dikosongkan]*

## **BAB V**

### **IMPLEMENTASI SISTEM**

Pada bab ini, dibahas mengenai implementasi proyek yang mengacu pada rancangan sistem yang telah dijelaskan sebelumnya. Penjelasan akan dibagi menjadi tiga bagian, yaitu implementasi Dashboard APNR, implementasi perbaikan Dashboard Damping Force, dan implementasi Dashboard PPC Planning Assy.

#### **5.1. Implementasi Dashboard APNR**

##### **5.1.1. Implementasi Model Deteksi Objek**

Langkah pertama yang dilakukan adalah menyiapkan dataset yang nantinya akan digunakan dalam proses *fine-tuning* sebuah model dari TensorFlow Model Zoo, yaitu SSD MobileNet V2. Sejumlah 1000 gambar kendaraan bermotor (mobil dan sepeda motor) dikumpulkan, diubah formatnya menjadi file JPG, dan diubah ukurannya menjadi 640 x 640 piksel. Setelah itu, gambar-gambar tersebut dibagi menjadi dataset untuk training dan testing dengan perbandingan 9:1. Setelah dibagi, setiap gambar dianotasi (menandai letak plat nomor pada gambar dengan sebuah kotak) secara manual dengan `labelImg` yang kemudian akan menghasilkan file XML untuk setiap gambar dan file-file XML tersebut akan menjadi label dari dataset.



**Gambar 5. 1** Sampel Dataset Model Deteksi Objek



**Gambar 5. 2** Penganotasian Gambar

```

<annotation verified="yes">
  <folder>train</folder>
  <filename>platomor425.jpg</filename>
  <path>platomor425.jpg</path>
  <source>
    <database>Unknown</database>
  </source>
  <size>
    <width>900</width>
    <height>1200</height>
    <depth>3</depth>
  </size>
  <segmented>0</segmented>
  <object>
    <name>plat-nomor</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <bndbox>
      <xmin>154</xmin>
      <ymin>384</ymin>
      <xmax>334</xmax>
      <ymax>487</ymax>
    </bndbox>
  </object>
</annotation>

```

**Gambar 5. 3** Contoh File Label dalam Format PascalVOC

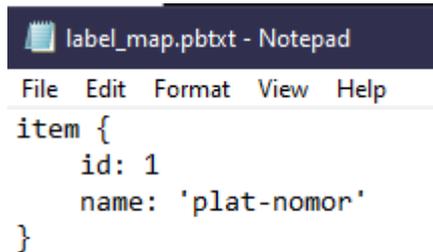
Langkah kedua adalah menyiapkan sebuah folder yang memiliki struktur sebagai berikut.

```
training_dataset/  
-- annotations/  
  -- label_map.pbtxt  
  -- test.record  
  -- train.record  
-- exported-models/  
-- images/  
  -- test/  
  -- train/  
-- models/  
  -- my_ssd_mobilenet_v2/  
-- pre-trained-models/  
  -- ssd_mobilenet_v2_320x320_coco17_tpu-8/  
-- exporter_main_v2.py  
-- model_main_tf2.py
```

**Gambar 5.4** Struktur Folder untuk Proses Fine-Tuning Model Deteksi Objek

Dataset yang telah disiapkan sebelumnya diletakkan pada folder images. Pada folder annotations, terdapat file label\_map.pbtxt yang berisi kelas-kelas yang akan diprediksi oleh model, yaitu plat nomor. File test.record dan train.record merupakan file dengan format TFRecord yang dibuat dari dataset yang ada di folder images. Dataset diubah menjadi file TFRecord agar proses *fine tuning* nantinya menjadi lebih efisien. Folder pre-trained-models berisi model deteksi objek yang akan digunakan sebagai model dasar untuk proses *fine tuning* dengan dataset yang diinginkan. Model yang akan digunakan adalah SSD MobileNet V2 yang diperoleh dari TensorFlow Model Zoo. Folder models berisi file config dari model yang ada di folder pre-trained-models yang

telah disesuaikan pengaturannya untuk proses *fine tuning* nanti. Folder `exported-models` akan digunakan sebagai tempat hasil model baru yang telah dibuat. File `model_main_tf2.py` dan `exporter_main_v2.py` diperoleh dari GitHub TensorFlow Object Detection API dan secara berurutan digunakan untuk melakukan proses *fine tuning* model serta mengeksport model.



```
label_map.pbtxt - Notepad
File Edit Format View Help
item {
  id: 1
  name: 'plat-nomor'
}
```

**Gambar 5. 5** Kelas yang akan Dideteksi Model

Setelah struktur folder selesai dibuat, langkah selanjutnya adalah membuat sebuah virtual environment Python di Google Colab dan memasang beberapa pustaka, yaitu TensorFlow, TensorFlow Object Detection API, dan Protobuf. Setelah itu, proses *fine tuning* dapat dimulai dengan menjalankan file `model_main_tf2.py` beserta argumennya lalu mengeksport model yang telah dibuat dengan menjalankan file `exporter_main_v2.py` beserta argumennya. Hasil model yang diekspor akan disimpan di dalam folder `exported-models`.

```

import os
import pathlib
!pip install tensorflow==2.13.0

folder_name = 'training_dataset'
model_name = 'my_ssd_mobilenet_v2'

if "models" in pathlib.Path.cwd().parts:
    while "models" in pathlib.Path.cwd().parts:
        os.chdir('..')
elif not pathlib.Path('models').exists():
    !git clone --depth 1
    https://github.com/tensorflow/models

%%bash
cd models/research/
protoc object_detection/protos/*.proto --
python_out=.
cp object_detection/packages/tf2/setup.py .
python -m pip install .

!python model_main_tf2.py
--model_dir={folder_name}/models/{model_name}
--
pipeline_config_path={folder_name}/models/{model_n
ame}/pipeline.config

!python exporter_main_v2.py
--input_type image_tensor --pipeline_config_path
{folder_name}/models/{model_name}/pipeline.config
--trained_checkpoint_dir
{folder_name}/models/{model_name}/
--output_directory {folder_name}/exported-
models/{model_name}

```

### **Kode Sumber 5.1** Proses *Fine Tuning* dan Ekspor Model Deteksi Objek

Setelah diperoleh model deteksi objek yang dapat mendeteksi plat nomor pada gambar, model tersebut

dioptimalkan menggunakan Intel OpenVINO agar model dapat dijalankan dengan lebih ringan nantinya.

```
%pip install -q "openvino>=2023.1.0" "numpy>=1.21.0"
"opencv-python" "matplotlib>=3.4,<3.5.3"

from pathlib import Path

import openvino as ov

model_dir = Path("model")
model_dir.mkdir(exist_ok=True)

tf_model_dir = model_dir /
"extraction_oda_my_ssd_mobilenet_v2_trained_01_11_2023
_21_06_10_only_exported/saved_model"
tf_model_dir.mkdir(exist_ok=True)

ir_model_dir = model_dir / "ir"
ir_model_dir.mkdir(exist_ok=True)

model_name = # Nama model

openvino_ir_path = ir_model_dir / f"{model_name}.xml"

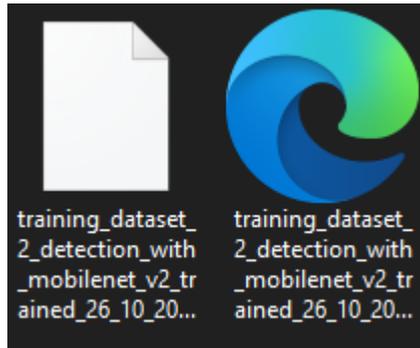
ov_model = ov.convert_model(
    tf_model_dir,
    input=[[1, 640, 640, 3]],
)

ov.save_model(ov_model, openvino_ir_path)
```

### **Kode Sumbur 5. 2** Proses Konversi ke OpenVINO

Proses konversi model TensorFlow menjadi OpenVINO menghasilkan dua file, yaitu file XML dan

BIN yang dapat digunakan nantinya pada *backend* dashboard APNR.



**Gambar 5. 6** Model Deteksi Objek Plat Nomor

### **5.1.2. Implementasi Model OCR**

Langkah pertama yang dilakukan adalah menyiapkan dataset yang nantinya akan digunakan dalam proses *fine-tuning* sebuah model bernama TrOCR. Sebanyak 915 gambar plat nomor dikumpulkan dan diubah formatnya menjadi file JPG. Seluruh gambar tersebut akan digunakan sebagai dataset training untuk proses *fine tuning* nantinya. Setelah itu, dibuat sebuah file TXT dengan nama dataset.txt yang menyimpan label untuk setiap gambar dataset.



**Gambar 5. 7** Sampel Dataset Model OCR

```

dataset.txt - Notepad
File Edit Format View Help
image label
platnomor0.jpg E 4092 PAE
platnomor1.jpg F 2716 FZ
platnomor2.jpg A 7736 HX
platnomor3.jpg D 5431 VB
platnomor4.jpg D 5624 KB
platnomor5.jpg D 800 DRI
platnomor6.jpg H 6202 ZE
  
```

**Gambar 5. 8** Label Dataset untuk Fine Tuning Model OCR

Langkah selanjutnya adalah melakukan *preprocessing* dataset dan memulai proses *fine tuning*. Setelah selesai, diperoleh model OCR yang dapat mengekstrak teks nomor polisi pada gambar plat nomor.

```

folder_name =
'training_extraction_using_trocr_dataset'

df = pd.read_csv('dataset.txt', sep='\t')
train_df, test_df = train_test_split(df,
test_size=0.2)
train_df.reset_index(drop=True, inplace=True)
test_df.reset_index(drop=True, inplace=True)

@dataclass(frozen=True)
class TrainingConfig:
    BATCH_SIZE:    int = 8
    EPOCHS:        int = 50
    LEARNING_RATE: float = 0.00005

@dataclass(frozen=True)
class DatasetConfig:
    DATA_ROOT:    str = folder_name

@dataclass(frozen=True)
class ModelConfig:
    MODEL_NAME:    str = 'microsoft/trocr-small-printed'

train_transforms = transforms.Compose([
    transforms.ColorJitter(brightness=.4, hue=.2),
    transforms.RandomPerspective(distortion_scale=.2,
p=.5)
])

```

```

class CustomOCRDataset(Dataset):
    def __init__(self, root_dir, df, processor,
max_target_length=128):
        self.root_dir = root_dir
        self.df = df
        self.processor = processor
        self.max_target_length = max_target_length

    def __len__(self):
        return len(self.df)

    def __getitem__(self, idx):
        file_name = self.df['image'][idx]
        text = self.df['label'][idx]
        image = Image.open(self.root_dir +
file_name).convert('RGB')
        image = train_transforms(image)
        pixel_values = self.processor(image,
return_tensors='pt').pixel_values
        labels = self.processor.tokenizer(
            text,
            padding='max_length',
            max_length=self.max_target_length
        ).input_ids
        labels = [label if label !=
self.processor.tokenizer.pad_token_id else -100
for label in labels]
        encoding = {"pixel_values":
pixel_values.squeeze(), "labels":
torch.tensor(labels)}
        return encoding

```

```

processor =
TrOCRProcessor.from_pretrained(ModelConfig.MODEL_NAME)
train_dataset = CustomOCRDataset(
    root_dir=DatasetConfig.DATA_ROOT + '/',
    df=train_df,
    processor=processor
)
valid_dataset = CustomOCRDataset(
    root_dir=DatasetConfig.DATA_ROOT + '/',
    df=test_df,
    processor=processor
)

model =
VisionEncoderDecoderModel.from_pretrained(ModelConfig.MODEL_NAME)
model.to(device)

model.config.decoder_start_token_id =
processor.tokenizer.cls_token_id
model.config.pad_token_id =
processor.tokenizer.pad_token_id
model.config.vocab_size =
model.config.decoder.vocab_size
model.config.eos_token_id =
processor.tokenizer.sep_token_id
model.config.max_length = 64
model.config.early_stopping = True
model.config.no_repeat_ngram_size = 3
model.config.length_penalty = 2.0
model.config.num_beams = 4

optimizer = optim.AdamW(
    model.parameters(),
    lr=TrainingConfig.LEARNING_RATE,
    weight_decay=0.0005
)

```

```

cer_metric = evaluate.load('cer')

def compute_cer(pred):
    labels_ids = pred.label_ids
    pred_ids = pred.predictions

    pred_str = processor.batch_decode(pred_ids,
skip_special_tokens=True)
    labels_ids[labels_ids == -100] =
processor.tokenizer.pad_token_id
    label_str = processor.batch_decode(labels_ids,
skip_special_tokens=True)

    cer = cer_metric.compute(predictions=pred_str,
references=label_str)

    return {"cer": cer}

training_args = Seq2SeqTrainingArguments(
    predict_with_generate=True,
    evaluation_strategy='epoch',

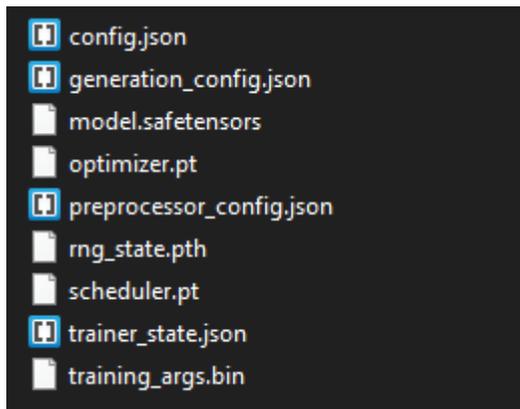
per_device_train_batch_size=TrainingConfig.BATCH_S
IZE,

per_device_eval_batch_size=TrainingConfig.BATCH_SI
ZE,
    fp16=True,
    output_dir='seq2seq_model_printed/',
    logging_strategy='epoch',
    save_strategy='epoch',
    save_total_limit=5,
    num_train_epochs=TrainingConfig.EPOCHS
)

```

```
trainer = Seq2SeqTrainer(  
    model=model,  
    tokenizer=processor.feature_extractor,  
    args=training_args,  
    compute_metrics=compute_cer,  
    train_dataset=train_dataset,  
    eval_dataset=valid_dataset,  
    data_collator=default_data_collator  
)  
  
res = trainer.train()
```

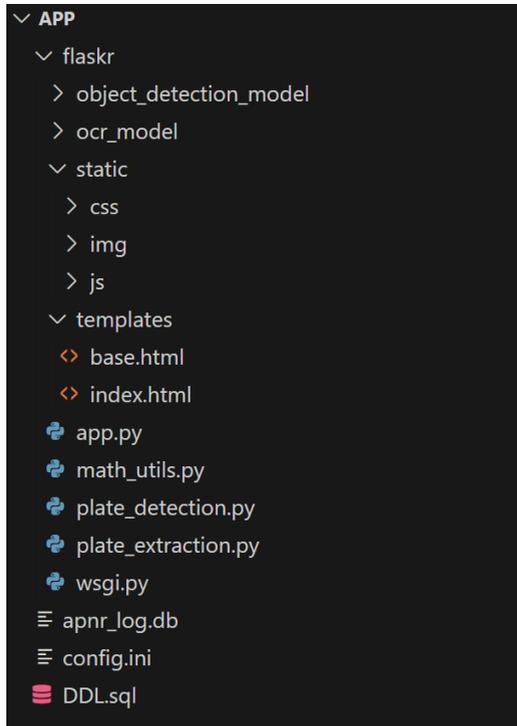
**Kode Sumber 5. 3** Proses Fine Tuning Model OCR



**Gambar 5. 9** Model OCR Plat Nomor

### 5.1.3. Implementasi Dashboard

Aplikasi dashboard APNR memiliki struktur seperti pada Gambar 5.10.



**Gambar 5. 10** Struktur Kode Dashboard APNR

Aplikasi utama berada pada folder flaskr. Di dalamnya, terdapat beberapa folder dan file. Folder static berisi file pustaka CSS, pustaka JavaScript, dan gambar. Folder template berisi halaman HTML dari dashboard. Folder object\_detection\_model dan ocr\_model berisi model untuk deteksi dan ekstraksi plat nomor. File app.py berfungsi sebagai program utama dashboard. File math\_utils.py berisi implementasi fungsi-fungsi matematika. File plate\_detection.py dan plate\_extraction.py berisi implementasi fitur deteksi dan ekstraksi plat nomor dari dashboard. File wsgi.py berfungsi untuk menjalankan aplikasi. File apnr\_log.db

merupakan basis data SQLite3 untuk aplikasi. File config.ini berisi pengaturan aplikasi. File DDL.sql berisi implementasi DDL untuk membuat tabel-tabel pada basis data aplikasi.

Beberapa kode sumber python dan SQL akan dijelaskan pada Kode Sumber 5.4 hingga Kode Sumber 5.7 dalam bentuk pseudocode.

```
PROGRAM app.py

Impor pustaka

// Membaca config
READFILE "config.ini", cfg
cfg := Map(cfg)

// Kamera
DECLARE cam1 := instansiasi objek kamera OpenCV
DECLARE cam2 := instansiasi objek kamera OpenCV

// Inisialisasi aplikasi Flask
DECLARE app := instansiasi objek aplikasi Flask

// Pengaturan database SQLite3
Membuat koneksi ke database

// Fungsi untuk memasukkan data tangkapan ke log
PROCEDURE db_insert_log(result_number, result_image,
cam_id)
result_number := ambil nomor polisi dari string
result_number

IF result_number bukan nomor polisi THEN
    result_number := "TIDAK TERDETEKSI"
```

```

ENDIF

DECLARE curr_datetime := tanggal dan waktu saat ini
dalam format m/d/Y H:M:S
DECLARE owner

IF result_number adalah milik seorang karyawan THEN
    owner := "KARYAWAN"
ELSE
    owner := "TAMU"
ENDIF

// Logika untuk kamera 1 (kamera pintu masuk)
IF cam_id adalah id kamera 1 THEN
    IF result_number belum tercatat di log dan tidak
berstatus CHECK_IN THEN
        Melakukan insert data kendaraan yang masuk
ke tabel log
    ENDIF

    Menampilkan data kendaraan yang masuk pada
dashboard

// Logika untuk kamera 2 (kamera pintu keluar)
ELSE
    IF result_number tercatat di log dan berstatus
CHECK_IN THEN
        Melakukan update data waktu keluar dan
status kendaraan dari kendaraan tersebut
    ENDIF

    Menampilkan data kendaraan yang keluar pada
dashboard
ENDIF
ENDPROCEDURE

```

```

PROCEDURE generates_frames_async(cam, cam_id)
DECLARE plate_detection := instansiasi objek kelas
PlateDetection
DECLARE plate_extraction := instansiasi objek kelas
PlateExtraction
DECLARE result
DECLARE cropped_plate
DECLARE result_number

WHILE cam masih mengembalikan gambar
    result := Hasil prediksi letak plat nomor pada
gambar oleh plate_detection
    Menampilkan result pada dashboard

    IF jarak plat nomor yang dideteksi dekat dengan
kamera THEN
        cropped_plate := Potongan gambar plat nomor
yang diprediksi
        result_number := Hasil ekstraksi nomor
polisi dari cropped_plate oleh plate_extraction

        CALL db_insert_log(result_number,
cropped_plate, cam_id)

    ENDWHILE
ENDPROCEDURE

// Route dashboard
// Rute untuk mengambil data prediksi
GET /new_log_data
// Rute untuk mengambil halaman index
GET /
// Rute untuk mengambil video kamera 1
GET /camera1
// Rute untuk mengambil video kamera 2
GET /camera2

END

```

#### Kode Sumber 5. 4 Pseudocode app.py

```

PROGRAM plate_detection.py

Impor pustaka

CLASS PlateDetection
PUBLIC compiled_model

PUBLIC PROCEDURE NEW(model_xml)
compiled_model := Membaca model deteksi plat nomor
dengan API OpenVINO
ENDPROCEDURE

PUBLIC FUNCTION predict(frame)
DECLARE result := hasil prediksi model terhadap
input frame
RETURN result
ENDFUNCTION

PUBLIC FUNCTION preprocess_frame(frame)
RETURN Frame yang telah disesuaikan untuk input
model
ENDFUNCTION

PUBLIC FUNCTION postprocess_frame(frame, label_map,
boxes, class_ids, scores, is_normalized)
DECLARE selected_boxes : LIST
DECLARE selected_centroids : LIST

selected_boxes := boxes dengan skor minimal yang
telah ditentukan
selected_centroids := titik tengah dari setiap kotak
pada selected_boxes

// Menganotasi gambar dengan hasil prediksi
frame := visualize_inference_result(frame,
label_map, selected_boxes, class_ids, scores,
selected_centroids)

```

```

RETURN frame, selected_boxes, selected_centroids
ENDFUNCTION

PRIVATE FUNCTION add_detection_box(box, image,
label)
Menggambar persegi panjang pada image berdasarkan
koordinat dari box

IF label != NULL THEN
    Menggambar label prediksi pada image
ENDIF

RETURN image
ENDFUNCTION

PUBLIC visualize_inference_result(image, labels_map,
boxes, class_ids, scores, centroids)
Menggambar seluruh boxes, label, dan scores pada
image dengan add_detection_box
Menggambar seluruh centroids pada image

RETURN image
ENDFUNCTION

ENDCLASS

END

```

### **Kode Sumber 5. 5** Pseudocode plate\_detection.py

```

PROGRAM plate_extraction.py

Impor pustaka

CLASS PlateExtraction
PUBLIC processor
PUBLIC trained_model

PUBLIC PROCEDURE NEW(processor_name, trained_model)

```

```

processor := Memuat preprocessor TrOCR
trained_model := Memuat VisionEncoderDecoderModel
dari trained_model
ENDPROCEDURE

PUBLIC FUNCTION extract(img)
DECLARE generated_text
generated_text := Hasil ekstraksi teks dari img
RETURN generated_text
ENDFUNCTION

ENDCLASS

END

```

**Kode Sumber 5. 6** Pseudocode plate\_extraction.py

```

CREATE TABLE log(
    id_log INTEGER PRIMARY KEY AUTOINCREMENT,
    nomor_kendaraan VARCHAR(10) NOT NULL,
    waktu_masuk DATETIME NOT NULL,
    waktu_keluar DATETIME,
    pemilik VARCHAR(20) NOT NULL,
    status_kendaraan VARCHAR(20) NOT NULL
);

CREATE TABLE karyawan(
    id_karyawan INTEGER PRIMARY KEY
    AUTOINCREMENT,
    nama VARCHAR(20) NOT NULL,
    plat_nomor_kendaraan VARCHAR(8) NOT NULL
);

```

**Kode Sumber 5. 7** DDL Basis Data APNR

Gambar 5.11 merupakan tampilan halaman utama dari dashboard APNR. Halaman tersebut memiliki beberapa bagian penting, yaitu tayangan langsung dari

pintu masuk dan keluar PT Kayaba Indonesia serta data kendaraan yang tertangkap kamera melewati pintu masuk dan keluar perusahaan. Data kendaraan tersebut nantinya akan dimasukkan ke basis data ke dalam tabel log.

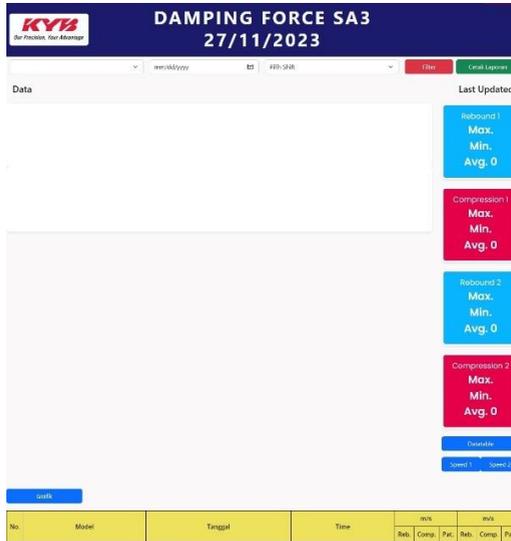


**Gambar 5. 11** Halaman Utama Dashboard APNR

## 5.2. Implementasi Perbaikan Dashboard Damping Force

### 5.2.1. Dashboard Damping Force sebelum Perbaikan

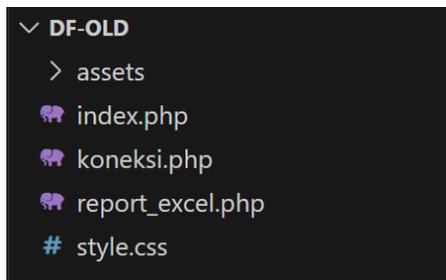
Sebelum diperbaiki, Dashboard Damping Force memiliki beberapa masalah, misalkan data-data yang tidak bisa muncul serta hanya dapat memilih data yang berasal dari *line Shock Absorber 3*. Pada Subbab 5.2.2 dan 5.2.3, akan dijelaskan implementasi dari perbaikan dashboard sesuai dengan rancangan pada Subbab 4.2.2.



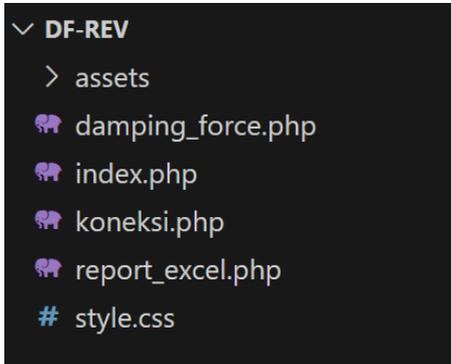
**Gambar 5. 12** Dashboard Damping Force sebelum Perbaikan

### 5.2.2. Implementasi Perbaikan Backend

Aplikasi Dashboard Damping Force memiliki struktur kode seperti pada Gambar 5.13 dan Gambar 5.14.



**Gambar 5. 13** Struktur Kode Dashboard Damping Force sebelum Perbaikan



**Gambar 5. 14** Struktur Kode Dashboard Damping Force setelah Perbaikan

Sebelum perbaikan, `index.php` merupakan halaman utama dashboard yang menampilkan visualisasi data pada *line Shock Absorber 3*. Setelah perbaikan, `index.php` merupakan halaman utama dashboard yang menampilkan menu untuk mengakses visualisasi data dari semua *line shock absorber* dan file `damping_force.php` menjadi halaman untuk menampilkan visualisasi data dari *line shock absorber* yang dipilih dari halaman utama. Folder `assets` berisi pustaka-pustaka, gambar, dan font. File `koneksi.php` berfungsi untuk menghubungkan *backend* dengan basis data. File `report_excel.php` berfungsi untuk mencetak file laporan dari data dengan filter-filter tertentu. File `style.css` berfungsi untuk mengatur CSS dari dashboard.

Selanjutnya, akan dijelaskan beberapa bagian *backend* dashboard yang diperbaiki agar dashboard dapat berfungsi dengan seharusnya. Pada Kode Sumber 5.8, dijelaskan pseudocode dari potongan kode `index.php`

untuk mengambil data *line shock absorber* yang dapat diakses visualisasi datanya.

### **Kode Sumber 5. 8** Pseudocode index.php Dashboard

```
PROGRAM index.php
DECLARE sa_line_n := 9 // Total line shock absorber
DECLARE title_date // Tanggal data terbaru pada sebuah
line

FOR i := 1 TO sa_line_n
    title_date := Query MAX(u_date) dari tabel data2
dengan line RP-SA ke-i

    IF title_date != NULL THEN
        Tampilkan menu untuk RP-SA ke-i
    ELSE
        Jangan tampilkan menu untuk RP-SA ke-i
NEXT i
END
```

### Damping Force untuk Menampilkan Menu

Untuk file *damping\_force.php*, terdapat beberapa bagian *backend* yang diperbaiki dari kode sebelumnya. Filter jenis model pada halaman visualisasi dan halaman cetak excel diperbaiki *query*-nya agar hanya menampilkan jenis model dari *line shock absorber* yang dipilih sebelumnya. Selain itu, *query* data dan pemrosesan data untuk grafik garis dan tabel juga diperbaiki. Sama seperti sebelumnya, setiap *query* diberikan filter agar hanya mengambil data dari *line shock absorber* yang dipilih sebelumnya. Selanjutnya, proses untuk menampilkan data grafik garis dan tabel diketahui masih kurang efisien, misalkan terdapat *query-query* dan logika-logika yang terlalu kompleks, sehingga memengaruhi performa dashboard. Maka dari itu, *query-query* dan logika-logika tersebut disederhanakan untuk meningkatkan performa dari dashboard.

### 5.2.3. Implementasi Perbaikan Frontend

Perbaikan pada sisi *frontend* dari dashboard meliputi penyesuaian warna elemen, tata letak elemen, ukuran elemen, dan penambahan fitur pencarian pada filter jenis model. Gambar 5.15 hingga Gambar 5.19 menunjukkan Dashboard Damping Force yang telah diperbaiki baik dari sisi *backend* maupun dari sisi *frontend*.



Gambar 5. 15 Menu Utama Dashboard Damping Force



Gambar 5. 16 Halaman Visualisasi Grafik Garis Dashboard Damping Force

**DAMPING FORCE SA8**  
12/10/2023 19:32:49

Model: D021, Tanggal: 12/10/2023, Last Updated Data DF: D021 05-10-2023 Shift 2

DF List Standard	"0.300" inch			"0.050" inch		
	Reb.	Comp.	Pat.	Reb.	Comp.	Pat.
Normal	900	155	0	195	81	6
Tolerance(±1)	130	15	30	75	48	20

No.	Model	Tanggal	Time	"0.300" inch			"0.050" inch		
				Reb.	Comp.	Pat.	Reb.	Comp.	Pat.
1	D021	2023-10-02	00:05:29	914	206	6.3	190	138	0
2	D021	2023-10-02	00:05:48	709	227	0	212	168	0
3	D021	2023-10-02	00:07:20	1024	484	46.5	270	142	13.3
4	D021	2023-10-02	00:10:26	990	180	0	235	63	0
5	D021	2023-10-02	00:20:41	990	184	0	233	67	0

Gambar 5. 17 Halaman Tabel Dashboard Damping Force

**Cetak Laporan Damping Force**

Pilih Model: D021D26A-F RH (BZE50)

Dari tanggal: 10/02/2023 Hingga tanggal: 10/10/2023

**Cetak**

Gambar 5. 18 Menu Cetak Laporan Dashboard Damping Force

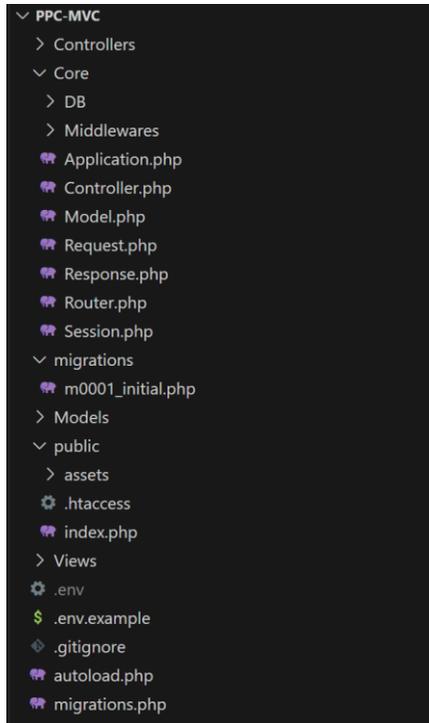
DATA DAMPING FORCE D021										Dari tanggal:	2023-10-02
										Sampai tanggal:	2023-10-11
ID	MODEL	TANGGAL	WAKTU	TEMP.	" 0.300"			" 0.050"			
					Rebound	Compression	Pattern	Rebound	Compression	Pattern	
1	D021	2023-10-02	01:30:28	29.8	967	166	6	212	46	6.8	
2	D021	2023-10-02	01:30:42	29.8	948	176	2.2	213	50	4.3	
3	D021	2023-10-02	01:30:56	29.9	983	180	0	217	62	0	
4	D021	2023-10-02	01:31:10	29.9	987	172	0	222	60	0	
5	D021	2023-10-02	01:31:25	29.8	977	167	3.4	166	36	24.8	
6	D021	2023-10-02	01:31:42	29.8	960	177	0.2	199	51	2.9	
7	D021	2023-10-02	01:31:56	29.8	962	170	3.6	221	46	8.3	
8	D021	2023-10-02	01:32:10	29.8	954	176	0	215	56	0	
9	D021	2023-10-02	01:32:24	29.8	966	172	3.4	211	48	9.7	
10	D021	2023-10-02	01:32:39	29.8	982	168	5	190	47	4.9	
11	D021	2023-10-02	01:32:57	29.8	951	173	0.4	208	55	0	
12	D021	2023-10-02	01:33:14	29.8	1008	182	0	220	60	0	
13	D021	2023-10-02	01:33:29	29.7	988	180	0	232	55	0	

Gambar 5. 19 Contoh Laporan Dashboard Damping Force

## 5.3. Implementasi Dashboard PPC Planning Assy

### 5.3.1. Struktur Kode

Struktur kode dashboard disusun berdasarkan arsitektur MVC yang memiliki empat bagian utama, yaitu Core, Model, View, dan Controller. Bagian Core terdiri dari kelas-kelas inti dari aplikasi MVC. Bagian Model terdiri dari dua jenis, yaitu model yang mewakili sebuah tabel pada basis data dan model yang mewakili sebuah data pada dashboard. Jenis model yang pertama antara lain DelvPlan, MasterModelFF, TablesUpdate, dan User. Jenis model yang kedua antara lain Historical (data riwayat plan yang telah dibuat), LoginForm (data login), PlanForm (data daftar plan yang dibuat), dan PlanRow (data baris plan). Bagian View terdiri dari halaman-halaman pada dashboard, yaitu halaman login, planning, historical, dan 404 Not Found. Bagian Controller terdiri dari AJAXController (mengatur *autofill* plan halaman planning dan *autorefresh* halaman historical), AuthController (mengatur proses login), dan SiteController (mengatur perpindahan halaman). Pada Gambar 5.20, ditunjukkan struktur kode secara lengkap dari dashboard.



**Gambar 5. 20** Struktur Kode Dashboard PPC Planning Assy

### 5.3.2. Implementasi Halaman Login

Halaman login dashboard memiliki tampilan yang umum yang ada pada halaman login. Data login admin dashboard disimpan di dalam basis data pada tabel users. Pada Gambar 5.21, ditunjukkan tampilan halaman login dari dashboard.



**Gambar 5. 21** Halaman Login Dashboard PPC Planning Assy

### 5.3.3. Implementasi Halaman Planning

Halaman planning dashboard memiliki beberapa bagian. Halaman ini memiliki tiga filter, yaitu line assy (model yang akan dibuatkan plan), shift (shift pembuatan plan), dan tanggal (tanggal pembuatan plan). Selain itu, terdapat input plan yang memiliki beberapa kolom, yaitu nomor, *part number*, *model name*, *lot size*, *planning*, dan *action*. Plan yang ditambahkan dapat dibatalkan dengan tombol batal atau dihapus seluruhnya dengan tombol *reset*. Daftar plan yang sudah dibuat dapat di-*submit* dan tersimpan di dalam basis data pada tabel *delv\_plan*. Pada Gambar 5.22, ditunjukkan tampilan halaman planning dari dashboard.

**KYB**  
Our Passion. Your Advantage

DASHBOARD PPC - PLANNING ASSY 17:57:37

PLANNING ASSY

LINE ASSY: RC-FAB SHIFT: Shift 3 DATE: Wednesday, 25/05/202

ID	PART NUMBER	MODEL NAME	LOT SIZE	PLANNING	ACTION
1	13300-12039-GP	2DP-GP	20	100	READY <input type="button" value="✖"/>
2	13300-11858-	84V-00	20	100	READY <input type="button" value="✖"/>
3	13300-11960 E	84V-10 EXP	20	100	READY <input type="button" value="✖"/>
4	13300-11858-	84V-00	20	100	READY <input type="button" value="✖"/>
5	13300-11858-GP	84V-00 GP	20	100	READY <input type="button" value="✖"/>

Input Part Number Auto Fill Auto Fill Input Planning CONFIRM

**Gambar 5. 22** Halaman Planning Dashboard PPC Planning Assy

### 5.3.4. Implementasi Halaman Historical

Halaman historical dashboard memiliki beberapa bagian. Halaman ini memiliki tiga filter yang sama dengan halaman planning untuk menyaring daftar plan yang telah dibuat sebelumnya. Tabel daftar plan memiliki beberapa kolom, seperti nomor, shift, *part number*, *model name*, *lot size*, *actual/plan*, *status*, dan *action*. Terdapat lima macam status yang dimiliki oleh sebuah plan sesuai dengan keadaan *actual* terhadap *plan*, yaitu *unrunning*, *running*, *completed*, *uncompleted*, dan *cancelled*. Plan dengan status *unrunning* dapat dibatalkan dengan menekan tombol batal pada kolom *action*. Halaman historical juga akan diperbarui secara otomatis jika terdapat data plan baru atau terdapat perubahan data. Pada Gambar 5.23, ditunjukkan tampilan halaman historical dari dashboard.

HISTORICAL PLAN ASSY

DATE: 
 LINE ASSY: 
 SHIFT:

**RC-FA3**

ID	SHIFT	PART NUMBER	MODEL NAME	LOT SIZE	ACTUAL   PLAN	STATUS	ACTION
1	3	13300-12029-GP	2DP-GP	20	0/100	Unloading	
2	3	13300-11858	84V-00	20	30/100	Running	
3	3	13300-11860-E	84V-10 EXP	20	100/100	Completed	
4	3	13300-11858	84V-00	20	90/100	Uncompleted	
5	3	13300-11858-GP	84V-00 GP	20	0/100	Cancelled	

**Gambar 5.23** Halaman Historical Dashboard PPC Planning Assy

*[Halaman ini sengaja dikosongkan]*

## **BAB VI**

### **PENGUJIAN DAN EVALUASI**

Bab ini menjelaskan tahap uji coba terhadap seluruh proyek yang telah dikerjakan. Pengujian dilakukan untuk mengukur seberapa sesuai hasil implementasi dengan kebutuhan dan rancangan yang telah ditetapkan.

#### **6.1. Tujuan Pengujian**

Tujuan pengujian terhadap proyek-proyek yang telah dikerjakan adalah untuk mengetahui kesesuaian hasil implementasi terhadap kebutuhan dan rancangan yang telah dirumuskan untuk setiap proyek tersebut.

#### **6.2. Kriteria Pengujian**

Pengukuran keberhasilan tujuan pengujian diperoleh dengan mengacu pada kriteria-kriteria berikut:

- a. Kesesuaian implementasi setiap sistem dengan kebutuhan dan perancangan yang telah didefinisikan
- b. Kemampuan setiap sistem untuk dijalankan pada server PT Kayaba Indonesia
- c. Kemampuan setiap sistem untuk diakses oleh pengguna yang bersangkutan

#### **6.3. Skenario Pengujian**

Skenario pengujian dilakukan dengan melihat cara pengguna menggunakan setiap sistem yang telah diimplementasikan. Langkah-langkah yang dilakukan adalah sebagai berikut:

1. Setiap sistem dijalankan pada server PT Kayaba Indonesia

2. Pengguna melakukan setiap kebutuhan yang telah didefinisikan untuk setiap sistem
3. Mengevaluasi umpan balik dari pengguna

#### 6.4. Evaluasi Pengujian

Hasil pengujian setiap proyek diberikan pada Tabel 6.1 hingga Tabel 6.3.

**Tabel 6. 1** Hasil Evaluasi Pengujian Dashboard APNR

Kriteria Pengujian	Hasil Pengujian
Admin dapat melihat video dari kamera pengawas secara <i>real-time</i>	Video yang ditampilkan memiliki FPS rendah karena kurangnya sumber daya di sisi <i>hardware</i>
Admin dapat melihat data hasil tangkapan dari kendaraan yang melintas	Terpenuhi
Admin dapat melihat riwayat data hasil tangkapan dari kendaraan yang melintas	Terpenuhi
Sistem dapat digunakan pada komputer dengan spesifikasi menengah ke bawah	Tidak terpenuhi, memerlukan komputer dengan spesifikasi menengah ke atas
Sistem memiliki antarmuka yang mudah dipahami	Terpenuhi
Sistem mendukung semua jenis browser	Terpenuhi

**Tabel 6. 2** Hasil Evaluasi Pengujian Dashboard Damping Force

Kriteria Pengujian	Hasil Pengujian
Admin dapat memilih <i>production line shock absorber</i> yang ingin dilihat datanya	Terpenuhi
Admin dapat melihat visualisasi data dalam bentuk grafik dan tabel	Terpenuhi
Admin dapat melakukan penyaringan data berdasarkan model, tanggal, dan shift	Terpenuhi
Admin dapat mencetak laporan dengan penyaringan data berdasarkan model, tanggal, dan shift	Terpenuhi
Sistem dapat digunakan pada komputer dengan spesifikasi menengah ke bawah	Terpenuhi
Sistem memiliki antarmuka yang mudah dipahami	Terpenuhi
Sistem mendukung semua jenis browser	Terpenuhi

**Tabel 6. 3** Hasil Evaluasi Pengujian Dashboard PPC Planning Assy

Kriteria Pengujian	Hasil Pengujian
Admin dapat membuat daftar plan produksi berdasarkan <i>assembly line</i> , shift, dan tanggal yang dipilih	Terpenuhi
Admin dapat melihat riwayat daftar plan yang telah dibuat	Terpenuhi
Admin dapat melakukan penyaringan data riwayat daftar plan berdasarkan tanggal, <i>assembly line</i> , dan shift	Terpenuhi
Admin dapat membatalkan plan dengan status “Unrunning” pada halaman riwayat	Terpenuhi
Sistem dapat digunakan pada komputer dengan spesifikasi menengah ke bawah	Terpenuhi
Sistem memiliki antarmuka yang mudah dipahami	Terpenuhi
Sistem mendukung semua jenis browser	Terpenuhi
Sistem dapat melakukan <i>refresh</i> halaman riwayat secara otomatis jika terdapat data yang diperbarui	Terpenuhi

*[Halaman ini sengaja dikosongkan]*

## **BAB VII**

### **KESIMPULAN DAN SARAN**

#### **7.1. Kesimpulan**

Kesimpulan yang diperoleh setelah melaksanakan magang/kerja praktik di *Section* Industry 4.0 Departemen MIS PT Kayaba Indonesia adalah sebagai berikut :

- a. Dashboard yang telah dikembangkan telah sesuai dengan yang dibutuhkan.
- b. Dashboard Automatic Plate Number Recognition dapat digunakan untuk mencatat plat nomor kendaraan bermotor yang keluar dan masuk PT Kayaba Indonesia. Akan tetapi, diperlukan sebuah komputer dengan spesifikasi tinggi agar aplikasi dapat berjalan dengan baik.
- c. Dashboard Damping Force telah diperbaiki sehingga aplikasi dapat memvisualisasikan data di seluruh *line shock absorber*.
- d. Dashboard PPC Planning Assy dapat berfungsi dengan baik sehingga dapat digunakan untuk membuat plan *production*.

#### **7.2. Saran**

Saran setelah melaksanakan magang/kerja praktik di *Section* Industry 4.0 Departemen MIS PT Kayaba Indonesia adalah sebagai berikut:

- a. Secara kompetensi atau penguasaan Project baik secara Flow Process ataupun Teknis sudah mumpuni. Akan tetapi, diperlukan peningkatan terkait kemampuan berkomunikasi (Pro Aktif). Sehingga komunikasi dua arah dapat berjalan secara efektif.

*[Halaman ini sengaja dikosongkan]*

## DAFTAR PUSTAKA

- Aufa Atila. (2021, Februari 17). *Mengenal Tentang Pengertian Product Planning Control*.  
<https://www.jojonomic.com/blog/product-planning-control/>
- Coursera Staff. (2023, November 30). *What Is Python Used For? A Beginner's Guide*.  
<https://www.coursera.org/in/articles/what-is-python-used-for-a-beginners-guide-to-using-python>
- Great Learning Team. (2022, Agustus 22). *Real-Time Object Detection Using TensorFlow*.  
<https://www.mygreatlearning.com/blog/object-detection-using-tensorflow/#:~:text=The%20TensorFlow%20Object%20Detection%20API,referred%20to%20as%20Model%20Zoo.>
- Hamdi, S. El. (2019). *Industry 4.0 : Fundamentals and Main Challenges*.
- Margaret Rouse. (2020, Agustus 31). *Web Development*.  
<https://www.techopedia.com/definition/23889/web-development>
- Matthew Deery. (2023, Agustus 30). *What Is Flask and How Do Developers Use It? A Quick Guide*.  
<https://careerfoundry.com/en/blog/web-development/what-is-flask/>
- Nico Klingler. (2024). *Automatic Number Plate Recognition (ANPR) – 2024 Guide*. <https://viso.ai/computer-vision/automatic-number-plate-recognition-anpr/>
- Oracle. (t.t.). *What Is a Database?* Diambil 30 Januari 2024, dari <https://www.oracle.com/id/database/what-is-database/>
- Tejpal Kumawat. (2023, Maret 5). *TrOCR — Transformer-based Optical Recognition Model*.

<https://medium.com/@tejpal.abhyuday/troc-r-transformer-based-optical-recognition-model-811f7b3217da>  
Unacademy. (t.t.). *Explanation of the Damping Forces*. Diambil 3 Februari 2024, dari <https://unacademy.com/content/jee/study-material/physics/damping-forces/#:~:text=Conclusion-,Damping%20forces%20are%20a%20special%20type%20of%20force%20that%20are,mec hanical%20nature%20or%20electrical%20nature.>  
Vidushi Meel. (t.t.). *What is OpenVINO? – The Ultimate Overview in 2024*. Diambil 30 Januari 2024, dari <https://viso.ai/computer-vision/intel-openvino-toolkit-overview/>  
Yash Agarwal. (2023, Juli 17). *Advantages of PHP over Other Programming Languages*. <https://www.scaler.com/topics/php-tutorial/advantages-of-php/>

*[Halaman ini sengaja dikosongkan]*

## **BIODATA PENULIS I**

Nama : Azzura Mahendra Putra Malinus  
Tempat, Tanggal Lahir : Surabaya, 19 Februari 2023  
Jenis Kelamin : Laki-laki  
Telepon : +6281231199416  
Email : 5025201211@student.its.ac.id

### **AKADEMIS**

Kuliah : Departemen Teknik Informatika –  
FTEIC, ITS  
Angkatan : 2020  
Semester : 7 (Tujuh)