



## Deteksi Hunian Di Tempat Parkir (Occupancy Detection In Parking Lot)

Echa Oktamiani Maulana

Universitas Islam 45 Bekasi

Jl. Cut Mutia No.83, RT.004/RW.009, Margahayu, Kec. Bekasi

Timur., Kota Bks, Jawa Barat 17113

Korespondensi: [echaoktamiani10@gmail.com](mailto:echaoktamiani10@gmail.com)

**Abstract.** MSIB (Certified Independent Study and Internship) is one of the activity programs at the Merdeka Campus which aims to help students improve their skills and develop themselves. MSIB appointed Orbit Future Academy as one of the partners in the Independent Study program. Founded in 2016 with the aim of improving the quality of life through innovation, education and skills training. In accordance with its mission, namely "We curate and localize international programs and courses for upskilling, re-skilling youth, and the workforce towards jobs of the future". Partners provide opportunities for students to take Artificial Intelligence programs and study online. Learning consists of eight material courses including Python Programming, AI Technology Logic and Concepts, AI Project Cycle, AI Research Methods, ChatGPT, Professional and Company Ethics, Financial Literacy and ending with a Final Project. The final project scope carried out is the Occupancy Detection in Parking Lot project. This project uses the Computer Vision domain with the selection of the YOLO model in detecting objects and pixel segmentation. The project begins with selecting a dataset using roboflow which then goes through data pre-processing for cloning, annotation and augmentation. Then the model is trained using machine learning and deep learning algorithms to understand patterns and characteristics related to parking spaces. Once trained, the AI model will be validated using test data. This aims to ensure that the model truly recognizes the presence of the vehicle. Next, form the application design in creating an informative interface using wireframes. Then enter the deployment stage so that the system can be accessed widely and easily via the web. Lastly, field testing is to find out the performance of the application that has been designed.

**Keywords:** Orbit Future Academy, Occupancy Detection in Parking Lot, Artificial Intelligence, Machine Learning, Deep Learning, YOLO

**Abstrak.** MSIB (Magang dan Studi Independen Bersertifikat) menjadi salah satu program kegiatan di Kampus Merdeka yang bertujuan membantu mahasiswa meningkatkan keterampilan dan mengembangkan diri. MSIB mengangkat Orbit Future Academy sebagai salah satu mitra dalam program Studi Independen. Didirikan pada tahun 2016 dengan tujuan untuk meningkatkan kualitas hidup melalui inovasi, edukasi, dan pelatihan keterampilan. Sesuai dengan misi-nya yaitu "We curate and localize international programs and courses for upskilling, re-skilling youth, and the workforce towards jobs of the future". Mitra memberikan kesempatan kepada mahasiswa untuk mengambil program Artificial Intelligence dan belajar secara daring. Pembelajaran terdiri dari delapan course materi diantaranya Pemrograman Python, Logika dan Konsep Teknologi AI, Siklus Proyek AI, Metode Penelitian AI, ChatGPT, Etika Profesi dan Perusahaan, Financial Literacy dan diakhiri Proyek Akhir. Lingkup proyek akhir yang dilakukan ialah proyek Occupancy Detection in Parking Lot. Pada proyek ini menggunakan domain Computer Vision dengan pemilihan model YOLO dalam mendeteksi objek dan segmentasi piksel. Proyek diawali dengan pemilihan dataset menggunakan roboflow yang selanjutnya melewati pre-processing data untuk di cloning, anotasi dan augmentasi. Kemudian model dilatih menggunakan algoritma machine learning dan deep learning untuk memahami pola dan ciri-ciri yang berkaitan dengan ruang parkir. Setelah dilatih, model AI akan divalidasi menggunakan data pengujian. Hal ini bertujuan untuk memastikan bahwa model benar-benar mengenali keberadaan kendaraan. Selanjutnya bentuk desain aplikasi dalam menciptakan antarmuka yang informatif dengan menggunakan wireframe. Lalu memasuki tahap deployment agar sistem dapat diakses secara luas dan mudah melalui web. Terakhir yaitu pengujian lapangan untuk mengetahui sejauh mana kinerja aplikasi yang telah dirancang.

**Kata kunci :** Orbit Future Academy, Deteksi Hunian di Tempat Parkir, Kecerdasan Buatan, Pembelajaran Mesin, Pembelajaran Mendalam, YOLO.

## **PENDAHULUAN**

Perkembangan pesat terjadi dalam berbagai bidang, mulai dari ekonomi hingga teknologi dan transportasi. Menghindari ketertinggalan, dibutuhkan upaya dalam meningkatkan pembangunan yang berkelanjutan, diantaranya dengan meningkatkan sumber daya manusia melalui pendidikan dan pelatihan. Untuk mendukung hal ini, pemerintah Indonesia memperkenalkan program MBKM (Merdeka Belajar Kampus Mengajar) atau Kampus Merdeka yang merupakan kebijakan dari Kementerian Pendidikan, Kebudayaan, Riset dan Teknologi.

MSIB (Magang dan Studi Independen Bersertifikat) menjadi salah satu program kegiatan di Kampus Merdeka yang bertujuan membantu mahasiswa meningkatkan keterampilan dan mengembangkan diri, sehingga dapat membantu mereka dalam membangun karirnya di masa depan. Dengan adanya program MBKM maupun MSIB diharapkan dapat membantu serta meningkatkan kualitas sumber daya manusia di Indonesia dan ikut berkontribusi terhadap pembangunan berkelanjutan.

Fasilitas yang diperoleh mahasiswa dalam program dan kegiatan yang mereka jalani berasal dari mitra. MSIB mengangkat Orbit Future Academy sebagai salah satu mitra dalam program Studi Independen. Mitra memberikan kesempatan kepada mahasiswa untuk mengambil program Artificial Intelligence dan belajar secara daring. Pembelajaran terdiri dari delapan course materi diantaranya Pemrograman Python, Logika dan Konsep Teknologi AI, Siklus Proyek AI, Metode Penelitian AI, ChatGPT, Etika Profesi dan Perusahaan, Financial Literacy dan diakhiri Proyek Akhir.

Proyek akhir merupakan implementasi pembelajaran siswa setelah mengikuti program MSIB Orbit Future Academy. Proyek akhir ini dapat dijadikan sebagai bentuk hasil pembelajaran mahasiswa, Proyek akhir tersebut merupakan sebuah sistem object detection pada hunian tempat parkir.

## **METODE PENELITIAN**

Pada proyek ini menggunakan domain Computer Vision dengan pemilihan model YOLO dalam mendeteksi objek dan segmentasi piksel. Proyek diawali dengan pemilihan dataset menggunakan roboflow yang selanjutnya melewati pre-processing data untuk di cloning, anotasi dan augmentasi. Kemudian model dilatih menggunakan algoritma machine learning dan deep learning untuk memahami pola dan ciri-ciri yang berkaitan dengan ruang parkir. Setelah dilatih, model AI akan divalidasi menggunakan data pengujian. Hal ini bertujuan untuk memastikan bahwa model benar-benar mengenali keberadaan kendaraan.

Selanjutnya bentuk desain aplikasi dalam menciptakan antarmuka yang informatif dengan menggunakan wireframe. Lalu memasuki tahap deployment agar sistem dapat diakses secara luas dan mudah melalui web. Terakhir yaitu pengujian lapangan untuk mengetahui sejauh mana kinerja aplikasi yang telah dirancang.

## **HASIL DAN PEMBAHASAN**

### **Proses Pelaksanaan Proyek Akhir**

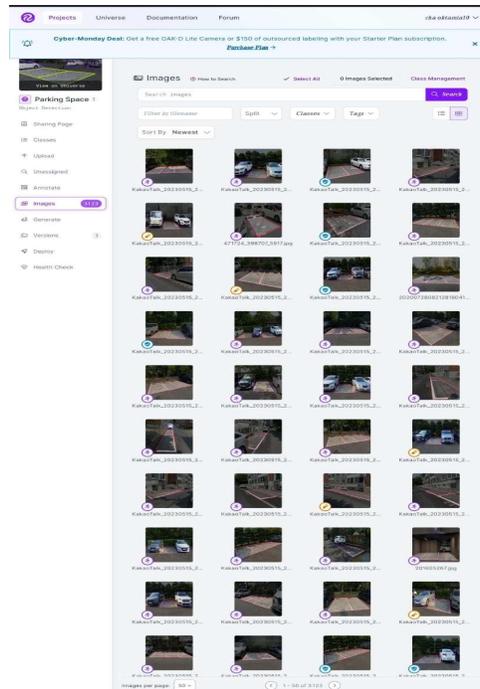
Pada bab ini, pengerjaan aplikasi Deteksi Hunian di Tempat Parkir, berikut adalah gambaran yang melibatkan penggunaan model AI dari awal hingga akhir. Proses ini menggabungkan teknologi machine learning dan computer vision untuk mencapai tujuan akhir, yaitu mengidentifikasi keberadaan kendaraan di area parkir.

#### **1. Pemilihan Dataset**

Dalam konteks pengembangan aplikasi Deteksi Hunian di Tempat Parkir, pemilihan dataset adalah tahap kunci dalam mencapai keberhasilan. Kami memutuskan untuk memanfaatkan layanan Roboflow untuk mengakses dataset yang beragam dan representatif. Roboflow memberikan akses ke koleksi dataset yang mencakup berbagai situasi di lokasi parkir, kondisi pencahayaan, dan posisi kamera yang berbeda. Pilihan dataset ini adalah langkah penting karena kualitas dataset mempengaruhi langsung kemampuan model deteksi objek kami untuk mengenali kendaraan di berbagai situasi. Dengan dataset yang diperoleh dari Roboflow, kami memastikan bahwa model kami dapat mengatasi tantangan nyata dalam deteksi hunian di tempat parkir dengan akurat.



**Gambar 1 Dataset Pada Roboflow**



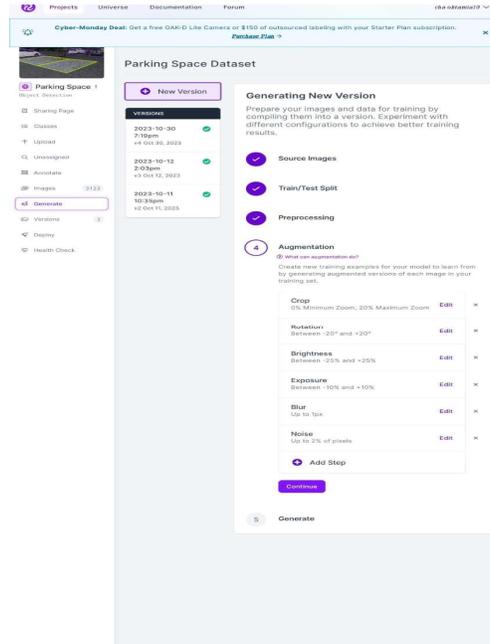
**Gambar 2 Jumlah Dataset Pada Roboflow**

Pada gambar tersebut, kami mengandalkan garis pada area parkir sebagai pembatas antar hunian parkir menggunakan metode Background Subtraction. Pendekatan ini memanfaatkan kumpulan data yang meliputi sudut pandang atas, samping, depan, dengan totalnya sebanyak 3.123. Garis-garis pembatas ini digunakan sebagai acuan utama untuk mendeteksi status hunian pada setiap tempat parkir. Background Subtraction memungkinkan kami untuk secara efisien mengidentifikasi perbedaan antara gambar aktual yang berisi kendaraan dan gambar latar belakang parkir yang seharusnya tanpa kendaraan. Dengan menggunakan garis pembatas ini, kami dapat dengan akurat memantau perubahan pada setiap slot parkir, sehingga memberikan informasi real-time tentang apakah tempat parkir tersebut sedang terisi atau kosong. Metode ini membantu meningkatkan akurasi deteksi hunian dan memberikan solusi yang andal untuk manajemen parkir yang lebih efisien.

## 2. Preprocessing Data

Proses pengembangan aplikasi deteksi hunian di tempat parkir dimulai dengan pencarian dataset yang sesuai di situs web RoboFlow. Dataset ini akan menjadi fondasi utama dalam melatih model AI. Setelah dataset terpilih, langkah berikutnya adalah melakukan cloning dataset ke dalam proyek yang telah dipilih. Proses berlanjut dengan melakukan anotasi tambahan pada dataset jika diperlukan, dengan tujuan meningkatkan akurasi deteksi objek. Selanjutnya, kami melakukan proses generasi gambar dengan berbagai kondisi dan augmentasi data untuk memperluas atau memperkaya kumpulan data gambar yang digunakan dalam

pelatihan model. Augmentasi melibatkan beberapa teknik seperti rotasi, pemangkasan, perubahan ukuran, pencerahan, dan penyesuaian warna untuk menciptakan variasi pada gambar yang ada. Roboflow menyediakan serangkaian alat untuk mengimplementasikan augmentasi ini secara otomatis, membantu dalam meningkatkan keberagaman data yang digunakan untuk melatih model pembelajaran mesin dataset.



**Gambar 3 Augmentasi Dataset Pada Roboflow**

### 3. Pelatihan Model

Pelatihan model Occupancy Detection in parking lot di Roboflow merupakan langkah penting dalam membangun sistem yang efisien dan cerdas untuk memantau penggunaan ruang parkir. Proses ini dimulai dengan pengumpulan dataset yang luas, terdiri dari berbagai gambar dan video yang merepresentasikan berbagai kondisi dan situasi di area parkir. Setelah dataset terkumpul, langkah selanjutnya adalah melakukan pra-pemrosesan data, seperti normalisasi gambar, penyesuaian ukuran, dan pemilihan fitur yang tepat untuk memastikan kualitas yang optimal sebelum masuk ke dalam tahap pelatihan.

Selama proses pelatihan, model diberikan informasi untuk memahami pola dan ciri-ciri yang berkaitan dengan ruang parkir yang kosong atau terisi. Algoritma machine learning dan deep learning digunakan untuk mengolah data, mengidentifikasi pola, dan memprediksi status parkir pada setiap lokasi dalam area yang diamati. Melalui iterasi dan penyesuaian berulang, model ini ditingkatkan performanya agar mampu memberikan prediksi yang akurat dalam berbagai kondisi pencahayaan, cuaca, dan tipe kendaraan yang parkir.

Penting untuk terus memvalidasi dan mengevaluasi model ini menggunakan dataset yang berbeda-beda serta menguji performanya di lingkungan nyata. Hal ini membantu untuk memastikan bahwa model mampu mengatasi variasi yang ada di lapangan dan dapat diandalkan dalam memberikan informasi yang akurat tentang ketersediaan tempat parkir. Selain itu, selama proses pelatihan, penerapan teknik-teknik seperti data augmentation dan fine-tuning menjadi kunci dalam meningkatkan kecerdasan serta ketepatan model dalam memahami situasi parkir.

Hasil dari pelatihan model Occupancy Detection ini akan memberikan manfaat yang signifikan dalam pengelolaan parkir secara efisien. Dengan kemampuan untuk secara akurat mendeteksi ketersediaan tempat parkir, pemanfaatan ruang parkir dapat dioptimalkan, mengurangi kemacetan lalu lintas, meningkatkan pengalaman pengguna, serta mendukung upaya untuk mengurangi emisi gas buang kendaraan karena penggunaan parkir yang lebih efisien.

#### 4. Validasi Model

Setelah pelatihan, model AI kami divalidasi menggunakan data pengujian. Kami mengukur akurasi, presisi, re call, dan F1-score untuk memastikan bahwa model dapat mengenali keberadaan kendaraan dengan tepat. Berikut adalah hasil dari dataset yang sudah di training



**Gambar 4 Hasil Training Dataset**

#### 5. Integrasi dengan Sistem Parkir

Proses integrasi model AI dengan sistem parkir adalah tahapan kunci dalam pengembangan aplikasi Deteksi Hunian di Tempat Parkir. Pada tahap ini, kamera-kamera yang telah dipasang di lokasi parkir berperan penting. Kamera-kamera ini terhubung ke model AI yang telah dilatih, dan secara berkala mengirimkan aliran gambar ke model. Model AI ini kemudian mengambil peran utama dalam melakukan deteksi objek secara real-time.

Model AI yang telah dilatih memiliki kemampuan untuk menganalisis setiap gambar yang diterimanya. Dengan menggunakan teknik computer vision, model ini mampu

mengidentifikasi keberadaan kendaraan pada setiap tempat parkir yang terpantau oleh kamera. Setiap tempat parkir akan melewati proses pengolahan gambar di model AI, yang kemudian mengeluarkan hasil berupa informasi apakah tempat parkir tersebut sudah terisi oleh kendaraan atau masih kosong.

Hasil deteksi objek yang dihasilkan oleh model AI ini diintegrasikan kembali ke dalam sistem parkir. Informasi ini dapat dengan mudah diakses oleh operator parkir atau pengguna yang mencari tempat parkir. Dengan bantuan model AI, sistem ini memberikan pembaruan status tempat parkir secara real-time, memungkinkan pengelola parkir untuk mengoptimalkan alokasi tempat parkir dan memastikan efisiensi parkir yang lebih baik.

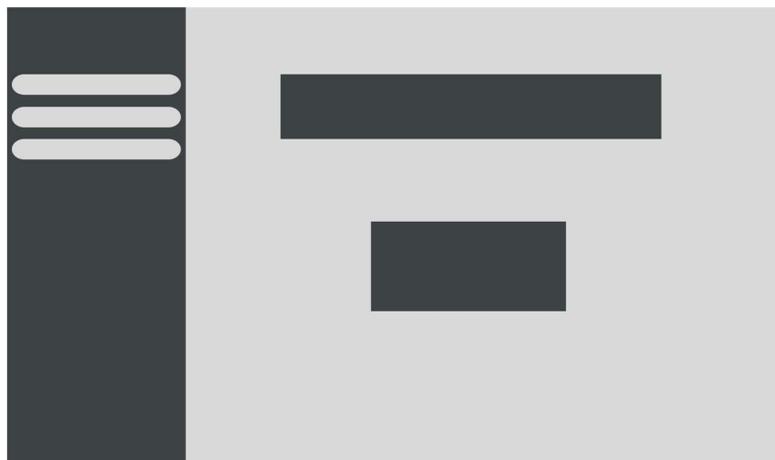
Dalam pengembangan aplikasi ini, integrasi model AI dengan sistem parkir merupakan titik fokus utama yang memungkinkan penerapan teknologi deteksi hunian di tempat parkir yang akurat dan efisien. Dengan kemampuan model AI untuk bekerja secara real-time, informasi yang diberikan kepada pengguna dan pengelola parkir menjadi lebih akurat dan dapat membantu mengurangi waktu mencari tempat parkir, meningkatkan pengalaman pengguna, dan mendukung pengelolaan parkir yang lebih efektif.

#### 6. Pengembangan Kerangka Dengan Wireframe

Dalam fase ini, kami juga membuat suatu kerangka kerja Wireframe yang memberikan gambaran awal tentang desain aplikasi. Kami fokus pada menciptakan antarmuka yang informatif dan mudah dipahami bagi pengguna. Di bawah ini adalah hasil akhir dari Wireframe yang telah kami buat.

##### a. Home

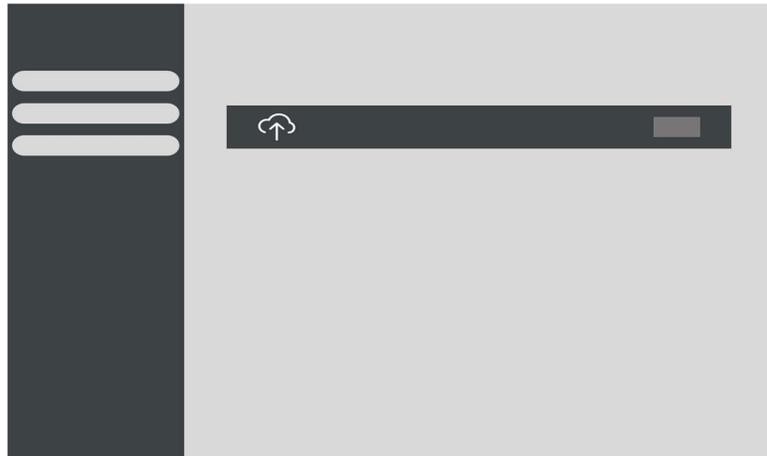
Pada halaman ini, menampilkan logo Orbit Future Academy dan penjelasan secara singkat mengenai aplikasi Park sense



**Gambar 5 Halaman Home**

b. Detection

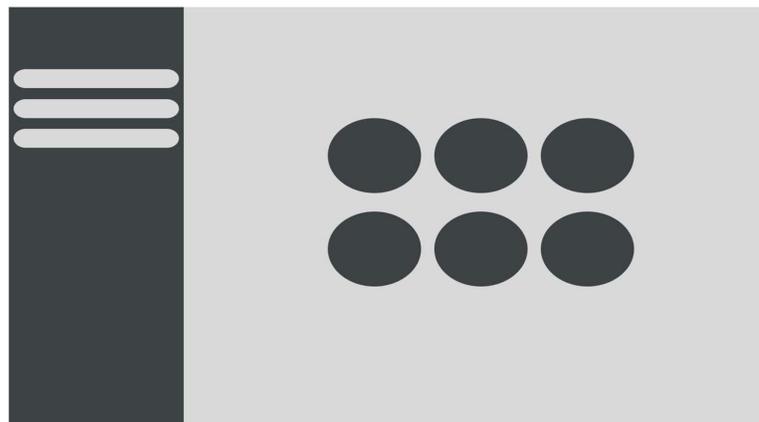
Pada halaman ini, menampilkan tombol browser file untuk mengupload gambar yang akan dideteksi oleh aplikasi Park sense



**Gambar 6 Halaman Detection**

c. About Team

Pada halaman ini, menampilkan profil anggota kelompok



**Gambar 7 Halaman About Team**

Dalam mendeteksi hunian di tempat parkir menggunakan Roboflow, beberapa hambatan yang mungkin dihadapi perlu diperhatikan adalah sebagai berikut :

- a. Perbedaan pencahayaan di area parkir dapat mempengaruhi kualitas gambar yang diambil oleh kamera. Hal ini dapat menyulitkan proses deteksi hunian karena kontras antara mobil dan latar belakang dapat berubah-ubah.
- b. Posisi mobil yang berbeda di dalam area parkir dapat menjadi tantangan, sebab beberapa mobil mungkin terhalang oleh kendaraan lain atau memiliki sudut pandang yang tidak optimal untuk deteksi.

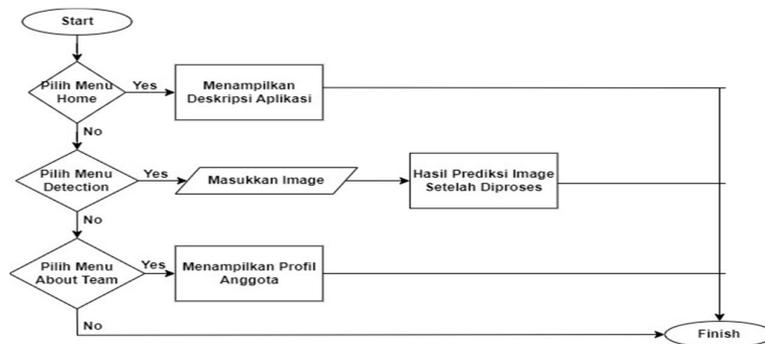
- c. Kualitas data pelatihan adalah faktor kunci; jika dataset pelatihan tidak mencakup berbagai situasi dan kondisi yang mungkin terjadi di tempat parkir, model deteksi mungkin tidak akurat.
- d. Perubahan tata letak parkir yang terjadi secara periodik dapat menyulitkan model deteksi yang sudah dilatih.

Untuk mengatasi hambatan-hambatan tersebut, ada beberapa langkah yang dapat diambil yaitu :

- a. Melakukan pembaruan dan pelatihan ulang model secara berkala untuk mengakomodasi perubahan dalam tata letak parkir atau kondisi lingkungan. Data baru harus ditambahkan ke dalam dataset pelatihan sehingga model dapat belajar dan mengenali pola baru.
- b. Memilih kamera yang memiliki kualitas gambar yang baik dan tahan terhadap perubahan cuaca atau penutupan. Kamera dengan fitur penyesuaian otomatis pencahayaan atau penghapusan bayangan dapat membantu meningkatkan akurasi deteksi.
- c. Penggunaan teknik pengolahan citra khusus, seperti filter atau teknik restorasi gambar, dapat digunakan untuk mengatasi gambar yang buram atau kabur akibat cuaca buruk, sehingga kualitas gambar dapat ditingkatkan. Dengan mengambil langkah-langkah ini, aplikasi deteksi hunian di tempat parkir menggunakan Roboflow dapat menjadi lebih tangguh dan andal dalam berbagai kondisi.

## 7. Flowchart Aplikasi

Flowchart berikut ini menunjukkan aliran kerja umum dari proses aplikasi deteksi hunian di tempat parkir:



**Gambar 8 Flowchart Aplikasi**

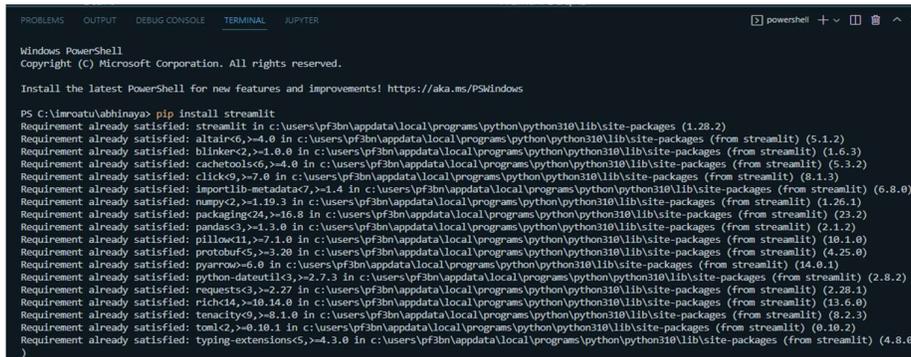
## 8. Deployment Streamlit

Deployment Streamlit pada aplikasi Park sense merupakan tahap krusial dalam menghadirkan solusi yang dapat diakses secara luas. Proses deployment ini memungkinkan aplikasi yang telah dibangun menggunakan Streamlit untuk dijalankan secara online,

memungkinkan pengguna dari berbagai lokasi untuk mengaksesnya dengan mudah melalui web browser.

Untuk melakukan deployment Streamlit pada aplikasi Park sense, terdapat beberapa package yang perlu diinstall untuk memastikan aplikasi dapat berjalan dengan lancar. Berikut adalah daftar package yang perlu diinstall agar aplikasi dapat berjalan :

a. pip install streamlit



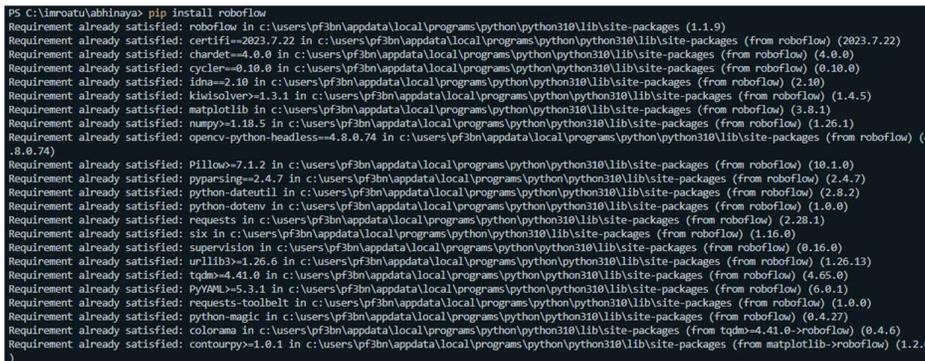
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Inroatu\labhinaya> pip install streamlit
Requirement already satisfied: streamlit in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (1.28.2)
Requirement already satisfied: altair<6,>=4.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (5.1.2)
Requirement already satisfied: blinker<2,>=1.0.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (1.6.3)
Requirement already satisfied: cachetools<6,>=4.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (5.3.2)
Requirement already satisfied: click<9,>=7.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (8.1.3)
Requirement already satisfied: importlib-metadata<7,>=1.4 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (6.8.0)
Requirement already satisfied: numpy<2,>=1.19.3 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (1.26.1)
Requirement already satisfied: packaging<24,>=16.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (22.2)
Requirement already satisfied: pandas<3,>=1.3.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (2.1.2)
Requirement already satisfied: pillow<11,>=7.1.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (10.1.0)
Requirement already satisfied: protobuf<5,>=3.20 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (4.25.0)
Requirement already satisfied: pyarrow<=6.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (14.0.1)
Requirement already satisfied: python-dateutil<3,>=2.7.3 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (2.8.2)
Requirement already satisfied: requests<3,>=2.27 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (2.28.1)
Requirement already satisfied: rich<14,>=10.14.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (13.6.0)
Requirement already satisfied: tenacity<9,>=8.1.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (8.2.3)
Requirement already satisfied: tomli<2,>=0.10.1 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (0.10.2)
Requirement already satisfied: typing-extensions<5,>=4.3.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from streamlit) (4.8.0)
```

Gambar 9 Install Library Streamlit

b. pip install roboflow



```
PS C:\Inroatu\labhinaya> pip install roboflow
Requirement already satisfied: roboflow in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (1.1.9)
Requirement already satisfied: certifi==2023.7.22 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (2023.7.22)
Requirement already satisfied: chardet==4.0.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (4.0.0)
Requirement already satisfied: cyclert==0.10.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (0.10.0)
Requirement already satisfied: idna==2.10 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (2.10)
Requirement already satisfied: kiwisolver==1.3.1 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (1.4.5)
Requirement already satisfied: matplotlib in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (3.8.1)
Requirement already satisfied: numpy>=1.19.5 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (1.26.1)
Requirement already satisfied: opencv-python-headless==4.8.0.74 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (4.8.0.74)
Requirement already satisfied: Pillow==7.1.2 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (10.1.0)
Requirement already satisfied: pyparsing==2.4.7 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (2.4.7)
Requirement already satisfied: python-dateutil in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (2.8.2)
Requirement already satisfied: python-dotenv in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (1.0.0)
Requirement already satisfied: requests in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (2.28.1)
Requirement already satisfied: six in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (1.16.0)
Requirement already satisfied: supervision in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (0.16.0)
Requirement already satisfied: urllib3==1.26.6 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (1.26.13)
Requirement already satisfied: tqdm==4.41.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (4.65.0)
Requirement already satisfied: PyYAML>=5.3.1 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (6.0.1)
Requirement already satisfied: requests-toolbelt in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (1.0.0)
Requirement already satisfied: python-magic in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (0.4.27)
Requirement already satisfied: colorama in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from tqdm==4.41.0->roboflow) (0.4.6)
Requirement already satisfied: contourpy==1.0.1 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from matplotlib->roboflow) (1.2.0)
```

Gambar 10 Install Library Roboflow

c. pip install roboflow streamlit Pillow



```
PS C:\Inroatu\labhinaya> pip install roboflow streamlit Pillow
Requirement already satisfied: roboflow in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (1.1.9)
Requirement already satisfied: streamlit in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (1.28.2)
Requirement already satisfied: Pillow in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (10.1.0)
Requirement already satisfied: certifi==2023.7.22 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (2023.7.22)
Requirement already satisfied: chardet==4.0.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (4.0.0)
Requirement already satisfied: cyclert==0.10.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (0.10.0)
Requirement already satisfied: idna==2.10 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (2.10)
Requirement already satisfied: kiwisolver==1.3.1 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (1.4.5)
Requirement already satisfied: matplotlib in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (3.8.1)
Requirement already satisfied: numpy>=1.18.5 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (1.26.1)
Requirement already satisfied: opencv-python-headless==4.8.0.74 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (4.8.0.74)
Requirement already satisfied: pyparsing==2.4.7 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (2.4.7)
Requirement already satisfied: python-dateutil in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (2.8.2)
Requirement already satisfied: python-dotenv in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (1.0.0)
Requirement already satisfied: requests in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (2.28.1)
Requirement already satisfied: six in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (1.16.0)
Requirement already satisfied: supervision in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (0.16.0)
Requirement already satisfied: urllib3==1.26.6 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (1.26.13)
Requirement already satisfied: tqdm==4.41.0 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (4.65.0)
Requirement already satisfied: PyYAML>=5.3.1 in c:\users\pf3bn\appdata\local\programs\python\python310\lib\site-packages (from roboflow) (6.0.1)
```

Gambar 11 Install Library Pillow

Setelah menyelesaikan semua konfigurasi yang diperlukan, aplikasi data dijalankan dengan menggunakan perintah "streamlit run <namafile.py>" pada terminal atau command

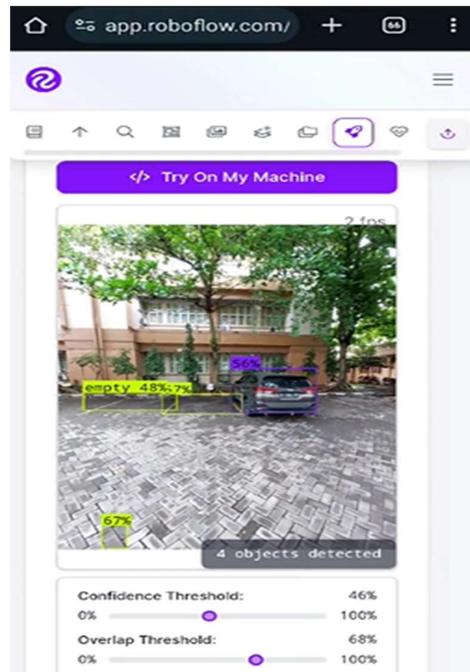
prompt, dan aplikasi akan dijalankan secara otomatis. Proses ini memungkinkan untuk segera melihat UI aplikasi Park sense yang telah dibuat menggunakan Streamlit.

```
PS C:\imroatu\abhinaya> streamlit run .\1_@Home.py  
  
You can now view your Streamlit app in your browser.  
  
Local URL: http://localhost:8501  
Network URL: http://192.168.1.15:8501
```

**Gambar 12 Perintah Untuk Menjalankan Aplikasi**

#### 9. Pengujian di Lapangan

Aplikasi yang telah selesai dikembangkan diuji di lapangan dengan menggunakan kamera smartphone secara real-time. Hasilnya dievaluasi untuk memastikan kinerja aplikasi sesuai dengan harapan.



**Gambar 13 Pengujian Secara Realtime**

#### Hasil Proyek Akhir

- *Uraian/deskripsi aplikasi atau model AI yang telah dikembangkan*

Membangun aplikasi Park sense dengan menggunakan platform Streamlit dapat menjadi langkah yang menarik. Streamlit adalah framework yang memudahkan pembuatan antarmuka pengguna (UI) yang responsif dan interaktif. Dengan menggunakan alat ini, proses memvisualisasikan dan menganalisis data untuk memantau status kapasitas parkir dapat dilakukan dengan lebih efisien.

Pertama-tama, dalam pengembangan aplikasi, langkah awalnya adalah memastikan integrasi yang solid antara Streamlit dengan algoritma atau model yang digunakan untuk mendeteksi occupancy dalam tempat parkir. Proses ini melibatkan penggunaan teknik computer vision, machine learning, atau sensor yang mampu mengidentifikasi mobil yang masuk atau keluar dari area parkir.

Setelah itu, Streamlit memungkinkan untuk membuat dashboard interaktif yang menampilkan informasi aktual tentang jumlah slot parkir yang kosong, slot yang terisi, serta estimasi waktu tunggu atau prediksi ketersediaan parkir di masa depan. Dengan integrasi yang tepat antara algoritma deteksi, tampilan visual yang responsif, serta fokus pada keamanan dan privasi, aplikasi Park sense menggunakan Streamlit memiliki potensi besar untuk memberikan solusi yang efektif dan efisien dalam mengelola tempat parkir.

● *Uraian/deskripsi aplikasi atau model AI yang telah dikembangkan*

a. Home

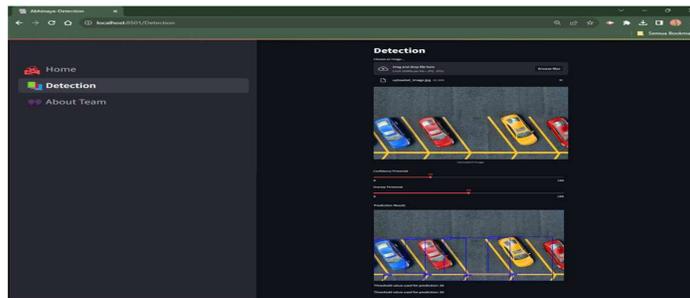
Pada halaman ini, menampilkan logo Orbit Future Academy dan penjelasan secara singkat mengenai aplikasi Park sense



Gambar 14 Tampilan Home

b. Detection

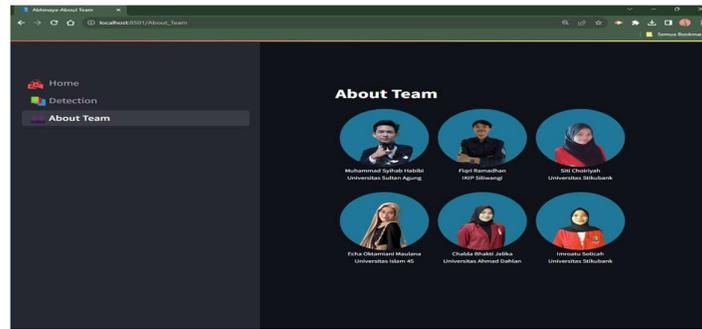
Pada halaman ini, menampilkan tombol browser file untuk mengupload gambar yang akan dideteksi oleh aplikasi Park sense



Gambar 15 Tampilan Detection

c. About Team

Pada halaman ini, menampilkan profil anggota kelompok



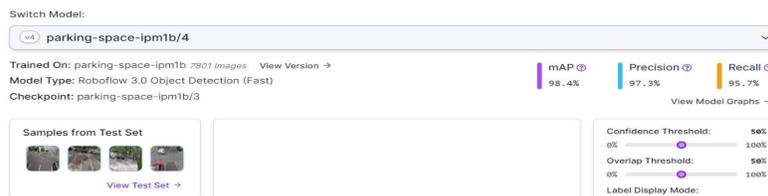
Gambar 16 Tampilan About Team

- Performa aplikasi atau model AI. Bisa didukung dengan tabel, gambar, atau grafik (hasil pengujian aplikasi atau model AI)

a. Nilai akurasi



Gambar 17 Grafik Training dataset



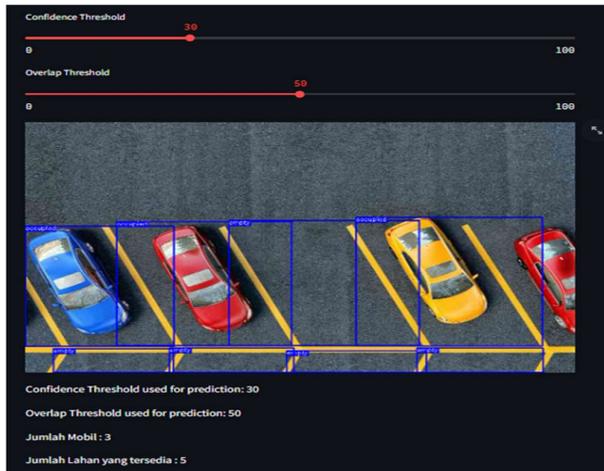
Gambar 18 Hasil Training Dataset

Gambar diatas menjelaskan tentang kesimpulan dari hasil training AI yang menunjukkan mAP sebesar 98.4%, Precision sebesar 97.3%, dan Recall sebesar 95.7%.

Keterangan

- mAP : mAP digunakan untuk mengukur rata-rata akurasi dari setiap class dalam model.
- Precision : Mengukur seberapa akurat model dalam proses training
- Recall : Seberapa akurat model dapat mendeteksi class dalam gambar.

b. Hasil deteksi



**Gambar 19 Hasil Deteksi Pada Aplikasi**

Slider confidence berguna untuk menentukan nilai minimum confidence dari AI. Sebelumnya, AI melakukan prediksi gambar yang menghasilkan bounding box dengan tingkat confidence yang beragam. Dengan confidence threshold, bisa menentukan nilai confidence yang berada diatas nilai confidence yang ditentukan pada slider.

Overlap pada Roboflow umumnya diukur sebagai nilai antara 0 hingga 1, di mana 0 berarti tidak ada tumpang tindih antara bounding box, sedangkan 1 berarti tumpang tindih penuh. Beberapa detektor objek menggunakan istilah "IoU" (Intersection over Union) untuk mengukur sejauh mana dua bounding box tumpang tindih. IoU dihitung sebagai rasio area tumpang tindih antara dua bounding box dibagi dengan total area yang dicakup oleh kedua bounding box tersebut

- *Kelebihan dan kelemahan aplikasi atau model AI yang telah dikembangkan*

**Tabel 1 Kelebihan dan kelemahan Aplikasi**

Kelebihan		Kelemahan	
Kemudahan Penggunaan	Aplikasi dirancang dengan fokus pada kesederhanaan, memungkinkan pengguna untuk dengan cepat membuat antarmuka pengguna yang interaktif tanpa harus memiliki pengetahuan yang mendalam tentang pengembangan web.	Ketergantungan pada Kualitas Data Input	Keakuratan deteksi occupancy parkir sangat bergantung pada kualitas data masukan. Jika terjadi gangguan atau kesalahan dalam sensor atau sumber data, hal ini dapat menyebabkan informasi yang tidak akurat tentang

			kapasitas parkir.
Responsif dan Interaktif	UI aplikasi yang dihasilkan menggunakan Streamlit sangat responsif dan memungkinkan interaksi yang lancar. Pengguna dapat dengan mudah memantau data occupancy parkir secara real-time dan berinteraksi dengan berbagai komponen aplikasi.	Kesulitan dalam Skalabilitas	Ketika aplikasi diperluas untuk menangani lebih banyak area parkir atau perangkat tambahan, mungkin sulit untuk mempertahankan kinerja yang optimal tanpa melakukan penyesuaian infrastruktur yang lebih besar.
Fleksibilitas	Aplikasi memungkinkan integrasi yang mudah dengan berbagai jenis model deteksi occupancy, baik itu berbasis computer vision, machine learning, atau teknologi sensor lainnya. Hal ini memberikan fleksibilitas dalam memilih pendekatan yang sesuai untuk deteksi parkir.	Kesulitan Pemeliharaan dan Dukungan	Dalam jangka panjang, aplikasi ini memerlukan pemeliharaan terus-menerus untuk memastikan bahwa algoritma deteksi tetap relevan dan optimal. Dukungan teknis dan pembaruan secara berkala juga menjadi krusial.
Kinerja yang Baik	Aplikasi yang dihasilkan memiliki kinerja yang baik. Hal ini memungkinkan pengguna untuk menjalankan aplikasi dengan cepat dan efisien.	Keterbatasan Keamanan dan Privasi Data	Aplikasi ini harus memperhatikan keamanan data, terutama jika melibatkan informasi sensitif tentang aktivitas parkir. Perlindungan data dan privasi pengguna menjadi prioritas yang harus diatasi.

- *Rencana pengembangan aplikasi atau model AI dikemudian hari (pengembangan lebih lanjut)*

Untuk meningkatkan kualitas aplikasi, rencana pengembangan selanjutnya bisa berfokus pada integrasi fitur-fitur baru yang akan memberikan pengguna pilihan lebih luas. Salah satunya adalah dengan menambahkan opsi deteksi berbasis waktu untuk memasukkan video, yang akan memberikan tiga pilihan deteksi: gambar, video yang telah di-input, dan deteksi real-time. Dengan menghadirkan opsi tersebut, pengguna dapat lebih leluasa dalam memilih metode deteksi yang sesuai dengan kebutuhan mereka, meningkatkan fleksibilitas dan kegunaan aplikasi secara keseluruhan.

## **KESIMPULAN DAN SARAN**

### **Kesimpulan**

Program Studi Independen Bersertifikat (MSIB) merupakan program yang diselenggarakan oleh Menteri Pendidikan dan Kebudayaan di Indonesia. Tujuan dari program ini adalah memberikan kesempatan kepada mahasiswa untuk mendapatkan pengalaman dunia kerja melalui proyek nyata yang dibimbing oleh tim dan pembimbing dari organisasi, startup, mitra, dan pemerintahan.

Salah satu mitra yang bekerja sama dengan program MSIB adalah Orbit future academy Artificial Intelligence 4 Jobs berpusat pada kompetensi untuk membangun AI Readiness dengan meningkatkan keterampilan individu dalam bidang AI yang dibutuhkan sebagai persiapan untuk memasuki dunia pekerjaan masa depan. Program ini juga memiliki cakupan dalam pengembangan mindset dan skill set untuk mencapai keberhasilan dalam perkembangan teknologi dunia yang sangat dinamis menuju pematangan dan ketergantungan di bidang AI.

Proyek capstone finally yang kami buat adalah Park Sense yang merupakan Sistem Pendeteksi hunian Pada tempat parkir Dengan Menggunakan teknologi seperti sensor-sensor, kamera, atau perangkat untuk memantau dan merekam status slot parkir sistem berbasis web yang bisa diakses diberbagai platform dan data yang diambil dari sumber yang bisa dipertanggung jawabkan Ini melibatkan penggunaan. deteksi hunian di tempat parkir untuk meningkatkan efisiensi penggunaan ruang parkir, mengurangi waktu yang dihabiskan untuk mencari tempat parkir, dan meningkatkan pengalaman pengguna serta manajemen parkir.

### **Saran**

- Pelaksanaan MSIB di Orbit Future Academy program AI 4 Jobs pembelajarannya sudah sesuai, dan tepat waktu.

- Untuk mitra Orbit Future Academy, saran dari penulis adalah semoga kedepannya lebih baik lagi, sekarang sudah sangat baik tapi besar harapan bagi untuk perkembangan yang lebih hebat lagi kedepannya.
- Kami menyadari masih banyak kekurangan di sistem kami, saran terhadap pengembangan projek akhir kedepannya bisa dihosting dan diperbanyak dataset dan fitur-fiturnya.

## DAFTAR PUSTAKA

- [1] P. Tjiptoherijanto, "Urbanisasi dan pengembangan kota di Indonesia," *Populasi*, vol. 10, no. 2, pp. 57–72, 1999.
- [2] R. Lianzah, "Efisiensi Pengelolaan Manajemen Parkir Terhadap Penerimaan Pendapatan Asli Daerah Kota Bandar Lampung." UIN Raden Intan Lampung, 2017.
- [3] M. R. Pamungkas, "Gedung parkir di sagan 'Konsep Desain di Lahan Terbatas Dengan Kapasitas Parkir Maksimal dan Aplikasi Vegetasi Pada Fasad Bangunan,'" 2015.
- [4] A. Affan, "Analisa Kebutuhan Perparkiran Pada Rumah Sakit Umum Pirngadi Medan." Universitas Medan Area, 2012.
- [5] J. C. Setiawan, R. Lim, and J. Andjarwirawan, "Implementasi Internet of Things Untuk Parkir Mobil Dengan Pembayaran Menggunakan QR Code," *J. Infra*, p. 7, 2020, [Online]. Available: <http://publication.petra.ac.id/index.php/teknik-informatika/article/view/9767>.
- [6] Edi Surya Negara, "Smart Government," 2021, [Online]. Available: [http://eprints.binadarma.ac.id/4426/1/Smart Government - Full-Ebook.pdf](http://eprints.binadarma.ac.id/4426/1/Smart%20Government%20-%20Full-Ebook.pdf).