

AI Mastery Course

Topic 06

AI Deployment

Subtopic 05

Contoh Deployment untuk Domain
Computer Vision



Learning Objectives

Pada akhir modul ini, Anda akan dapat:

- Mampu menjelaskan siklus proyek AI serta proses pembuatan model khususnya untuk domain Computer Vision (CV)
- Dapat mendeskripsikan proses pembuatan Front End untuk aplikasi AI
- Dapat mendeskripsikan proses pembuatan Back End untuk aplikasi AI



Agenda

01

DESKRIPSI APLIKASI

- Siklus Proyek AI
- Lingkup Masalah
- Dataset
- Demo Aplikasi

02

PEMBUATAN MODEL

- Proses Pembuatan Model
- Model yang Digunakan

03

FRONT END

- Desain Halaman Aplikasi
- Fungsi pada JavaScript

04

BACK END

- Membuat Aplikasi Web
- Desain Routing
- Membuat API

05

HANDS-ON

- Hands-On

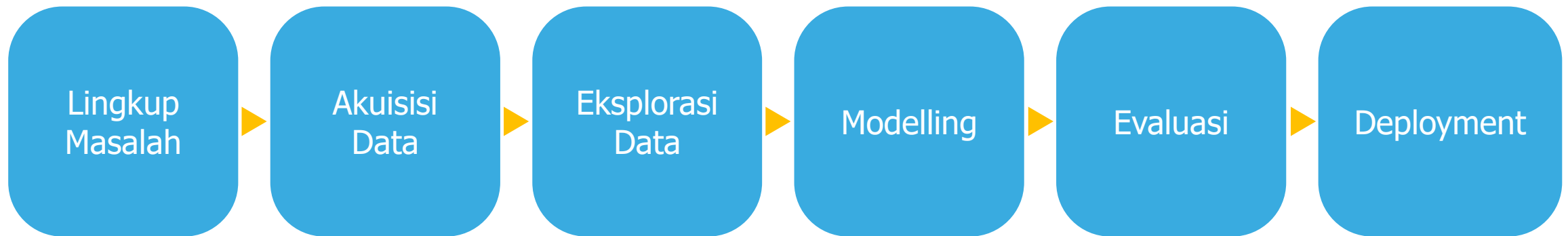


01

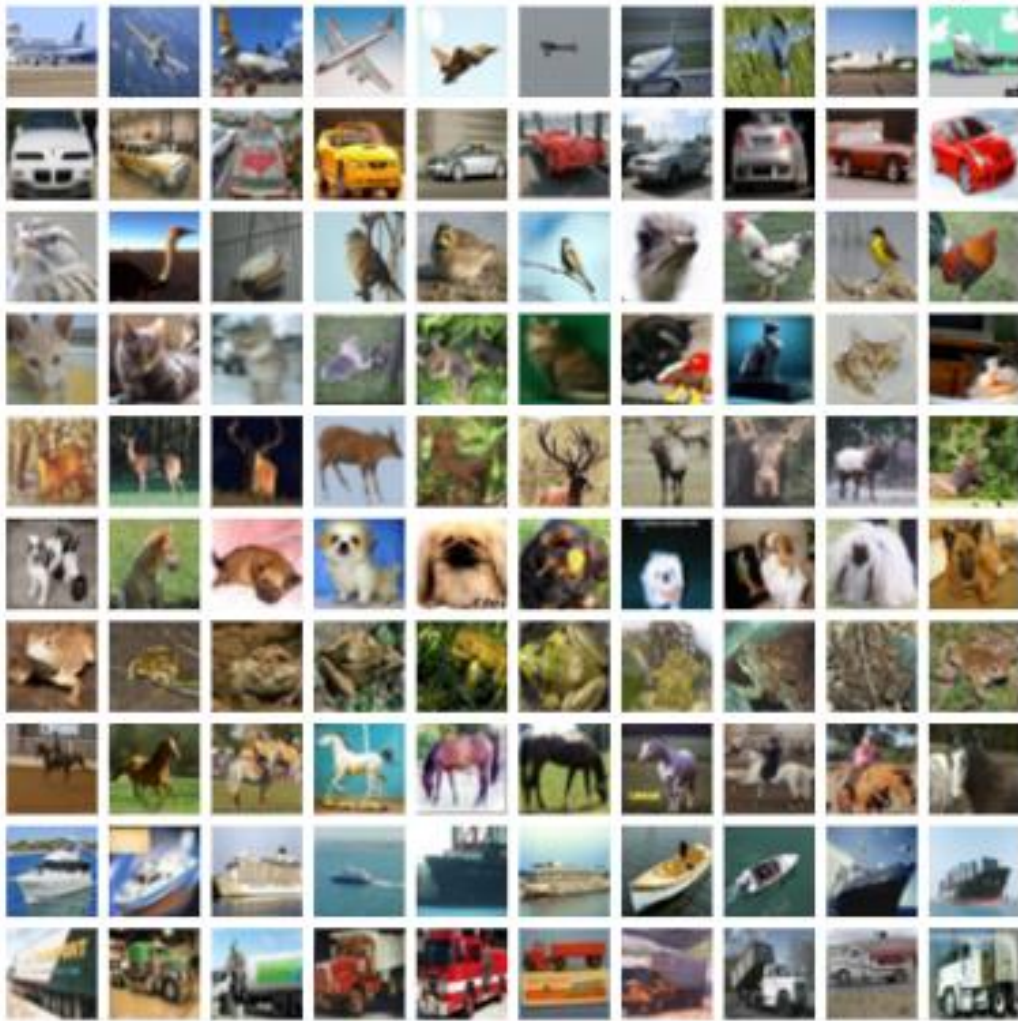
DESKRIPSI APLIKASI

- Siklus Proyek AI
- Lingkup Masalah
- Dataset
- Demo Aplikasi

Siklus Proyek AI



Lingkup Masalah



airplane



automobile



bird



cat



deer



dog



frog



horse



ship



truck

**Klasifikasi
10 Objek**

Dataset CIFAR 10

Dataset ***Canadian Institute for Advanced Research (CIFAR-10)*** terdiri dari **10 objek** yaitu : pesawat, mobil, burung, kucing, rusa, anjing, katak, kuda, kapal, dan truk.

Dataset CIFAR-10 memiliki 6000 gambar untuk setiap kelasnya.

5000 gambar digunakan untuk pelatihan dan sisanya 1000 gambar digunakan untuk pengujian.

Gambar pada dataset CIFAR-10 berukuran 32 x 32 pixel dengan 3 *channel*/warna.

airplane

automobile

bird

cat

deer

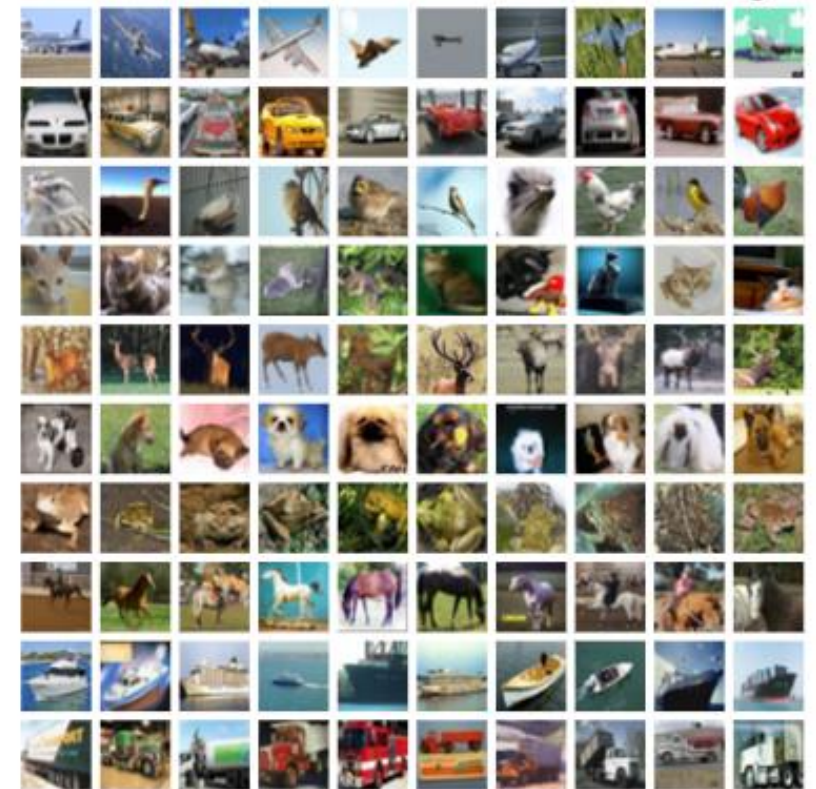
dog

frog

horse

ship

truck



Sumber : <https://www.cs.toronto.edu/~kriz/cifar.html>

Demo Aplikasi

CIC

Cifar-10 Image Classification

Aplikasi ini dibuat khusus untuk mendeteksi sebuah objek dari gambar yang diberikan. Adapun dataset yang digunakan untuk model klasifikasi gambar ini adalah CIFAR-10. Pada dataset CIFAR-10, terdapat 10 objek yang dapat diklasifikasi yaitu pesawat, mobil, burung, kucing, rusa, anjing, katak, kuda, kapal, dan truk. Untuk dapat menggunakan aplikasi ini, pengguna hanya mengunggah sebuah gambar dengan format 'JPG'. Hasil deteksi objek akan ditampilkan setelah pengguna mengunggah gambar tersebut.

Mulai

airplane

automobile

bird

cat

deer

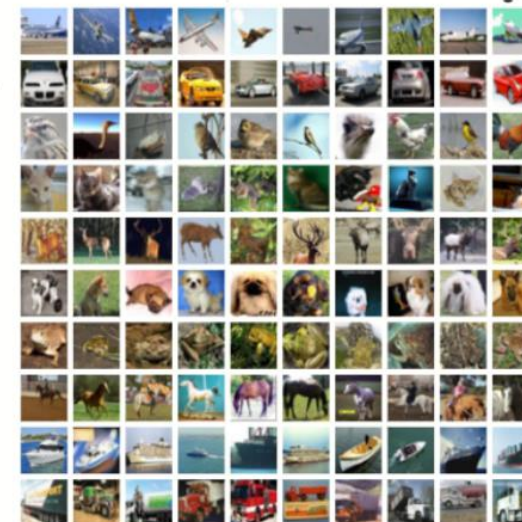
dog

frog

horse

ship

truck



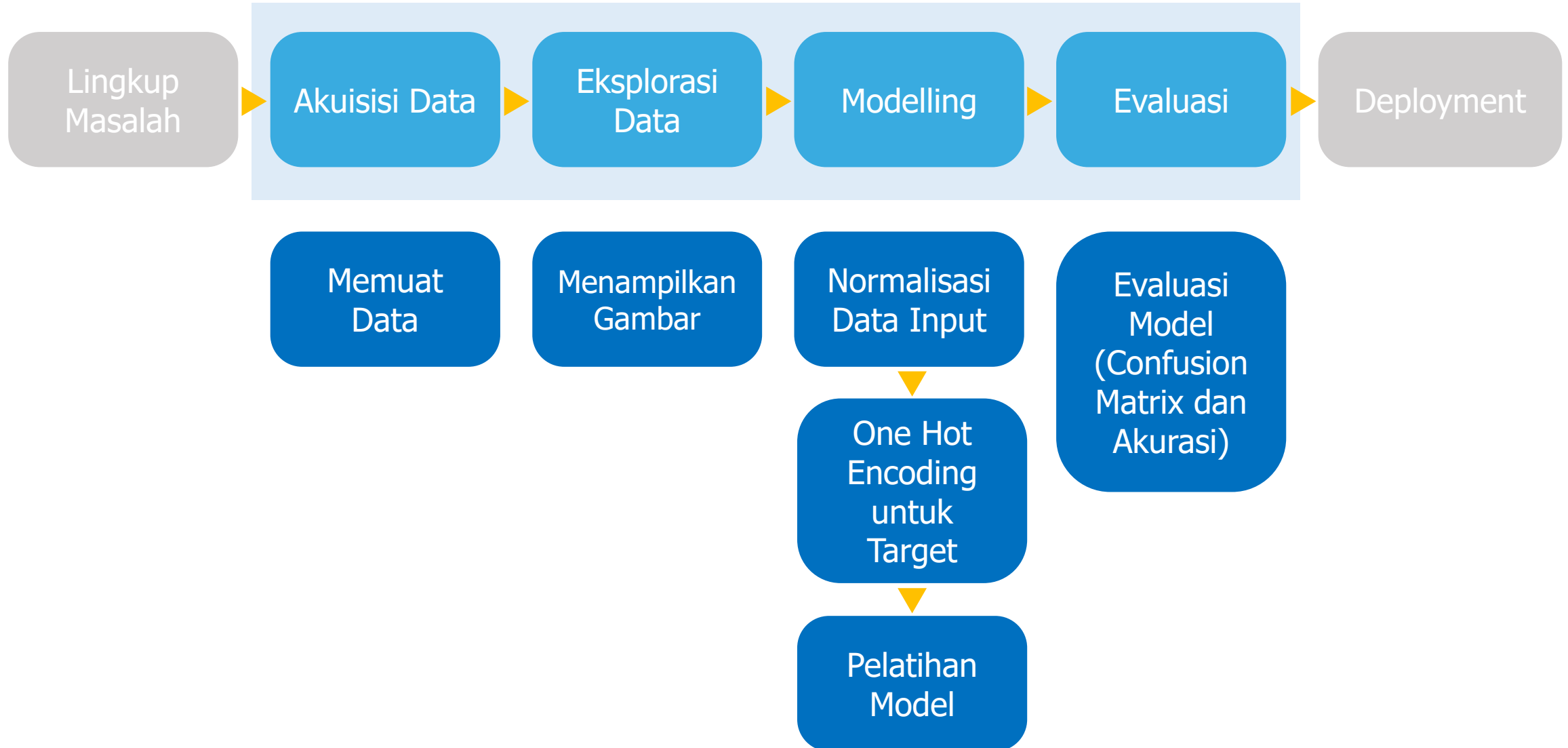


02

PPEMBUATAN MODEL

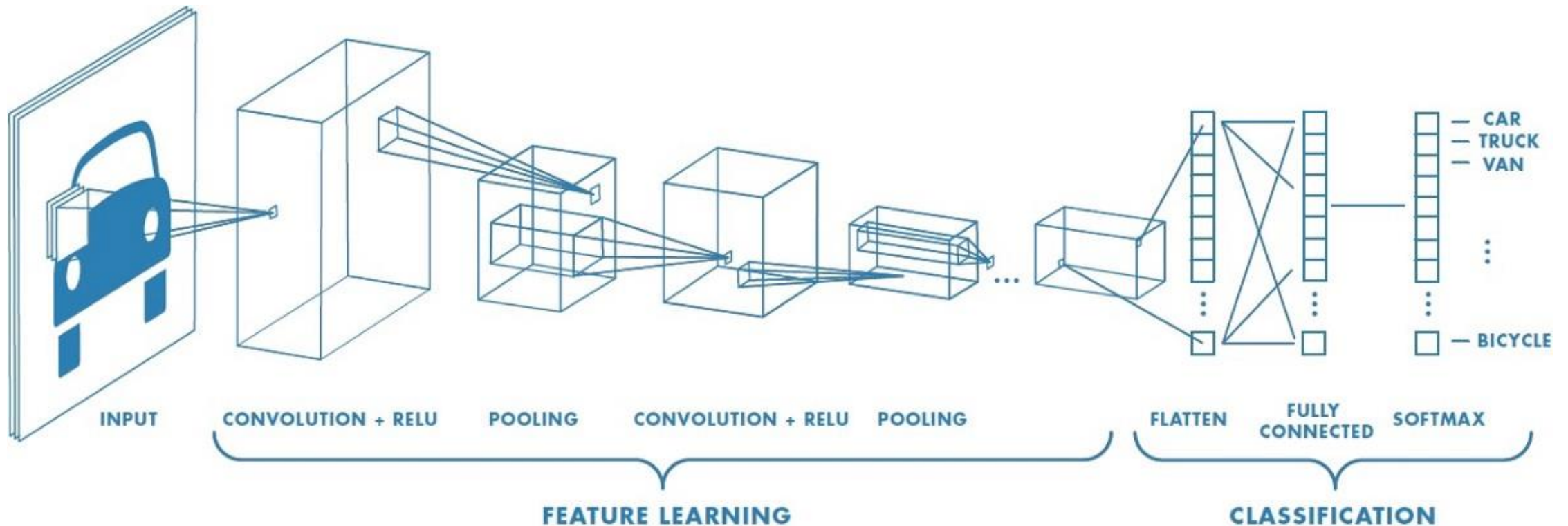
- Proses Pembuatan Model
- Model yang Digunakan

Proses Pembuatan Model



Model yang Digunakan

CNN



Sumber : <https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148>

Model yang Digunakan

CNN

```
def make_model():  
    model = Sequential()  
    model.add(Conv2D(16, (3, 3), input_shape=(32, 32, 3), padding='same'))  
    model.add(LeakyReLU(0.1))  
    model.add(Conv2D(32, (3, 3), padding='same'))  
    model.add(LeakyReLU(0.1))  
    model.add(MaxPooling2D(pool_size=(2, 2)))  
    model.add(Dropout(0.25))  
    model.add(Conv2D(32, (3, 3), padding='same'))  
    model.add(LeakyReLU(0.1))  
    model.add(Conv2D(64, (3, 3), padding='same'))  
    model.add(LeakyReLU(0.1))  
    model.add(MaxPooling2D(pool_size=(2, 2)))  
    model.add(Dropout(0.25))  
    model.add(Flatten())  
    model.add(Dense(256))  
    model.add(LeakyReLU(0.1))  
    model.add(Dropout(0.5))  
    model.add(Dense(10, activation='softmax'))  
  
    return model
```

Kode Python untuk Pelatihan Model

Klasifikasi Dataset CIFAR 10 (Model CNN)

Contoh Deployment untuk Domain Computer Vision (CV)

Orbit Future Academy - AI Mastery - KM Batch 3

Tim Deployment dan Tim CV

2022

Modules dan Packages

```
In [1]: 1 import numpy as np
        2 import matplotlib.pyplot as plt
        3 import tensorflow as tf
        4 import tensorflow.keras
        5 from tensorflow.keras import backend as K
        6 import tensorflow.keras.utils
        7 import ssl
        8 from tensorflow.keras.datasets import cifar10
        9 from tensorflow.keras.models import Sequential
       10 from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Activation, Dropout, LeakyReLU
       11 from sklearn.metrics import confusion_matrix, accuracy_score
       12
       13 %matplotlib inline
       14 K.clear_session()
```

(Notebook Link)



03

FRONT END

- Desain Halaman Aplikasi
- Fungsi pada JavaScript

Requirement untuk Front End



Google
Chrome



Bootstrap
(3.4.1)



jQuery
(3.6.0)



Fonts

(Lato & Glyphicons – Halflings Set)

Desain Halaman Aplikasi



Aplikasi Web yang dibangun memiliki 1 halaman web yang terdiri dari 5 bagian.

Note :
Aplikasi Web dirancang untuk dijalankan di Browser Chrome (*Best View*)

Desain Navigation Bar dan Informasi Aplikasi

CIC

Cifar-10 Image Classification

Aplikasi ini dibuat khusus untuk mendeteksi sebuah objek dari gambar yang diberikan. Adapun dataset yang digunakan untuk model klasifikasi gambar ini adalah CIFAR-10. Pada dataset CIFAR-10, terdapat 10 objek yang dapat diklasifikasi yaitu pesawat, mobil, burung, kucing, rusa, anjing, katak, kuda, kapal, dan truk. Untuk dapat menggunakan aplikasi ini, pengguna hanya mengunggah sebuah gambar dengan format 'JPG'. Hasil deteksi objek akan ditampilkan setelah pengguna mengunggah gambar tersebut.

Mulai

airplane

automobile

bird

cat

deer

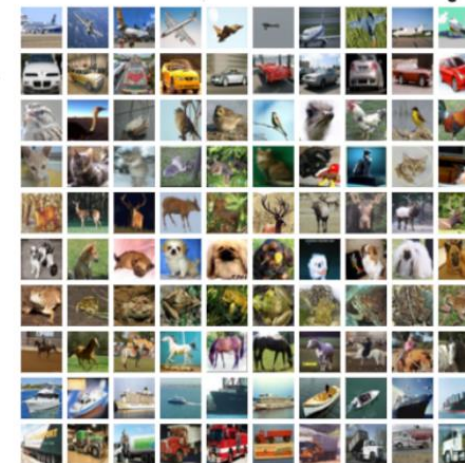
dog

frog

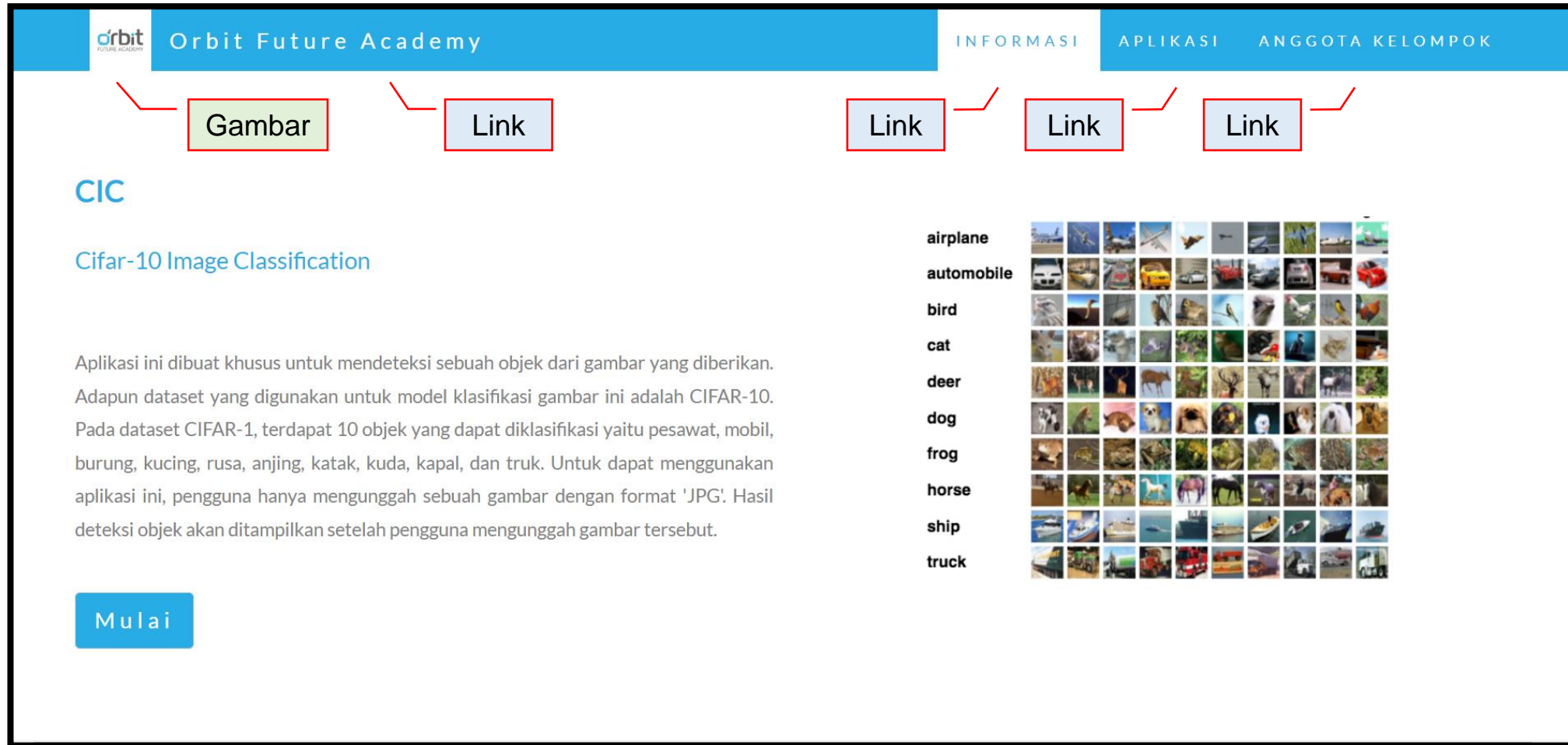
horse

ship

truck



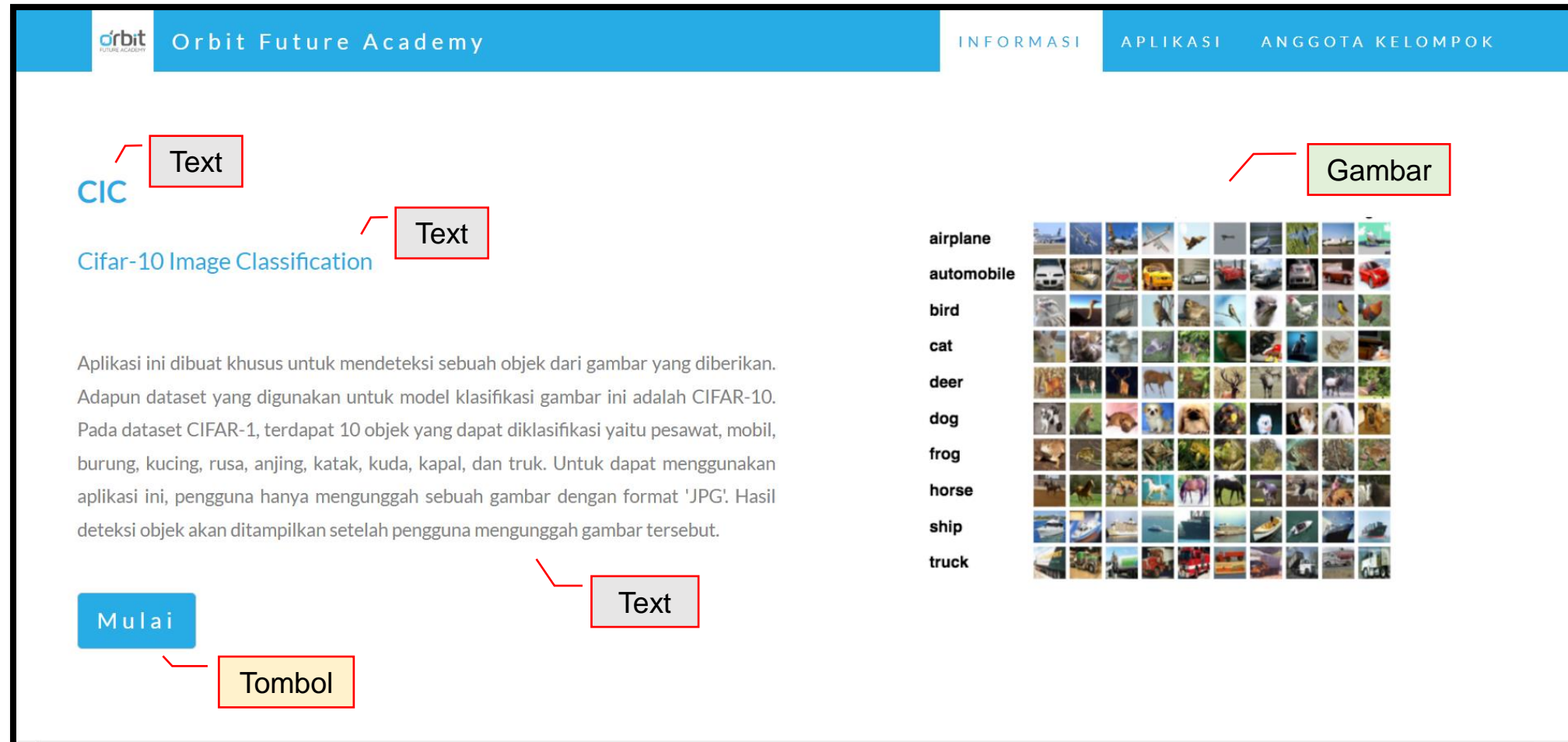
Desain Navigation Bar



Kode HTML untuk Navigation Bar

```
24 <!-- Navigation Bar -->
25 <nav class="navbar navbar-default navbar-fixed-top">
26   <div class="container">
27     <!-- Logo -->
28     <div class="navbar-header">
29       <img src={{ url_for('static', filename='/images/orbit_logo.jpg') }} alt="Orbit Logo"
30       width="50" height="50" align="left">
31       <a class="navbar-brand" href="#halamanku">Orbit Future Academy</a>
32     </div>
33
34     <!-- Daftar Halaman -->
35     <div class="collapse navbar-collapse" id="myNavbar">
36       <ul class="nav navbar-nav navbar-right">
37         <li><a href="#informasi">INFORMASI</a></li>
38         <li><a href="#aplikasi">APLIKASI</a></li>
39         <li><a href="#anggotaKelompok">ANGGOTA KELOMPOK</a></li>
40       </ul>
41     </div>
42   </div>
43 </nav>
44
```

Desain Halaman Bagian Informasi Aplikasi




Kode HTML untuk Halaman Bagian Informasi Aplikasi (1)

```
46 <!-- Bagian Halaman (Informasi) -->
47 <div id="informasi" class="container-fluid">
48     <br>
49     <br>
50     <div class="row">
51         <div class="col-sm-6">
52             <h2>CIC</h2>
53             <h4>Cifar-10 Image Classification</h4><br>
54             <p style="text-align: justify;text-justify: inter-word;">
55                 Aplikasi ini dibuat khusus untuk mendeteksi sebuah objek dari gambar yang diberikan.
56                 Adapun dataset yang digunakan untuk model klasifikasi gambar ini adalah CIFAR-10.
57                 Pada dataset CIFAR-1, terdapat 10 objek yang dapat diklasifikasi yaitu pesawat, mobil, burung, kucing,
58                 rusa, anjing, katak, kuda, kapal, dan truk.
59                 Untuk dapat menggunakan aplikasi ini, pengguna hanya mengunggah sebuah gambar dengan format 'JPG'.
60                 Hasil deteksi objek akan ditampilkan setelah pengguna mengunggah gambar tersebut.
61             </p>
62             <br>
63
64             <!-- Tombol Mulai -->
65             <nav id="navgo" class="navbar" style="background-color:#ffffff;z-index: 0;">
66                 <ul class="nav nav-pills">
67                     <li class="nav-item">
68                         <a class="btn btn-default btn-lg" href="#aplikasi" style="background-color:#28ace5">Mulai</a>
69                     </li>
70                 </ul>
71             </nav>
```

Kode HTML untuk Halaman Bagian Informasi Aplikasi (2)

```
64      <!-- Tombol Mulai -->
65      <nav id="navgo" class="navbar" style="background-color:#ffffff;z-index: 0;">
66          <ul class="nav nav-pills">
67              <li class="nav-item">
68                  <a class="btn btn-default btn-lg" href="#aplikasi" style="background-color:#28ace5">Mulai</a>
69              </li>
70          </ul>
71      </nav>
72
73  </div>
74  <div class="col-sm-6">
75      <br>
76      <br>
77      <!-- Gambar Deskripsi Aplikasi -->
78      <div align="center">
79          <img src={{ url_for('static', filename='/images/app-display.JPG') }} height="300" ></img>
80      </div>
81  </div>
82  </div>
83  <br>
84  </div>
```

Desain Halaman Bagian Aplikasi



**Orbit Future Academy**

INFORMASI

APLIKASI

ANGGOTA KELOMPOK

Upload Gambar

 airplane


silahkan upload sebuah gambar

Choose File

wikipedia_truck.jpg

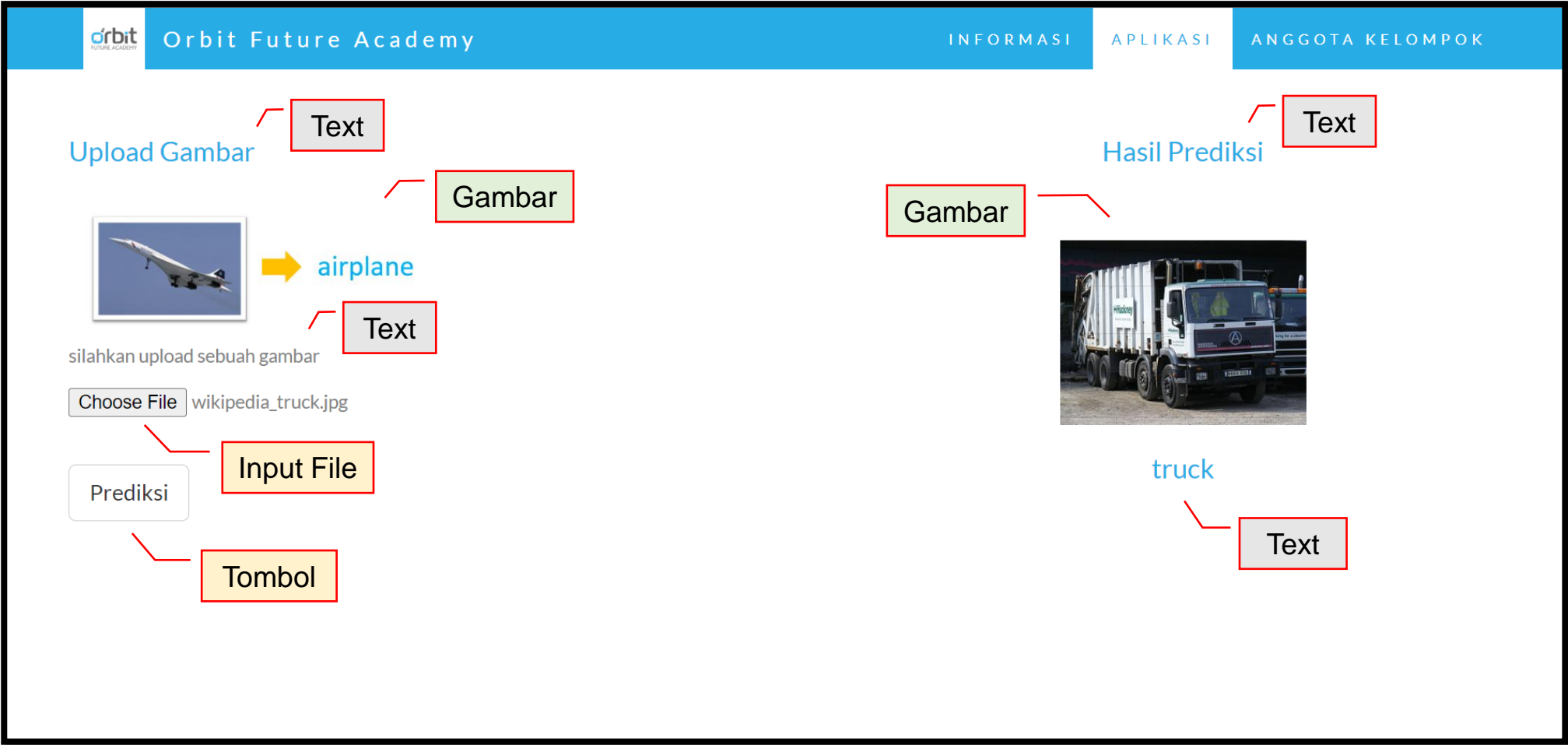
Prediksi

Hasil Prediksi



truck

Desain Halaman Bagian Aplikasi



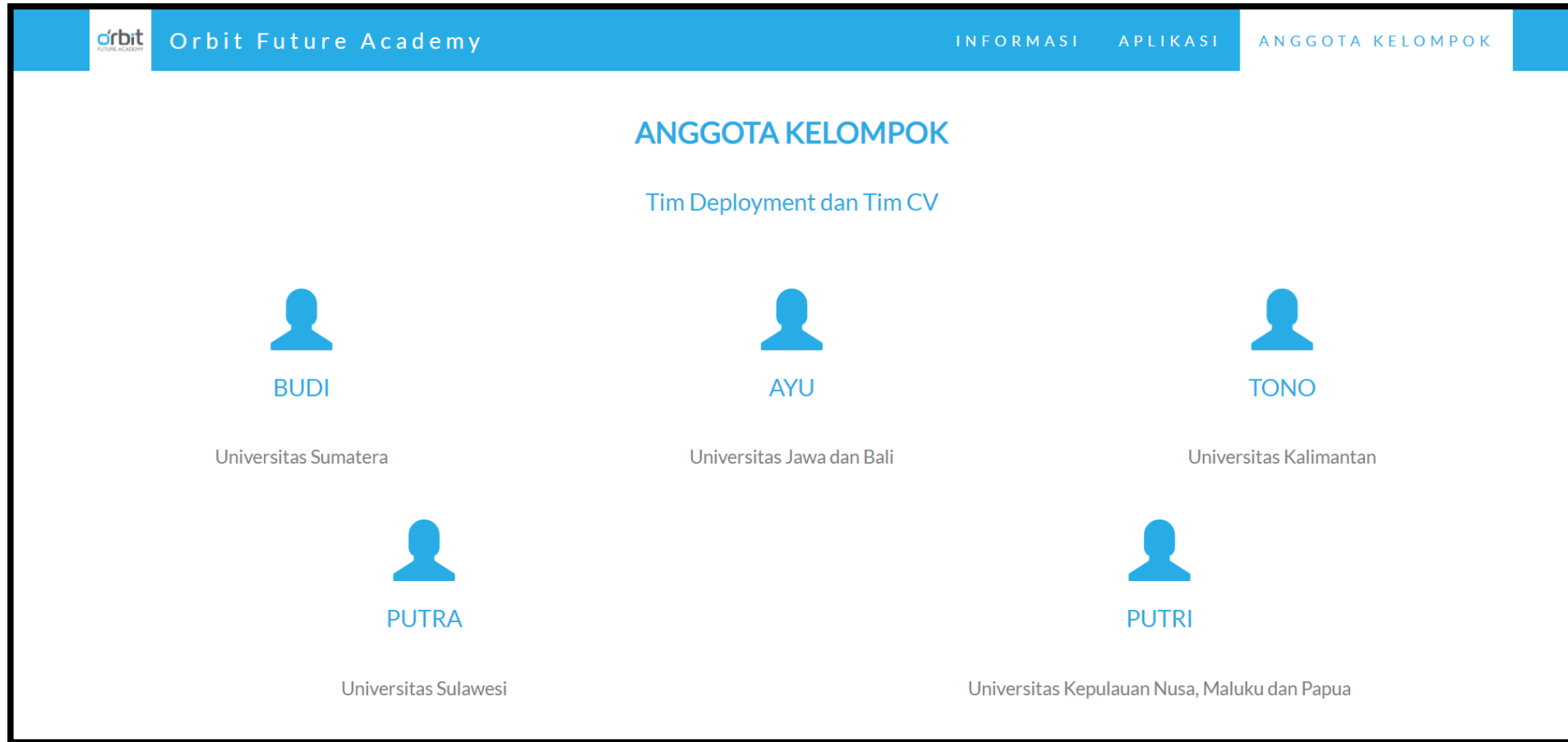
Kode HTML untuk Halaman Bagian Aplikasi (1)

```
87 <!-- Bagian Halaman (Aplikasi) -->
88 <div id="aplikasi" data-bs-spy="scroll" data-bs-target="#navgo" data-bs-offset="0" class="scrollspy-example" tabindex="0">
89     <br>
90     <br>
91     <div class="row">
92         <div class="col-sm-6">
93             <br>
94             <h3 style="margin-left: 50px;">Upload Gambar </h3>
95             <form style="margin-left: 50px;">
96                 <!-- Input atau Upload Gambar dari Pengguna -->
97                 <img src={{ url_for('static', filename='/images/ex-upload.JPG') }} width="300" ></img>
98                 <p>silahkan upload sebuah gambar</p>
99                 <p>
100                     <input type="file" id="input_gambar" name="file">
101                 </p>
102             </br>
103             <!-- Tombol Prediksi -->
104             <button type="submit" id="prediksi_submit" class="btn btn-default btn-lg">Prediksi</button>
105         </form>
106     </div>
```

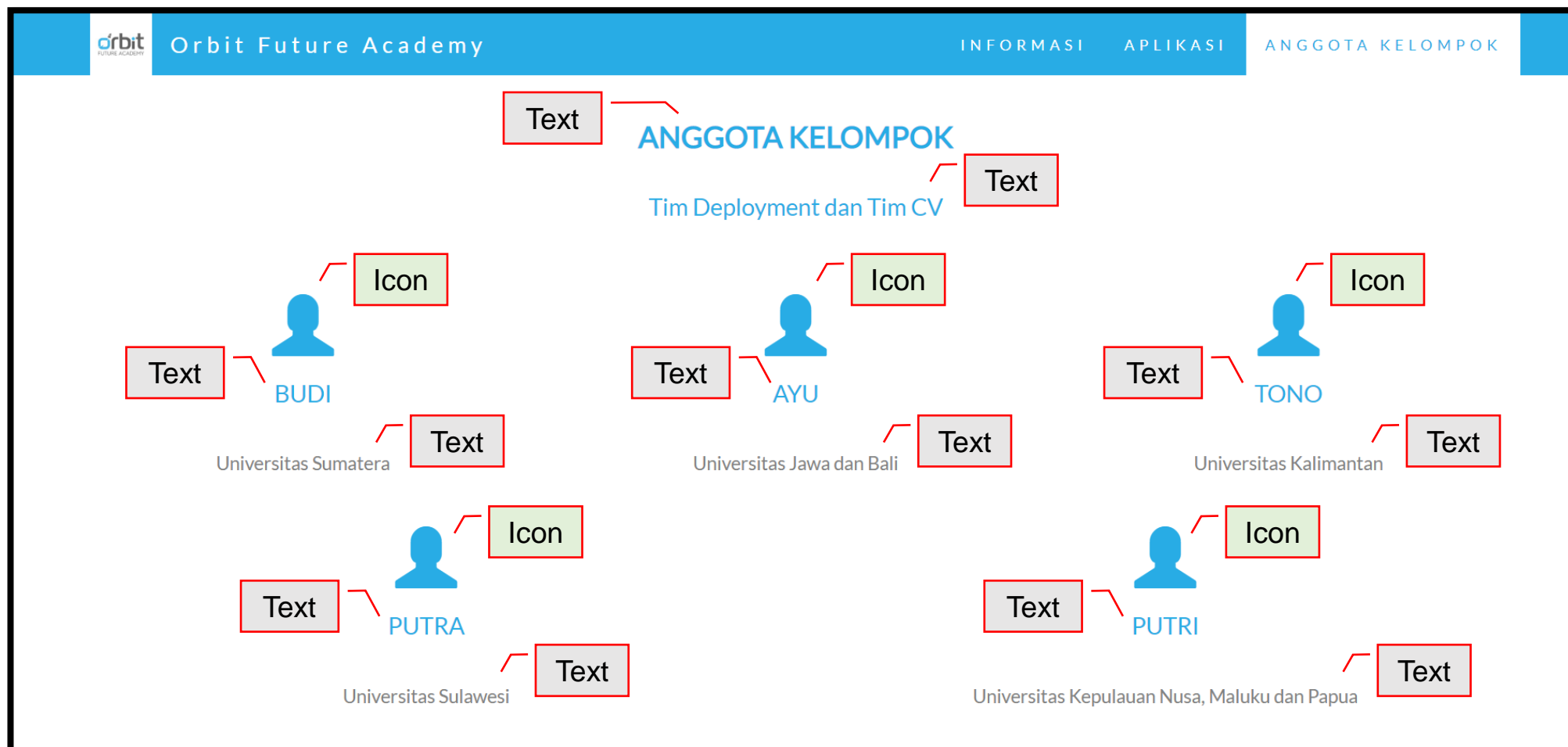
Kode HTML untuk Halaman Bagian Aplikasi (2)

```
103         <!-- Tombol Prediksi -->
104         <button type="submit" id="prediksi_submit" class="btn btn-default btn-lg">Prediksi</button>
105     </form>
106 </div>
107 <div class="col-sm-6">
108     <br>
109     <!-- Bagian Halaman untuk Hasil Prediksi -->
110     <div id="hasil_prediksi" align="center">
111         <!-- Hasil Prediksi akan ditampilkan di sini-->
112     </div>
113 </div>
114 </div>
115 </br>
116 </br>
117 </br>
118 </br>
119 </div>
```


Desain Halaman Bagian Anggota Kelompok



Desain Halaman Bagian Anggota Kelompok



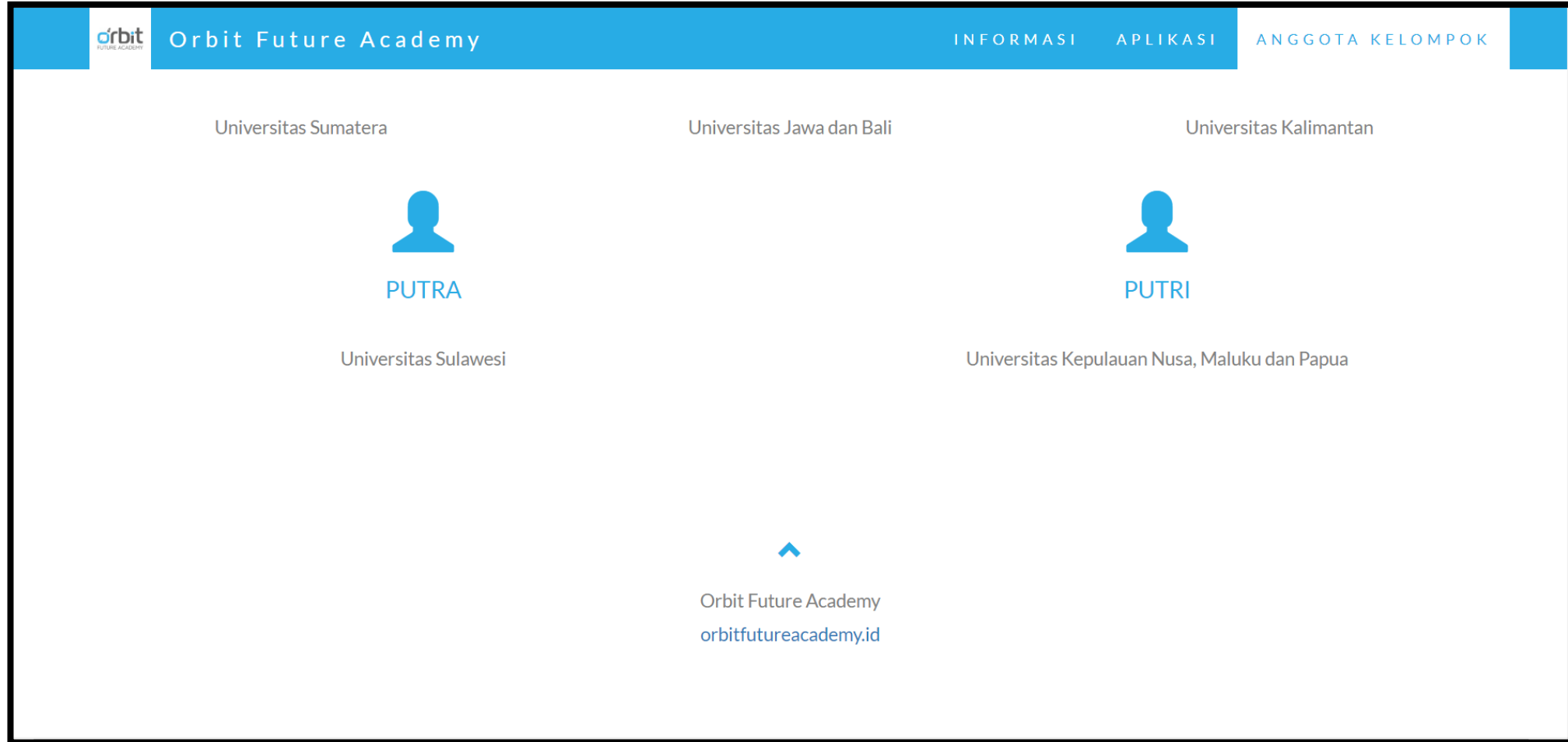
Kode HTML untuk Halaman Bagian Anggota Kelompok (1)

```
121 <!-- Bagian Halaman (Anggota Kelompok) -->
122 <div id="anggotaKelompok" class="container-fluid text-center">
123     <br>
124     <h2>ANGGOTA KELOMPOK</h2>
125     <h4>Tim Deployment dan Tim CV</h4>
126     <br>
127     <div class="row slideanim">
128         <div class="col-sm-4">
129             <span class="glyphicon glyphicon-user logo-small"></span>
130             <h4>BUDI</h4>
131             <p>Universitas Sumatera</p>
132         </div>
133         <div class="col-sm-4">
134             <span class="glyphicon glyphicon-user logo-small"></span>
135             <h4>AYU</h4>
136             <p>Universitas Jawa dan Bali</p>
137         </div>
138         <div class="col-sm-4">
139             <span class="glyphicon glyphicon-user logo-small"></span>
140             <h4>TONO</h4>
141             <p>Universitas Kalimantan</p>
142         </div>
143     </div>
144     <br>
```


Kode HTML untuk Halaman Bagian Anggota Kelompok (2)

```
138 <div class="col-sm-4">
139   <span class="glyphicon glyphicon-user logo-small"></span>
140   <h4>TONO</h4>
141   <p>Universitas Kalimantan</p>
142 </div>
143 </div>
144 <br>
145 <div class="row slideanim">
146   <div class="col-sm-6">
147     <span class="glyphicon glyphicon-user logo-small"></span>
148     <h4>PUTRA</h4>
149     <p>Universitas Sulawesi</p>
150   </div>
151   <div class="col-sm-6">
152     <span class="glyphicon glyphicon-user logo-small"></span>
153     <h4>PUTRI</h4>
154     <p>Universitas Kepulauan Nusa, Maluku dan Papua</p>
155   </div>
156 </div>
157 </div>
```

Desain Footer



Desain Footer




Orbit Future Academy

INFORMASI

APLIKASI

ANGGOTA KELOMPOK


Universitas Sumatera



PUTRA

Universitas Sulawesi

Universitas Jawa dan Bali



PUTRI

Universitas Kepulauan Nusa, Maluku dan Papua

Text

Tombol

Orbit Future Academy
orbitfutureacademy.id

Link

Kode HTML untuk Halaman Bagian Footer

```
159 <!-- Footer -->
160 <footer class="container-fluid text-center">
161     <a href="#halamanku" title="To Top">
162         <span class="glyphicon glyphicon-chevron-up"></span>
163     </a>
164     <p>Orbit Future Academy <br> <a href="https://orbitfutureacademy.id/">orbitfutureacademy.id</a></p>
165 </footer>
```

Fungsi pada JavaScript

Terdapat 3 Fungsi JavaScript yang dibuat

Animasi
Scroll

Memanggil API untuk
Prediksi Gambar
(AJAX)

Menampilkan Hasil
Prediksi Gambar

Fungsi JavaScript untuk Animasi Scroll

CIC

Cifar-10 Image Classification

Aplikasi ini dibuat khusus untuk mendeteksi sebuah objek dari gambar yang diberikan. Adapun dataset yang digunakan untuk model klasifikasi gambar ini adalah CIFAR-10. Pada dataset CIFAR-10, terdapat 10 objek yang dapat diklasifikasi yaitu pesawat, mobil, burung, kucing, rusa, anjing, katak, kuda, kapal, dan truk. Untuk dapat menggunakan aplikasi ini, pengguna hanya mengunggah sebuah gambar dengan format 'JPG'. Hasil deteksi objek akan ditampilkan setelah pengguna mengunggah gambar tersebut.

Mulai

airplane

automobile

bird

cat

deer

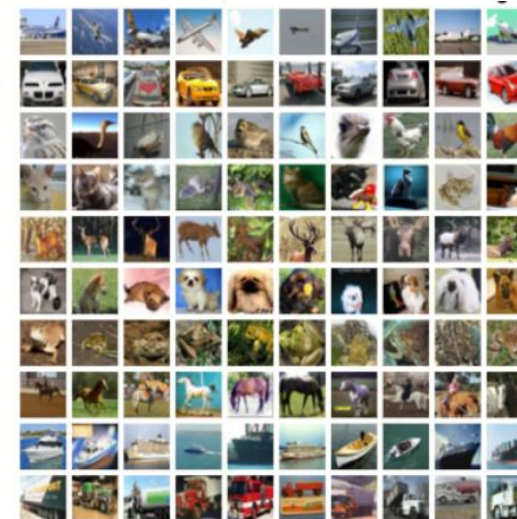
dog

frog

horse

ship


truck



Kode JavaScript untuk Animasi Scroll

```
3 // -[Animasi Scroll-----
4
5 $(".navbar a, footer a[href='#halamanku']").on('click', function(event) {
6     if (this.hash !== "") {
7         event.preventDefault();
8         var hash = this.hash;
9         $('html, body').animate({
10             scrollTop: $(hash).offset().top
11         }, 900, function(){
12             window.location.hash = hash;
13         });
14     }
15 });
16
17 $(window).scroll(function() {
18     $(".slideanim").each(function() {
19         var pos = $(this).offset().top;
20         var winTop = $(window).scrollTop();
21         if (pos < winTop + 600) {
22             $(this).addClass("slide");
23         }
24     });
25 });
```

Fungsi JavaScript untuk Memanggil API dan Menampilkan Hasil Prediksi Gambar



 Orbit Future Academy

INFORMASI

APLIKASI

ANGGOTA KELOMPOK

Upload Gambar

 airplane

silahkan upload sebuah gambar

Choose File No file chosen

Prediksi

Kode JavaScript untuk Memanggil API Prediksi Gambar (1)

```
27 // -[Prediksi Model]-----
28
29 // Fungsi untuk memanggil API ketika tombol prediksi ditekan
30 $("#prediksi_submit").click(function(e) {
31     e.preventDefault();
32
33     // Get File Gambar yg telah diupload pengguna
34     var file_data = $('#input_gambar').prop('files')[0];
35     var pics_data = new FormData();
36     pics_data.append('file', file_data);
37
38     // Panggil API dengan timeout 1 detik (1000 ms)
39     setTimeout(function() {
40         try {
41             $.ajax({
42                 url: "/api/deteksi",
43                 type: "POST",
44                 data: pics_data,
45                 processData: false,
46                 contentType: false,
47                 success: function(res) {
48                     // Ambil hasil prediksi dan path gambar yang diprediksi dari API
49                     res_data_prediksi = res['prediksi']
50                     res_gambar_prediksi = res['gambar_prediksi']
51                 }
41             });
42         }
43     });
44 }
```

Kode JavaScript untuk Memanggil API Prediksi Gambar (2)

```
38 // Panggil API dengan timeout 1 detik (1000 ms)
39 setTimeout(function() {
40     try {
41         $.ajax({
42             url: "/api/deteksi",
43             type: "POST",
44             data: pics_data,
45             processData: false,
46             contentType: false,
47             success: function(res) {
48                 // Ambil hasil prediksi dan path gambar yang diprediksi dari API
49                 res_data_prediksi = res['prediksi']
50                 res_gambar_prediksi = res['gambar_prediksi']
51
52                 // Tampilkan hasil prediksi ke halaman web
53                 generate_prediksi(res_data_prediksi, res_gambar_prediksi);
54             }
55         });
56     }
57     catch(e) {
58         // Jika gagal memanggil API, tampilkan error di console
59         console.log("Gagal !");
60         console.log(e);
61     }
62 }, 1000)
63 })
```

Kode JavaScript untuk Menampilkan Hasil Prediksi Gambar

```
65 // Fungsi untuk menampilkan hasil prediksi model
66 function generate_prediksi(data_prediksi, image_prediksi) {
67     var str="";
68
69     if(image_prediksi == "(none)") {
70         str += "<h3>Hasil Prediksi </h3>";
71         str += "<br>";
72         str += "<h4>Silahkan masukkan file gambar (.jpg)</h4>";
73     }
74     else {
75         str += "<h3>Hasil Prediksi </h3>";
76         str += "<br>";
77         str += "<img src='" + image_prediksi + "' width=\"200\"></img>";
78         str += "<h3>" + data_prediksi + "</h3>";
79     }
80     $("#hasil_prediksi").html(str);
81 }
```




04

BACK END

- Membuat Aplikasi Web
- Desain Routing
- Membuat API

Membuat Aplikasi Web

1. Memuat Module dan Package
2. Definisikan Variabel Global
3. Membuat Routing
4. Memuat Model ML yang telah dilatih
5. Menjalankan Aplikasi Web

```
8  # =[Modules dan Packages]=====
9
10 from flask import Flask,render_template,request,jsonify
11 from werkzeug.utils import secure_filename
12 import pandas as pd
13 import numpy as np
14 import os
15 import tensorflow as tf
16 from tensorflow.keras.models import Sequential
17 from tensorflow.keras.layers import Conv2D, MaxPooling2D, \
18 Flatten, Dense, Activation, Dropout,LeakyReLU
19 from PIL import Image
20 from fungsi import make_model
21
22 # =[Variabel Global]=====
23
24 app = Flask(__name__, static_url_path='/static')
25
26 app.config['MAX_CONTENT_LENGTH'] = 1024 * 1024
27 app.config['UPLOAD_EXTENSIONS'] = ['.jpg', '.JPG']
28 app.config['UPLOAD_PATH']      = './static/images/uploads/'
29
30 model = None
31
32 NUM_CLASSES = 10
33 cifar10_classes = ["airplane", "automobile", "bird", "cat", "deer",
34                   "dog", "frog", "horse", "ship", "truck"]
35
```

1

2

Membuat Aplikasi Web

1. Memuat Module dan Package
2. Definisikan Variabel Global
3. Membuat Routing
4. Memuat Model ML yang telah dilatih
5. Menjalankan Aplikasi Web

```
36 # =[Routing]=====
37
38 # [Routing untuk Halaman Utama atau Home]
39 @app.route("/")
40 def beranda():
41
42
43 # [Routing untuk API]
44 @app.route("/api/deteksi", methods=['POST'])
45 def apiDeteksi():
46
47
48 # =[Main]=====
49
50 if __name__ == '__main__':
51
52     # Load model yang telah ditraining
53     model = make_model()
54     model.load_weights("model_cifar10_cnn_tf.h5")
55
56     # Run Flask di localhost
57     app.run(host="localhost", port=5000, debug=True)
```

3

4

5

Desain Routing

Routing ditujukan untuk memetakan alamat URL dengan suatu fungsi.

Terdapat 2 URL yang akan digunakan :

1. Halaman Utama (/)
2. API Prediksi Gambar (/api/deteksi)

untuk API Prediksi Gambar, digunakan metode POST

Desain Routing

```
36 # =[Routing]=====
37
38 # [Routing untuk Halaman Utama atau Home]
39 @app.route("/")
40 def beranda():
41     return render_template('index.html')
42
43 # [Routing untuk API]
44 @app.route("/api/deteksi",methods=['POST'])
45 def apiDeteksi():
46     # Set nilai default untuk hasil prediksi dan gambar yang diprediksi
47     hasil_prediksi = '(none)'
48     gambar_prediksi = '(none)'
49
50     # Get File Gambar yg telah diupload pengguna
51     uploaded_file = request.files['file']
52     filename      = secure_filename(uploaded_file.filename)
53
```

Membuat API (1)

```
43 # [Routing untuk API]
44 @app.route("/api/deteksi",methods=['POST'])
45 def apiDeteksi():
46     # Set nilai default untuk hasil prediksi dan gambar yang diprediksi
47     hasil_prediksi = '(none)'
48     gambar_prediksi = '(none)'
49
50     # Get File Gambar yg telah diupload pengguna
51     uploaded_file = request.files['file']
52     filename      = secure_filename(uploaded_file.filename)
53
54     # Periksa apakah ada file yg dipilih untuk diupload
55     if filename != '':
56
57         # Set/mendapatkan extension dan path dari file yg diupload
58         file_ext      = os.path.splitext(filename)[1]
59         gambar_prediksi = '/static/images/uploads/' + filename
60
61         # Periksa apakah extension file yg diupload sesuai (jpg)
62         if file_ext in app.config['UPLOAD_EXTENSIONS']:
63
64             # Simpan Gambar
65             uploaded_file.save(os.path.join(app.config['UPLOAD_PATH'], filename))
66
67             # Memuat Gambar
68             test_image      = Image.open('.') + gambar_prediksi
69
```

Membuat API (2)

```
67 # Memuat Gambar
68 test_image = Image.open('.') + gambar_prediksi
69
70 # Mengubah Ukuran Gambar
71 test_image_resized = test_image.resize((32, 32))
72
73 # Konversi Gambar ke Array
74 image_array = np.array(test_image_resized)
75 test_image_x = (image_array / 255) - 0.5
76 test_image_x = np.array([image_array])
77
78 # Prediksi Gambar
79 y_pred_test_single = model.predict_proba(test_image_x)
80 y_pred_test_classes_single = np.argmax(y_pred_test_single, axis=1)
81
82 hasil_prediksi = cifar10_classes[y_pred_test_classes_single[0]]
83
84 # Return hasil prediksi dengan format JSON
85 return jsonify({
86     "prediksi": hasil_prediksi,
87     "gambar_prediksi" : gambar_prediksi
88 })
89 else:
90     # Return hasil prediksi dengan format JSON
91     gambar_prediksi = '(none)'
92     return jsonify({
93         "prediksi": hasil_prediksi,
94         "gambar_prediksi" : gambar_prediksi
95     })
96
```



05

HANDS-ON

- Hands-On

Hands-On

(Source Code Link)

