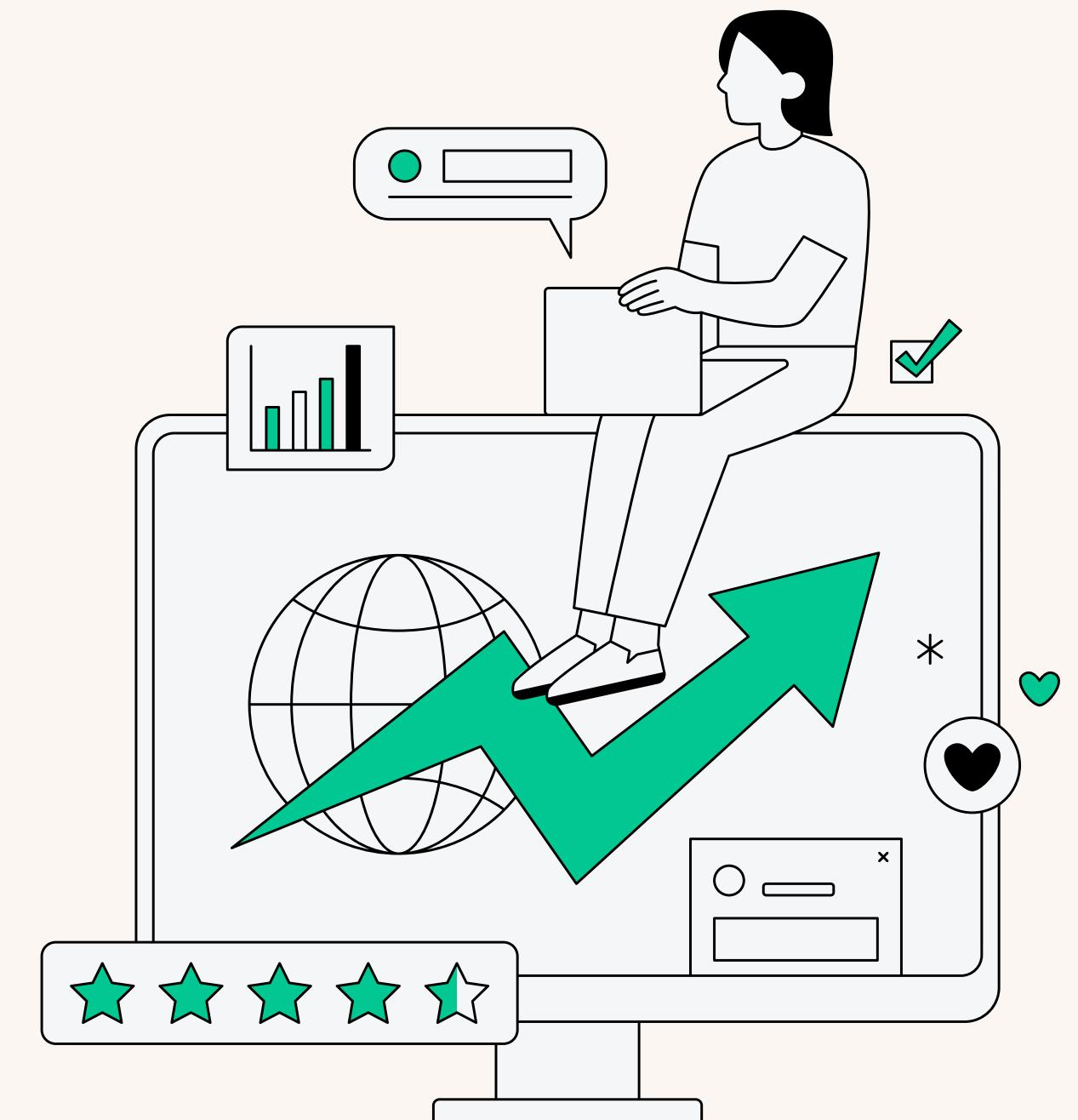


Tugas Besar Dasar Kecerdasan Artifisial Penerapan Fuzzy System

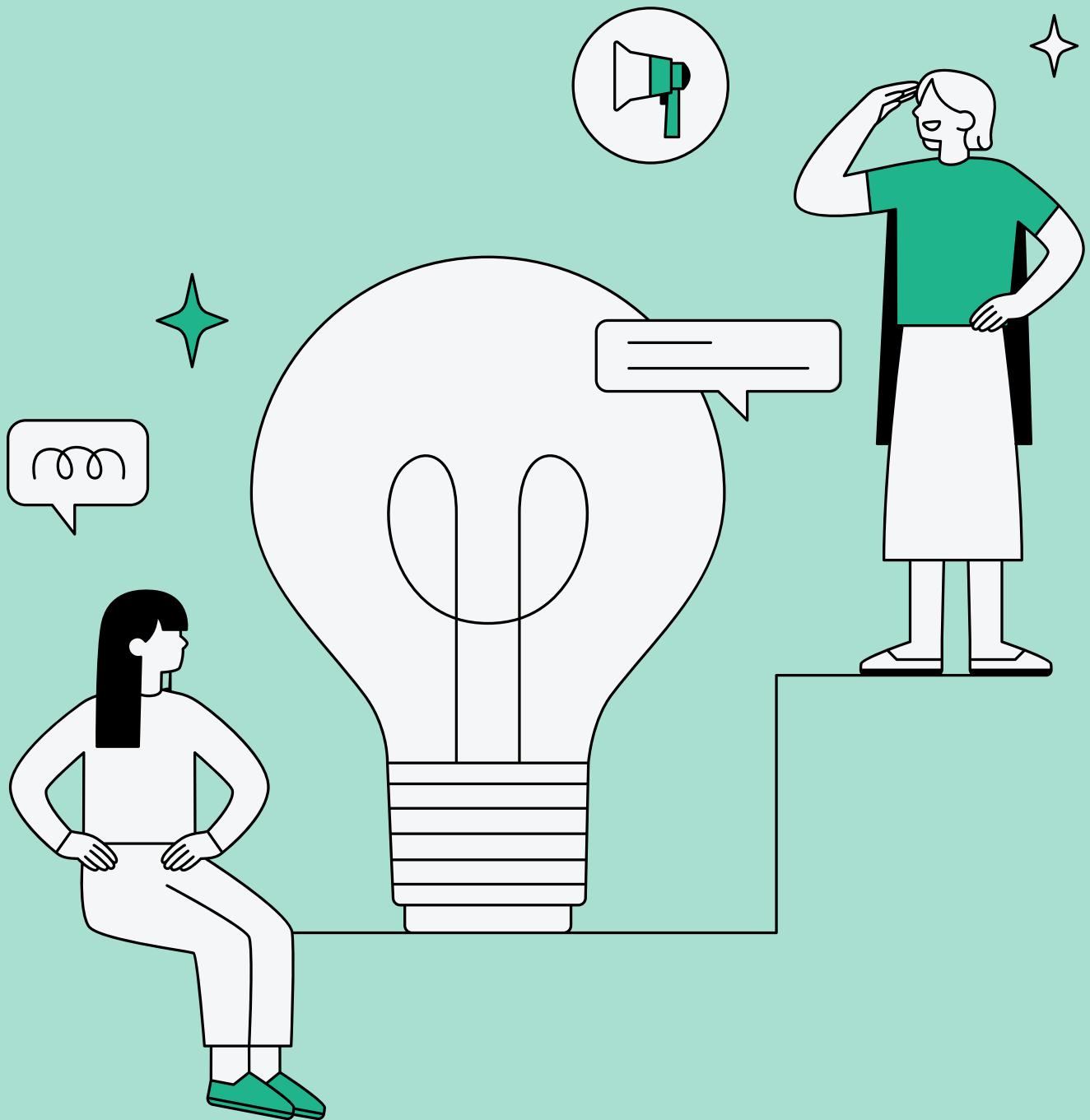
Klasifikasi Pemain NBA Musim
2023-2024

Oleh:
Muhammad Arya Dwi Kesuma - 103012300165
Ezra Mangasi Andika Sibuea - 103012330117

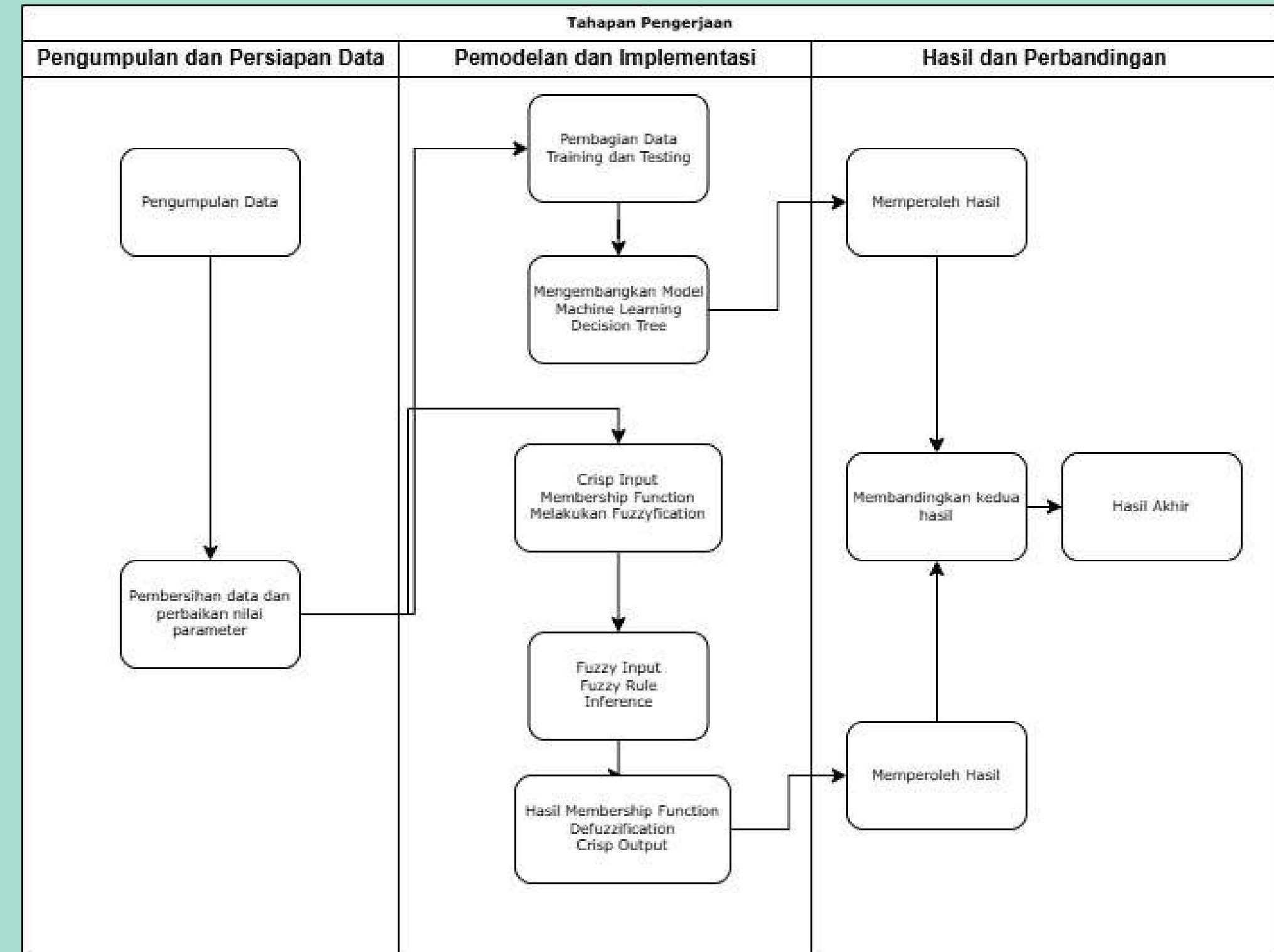


Latar Belakang

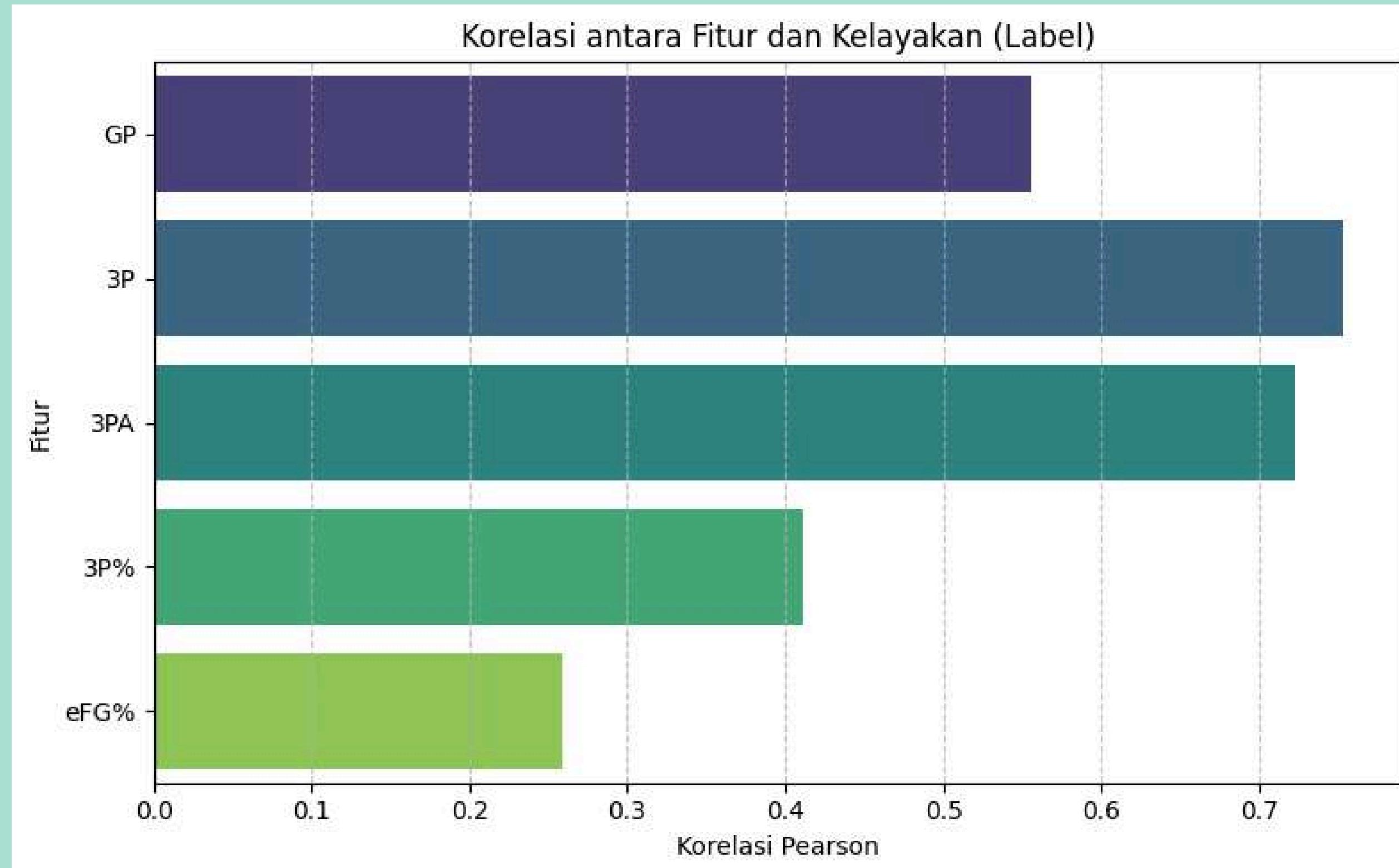
Dengan memperhatikan bahwa selama 20 tahun terakhir partisipasi masyarakat berolahraga masih rendah, lanjut Wapres, yaitu di sekitar 30 persen, maka perlu ditingkatkan kesadaran masyarakat dalam berolahraga agar tercapai angka partisipasi sebesar 40 persen pada tahun 2024, dan perlu terus ditingkatkan sehingga olahraga menjadi bagian dari gaya hidup masyarakat



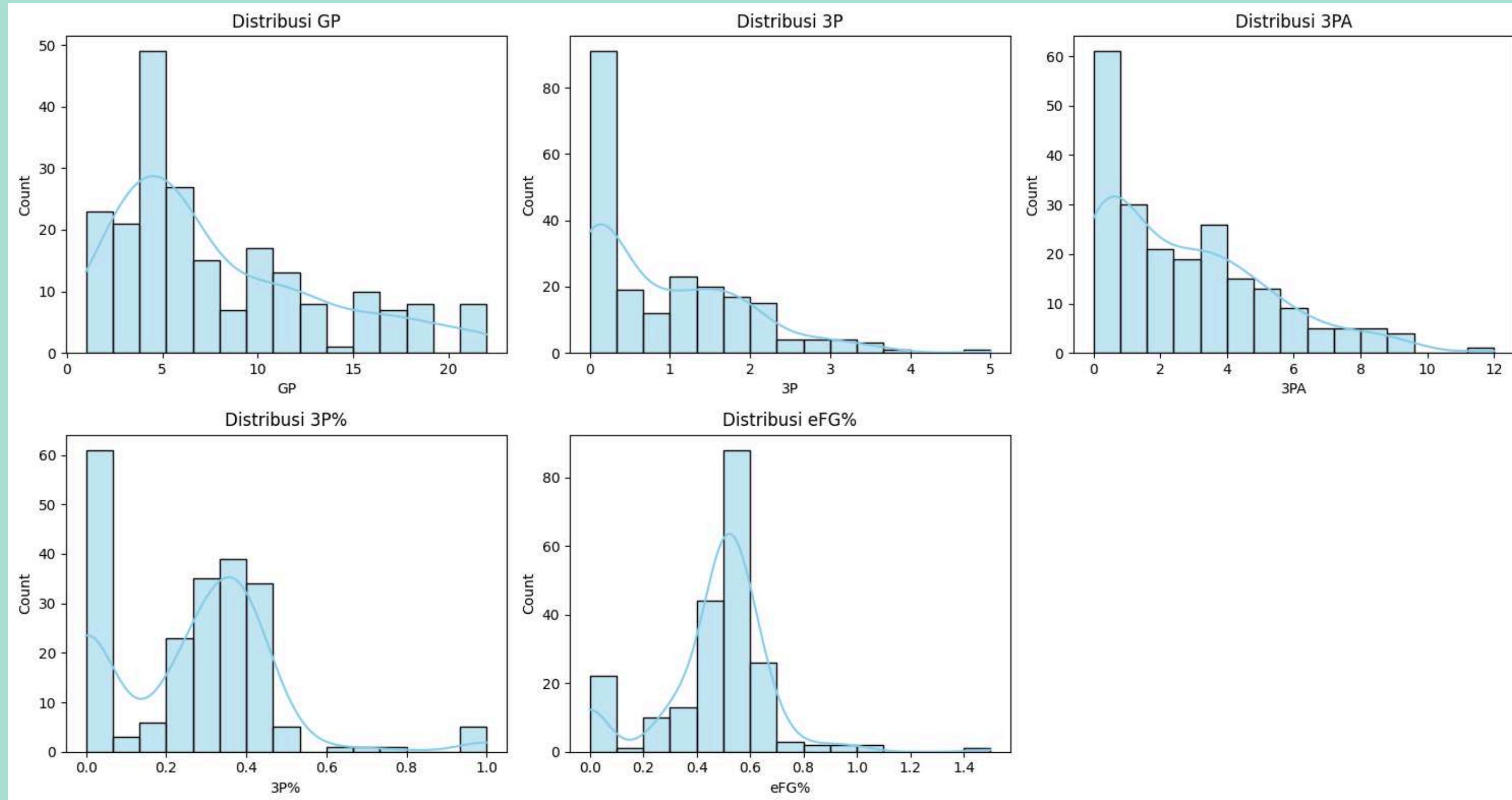
Alur Penggerjaan



Korelasi Fitur dan Label



Distribusi Fitur pada Dataset



Decision Tree

```
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, classification_report
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier

data = data.drop(columns=["Player"])

X = data.drop(columns=["Label"])
y = data["Label"]

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42, stratify=y
)

dt_model = DecisionTreeClassifier(random_state=42)
dt_model.fit(X_train, y_train)

y_pred = dt_model.predict(X_test)
print("Akurasi:", accuracy_score(y_test, y_pred))
print("Presisi:", precision_score(y_test, y_pred))
print("Recall:", recall_score(y_test, y_pred))
print("F1:", f1_score(y_test, y_pred))
print("\nClassification Report:\n", classification_report(y_test, y_pred))
```

Pembagian
Dataset dan
Simulasi

Nilai Linguistik

Atribut	Jumlah Nilai Linguistik	Jenis Linguistik
GP	3	Sedikit, Sedang Banyak
3P	3	Rendah, Sedang, tinggi
3PA	3	Rendah, Sedang, tinggi
3P%	3	Rendah, Sedang, tinggi
eFG%	3	Rendah, Sedang, tinggi
Kelayakan	2	Layak, Tidak Layak

```

GP = ctrl.Antecedent(np.arange(1, 29, 1), 'GP')
threeP = ctrl.Antecedent(np.arange(0.0, 10.1, 0.1), 'threeP')
threePA = ctrl.Antecedent(np.arange(0.0, 15.1, 0.1), 'threePA')
threeP_pct = ctrl.Antecedent(np.arange(0.0, 1.01, 0.01), 'threeP_pct')
efg = ctrl.Antecedent(np.arange(0.00, 1.51, 0.01), 'efg')
kelayakan = ctrl.Consequent(np.arange(0, 1.1, 0.01), 'kelayakan')

GP['sedikit'] = fuzz.trimf(GP.universe, [1, 1, 8])
GP['cukup'] = fuzz.trimf(GP.universe, [1, 8, 14])
GP['banyak'] = fuzz.trimf(GP.universe, [8, 28, 28])
GP.view()

threeP['rendah'] = fuzz.trimf(threeP.universe, [0.0, 0.0, 5.0])
threeP['sedang'] = fuzz.trimf(threeP.universe, [0.0, 5.0, 10.0])
threeP['tinggi'] = fuzz.trimf(threeP.universe, [5.0, 10.0, 10.0])
threeP.view()

threePA['sedikit'] = fuzz.trimf(threePA.universe, [0.0, 0.0, 7.5])
threePA['sedang'] = fuzz.trimf(threePA.universe, [0.0, 7.5, 15.0])
threePA['banyak'] = fuzz.trimf(threePA.universe, [7.5, 15.0, 15.0])
threePA.view()

threeP_pct['rendah'] = fuzz.trimf(threeP_pct.universe, [0.0, 0.0, 0.5])
threeP_pct['cukup'] = fuzz.trimf(threeP_pct.universe, [0.0, 0.5, 1.0])
threeP_pct['tinggi'] = fuzz.trimf(threeP_pct.universe, [0.5, 1.0, 1.0])
threeP_pct.view()

efg['rendah'] = fuzz.trimf(efg.universe, [0.0, 0.0, 0.75])
efg['cukup'] = fuzz.trimf(efg.universe, [0.0, 0.75, 1.5])
efg['tinggi'] = fuzz.trimf(efg.universe, [0.75, 1.5, 1.5])
efg.view()

kelayakan['tidak'] = fuzz.trimf(kelayakan.universe, [0.0, 0.0, 0.5])
kelayakan['layak'] = fuzz.trimf(kelayakan.universe, [0.5, 1.0, 1.0])
kelayakan.view()

```

Fungsi Keanggotaan dan Batas Nilai Linguistik

Variable dan Membership Function



Fuzzy Rule

```
rules = [
    ctrl.Rule(threeP['rendah'], kelayakan['tidak']),
    ctrl.Rule(threeP['sedang'] & GP['sedikit'], kelayakan['tidak']),
    ctrl.Rule(threeP['sedang'] & GP['banyak'], kelayakan['layak']),
    ctrl.Rule(threeP['tinggi'] & GP['sedikit'] & threeP['rendah'] & threePA['sedikit'], kelayakan['tidak']),
    ctrl.Rule(threeP['tinggi'] & GP['sedikit'] & threeP['rendah'] & threePA['sedang'], kelayakan['layak']),
    ctrl.Rule(threeP['tinggi'] & GP['sedikit'] & threeP['rendah'] & threePA['banyak'], kelayakan['tidak']),
    ctrl.Rule(threeP['tinggi'] & GP['sedikit'] & threeP['tinggi'], kelayakan['layak']),
    ctrl.Rule(threeP['tinggi'] & GP['banyak'] & efg['rendah'], kelayakan['tidak']),
    ctrl.Rule(threeP['tinggi'] & GP['banyak'] & efg['cukup'] & GP['sedikit'], kelayakan['tidak']),
    ctrl.Rule(threeP['tinggi'] & GP['banyak'] & efg['cukup'] & GP['banyak'] & threeP_pct['rendah'], kelayakan['tidak']),
    ctrl.Rule(threeP['tinggi'] & GP['banyak'] & efg['cukup'] & GP['banyak'] & threeP_pct['cukup'], kelayakan['layak']),
    ctrl.Rule(threeP['tinggi'] & GP['banyak'] & efg['tinggi'], kelayakan['layak'])
]
```

Perhitungan Fuzzy

```
fuzzy_preds = []
for i, row in nba_df.iterrows():
    try:
        simulator.input['threeP'] = row['3P']
        simulator.input['threePA'] = row['3PA']
        simulator.input['threeP_pct'] = row['3P%']
        simulator.input['efg'] = row['eFG%']
        simulator.input['GP'] = row['GP']
        simulator.compute()
        score = simulator.output['kelayakan']
        pred = 1 if score >= 0.5 else 0 #Output Kelayakan harus bernilai biner
    except Exception as e:
        pred = 0
    fuzzy_preds.append(pred)

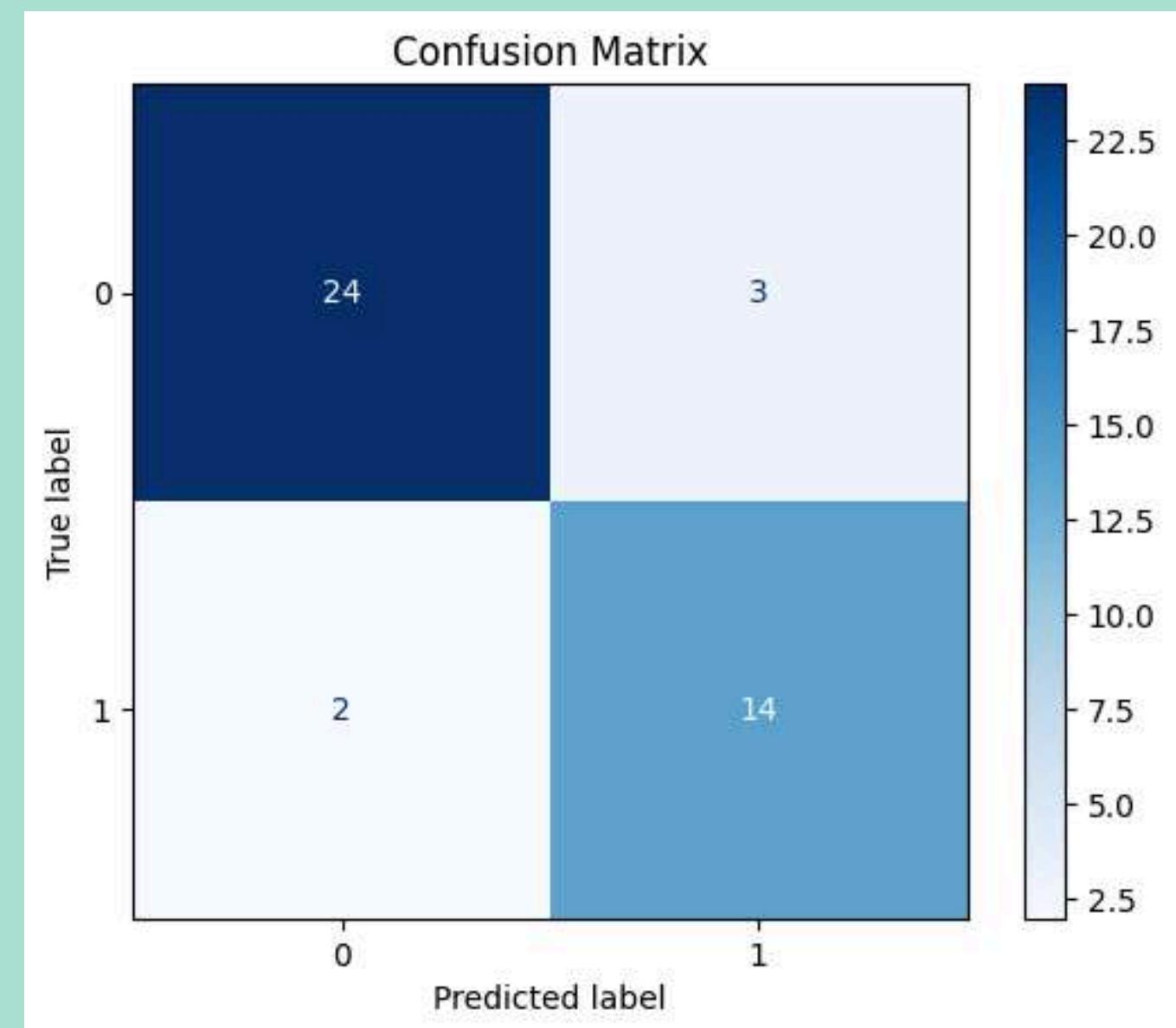
nba_df['Fuzzy_Pred'] = fuzzy_preds

print("Akurasi:", accuracy_score(nba_df['Label'], nba_df['Fuzzy_Pred']))
print("Presisi:", precision_score(nba_df['Label'], nba_df['Fuzzy_Pred']))
print("Recall:", recall_score(nba_df['Label'], nba_df['Fuzzy_Pred']))
print("F1:", f1_score(nba_df['Label'], nba_df['Fuzzy_Pred']))
print("\nClassification Report:\n", classification_report(nba_df['Label'], nba_df['Fuzzy_Pred']))
```

Hasil Decision Tree

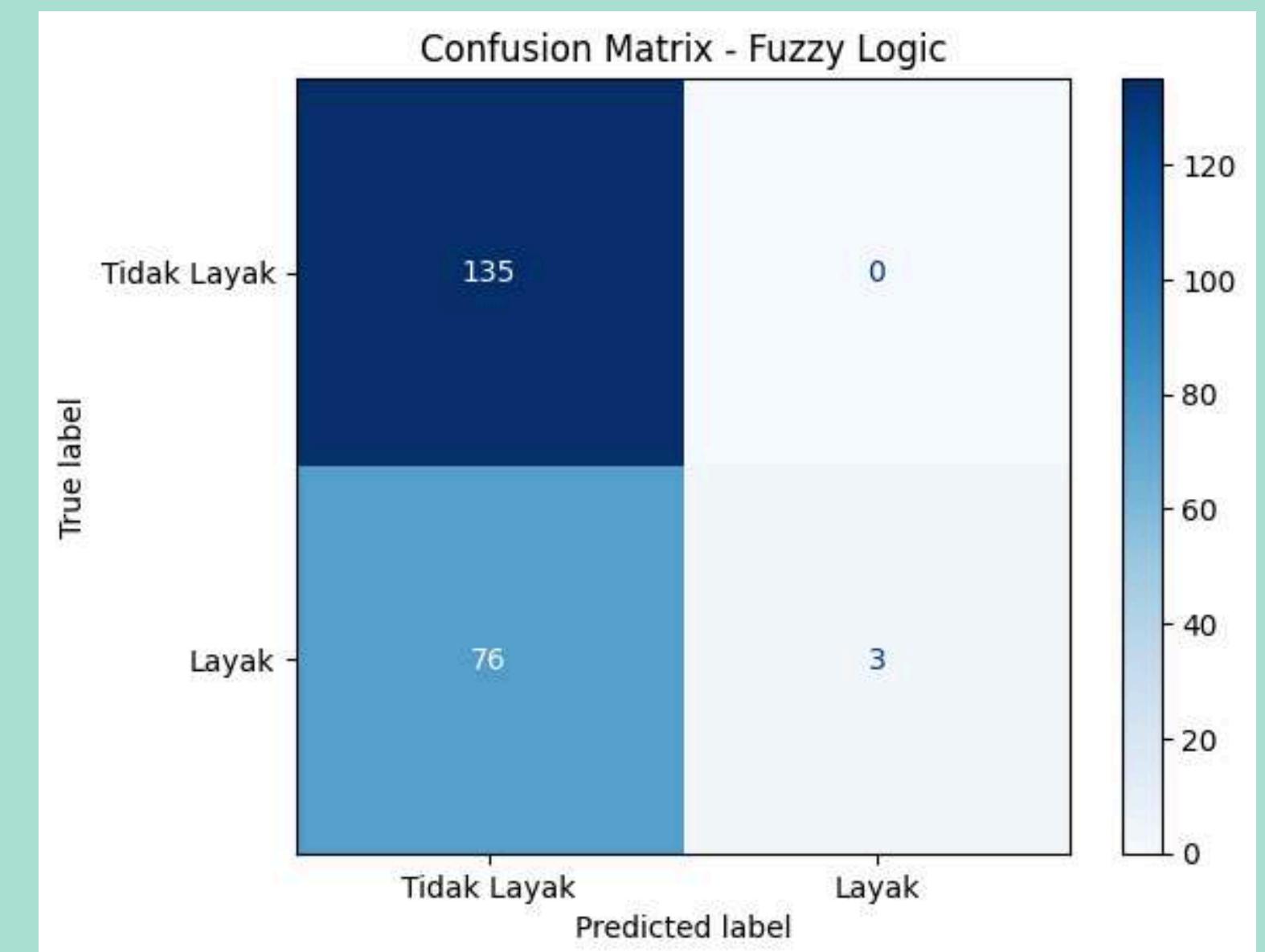
Akurasi: 0.8837209302325582
Presisi: 0.8235294117647058
Recall: 0.875
F1: 0.8484848484848485

Classification Report:				
	precision	recall	f1-score	support
0	0.92	0.89	0.91	27
1	0.82	0.88	0.85	16
accuracy			0.88	43
macro avg	0.87	0.88	0.88	43
weighted avg	0.89	0.88	0.88	43



Hasil Fuzzy Logic

	precision	recall	f1-score	support
0	0.64	1.00	0.78	135
1	1.00	0.04	0.07	79
accuracy			0.64	214
macro avg	0.82	0.52	0.43	214
weighted avg	0.77	0.64	0.52	214
Akurasi Fuzzy Mamdani:	64.49 %			
Presisi Fuzzy Mamdani:	100.0 %			
Recall Score Fuzzy Mamdani:	3.8 %			
F1 Score Fuzzy Mamdani:	7.32 %			



Perbandingan

Metrik	Decision Tree	Fuzzy Logic
Accuracy	88%	64%
Precision 0/Tidak Layak	0.92	0.64
Recall 0/Tidak Layak	0.89	1.00
F1 0/Tidak Layak	0.91	0.78
Precision 1/Layak	0.82	1.00
Recall 1/Layak	0.88	0.04
F1 1/Layak	0.85	0.07



Hasil Akhir

1. Decision Tree unggul secara keseluruhan
 2. Fuzzy Logic hanya unggul pada Recall kelas 0
 3. Ketimpangan sensitivitas pada Fuzzy Logic
- 



Kesimpulan

Model Decision Tree memiliki performa klasifikasi yang lebih seimbang dan akurat dibandingkan dengan model Fuzzy Logic. Meskipun Fuzzy Logic menunjukkan kemampuan sangat baik dalam mengidentifikasi pemain bukan penembak 3 poin, kemampuannya dalam mengenali pemain layak masih sangat terbatas. Hal ini menunjukkan bahwa rule-based fuzzy system lebih cocok untuk interpretasi dan penalaran yang transparan, tetapi kurang efektif untuk prediksi akurat tanpa tuning lebih lanjut. Decision Tree lebih akurat karena dia data-driven, otomatis, dan mengoptimalkan pemisahan kelas. Fuzzy logic lebih bersifat penalaran manusia dan tidak mengoptimalkan aturan dari data secara langsung.

