

# **Skill Gap Analysis & Prediction**

**A PROJECT REPORT**

*Submitted by*

**DEBASHISH PANDA**

**PRIYADARSANI PANDA**

*in partial fulfillment for the award of the*

*degree of*

**MASTER**

*in*

**COMPUTER APPLICATIONS**



**Centurion  
UNIVERSITY**

*Shaping Lives...  
Empowering Communities...*

**SCHOOL OF APPLIED SCIENCE**

**BHUBANESWAR CAMPUS**

**CENTURION UNIVERSITY OF TECHNOLOGY AND MANAGEMENT  
ODISHA**

**NOV 2026**

DEPARTMENT OF MASTER IN COMPUTER APPLICATIONS  
SCHOOL OF APPLIED SCIENCE  
BHUBANESWAR CAMPUS

**BONAFIDE CERTIFICATE**

Certified that this project report **Skill Gap Analysis & Prediction** is the bonafide work of “DEBASHISH PANDA”, “PRIYADARSANI PANDA” who carried out the project work under my supervision. This is to further certify to the best of my knowledge, that this project has not been carried out earlier in this institute and the university.

<<Signature of the Supervisor>>  
**SIGNATURE**

**(Prof. Saroj Kumar Sahoo)**  
**Professor of Computer Science and Engineering**

*Certified that the above-mentioned project has been duly carried out as per the norms of the college and statutes of the university.*

**SIGNATURE**  
**(Prof. Rakesh Kumar Ray)**  
**HEAD OF THE DEPARTMENT**  
**Professor of Computer Science and Engineering**

DEPARTMENT SEAL

## **DECLARATION**

I hereby declare that the project entitled “**Skill Gap Analysis & Prediction**” submitted for the “Major Project” of 3rd semester MCA in Computer Science and Engineering is my original work and the project has not formed the basis for the award of any Degree or any other similar titles in any other University / Institute.

**Name of the Student: Debashish Panda  
Priyadarsani Panda**

**Signature of the Student:**

**Registration No:240720100091  
240720100059**

**Place: CUTM, BHUBNESWAR CAMPUS**

**Date:**

## **ACKNOWLEDGEMENT**

I wish to express my profound and sincere gratitude to Mrs. Maiyetree Mohanty, Department of Computer Science and Engineering, SoET, Bhubaneswar Campus, who guided me into the intricacies of this project nonchalantly with matchless magnanimity.

I thank Prof. Rakesh Kumar Ray, Head of the Department of Master in Computer Applications, SoET, Bhubaneswar Campus and Prof. Sujata Chakravarthy, Dean, School of Engineering and Technology, Bhubaneswar Campus for extending their support during Course of this investigation.

I am highly grateful to Mrs. Maiyetree Mohanty who evinced keen interest and invaluable support in the progress and successful completion of my project work.

I am indebted to Mr. Trilochan Sahoo for their constant encouragement, co-operation and help. Words of gratitude are not enough to describe the accommodation and fortitude which they have shown throughout my endeavor.

**Name of the Student: Debashish Panda  
Priyadarsani Panda**

**Signature of the Student:**

**Registration No:240720100091  
240720100059**

**Place: CUTM, BHUBNESWAR CAMPUS**

## **TABLE OF CONTENTS**

<b>CHAPTER NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
	<b>CERTIFICATE</b>	<b>i</b>
	<b>DECLARATION</b>	<b>ii</b>
	<b>ACKNOWLEDGEMENT</b>	<b>iii</b>
<b>1</b>	<b>ABSTRACT</b>	<b>01</b>
<b>2</b>	<b>INTRODUCTION</b>	<b>02</b>
<b>3</b>	<b>INTRODUCTION ABOUT DATASET</b>	<b>03</b>
<b>4</b>	<b>LIBRARIES USED</b>	<b>04-05</b>
<b>5</b>	<b>CLASSIFICATION AND VISUALIZATION</b>	<b>06-12</b>
<b>6</b>	<b>SPLITTING AND MODEL IMPLEMENTATION</b>	<b>13</b>
<b>7</b>	<b>ACCURACY OF ALL MODELS</b>	<b>13</b>
<b>8</b>	<b>BUILDING A USERINPUT PREDICTIVE SYSTEM</b>	<b>14-15</b>
<b>9</b>	<b>CONCLUSION</b>	<b>16</b>
<b>10</b>	<b>FUTURE SCOPE</b>	<b>17</b>
<b>11</b>	<b>REFERENCE</b>	<b>18</b>

# **1. Abstract**

This report presents the major-project assigned to Third semester MCA student in the field of Machine Learning for Predictive Analysis, under the guidance of the Department of Computer Science and Engineering. Diabetes is one of the fastest-growing health challenges in the world, affecting millions of people across all demographics. Early diagnosis and intervention can significantly help in managing the disease and preventing severe complications such as cardiovascular issues, kidney failure, and nerve damage. However, due to the chronic nature of the disease and the lack of resources for continuous patient monitoring, early detection remains a challenge. In this project, we have explored and implemented machine learning models to predict diabetes based on various health attributes of patients. Using publicly available datasets from Kaggle, we applied algorithms such as logistic regression, decision trees, and random forest. The performance of these models was evaluated using accuracy scores, confusion matrices, and cross-validation techniques. This project highlights the potential of machine learning in developing efficient and accessible diagnostic tools for diabetes prediction, which can support healthcare providers and help in early intervention for at-risk individuals.

## **2.Introduction**

According to the World Health Organization, diabetes is one of the leading causes of death and disability worldwide, with millions of new cases diagnosed every year. The burden of diabetes has been increasing rapidly across both developed and developing nations due to lifestyle changes, unhealthy diets, and genetic factors. Numerous studies have been conducted to identify the key risk factors and to develop reliable methods for predicting the onset of diabetes in individuals. Diabetes is often called a silent condition, as it may remain undiagnosed for years while gradually causing serious health issues. The early diagnosis of diabetes plays a crucial role in taking preventive actions and adopting lifestyle modifications to manage the condition effectively and reduce complications. This project aims to predict the likelihood of diabetes by analyzing patient data using machine learning algorithms that classify whether an individual is diabetic or not. One of the major challenges in diabetes management is its early detection, especially among asymptomatic individuals. Although there are several medical tests available for diabetes detection, they can be time-consuming, expensive, or inaccessible in remote areas. With the abundance of healthcare data available today, machine learning techniques offer an efficient way to identify hidden patterns that contribute to the onset of diabetes. In this project, we utilize various algorithms to build predictive models that can assist in early detection and help medical professionals provide timely intervention. This approach not only supports better health outcomes but also has the potential to significantly reduce the economic and clinical burden of diabetes on society.

### **3. INFORMATION ABOUT DATASET:**

- Data used: -Diabetes Dataset
- Size: -68 MB
- SOURCE OF COLLECTION  
KAGGLE  
<https://www.kaggle.com/datasets/mathchi/diabetes-data-set>
- TOTAL SIZES 1 FILE



## 4. Libraries used

- **Pandas**
- **NumPy**
- **Matplotlib**
- **Sklearn**



Pandas is a popular Python library for data analysis. It is not directly related to Machine Learning. As we know that the dataset must be prepared before training. In this case, Pandas comes handy as it was developed specifically for data extraction and preparation. It provides high-level data structures and wide variety tools for data analysis. It provides many inbuilt methods for grouping, combining and filtering data.



NumPy is a very popular python library for large multi-dimensional array and matrix processing, with the help of a large collection of high-level mathematical functions. It is very useful for fundamental scientific computations in Machine Learning. It is particularly useful for linear algebra, Fouriertransform, and random number capabilities.



Matplotlib is a very popular Python library for data visualization. Like Pandas, it is not directly related to Machine Learning. It particularly comes in handy when a programmer wants to visualize the patterns in the data. It is a 2D plotting library used for creating 2D graphs and plots. A module named pyplot makes it easy for programmers for plotting as it provides features to control line styles, font properties, formatting axes, etc. It provides various kinds of graphs and plots for data visualization, viz., histogram, error charts, bar charts etc.



Scikit-learn is one of the most popular ML libraries for classical ML algorithms. It is built on top of two basic Python libraries, viz., NumPy and SciPy. Scikit-learn supports most of the supervised and unsupervised learning algorithms. Scikit-learn can also be used for data-mining and data-analysis, which makes it a great tool who is starting out with ML.

# 5.CLASSIFICATION AND VISUALIZATION PART

SCREEN SHOT DATA SET:-

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC
1	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target															
2	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1															
3	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1															
4	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1															
5	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1															
6	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1															
7	57	1	0	140	192	0	1	148	0	0.4	1	0	1	1															
8	56	0	1	140	294	0	0	153	0	1.3	1	0	2	1															
9	44	1	1	120	263	0	1	173	0	0	2	0	3	1															
10	52	1	2	172	199	1	1	162	0	0.5	2	0	3	1															
11	57	1	2	150	168	0	1	174	0	1.6	2	0	2	1															
12	54	1	0	140	239	0	1	160	0	1.2	2	0	2	1															
13	48	0	2	130	275	0	1	139	0	0.2	2	0	2	1															
14	49	1	1	130	266	0	1	171	0	0.6	2	0	2	1															
15	64	1	3	110	211	0	0	144	1	1.8	1	0	2	1															
16	58	0	3	150	283	1	0	162	0	1	2	0	2	1															
17	50	0	2	120	219	0	1	158	0	1.6	1	0	2	1															
18	58	0	2	120	340	0	1	172	0	0	2	0	2	1															
19	66	0	3	150	226	0	1	114	0	2.6	0	0	2	1															
20	43	1	0	150	247	0	1	171	0	1.5	2	0	2	1															
21	69	0	3	140	239	0	1	151	0	1.8	2	2	2	1															
22	59	1	0	135	234	0	1	161	0	0.5	1	0	3	1															
23	44	1	2	130	233	0	1	179	1	0.4	2	0	2	1															
24	42	1	0	140	226	0	1	178	0	0	2	0	2	1															
25	61	1	2	150	243	1	1	137	1	1	1	0	2	1															
26	40	1	3	140	199	0	1	178	1	1.4	2	0	3	1															
27	71	0	1	160	302	0	1	162	0	0.4	2	2	2	1															
28	59	1	2	150	212	1	1	157	0	1.6	2	0	2	1															
29	51	1	2	110	175	0	1	123	0	0.6	2	0	2	1															
30	65	0	2	140	417	1	0	157	0	0.8	2	1	2	1															
31	53	1	2	130	197	1	0	152	0	1.2	0	0	2	1															
32	41	0	1	105	198	0	1	168	0	0	2	1	2	1															
33	65	1	0	120	177	0	1	140	0	0.4	2	0	3	1															
34	44	1	1	130	219	0	0	188	0	0	2	0	2	1															
35	54	1	2	125	273	0	0	152	0	0.5	0	1	2	1															
36	51	1	3	125	213	0	0	125	1	1.4	2	1	2	1															

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn import datasets
import streamlit as st
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
from sklearn.preprocessing import MultiLabelBinarizer, OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import MultiLabelBinarizer, OneHotEncoder
import logging
import io
import re

```

```

import pandas as pd
df = pd.read_excel('/content/Final_Updated_DMA_DATASET_Indian_Names.xlsx')
X = df.drop("job_role_aspiration", axis=1)
y = df["job_role_aspiration"]

# Data check karne ke liye print karein
print("----- FEATURES (X) -----")
print(X.head())

print("\n----- TARGET (y) -----")
print(y.head())

```

```

----- FEATURES (X) -----

```

	name	email_id	year	\
0	Sanjana Nair	sanjananair@pagesperso-orange.fr	3rd Year	
1	Ananya Verma	ananyaverma@globo.com	2nd Year	
2	Rohan Joshi	rohanjoshi@dmoz.org	Final Year	
3	Vikram Gupta	vikramgupta@mysql.com	1st Year_X	
4	Ayan Sayyed	ayan.sayyed@gmail.com	3rd Year	

	current_course	technical_skills	\
0	M.Tech AI	Cybersecurity, Cloud Computing_X	
1	B.Tech IT	Python, Java, Cloud Computing, SQL	
2	B.Sc Data Science	Machine Learning, Cybersecurity, Java, SQL_X	
3	BBA	Cloud Computing, Blockchain, Java	
4	B.Tech	Data Analysis, Cloud Computing_X	

	programming_languages	rating	soft_skills	\
0	Java, C#, R	2.930187	Teamwork, Leadership, Communication	
1	Java, JavaScript	2.912444	Communication, Time Management	
2	R	2.044163	Teamwork, Communication, Adaptability	
3	C#, Python	2.112412	Problem-Solving, Teamwork, Time Management	
4	C++	1.002671	Communication, Teamwork, Time Management	

```
df.head()
```

Python

	name	email_id	year	current_course	technical_skills	programming_languages	rating	soft_skills	rating.1	projects	job_role_aspiration	challenges_faced	career_support_needed	preferred_learning_method
0	Sanjana Nair	sanjananair@pagesperso-orange.fr	3rd Year	M.Tech AI	Cybersecurity, Cloud Computing, X	Java, C#, R	2.930187	Teamwork, Leadership, Communication	4.052759	Yes	Software Developer	Limited guidance	Mentorship programs	Online
1	Ananya Verma	ananyaverma@globo.com	2nd Year	B.Tech IT	Python, Java, Cloud Computing, SQL	Java, JavaScript	2.912444	Communication, Time Management	2.970611	Yes	AI Engineer_X	Lack of experience_X	Technical skill training_X	Bootcamp_X
2	Rohan Joshi	rohanjoshi@dmoz.org	Final Year	B.Sc Data Science	Machine Learning, Cybersecurity, Java, SQL_X		2.044163	Teamwork, Communication, Adaptability	0.962405	Yes	Cybersecurity Expert	Interview anxiety	Mentorship programs	Self-study
3	Vikram Gupta	vikramgupta@mysql.com	1st Year_X	BBA	Cloud Computing, Blockchain, Java	C#, Python	2.112412	Problem-Solving, Teamwork, Time Management	2.856560	Yes	Cybersecurity Expert	No networking opportunities	Internship opportunities	Bootcamp
4	Ayan Sayyed	ayansayyed@gmail.com	3rd Year	B.Tech	Data Analysis, Cloud Computing, X	C++	1.002671	Communication, Teamwork, Time Management	4.171992	No	AI Engineer	Difficulty in resume building	Mentorship programs	Online

```
df.tail()
```

Python

	name	email_id	year	current_course	technical_skills	programming_languages	rating	soft_skills	rating.1	projects	job_role_aspiration	challenges_faced	career_support_needed	preferred_learning_meth
2495	Arjun Verma	arjunverma@over-blog.com	1st Year	BBA	Blockchain, Java, Cybersecurity, SQL	JavaScript	1.033406	Leadership, Teamwork	1.024478	No	Data Scientist_X	Difficulty in resume building	Internship opportunities	College curricul
2496	Rohan Nair	rohanair@nymag.com	1st Year	MBA	Python, Blockchain	R, JavaScript, Java	2.098272	Communication, Problem-Solving, Time Management	4.985840	Yes	Cloud Architect	Limited guidance	Mentorship programs	College curricul
2497	Karthik Gupta_X	karthikgupta@friendfeed.com_X	3rd Year	B.Sc Data Science	Blockchain, Python, Cloud Computing	Python	2.073156	Teamwork, Problem-Solving	4.038051	Yes	AI Engineer	Lack of experience	Resume workshops	College curricul
2498	Priya Joshi	priyajoshi@geocities.jp	Final Year	MBA_X	Machine Learning, Java	C++, Java	5.078693	Teamwork, Time Management, Adaptability	1.048380	Yes	AI Engineer	Difficulty in resume building	Resume workshops	Bootcan
2499	Rohan Verma	rohanverma@squarespace.com_X	Final Year	B.Tech CSE	Cybersecurity, Python, SQL	Java, R_X	3.050335	Leadership, Problem-Solving, Time Management	0.929470	No	Software Developer	Interview anxiety	Technical skill training	Onli

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 2500 entries, 0 to 2499  
Data columns (total 14 columns):  
#   Column                                Non-Null Count  Dtype  
---  ---                                -  
0   name                                2500 non-null   object  
1   email_id                           2500 non-null   object  
2   year                               2500 non-null   object  
3   current_course                     2500 non-null   object  
4   technical_skills                   2500 non-null   object  
5   programming_languages              2500 non-null   object  
6   rating                             2500 non-null   float64  
7   soft_skills                        2500 non-null   object  
8   rating.1                           2500 non-null   float64  
9   projects                           2500 non-null   object  
10  job_role_aspiration                2500 non-null   object  
11  challenges_faced                   2500 non-null   object  
12  career_support_needed              2500 non-null   object  
13  preferred_learning_method          2500 non-null   object  
dtypes: float64(2), object(12)  
memory usage: 273.6+ KB
```

```
df.isnull().sum()
```

```
0  
name 0  
email_id 0  
year 0  
current_course 0  
technical_skills 0  
programming_languages 0  
rating 0  
soft_skills 0  
rating.1 0  
projects 0  
job_role_aspiration 0  
challenges_faced 0  
career_support_needed 0  
preferred_learning_method 0  
dtype: int64
```

```
df.describe()
```

	rating	rating.1
count	2500.000000	2500.000000
mean	2.943735	2.945711
std	1.408610	1.442212
min	0.785041	0.820915
25%	1.950717	1.914116
50%	2.983548	2.993185
75%	4.028709	4.045471
max	5.236144	5.247404

```
df['job_role_aspiration'].value_counts()
```

	count
job_role_aspiration	
Cloud Architect	424
Software Developer	406
Data Scientist	379
Cybersecurity Expert	357
Business Analyst	353
AI Engineer	335
Software Developer_X	51
Cybersecurity Expert_X	41
Data Scientist_X	41
Cloud Architect_X	41
AI Engineer_X	40
Business Analyst_X	32

dtype: int64

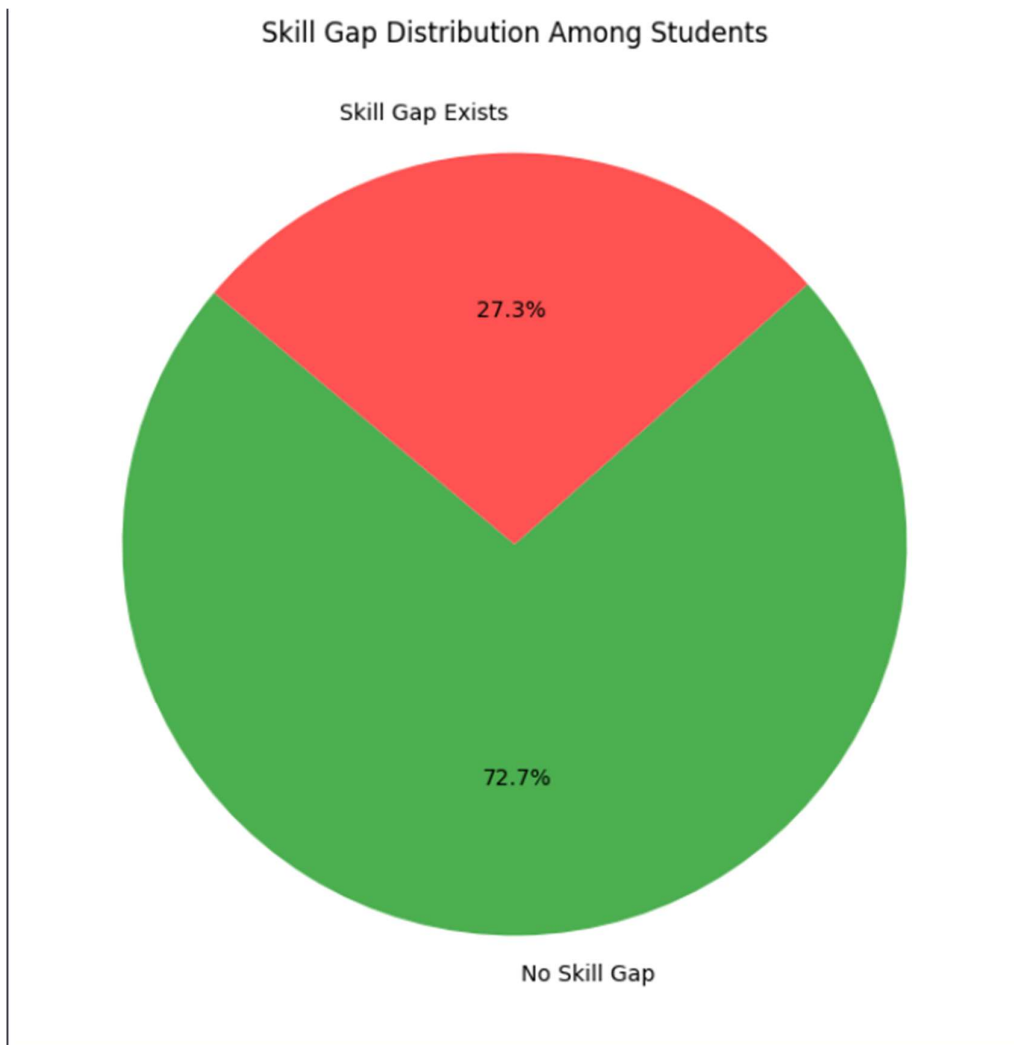
```
df['job_role_aspiration'].describe()
```

	job_role_aspiration
count	2500
unique	12
top	Cloud Architect
freq	424

dtype: object

```
print("\n--- 📊 Data Visualization 📊 ---")
gap_counts = df['skill_gap'].value_counts()
labels = ['No Skill Gap', 'Skill Gap Exists']

plt.figure(figsize=(8, 8))
plt.pie(gap_counts, labels=labels, autopct='%1.1f%%', startangle=140, colors=['#4CAF50', '#FF5252'])
plt.title('Skill Gap Distribution Among Students')
plt.show()
```





```

# Skills ko process karne ke liye ek 'all_skills' column banayein
df['all_skills'] = df['technical_skills'].str.cat(df['soft_skills'], sep=',')

# Features (X) aur Target (y) ko alag karein
X = df[['current_course', 'year', 'all_skills']]
y = df['skill_gap']

# Har skill ko alag column banayein
X.loc[:, 'all_skills'] = X['all_skills'].apply(lambda x: [skill.strip() for skill in str(x).split(',')])
mlb = MultilabelBinarizer()
skills_encoded = mlb.fit_transform(X['all_skills'])
skills_df = pd.DataFrame(skills_encoded, columns=mlb.classes_, index=X.index)

# Course aur Year ko numbers mein badlein
categorical_features = ['current_course', 'year']
one_hot = OneHotEncoder(handle_unknown='ignore', sparse_output=False)
encoded_cats = one_hot.fit_transform(X[categorical_features])
encoded_cats_df = pd.DataFrame(encoded_cats, columns=one_hot.get_feature_names_out(categorical_features), index=X.index)

# Sabhi taiyaar features ko jodein
X_processed = pd.concat([encoded_cats_df, skills_df], axis=1)

print("Data is Ready for Model BUilt")
print(f"Total features: {X_processed.shape[1]}")

```

```

Data is Ready for Model BUilt
Total features: 50

```

## 6. SPLITTING AND MODEL IMPLEMENTATION &

## 7. ACCURACY OF ALL IMPLEMENTED MODELS

```
# Data ko training aur testing hisson mein baantein
X_train, X_test, y_train, y_test = train_test_split(X_processed, y, test_size=0.25, random_state=42)

# Model banayein aur train karein
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# Model ki performance check karein
y_pred = model.predict(X_test)
print("--- Model Evaluation Results ---")
print(f"Accuracy: {accuracy_score(y_test, y_pred):.2f}")
print(classification_report(y_test, y_pred, zero_division=0))
```

--- Model Evaluation Results ---

Accuracy: 0.87

	precision	recall	f1-score	support
0	0.84	0.70	0.76	184
1	0.88	0.94	0.91	441
accuracy			0.87	625
macro avg	0.86	0.82	0.84	625
weighted avg	0.87	0.87	0.87	625

## 8. BUILDING A USER INPUT PREDICTIVE SYSTEM

```
# --- Interactive Input Section ---

# Prepare options to show the user from the dataset
course_options = df['current_course'].unique()
year_options = df['year'].unique()
job_options = list(job_skill_requirements.keys())

print("--- Predict Your Skill Gap ---")
print("\nPlease provide the information below.")

# Display options for Job Aspiration and get user input
print("\nWhich of these Job Roles are you aspiring for?")
for job in job_options:
    print(f"- {job}")
job_aspiration = input("Type your answer here: ")

# Get input for the course
course = input(f"\nWhat is your current course? (e.g., {course_options[0]}): ")

# Get input for the year
year = input(f"Which year are you in? (e.g., {year_options[0]}): ")

# Get input for skills
tech_skills = input("\nWhat Technical Skills do you have? (separate with a comma, e.g., Python, Java, SQL): ")
soft_skills = input("What Soft Skills do you have? (e.g., Leadership, Teamwork): ")

# Create a dictionary from the user's input
new_student_data = {
    'current_course': course,
    'year': year,
    'job_role_aspiration': job_aspiration,
    'technical_skills': tech_skills,
    'soft_skills': soft_skills
}

# --- Prediction Section (No changes here) ---

# Preprocess the new student's data to match the model's format
new_student_skills_list = [s.strip() for s in (new_student_data['technical_skills'] + ',' + new_student_data['soft_skills']).split(',')]
new_df = pd.DataFrame({'current_course': [new_student_data['current_course']], 'year': [new_student_data['year']], 'all_skills': [new_student_skills_list]})
new_skills_encoded = mlb.transform(new_df['all_skills'])
new_cats_encoded = one_hot.transform(new_df[['current_course', 'year']])
new_processed_df = pd.concat([pd.DataFrame(new_cats_encoded, columns=one_hot.get_feature_names_out()), pd.DataFrame(new_skills_encoded, columns=mlb.classes_)], axis=1)
new_processed_df = new_processed_df.reindex(columns=X_processed.columns, fill_value=0)

# Make the prediction
prediction = model.predict(new_processed_df)
proba = model.predict_proba(new_processed_df)
```

```
# --- Display Results ---

print("\n-----")
print("\n--- Your Prediction & Recommendation ---")
print(f"Your Aspiration: {new_student_data['job_role_aspiration']}")

if new_student_data['job_role_aspiration'] not in job_skill_requirements:
    print("\nSorry, we do not have data for this job role yet.")
elif prediction[0] == 1:
    print(f"Prediction: **Skill Gap Exists** (Probability: {proba[0][1]:.2f})")
    required = set(s.lower() for s in job_skill_requirements[new_student_data['job_role_aspiration']])
    owned = set(s.lower() for s in new_student_skills_list)
    recommendations = list(required - owned)
    print("\n**Recommendations to bridge the gap:**")
    if recommendations:
        for skill in recommendations:
            print(f" - Learn **{skill.title()}**")
    else:
        print(" You have the necessary skills, but you should practice them more.")
else:
    print(f"Prediction: **No Significant Skill Gap** (Probability: {proba[0][0]:.2f})")
    print("\nYou are on the right track! Keep improving your skills.")

print("\n-----")
```

```
--- Predict Your Skill Gap ---

Please provide the information below.

Which of these Job Roles are you aspiring for?
- Software Developer
- AI Engineer
- Data Scientist
- Cloud Architect
- Cybersecurity Expert
Type your answer here: Software Developer

What is your current course? (e.g., M.Tech AI): MCA
Which year are you in? (e.g., 3rd Year): 2nd year

What Technical Skills do you have? (separate with a comma, e.g., Python, Java, SQL): python,SQL
What Soft Skills do you have? (e.g., Leadership, Teamwork): Leadership

-----

--- Your Prediction & Recommendation ---
Your Aspiration: Software Developer
Prediction: **Skill Gap Exists** (Probability: 0.61)

**Recommendations to bridge the gap:**
- Learn **Teamwork**
- Learn **Data Structures**
- Learn **Java**
- Learn **Problem-Solving**

-----
```

## **9. Conclusion**

The early prognosis of cardiovascular diseases can aid in making decisions on lifestyle changes in high-risk patients and in turn reduce the complications, which can be a great milestone in the field of medicine. This project resolved the feature selection with the models used like DecisionTreeClassifier, Logistics Regression, KNeighborsClassifier, Gaussian NB and Random Forest Classifier. It successfully predicts the heart disease, with 85% accuracy. Further for its enhancement, we can train on models and predict the types of cardiovascular diseases providing recommendations to the users, and also use more enhanced models.

## **10. FUTURE SCOPE**

The future scope of a heart disease prediction project using machine learning (ML) is promising and multifaceted. Advancements in ML techniques will likely lead to more accurate predictive models, incorporating sophisticated algorithms and ensemble methods. Integration of medical imaging, personalized medicine, and continuous real-time monitoring through wearables and IoT devices could enhance predictive accuracy. Future projects may prioritize explainable AI, addressing data privacy, and collaborating closely with healthcare providers for seamless integration into clinical practice. Emphasizing continuous learning and model updating, alongside considerations for public health initiatives and global health applications, can contribute to broader impacts on cardiovascular health. The success of these projects depends not only on technical advancements but also on ethical considerations and collaboration with healthcare professionals to ensure real-world effectiveness and adherence to regulatory standards.

## **11. REFERENCE**

1. <https://www.analyticsvidhya.com/blog/2022/02/heart-disease-prediction-using-machine-learning/>
2. J. Ma, W. Gao, P. Mitra, S. Kwon, B. J. Jansen, K.-F. Wong, and M. Cha, "Heart diseases prediction.," in IJCAI, pp. 3818–3824, 2016.
3. Z. Jin, J. Cao, H. Guo, Y. Zhang, Y. Wang, and J. Luo, "Heart diseases prediction," arXiv preprint arXiv:1701.06250, 2017.
4. L. Derczynski, K. Bontcheva, M. Liakata, R. Procter, G. W. S. Hoi, and A. Zubiaga, "Semeval-2017 task 8: Rumoureal: Determining Heart diseases prediction," arXiv preprint arXiv:1704.05972, 2017.
5. X. Cao, T. Le, and J. Zhang, "Machine learning based detection of Heart diseases prediction,"

### **ASSESSMENT**

**Internal:**

<b>SL NO</b>	<b>RUBRICS</b>	<b>FULL MARK</b>	<b>MARKS OBTAINED</b>	<b>REMARKS</b>
1	Understanding the relevance, scope and dimension of the project	10		
2	Methodology	10		
3	Quality of Analysis and Results	10		
4	Interpretations and Conclusions	10		
5	Report	10		
	<b>Total</b>	<b>50</b>		

**Date:**

**Signature of the Faculty**

## COURSE OUTCOME (COs) ATTAINMENT

### ➤ Expected Course Outcomes (COs):

(Refer to COs Statement in the Syllabus)

---

---

---

---

### ➤ Course Outcome Attained:

How would you rate your learning of the subject based on the specified COs?

☐

1

☐

2

☐

3

☐

4

☐

5

☐

6

☐

7

☐

8

☐

9

☐

10

LOW

HIGH

### ➤ Learning Gap (if any):

---

---

---

---

### ➤ Books / Manuals Referred:

---

---

---

---

Date:

Signature of the Student

### ➤ Suggestions / Recommendations:

(By the Course Faculty)

---

---

---

---

Date:

Signature of the Faculty