

## DESKRIPSI PROGRAM DAN CARA IMPLEMENTASI

### WisataBot:

**Aplikasi Chatbot untuk Pariwisata di Provinsi Nusa Tenggara Timur  
Menggunakan Algoritme Deep Learning**



**PROGRAM STUDI TEKNOLOGI REKAYASA JARINGAN  
TELEKOMUNIKASI  
JURUSAN TEKNIK ELEKTRO  
POLITEKNIK NEGERI UJUNG PANDANG  
MAKASSAR  
2022**

## Program Komputer:

→ Coding Pemograman dapat diakses pada link yang disediakan.

## Deskripsi Program Komputer

### A. Aplikasi WisataBot

Nusa Tenggara Timur sebagai salah satu provinsi di Indonesia yang memiliki potensi Daya Tarik Wisata (DTW) yang tinggi. Tercatat total potensi DTW alam di Provinsi Nusa Tenggara Timur berjumlah 643 DTW dengan 22 kabupaten yang menyajikan berbagai objek wisata alam (hutan, pantai, gunung, gua, danau, dan lainnya). Banyaknya alternatif pilihan destinasi wisata memungkinkan para wisatawan merasa kesulitan untuk memutuskan tempat wisata yang akan dipilih, sehingga muncul salah satu solusi untuk memecahkan permasalahan tersebut yaitu dengan bantuan chatbot sistem rekomendasi objek wisata.

Aplikasi Wisatabot adalah suatu aplikasi chatbot berbasis web dan mobile yang membantu para wisatawan untuk mengetahui destinasi wisata di Nusa Tenggara Timur (NTT) secara lengkap sehingga memberi kemudahan bagi pengguna untuk menentukan tempat wisata yang diinginkan. Adapun gambaran aplikasi wisatabot ditampilkan di bawah ini:

#### 1. Halaman Utama Aplikasi Wisatabot



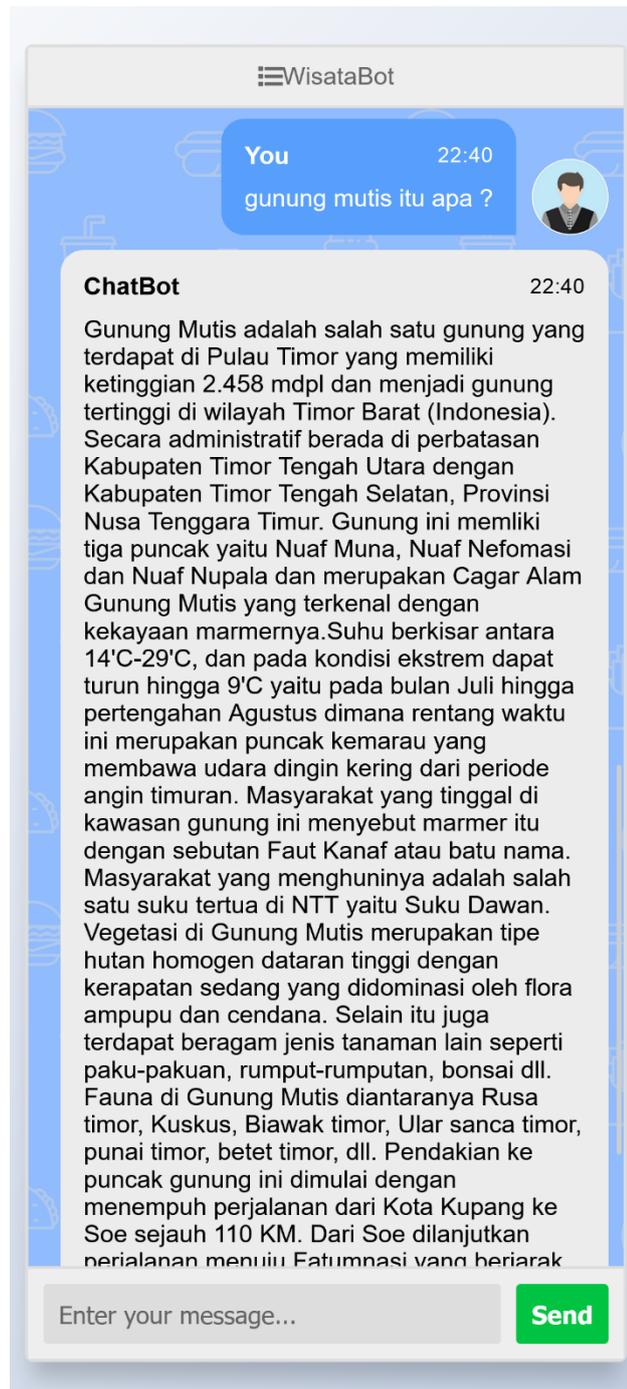
Gambar 1. Halaman Utama

## 2. Halaman Room Chat



Gambar 2. Room Chat

### 3. Halaman Room Chat Lanjutan



Gambar 3. Room Chat Lanjutan

#### 4. Tampilan Desktop Aplikasi WisataBot



Gambar 4. Aplikasi Wisatabot berbasis Desktop

#### 5. Potongan Source Code Back-End Aplikasi Wisatabot

```
app.py > ...
1 import nltk
2 #nltk.download('popular')
3 from nltk.stem import WordNetLemmatizer
4 lemmatizer = WordNetLemmatizer()
5 import pickle
6 import numpy as np
7
8 from keras.models import load_model
9 model = load_model('model.h5')
10 import json
11 import random
12 intents = json.loads(open('wisata_data.json').read())
13 words = pickle.load(open('texts.pkl', 'rb'))
14 classes = pickle.load(open('labels.pkl', 'rb'))
15
16 def clean_up_sentence(sentence):
17     # tokenize the pattern - split words into array
18     sentence_words = nltk.word_tokenize(sentence)
19     # stem each word - create short form for word
20     sentence_words = [lemmatizer.lemmatize(word.lower()) for word in sentence_words]
21     return sentence_words
22
23 # return bag of words array: 0 or 1 for each word in the bag that exists in the sentence
24
25 def bow(sentence, words, show_details=True):
26     # tokenize the pattern
27     sentence_words = clean_up_sentence(sentence)
28     # bag of words - matrix of N words, vocabulary matrix
29     bag = [0]*len(words)
30     for s in sentence_words:
31         for i,w in enumerate(words):
32             if w == s:
```

Gambar 5. Potongan Source Code Back-End Aplikasi Wisatabot

## 6. Potongan *Source Code Front-End* Aplikasi Wisatabot

```
app.py x training.py datajson wisata_data.json style.css index.html x
templates > index.html
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5 <meta charset="UTF-8">
6 <title>WisataBot</title>
7 <meta charset="UTF-8">
8 <meta name="viewport" content="width=device-width, initial-scale=1.0">
9 <meta http-equiv="X-UA-Compatible" content="ie=edge">
10 <link rel="stylesheet" href="{{ url_for('static', filename='styles/style.css') }}">
11 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
12 </head>
13
14 <body>
15 <!-- partial:index.partial.html -->
16 <section class="msgsr">
17 <header class="msgsr-header">
18 <div class="msgsr-header-title">
19 <i class="fas fa-list"></i>WisataBot</i>
20 </div>
21 </header>
22
23 <main class="msgsr-chat">
24 <div class="msg left-msg">
25 <div class="msg-img" style="background-image: url(static/images/robot.png)"></div>
26
27 <div class="msg-bubble">
28 <div class="msg-info">
29 <div class="msg-info-name">WisataBot</div>
30 <div class="msg-info-time"></div>
31 </div>
32
```

Gambar 6. Potongan *Source Code Front-End* Aplikasi Wisatabot

### B. Panduan Penggunaan Program

#### 1. Spesifikasi Personal Computer (PC) atau Laptop yang direkomendasikan:

##### a. Sistem Operasi:

Minimum Windows 8,10, MacOS Mojave, Linux Ubuntu 16.04.

##### b. Processor Intel Core i5

##### c. RAM 4 GB

#### 2. Instalasi Program

##### a. Pastikan PC anda ter-*install* Python 3.7 dan Flask (*Web Framework*)

##### b. *Extract* file wisatabot.zip di folder flask.



##### c. Buka folder wisatabot



d. *Install library* yang dibutuhkan dengan perintah di bawah ini pada terminal:

➤ Pip install -r requirements.txt

e. Setelah itu, program dapat dijalankan. Pastikan semua *library* ter-*install*.

### C. Menjalankan Program

1. Buka Terminal dan masuk ke folder flask (lokasi folder wisatabot disimpan) – masuk ke folder wisatabot dan ketik perintah di bawah pada terminal:

➤ cd flask

➤ cd wisatabot

➤ python3 app.py

```
2022-12-13 22:32:19.046216: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:169] retrieving CUDA diagnostic information for host: OFAJKT-010042
2022-12-13 22:32:19.047129: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:176] hostname: OFAJKT-010042
2022-12-13 22:32:19.048199: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
```

2. Buka *browser* (*Mozilla Firefox*, *Safari*, atau *Google Chrome*) dan masukkan IP Address sesuai hasil di atas:

➤ http://127.0.0.1:5000/



3. Aplikasi siap digunakan ketika halaman utama muncul pada layer komputer seperti Gambar 7. Kemudian, silahkan chat dengan wisatabot. Wisatabot akan menjawab secara otomatis apa yang anda tanyakan.



Gambar 7. Halaman Utama Aplikasi Wisatobot

Pada Gambar 7, pengguna harus memberikan pertanyaan yang ingin ditanyakan. Setelah itu, aplikasi akan menjawab sesuai dengan pertanyaan pengguna.

Untuk keluar dari program, tekan CTRL+C pada terminal.

```
* Serving Flask app 'app'  
* Debug mode: on  
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.  
* Running on http://127.0.0.1:5000  
Press CTRL+C to quit  
* Restarting with stat
```