



# Portfolio

---

Rafi Gumilang



# About Me

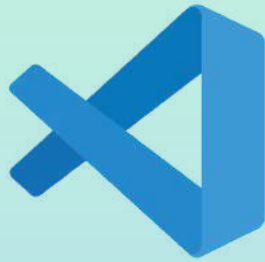
Hi,  
I am Rafi!



---

I am a final year undergraduate Physics student at the University of Indonesia, specializing in Instrumentation Physics. I have experience in the fields of machine learning, web development and have created applications while I participated in the Bangkit Academy program led by Google, Tokopedia, Gojek, & Traveloka. This allows me to work in a team in a fast-paced environment and requires good communication. Currently, I am looking for new opportunities to expand my knowledge and skills as a data analyst and machine learning.

# Tools



Google Calendar



Google Drive



Google Forms



Google Sheets

G Suite



Google Docs



Google Slides



Gmail

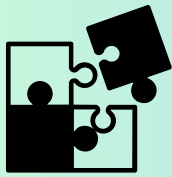


Google Hangouts

# Skills



Critical Thinking



Problem Solving



Time Management



Leadership



Project Management



Programing



Public Speaking



Content Writing

# Project Management

## Data Analyst with Tableau



AutoSave: Off | HR... Saved to this... | Search

File Home Insert Page Layout Formulas Data Review View Automate Add-ins Help Table Design

Clipboard Font Alignment Number Styles Cells Editing Sensitivity Add-ins

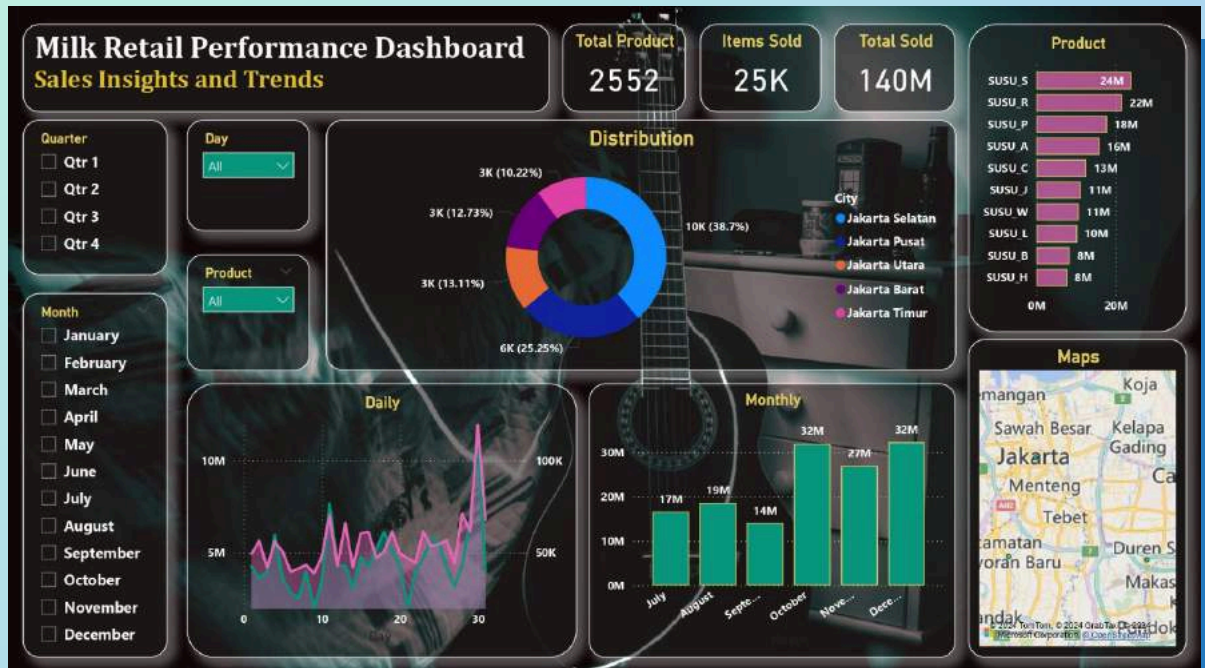
	L	M	N	O	P	Q	R	S	T	U
1	Over Time	Over18	Training Times Last Year	Age	CF_current Employee	Daily Rate	Distance From Home	Education	Employee Count	Environment Satisfaction
2	Yes	Y		0	41	0	1102	1 Associates Dr	1	2
3	No	Y		3	49	1	279	8 High School	1	3
4	Yes	Y		3	37	0	1373	2 Associates Dr	1	4
5	Yes	Y		3	33	1	1392	3 Master's Deg	1	4
6	No	Y		3	27	1	591	2 High School	1	1
7	No	Y		2	32	1	1005	2 Associates Dr	1	4
8	Yes	Y		3	59	1	1324	3 Bachelor's De	1	3
9	No	Y		2	30	1	1358	24 High School	1	4
10	No	Y		2	38	1	216	23 Bachelor's De	1	4
11	No	Y		3	36	1	1299	27 Bachelor's De	1	3
12	No	Y		5	35	1	809	16 Bachelor's De	1	1
13	Yes	Y		3	29	1	153	15 Associates Dr	1	4
14	No	Y		1	31	1	670	26 High School	1	1
15	No	Y		2	34	1	1346	19 Associates Dr	1	2
16	Yes	Y		4	28	0	103	24 Bachelor's De	1	3
17	No	Y		1	29	1	1389	21 Master's Deg	1	2
18	Yes	Y		5	32	1	334	5 Associates Dr	1	1

HR data



# Project Management

## Data Analyst with PowerBI



```
city_uniques = df["city"].unique()
city_uniques

array(['Jakarta Selatan', 'Jakarta Utara', 'Jakarta Barat',
      'Jakarta Timur', 'Surakarta', 'Tangerang', 'Denpasar', 'Surabaya',
      'Jakarta Pusat', 'Bogor', 'Bandung', 'Yogyakarta', 'Malang',
      'Bekasi', 'Depok', 'Semarang', 'Purwokerto', 'Ubud'], dtype=object)

monthly_order = df.groupby(["order_month", "city"])["gmv"].sum().reset_index()
monthly_order

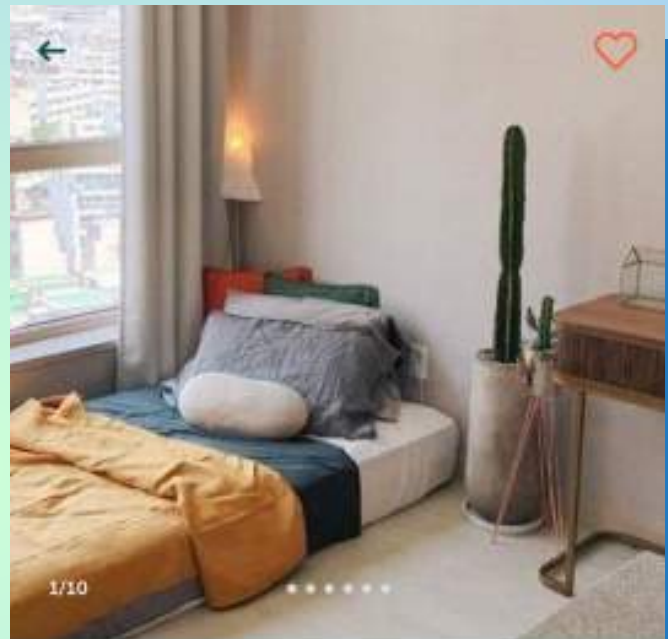
order_month  city  gmv
0  2022-07  Bandung  889340
1  2022-07  Bekasi  1502740
2  2022-07  Bogor  1509770
3  2022-07  Denpasar  767100
4  2022-07  Depok  294300
...  ...  ...
99  2022-12  Surabaya  2058390
100  2022-12  Surakarta  5920790
```

# Project Management

## Kostify(An Android App for Boarding House Rental)

**Kostify** 

*Temukan kos impianmu  
bersama kami*




### Kos Kartika

Kos Kartika Jaya Residence II  
Queesize Bedroom Kelapa Belalai  
Jakarta Barat 158GA

Campur

Rp1.420.000/bulan


#### Spesifikasi Tipe Kamar

 3.2 x 3.2 Meter

 Termasuk Listrik

#### Fasilitas Kamar

 AC

 Kasur

 Lemari

 Wifi

#### Fasilitas Kamar Mandi

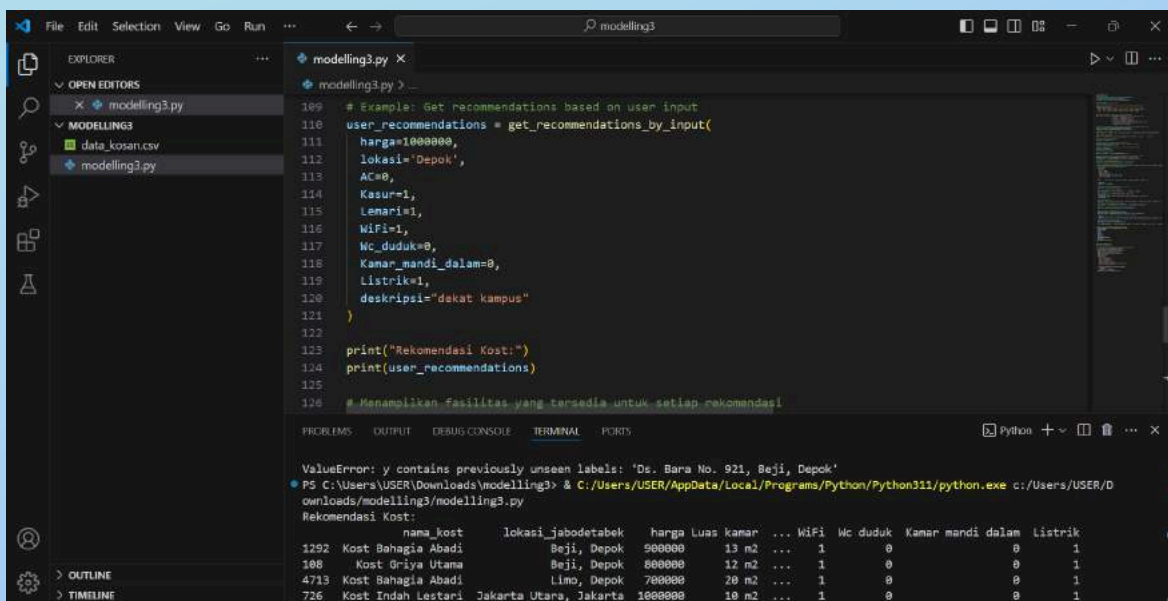
 K. Mandi Dalam

 Kloset Duduk

# Project Management

## Coding

```
1 import pandas as pd
2 import numpy as np
3 from sklearn.feature_extraction.text import TfidfVectorizer
4 from sklearn.preprocessing import LabelEncoder
5 from sklearn.metrics.pairwise import cosine_similarity
6 import tensorflow as tf
7 from tensorflow.keras import layers
8 from tensorflow.keras import Model
9
10 # Load the new dataset
11 data = pd.read_csv('data_kosan.csv')
12
13 # Custom list of Indonesian stop words (simplified for demonstration purposes)
14 indonesian_stop_words = [
15     "yang", "untuk", "dengan", "pada", "ini", "dan", "di", "ke", "dari", "adalah",
16     "itu", "atau", "oleh", "seperti", "jika", "karena", "sudah", "dalam", "akan",
17     "juga", "kami", "sangat", "hanya", "mereka", "saja", "agar", "tetapi"
18 ]
19
20 # Combine the relevant features into a single text feature
21 data['combined_features'] = data.apply(lambda row: f"{'AC' if row['AC'] else ''} "
22                                     f"{'Kasur' if row['Kasur'] else ''} "
23                                     f"{'Lemari' if row['Lemari'] else ''} "
24                                     f"{'Wifi' if row['Wifi'] else ''} "
```



```
109 # Example: Set recommendations based on user input
110 user_recommendations = get_recommendations_by_input(
111     harga=1000000,
112     lokasi='Depok',
113     AC=0,
114     Kasur=1,
115     Lemari=1,
116     Wifi=1,
117     Wc_duduk=0,
118     Kamar_mandi_dalam=0,
119     Listrik=1,
120     deskripsi="dekat kampus"
121 )
122
123 print("Rekomendasi Kost:")
124 print(user_recommendations)
125
126 # Menampilkan fasilitas yang tersedia untuk setiap rekomendasi
```

ValueError: y contains previously unseen labels: 'Ds. Bara No. 921, Beji, Depok'

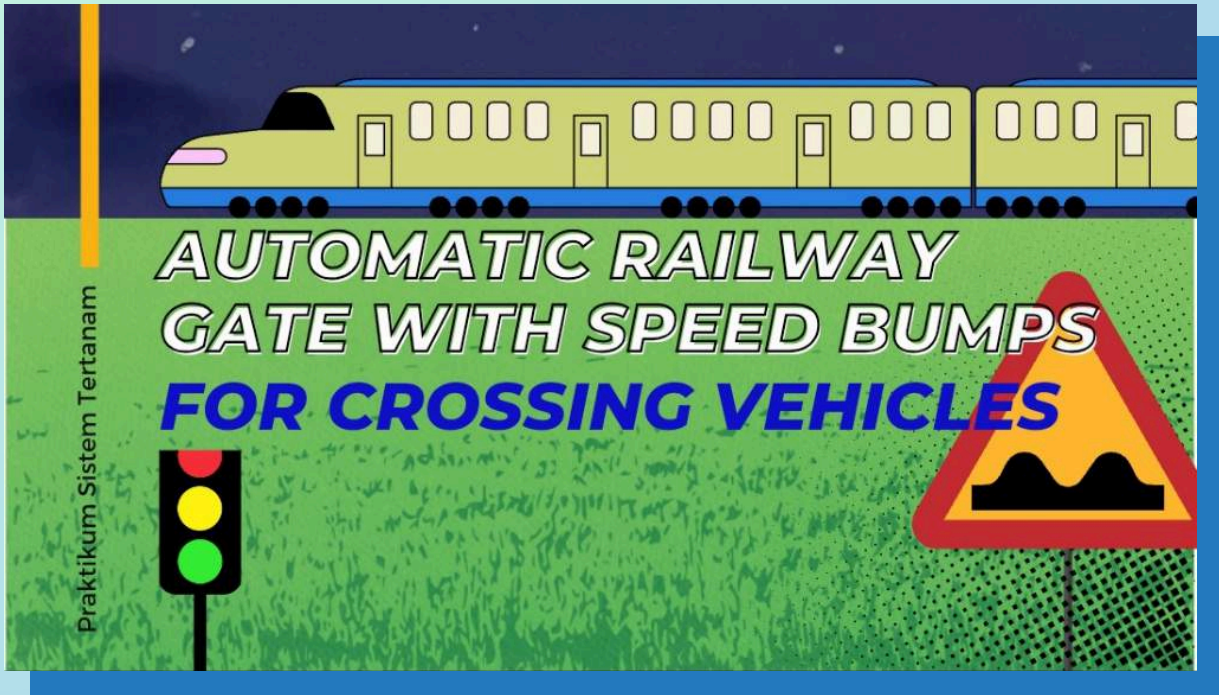
PS C:\Users\USER\Downloads\modelling3> & C:/Users/USER/AppData/Local/Programs/Python/Python311/python.exe c:/Users/USER/Downloads/modelling3/modelling3.py

Rekomendasi Kost:

	nama_kost	lokasi_jabodetabek	harga	Luas kamar	...	Wifi	Wc duduk	Kamar mandi dalam	Listrik
1292	Kost Bahagia Abadi	Beji, Depok	900000	13 m2	...	1	0	0	1
108	Kost Griya Utama	Beji, Depok	800000	12 m2	...	1	0	0	1
4713	Kost Bahagia Abadi	Limo, Depok	700000	20 m2	...	1	0	0	1
726	Kost Indah Lestari	Jakarta Utara, Jakarta	1000000	10 m2	...	1	0	0	1



# Programing Project



```
code_palang_pintu | Arduino IDE 2.2.1
File Edit Sketch Tools Help

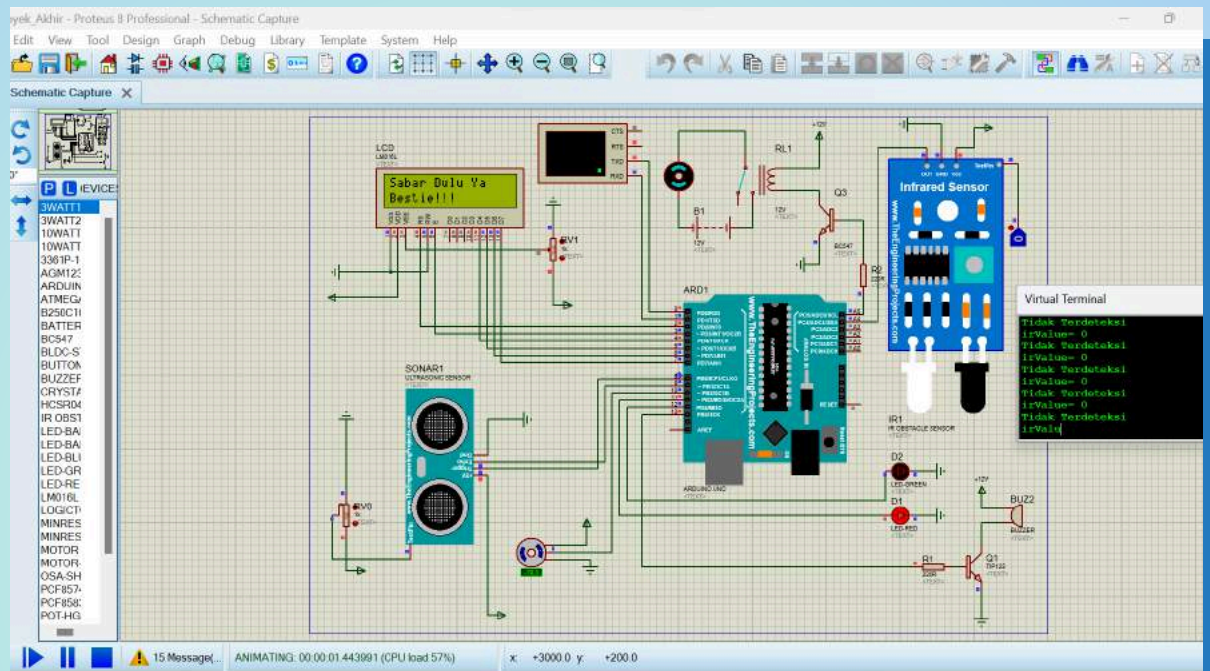
code_palang_pintu.ino
38
39
40 if (distance < 20 && !is_closed) {
41   // tutup pintu secara perlahan
42   for (int i = angle; i <= 93; i++) { // ubah batas atas menjadi 270
43     servo.write(i);
44     delay(20); // delay in milliseconds, ubah jadi lebih kecil untuk kecepatan putaran yang lebih tinggi
45   }
46   is_closed = true; // pintu tertutup sekarang
47 }
48 else if (distance > 20 && is_closed) {
49   // buka pintu setelah kereta melewati sensor
50   delay(1950); // delay selama 20 detik
51   // buka pintu secara perlahan
52   for (int i = 93; i >= angle; i--) { // ubah batas bawah menjadi 270
53     servo.write(i);
54     delay(20); // delay in milliseconds, ubah jadi lebih kecil untuk kecepatan putaran yang lebih tinggi
55   }
56 }

Output: Serial Monitor x
Not connected. Select a board and a port to connect automatically. New Line

Ln 52, Col 22 Arduino Uno on COM4 [not connected]
```

# Programming

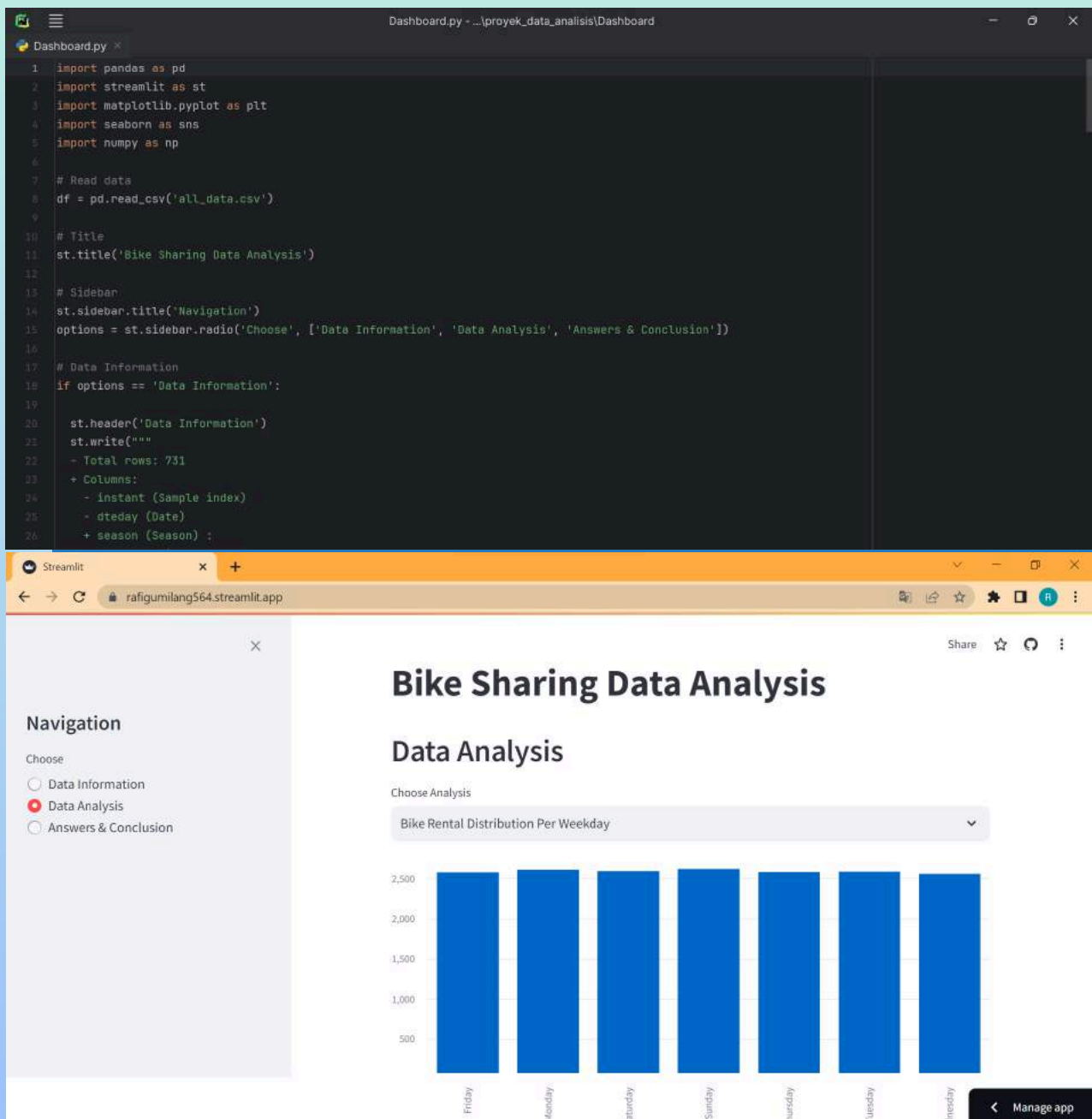
## Simulation and Miniature





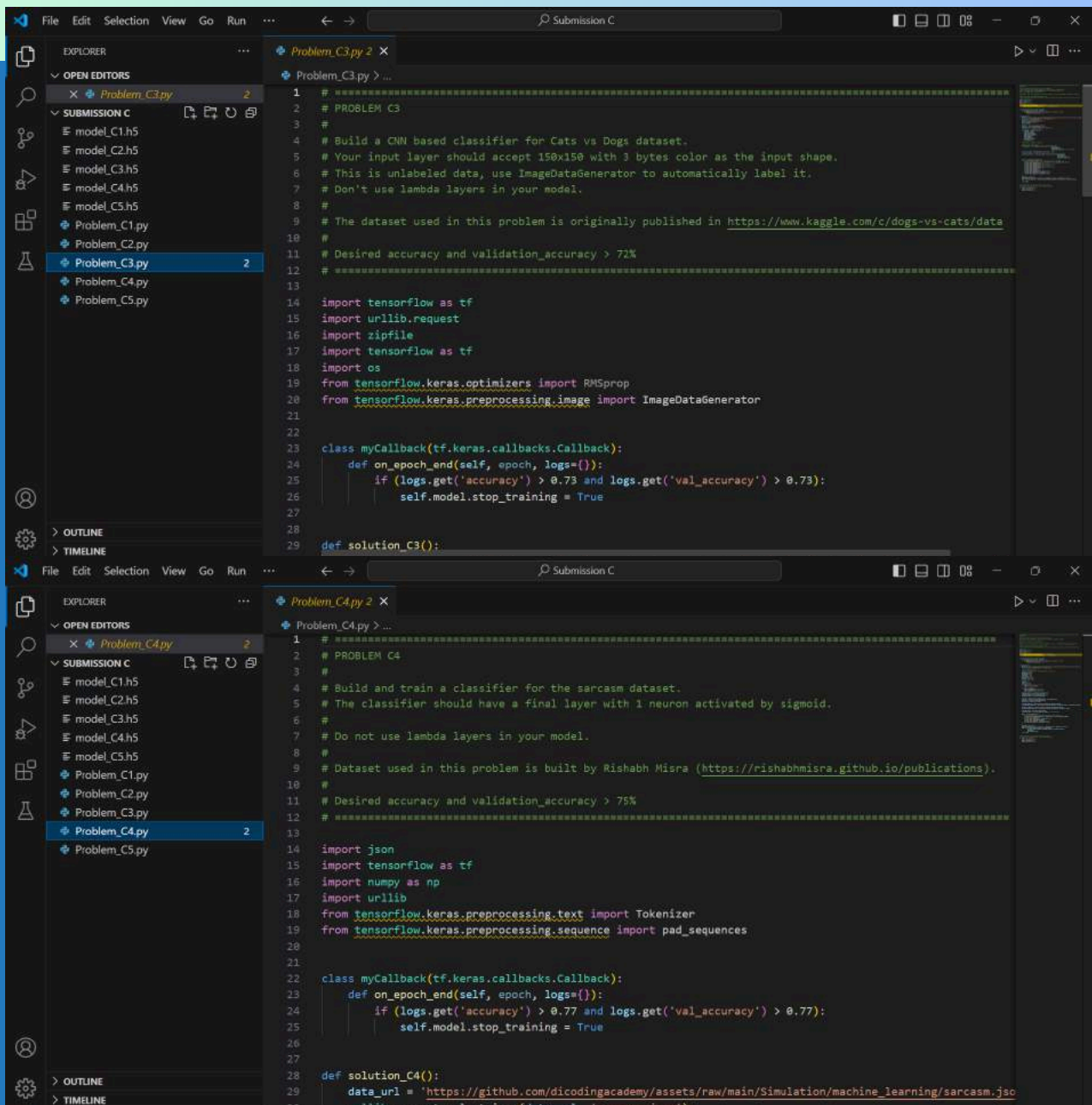
# Programing

## Data analysis



# Programing

## Simulasi Ujian TensorFlow Developer



The image displays a code editor interface with two open files, `Problem_C3.py` and `Problem_C4.py`, both under the `Submission C` project. The editor shows the Explorer, Outline, and Timeline panels on the left. The main editor area displays the code for both files.

**Problem\_C3.py**

```
1 # =====  
2 # PROBLEM C3  
3 #  
4 # Build a CNN based classifier for Cats vs Dogs dataset.  
5 # Your input layer should accept 150x150 with 3 bytes color as the input shape.  
6 # This is unlabeled data, use ImageDataGenerator to automatically label it.  
7 # Don't use lambda layers in your model.  
8 #  
9 # The dataset used in this problem is originally published in https://www.kaggle.com/c/dogs-vs-cats/data  
10 #  
11 # Desired accuracy and validation_accuracy > 72%  
12 # =====  
13  
14 import tensorflow as tf  
15 import urllib.request  
16 import zipfile  
17 import tensorflow as tf  
18 import os  
19 from tensorflow.keras.optimizers import RMSprop  
20 from tensorflow.keras.preprocessing.image import ImageDataGenerator  
21  
22  
23 class myCallback(tf.keras.callbacks.Callback):  
24     def on_epoch_end(self, epoch, logs={}):  
25         if (logs.get('accuracy') > 0.73 and logs.get('val_accuracy') > 0.73):  
26             self.model.stop_training = True  
27  
28  
29 def solution_C3():
```

**Problem\_C4.py**

```
1 # =====  
2 # PROBLEM C4  
3 #  
4 # Build and train a classifier for the sarcasm dataset.  
5 # The classifier should have a final layer with 1 neuron activated by sigmoid.  
6 #  
7 # Do not use lambda layers in your model.  
8 #  
9 # Dataset used in this problem is built by Rishabh Misra (https://rishabhmisra.github.io/publications).  
10 #  
11 # Desired accuracy and validation_accuracy > 75%  
12 # =====  
13  
14 import json  
15 import tensorflow as tf  
16 import numpy as np  
17 import urllib  
18 from tensorflow.keras.preprocessing.text import Tokenizer  
19 from tensorflow.keras.preprocessing.sequence import pad_sequences  
20  
21  
22 class myCallback(tf.keras.callbacks.Callback):  
23     def on_epoch_end(self, epoch, logs={}):  
24         if (logs.get('accuracy') > 0.77 and logs.get('val_accuracy') > 0.77):  
25             self.model.stop_training = True  
26  
27  
28  
29 def solution_C4():  
30     data_url = 'https://github.com/dicodingacademy/assets/raw/main/Simulation/machine_learning/sarcasm.json'  
31     urllib.request.urlretrieve(data_url, 'sarcasm.json')
```



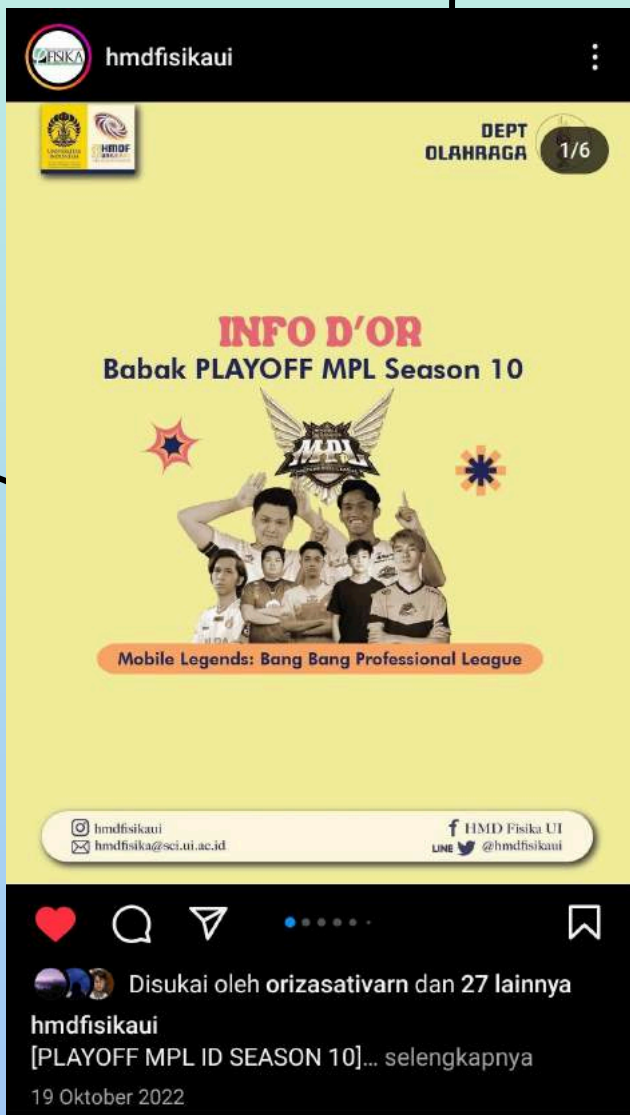
# Public Speaking

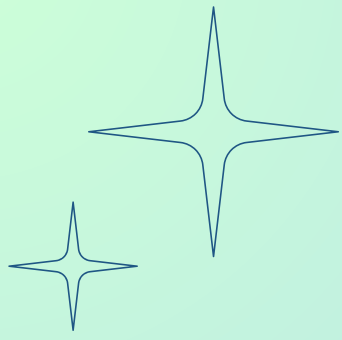
Giving orations and teaching  
at high school



# Content Writing

## Article





# Thank you!



## Let's get in touch with me through:



rafigumilang8@gmail.com



[http://www.linkedin.com/in/  
rafigumilang](http://www.linkedin.com/in/rafigumilang)



087820505374