IDENTIFYING PATTERN AND TRENDS IN CAMOUS PLACEMENT DATA USING MACHINE LEARNING

The project submitted to Smart Internz

By

BHARATHIRAJA.S (2022K1991)

BHUVANESHWARI.R(2022K1992)

BOOPATHI.K (2022K1993)

DHANISHKUMAR (2022K1994)

KAVINKUMAR.T(2022K1852)

Department of Computer Science

GOVERNMENT ARTS AND SCIENCE COLLEGE

MODAKURICHI

(Affiliated to Bharathiar University)

ERODE - 638104

1.INTRODUCTION

1.1 Overview

Campus recruitment is a strategy for sourcing, engaging and hiring young talent for internship and entry-level positions. College recruiting is typically a tactic for medium- to largesized companies with high-volume recruiting needs, but can range from small efforts (like working with university career centers to source potential candidates) to large-scale operations (like visiting a wide array of colleges and attending recruiting events throughout the spring and fall semester).Campus recruitment often involves working with university career services centers and attending career fairs to meet in-person with college students and recent graduates.Our solution revolves around the placement season of a Business School in India. Where it has various factors on candidates getting hired such as work experience,exam percentage etc., Finally it contains the status of recruitment and remuneration details.

We will be using algorithms such as KNN, SVM and ANN. We will train and test the data with these algorithms. From this the best model is selected and saved in .pkl format. We will be doing flask integration and IBM deployment.

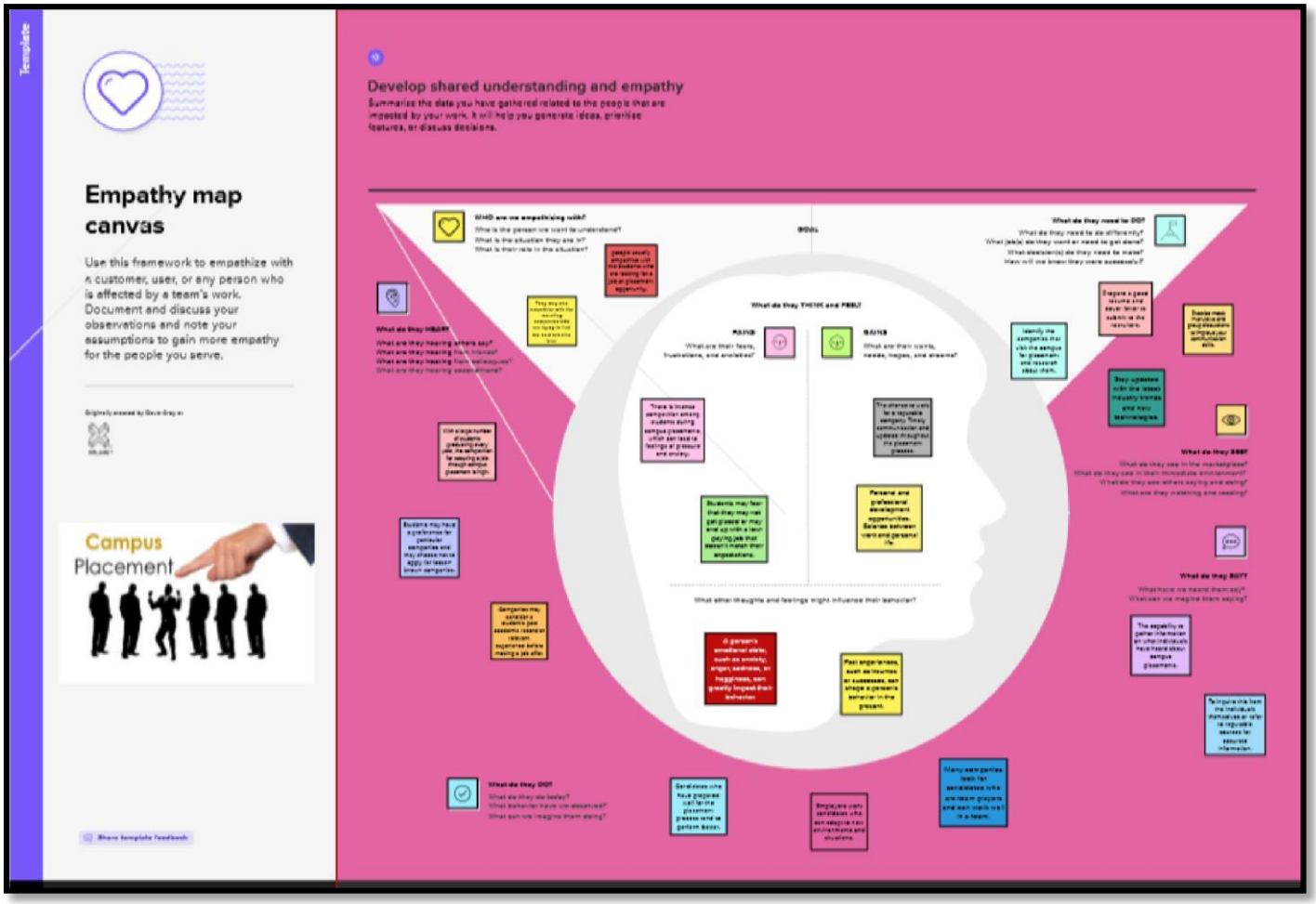
1.2 Purpose

Campus placement or campus recruiting is a program conducted within universities or other educational institutions to provide jobs to students nearing completion of their studies. In this type of program, the educational institutions partner with corporations who wish to recruit from the student population.

2.PROBLEM DEFINITION & DESIGN THINKING

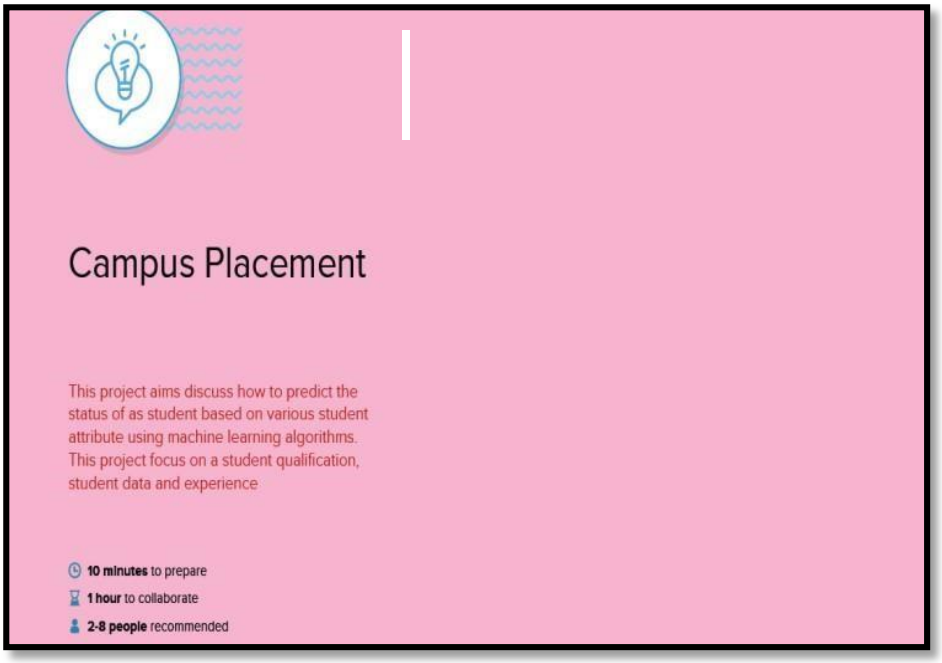
2.1 Empathy Map

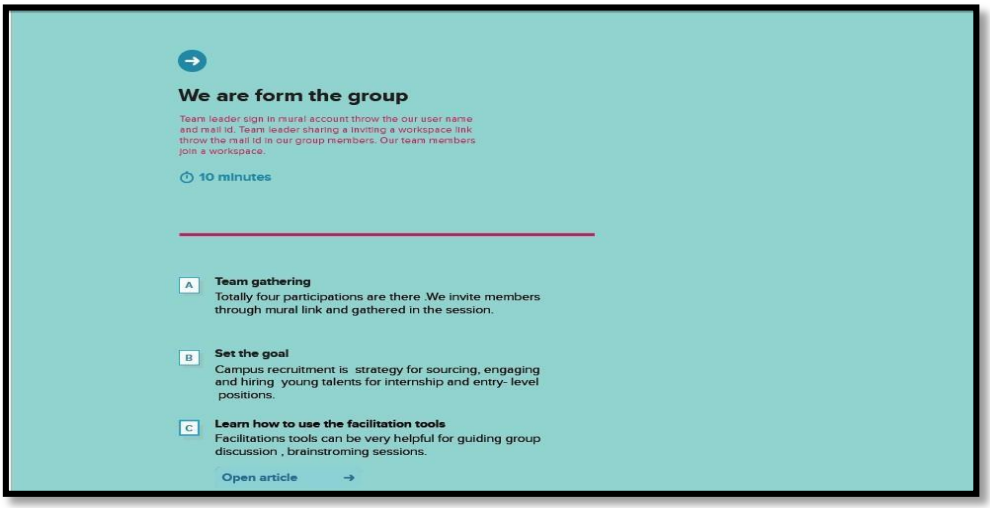
Empathy in this case can refer to the ability of the machine learning algorithm to understand the context and nuances of the data it is analyzing. This includes understanding the factors that may impact the placement of students, such as their academic performance, background, and the current job market.

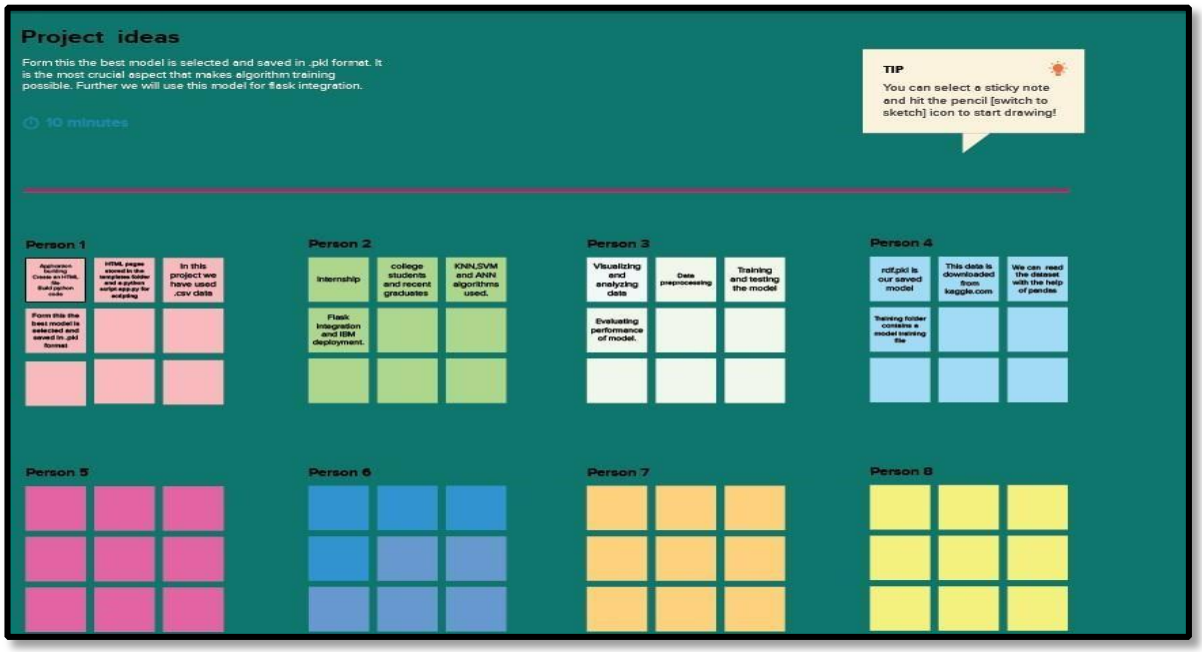


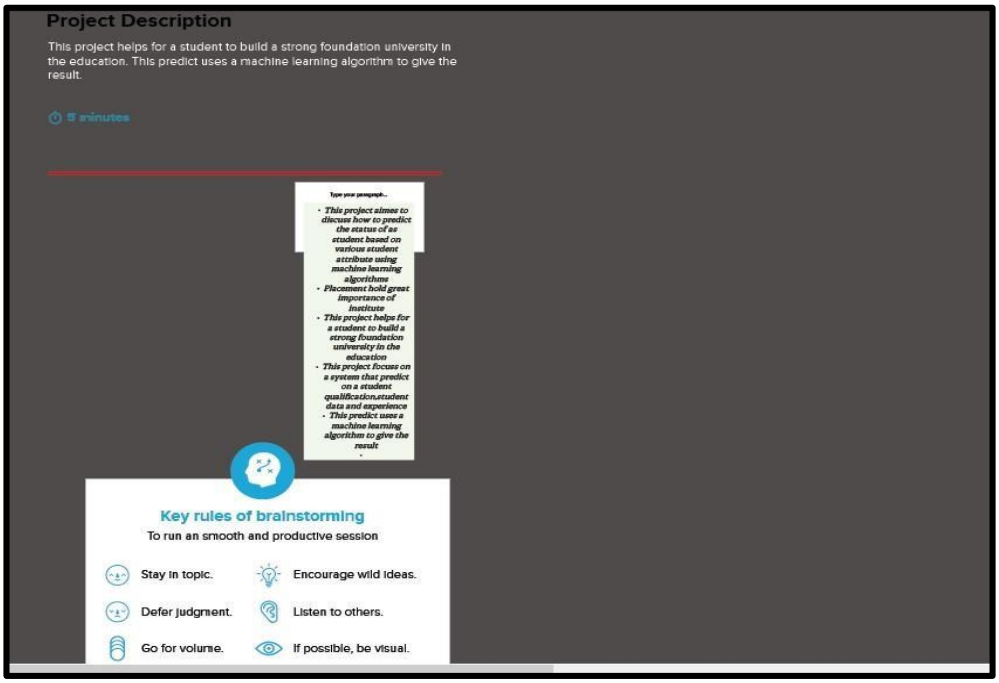
2.2 Ideation &Brainstorming Map

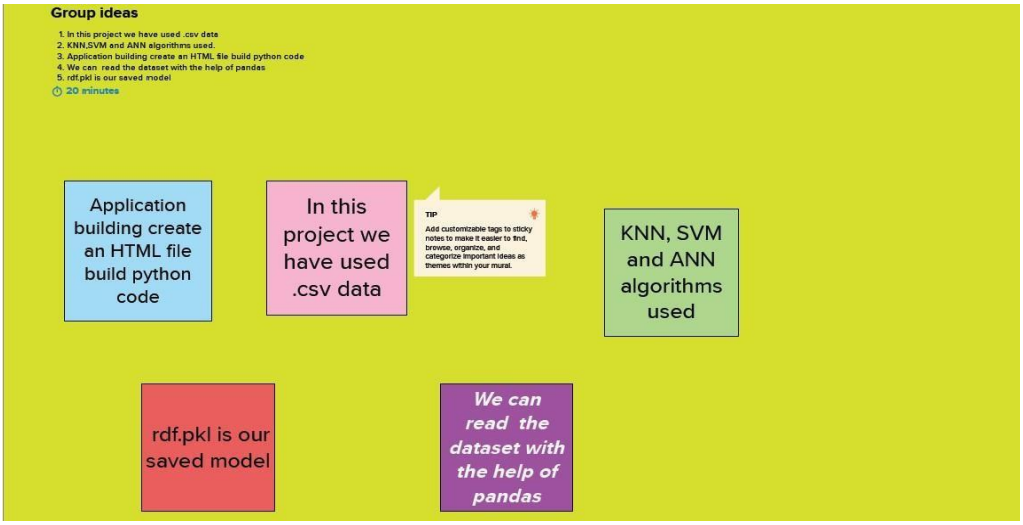
Brainstorm Map for Identifying Patterns and Trends in Campus Placement Data using Machine Learning

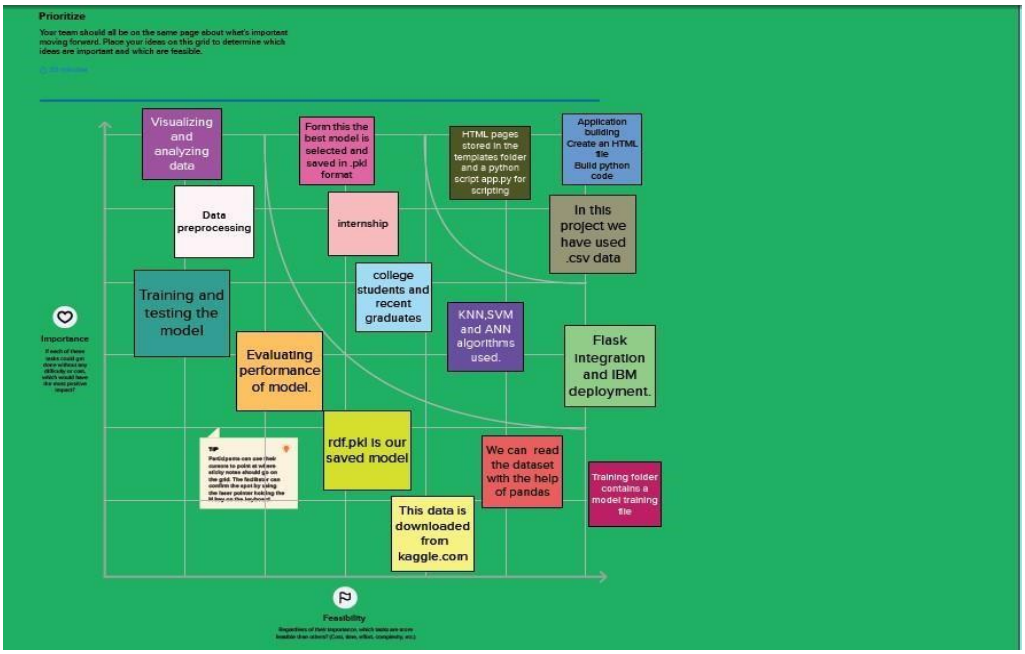

Campus Placement
This project aims discuss how to predict the status of as student based on various student attribute using machine learning algorithms. This project focus on a student qualification, student data and experience
10 minutes to prepare
1 hour to collaborate
2-8 people recommended

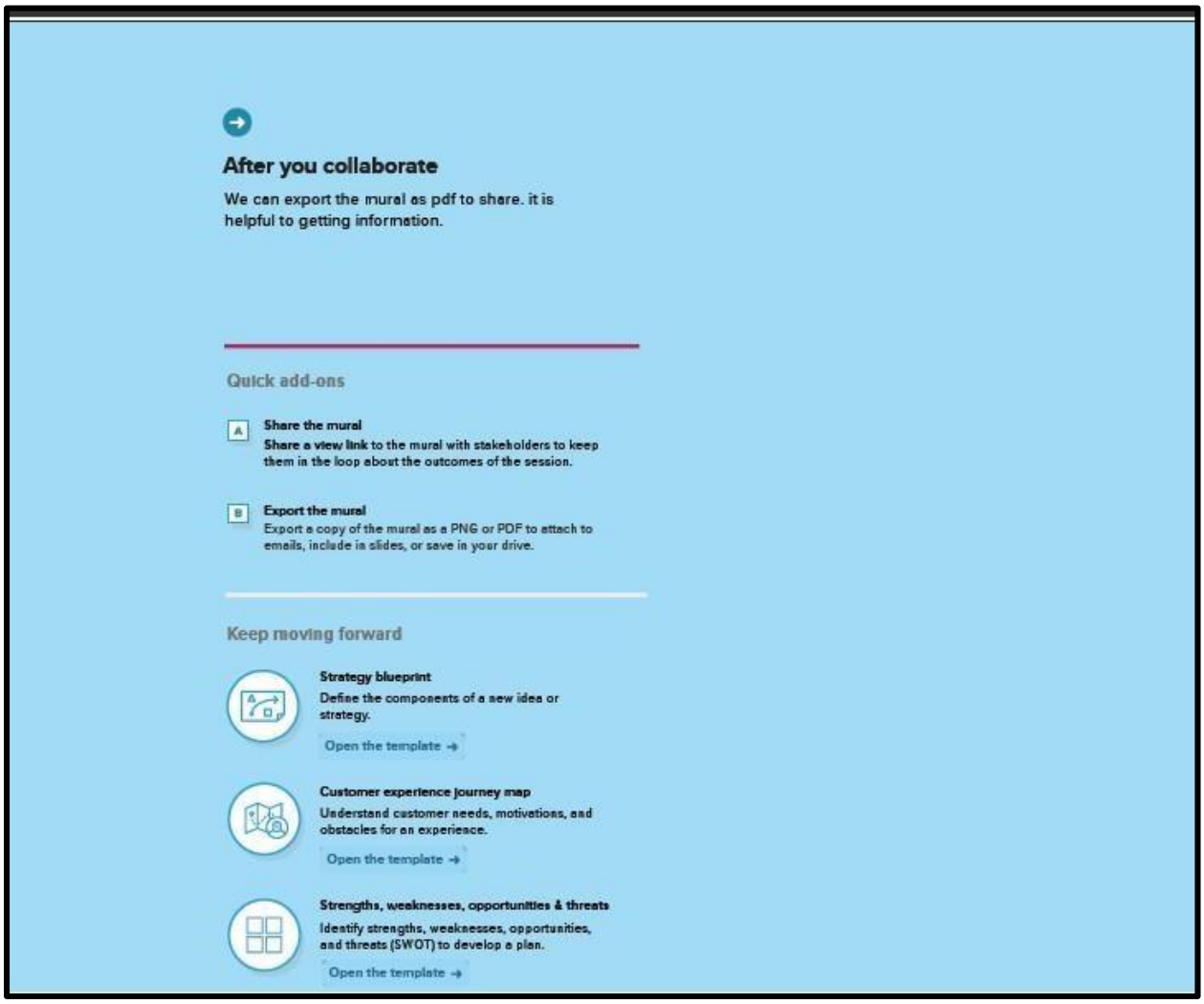

We are form the group
Team leader sign in mural account throw the our user name and mail id. Team leader sharing a inviting a workspace link throw the mail id in our group members. Our team members join a workspace.
10 minutes
A
Team gathering
Totally four participations are there .We invite members through mural link and gathered in the session.
B
Set the goal
Campus recruitment is strategy for sourcing, engaging and hiring young talents for internship and entry- level positions.
C
Learn how to use the facilitation tools
Facilitations tools can be very helpful for guiding group discussion , brainstroming sessions.
Open article


Project ideas
Form this the best model is selected and saved in .pkl format. It is the most cruciel aspect that makes algorithm training possible. Further we will use this model for flask integration.
10 minutes
TIP
You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!
Person 1
In this project we have used .csv data
Form this the best model is selected and saved in .pkl format
Person 2
Internship
college students and recent graduates
KNN,SVM and ANN algorithms used.
Flask Integration and IBM deployment.
Person 3
Visualizing and analyzing data
Data preprocessing
Training and testing the model
Evaluating performance of model.
Person 4
rdf.pkl is our saved model
This data is downloaded from kaggle.com
We can read the dataset with the help of pandas
Person 5
Person 6
Person 7
Person 8


Project Description
This project helps for a student to build a strong foundation university in the education. This predict uses a machine learning algorithm to give the result.
5 minutes
Type your paragraph...
This project aimes to discuse how to predict the status of as student based on various student attribute using machine learning algorithms
Placement hold great importance of institute
This project helps for a student to build a strong foundation university in the education
This project focuss on a system that predict on a student qualification,student data and experience
This predict uses a machine learning algorithm to give the result
Key rules of brainstorming
To run an smooth and productive session
Stay in topic.
Encourage wild ideas.
Defer judgment.
Listen to others.
Go for volume.
If possible, be visual.


Group ideas
1. In this project we have used .csv data
2. KNN,SVM and ANN algorithms used.
3. Application building create an HTML file build python code
4. We can read the dataset with the help of pandas
5. rdf.pkl is our saved model
20 minutes
Application building create an HTML file build python code
In this project we have used .csv data
TIP
Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.
KNN, SVM and ANN algorithms used
rdf.pkl is our saved model
We can read the dataset with the help of pandas

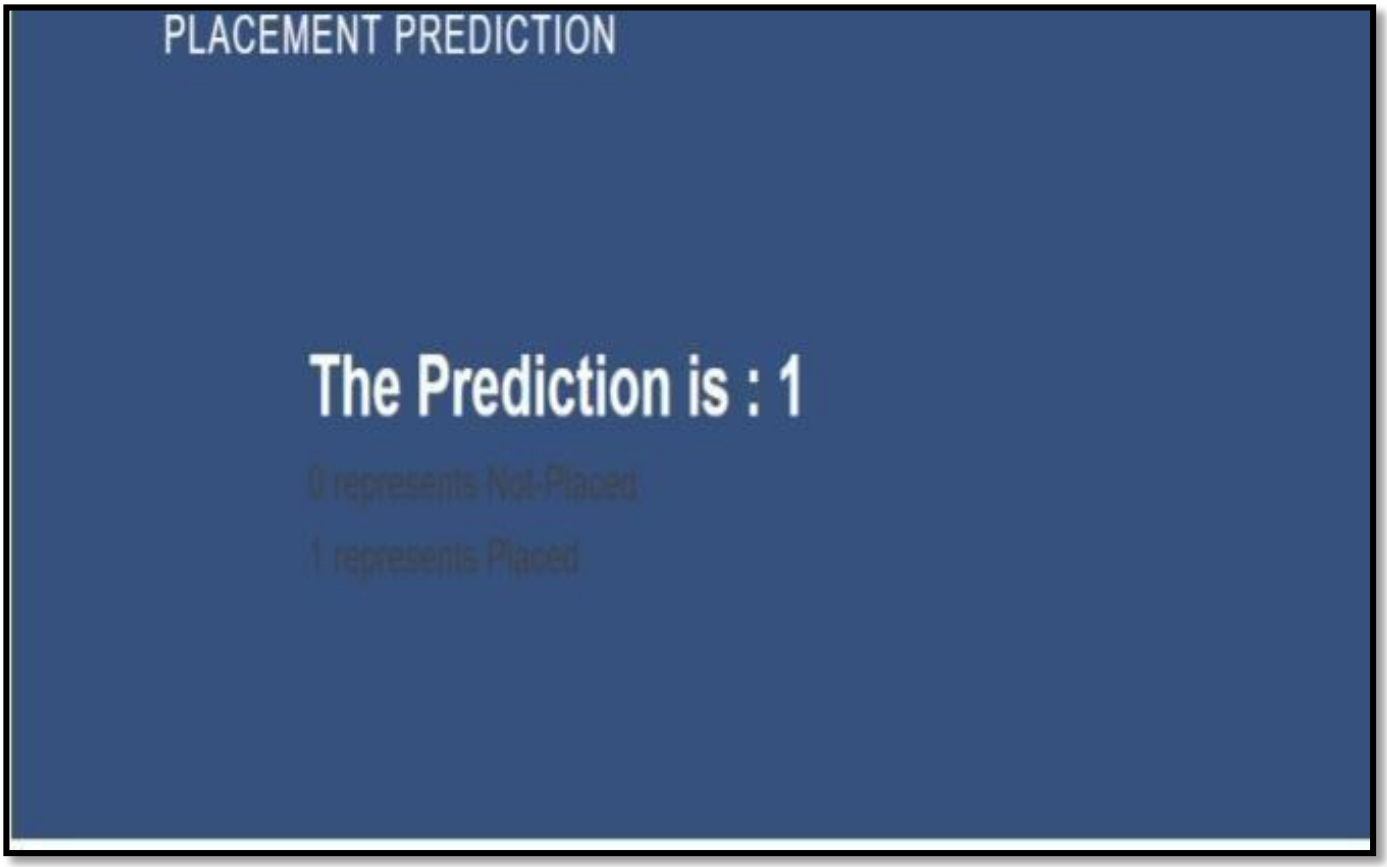

Prioritize
Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.
Visualizing and analyzing data
Form this the best model is selected and saved in .pkl format
HTML pages stored in the templates folder and a python script app.py for scripting
Application building Create an HTML file Build python code
Data preprocessing
internship
In this project we have used .csv data
Training and testing the model
college students and recent graduates
KNN,SVM and ANN algorithms used.
Flask integration and IBM deployment.
Evaluating performance of model.
rdf.pkl is our saved model
We can read the dataset with the help of pandas
Training folder contains a model training file
This data is downloaded from kaggle.com
Importance
Feasibility


After you collaborate
We can export the mural as pdf to share. it is helpful to getting information.
Quick add-ons
A
Share the mural
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.
B
Export the mural
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.
Keep moving forward
Strategy blueprint
Define the components of a new idea or strategy.
Open the template
Customer experience journey map
Understand customer needs, motivations, and obstacles for an experience.
Open the template
Strengths, weaknesses, opportunities & threats
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.
Open the template

3.Result


PLACEMENT PREDICTION
Get Started
Identifying Patterns and Trends
in Campus Placement Data
using Machine Learning


FILL THE DETAILS
22
0
2
1
8
1
Submit

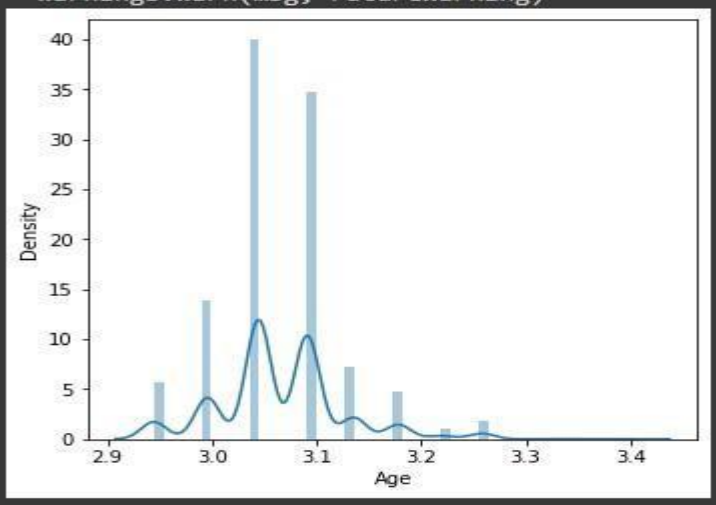


Handling Outliers

```
def transformationplot(feature):
    plt.figure(figsize=(12,5))
    plt.subplot(1,2,1)
    sns.distplot(feature)

transformationplot(np.log(df['Age']))
```

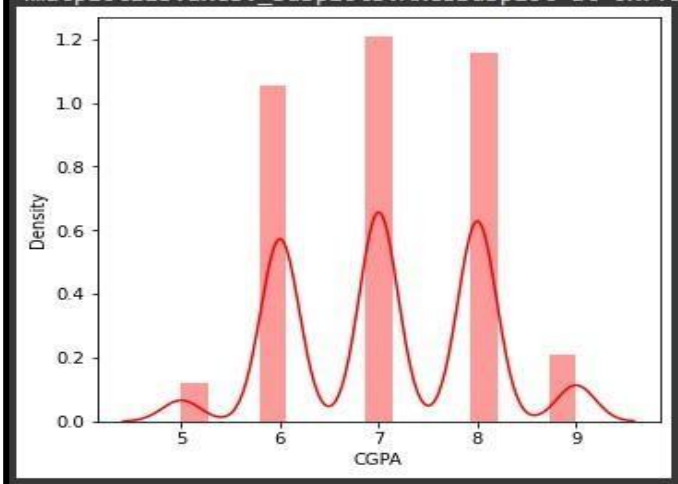
```
/usr/local/lib/python3.8/dist-packages/seaborn/distribut
  warnings.warn(msg, FutureWarning)
```



Univariate analysis

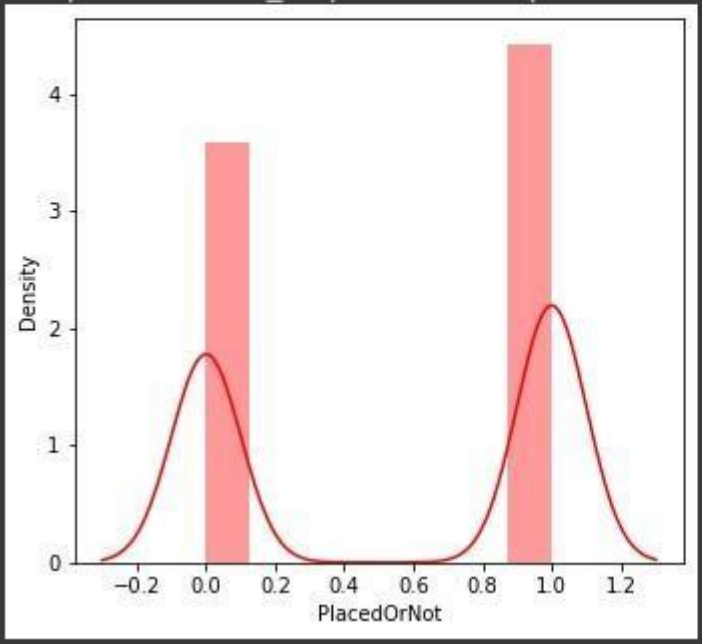
```
plt.figure(figsize=(12,5))
plt.subplot(121)
sns.distplot(df['CGPA'],color='r')
```

```
/usr/local/lib/python3.8/dist-packages/seaborn/distributions.py:261
  warnings.warn(msg, FutureWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7f5463e50d00>
```



```
plt.figure(figsize=(12,5))
plt.subplot(121)
sns.distplot(df['PlacedOrNot'],color='r')
```

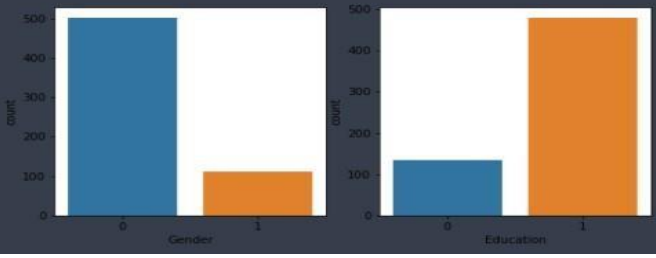
```
/usr/local/lib/python3.8/dist-packages/seaborn/distributions.py:2619: Futu
  warnings.warn(msg, FutureWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7f5463d95790>
```



Bivariate analysis

```
#plotting the count plot
plt.figure(figsize=(18,4))
plt.subplot(1,4,1)
sns.countplot(data['Gender'])
plt.subplot(1,4,2)
sns.countplot(data['Education'])
plt.show()
```

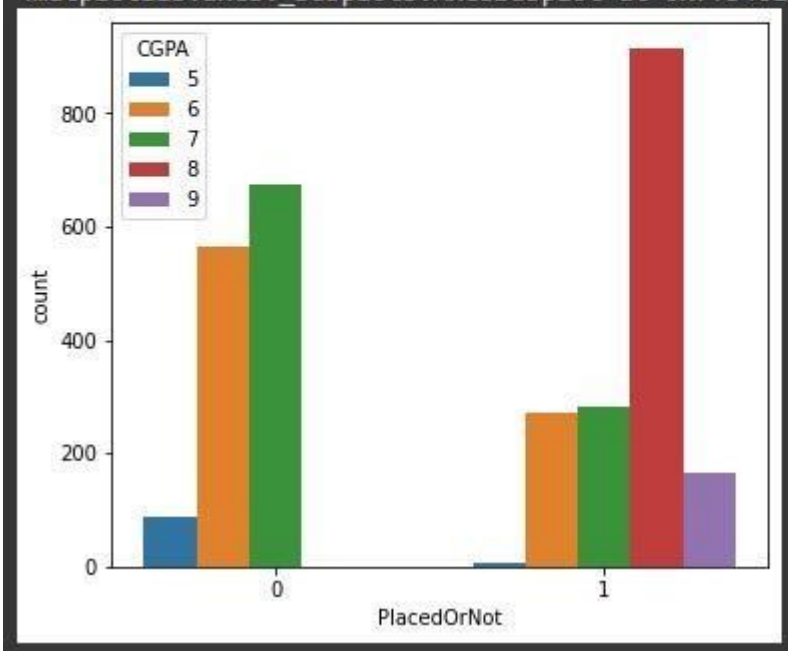
```
C:\Users\HP\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From ve
rsion 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an e
rror or misinterpretation.
  warnings.warn(
C:\Users\HP\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From ve
rsion 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an e
rror or misinterpretation.
  warnings.warn(
```



Multivariate analysis

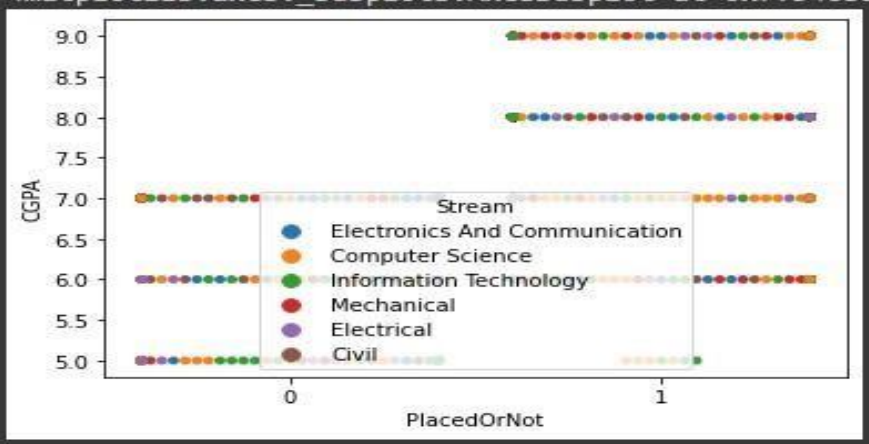
```
plt.figure(figsize=(20,5))
plt.subplot(131)
sns.countplot(df["PlacedOrNot"],hue=df['CGPA'])
```

```
/usr/local/lib/python3.8/dist-packages/seaborn/_decorators.py:36: Futur
  warnings.warn(
<matplotlib.axes._subplots.AxesSubplot at 0x7f5461cf85b0>
```




```
sns.swarmplot(df['PlacedOrNot'],df['CGPA'],hue=df['Stream'])
```

```
/usr/local/lib/python3.8/dist-packages/seaborn/_decorators.py:36:
  warnings.warn(
/usr/local/lib/python3.8/dist-packages/seaborn/categorical.py:129
  warnings.warn(msg, UserWarning)
/usr/local/lib/python3.8/dist-packages/seaborn/categorical.py:129
  warnings.warn(msg, UserWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7f5463d06df0>
```



4.Advantages

- Saves Time & Efforts
- Improved Retention Rates
- Getting New Knowledge & Skills
- Quick Learners & Multi-tasking candidates
- Good relationship between Organization & Campus
- High Volume of Talent Pool
- Resumes are the only way to select a candidate
- Limited Staff & Time

Disadvantages

Campus recruitment is an expensive affair for majority of the companies as it adds up costs to the bottom line. Companies incur different expenses related to travel, boarding, training etc while conducting campus selection process. The experienced and skilled candidates having practical job exposures cannot be recruited through campus placements. Fresh candidates selected through campus placements require adequate training for work.

This is an additional expense for the company. Also, students can't work with their dream company and will have to remain satisfied with the company that recruits them during campus selection.

5. Applications

Artificial Intelligence is a very popular topic which has been discussed around the world. Machine learning is one of the most exciting technologies of AI that gives systems the ability to think and act like humans. machine learning is a subfield of AI and has its various application which helps to make a prediction, analysis, classification, etc. that is recognized by the companies across several industries(like Financial Service, Government, Healthcare, Transportation, etc.) that deal with huge volumes of data needed by the organizations in running their business effectively and to get an edge over their competitors.

6.Conclusion

An effective recruitment and selection process reduces turnover, we also get much better results in our recruitment process if we advertise specific criteria that are relevant to the job. Include all necessary skills, and include a list of desired skills that are not necessary but that would enhance the candidate's chances. If we fail to do this, we might end up with a low-quality pool of candidates and wind up with limited choices to fill the open position. When we choose a candidate based upon the qualifications demonstrated in the resume, the interview, employment history and background check, we will land the best fit for the position. Based on our decisions about a specific candidate upon specific evidence rather than any gut instincts. If we hire people who can do the job instead of people we merely like, we will have higher productivity and quality in our products or services.

The recruitment and selection process is the time we not only identify a candidate who has the experience and aptitude to do the job that we are looking to fill, but also to find someone who shares and endorses our company's core values. The candidate will need to fit in well within our company's culture. The selection and recruitment process should provide our company with an employee who adapts and works well with others in our business.

Failure to recruit and select for the long term can result in high turnover.

7.Future Scope

Future enhancement for Identifying Patterns and Trends in Campus Placement Data using Machine Learning

There are several potential enhancements that could be made to identify patterns and trends in campus placement data using machine learning. Here are a few ideas: Incorporate natural language processing (NLP): Many campus placement reports include written feedback from both employers and students.

By incorporating NLP techniques, machine learning algorithms could extract insights from this unstructured data to identify patterns and trends in what employers are looking for in candidates and how students are responding to their job offers.

Use graph analysis techniques: Campus placement data typically involves complex relationships between multiple variables such as colleges, companies, job roles, and students. Graph analysis techniques such as network analysis and graph clustering could be used to identify patterns and trends in these relationships.

8.Appendix

8.1 Source code

Sample Coding: import numpy as np import

pandas as pd import os import seaborn as sns

import matplotlib.pyplot as plt from sklearn import

svm from sklearn.metrics import accuracy_score

from sklearn.neighbors import KNeighborsClassifier

from sklearn import metrics from

sklearn.model_selection import cross_val_score

from sklearn import preprocessing from

sklearn.model_selection import train_test_split

from sklearn.preprocessing import StandardScaler

import joblib from sklearn.metrics import

accuracy_score df

=pd.read_csv(r'/content/collegePlace.csv') df.head()

df.info()

df.isnull().sum() def

transformationplot(feature):

plt.figure(figsize=(12,5)) plt.subplot(1,2,1)

sns.distplot(feature)

transformationplot(np.log(df['Age']))

df=df.replace(['Male'],[0]) df=df.replace(['Female'],[1])

```
df=df.replace(['Computer Science','InformationTechnology','Electronics And
Communication','Mechanical','Electrical','Civil'],

        [0,1,2,3,4,5])

df=df.drop(['Hostel'],axis=1)   Df

plt.figure(figsize=(12,5))

plt.subplot(121)

sns.distplot(df['CGPA'],color='r')

plt.figure(figsize=(12,5))

plt.subplot(121)

sns.distplot(df['PlacedOrNot'],color='r')

plt.figure(figsize=(18,4))

plt.subplot(1,4,1)

sns.countplot(df['Gender'])

plt.subplot(1,4,2)

sns.countplot(df['Stream'])   plt.show()


plt.figure(figsize=(20,5))


plt.subplot(131)  sns.countplot(x='PlacedOrNot',hue='CGPA',data=df)

sns.swarmplot(x='PlacedOrNot',y='CGPA',hue=df['Stream'],data=df)



sc=StandardScaler()


x_bal=np.random.rand(100,10)


 names =[f'feature_{i}'for i in range(x_bal.shape[1])]
```

```
x_bal=sc.fit_transform(x_bal)

x_bal=pd.DataFrame(x_bal,columns=names)

  x=df.drop('PlacedOrNot',axis=1)

  y=df['PlacedOrNot']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,stratify=y,random_state=2)
classifier=svm.SVC(kernel='linear')

classifier.fit(x_train,y_train)

x_train_prediction=classifier.predict(x_train)
training_data_accuracy=accuracy_score(x_train_prediction,y_train)

print('Accuracy score of the training data:',training_data_accuracy)

best_k={"Regular":0}

best_score={"Regular":0}

for k in range(3,50,2):

knn_temp = KNeighborsClassifier(n_neighbors=k)   knn_temp.fit(x_train,y_train)

knn_temp_pred = knn_temp.predict(x_test)

  score = metrics.accuracy_score(y_test, knn_temp_pred)*100
if score >= best_score["Regular"]and score < 100:

best_score["Regular"]=score       best_k["Regular"]=k
```

```
print("---Results---\nk: {}".format(best_k,best_score))

knn=KNeighborsClassifier(n_neighbors=best_k["Regular"])

knn.fit(x_train,y_train)

knn_pred=knn.predict(x_test)

testd = accuracy_score(knn_pred,y_test)

 import tensorflow as tf

from tensorflow import keras

 from keras.models import Sequential

 from tensorflow.keras import layers

 classifier = Sequential()  classifier.add(keras.layers.Dense(6,activation
= 'relu',input_dim=6))  classifier.add(keras.layers.Dropout(0.50))
classifier.add(keras.layers.Dense(6,activation='relu'))
classifier.add(keras.layers.Dropout(0.50))
classifier.add(keras.layers.Dense(1,activation='sigmoid'))
loss_1=tf.keras.losses.BinaryCrossentropy()
classifier.compile(optimizer = 'Adam',loss = loss

metrics=['accuracy'])
classifier.fit(x_train,y_train,batch_size=20,epochs = 100)

 import pickle
```

pickle.dump(knn,open("placement.pkl",'wb')) model=pickle.load(open('placement.pkl','rb'))

1)index.html

```
<section id="hero" class="d-flex-column justify-content-center">

<div class="container">

<div class="row justify-content-center">

<div class="col-xl-8">

<h1>Identifying Patterns and Trends</h1>

<h1>in Campus placement Data</h1>

<h1> using Machine Learning</h1>

</div>

</div>

</div>

</section>
```

2.index1.html:

```
<section id="about"class="about">

<div class="container">

<div class="section-title">

<h2>Fill the deatails</h2>

</div>
<div.class="row content">
```

<form action="{{ url_for('y_predict')}}"method="POST">

<input type="number"id="sen1"name="sen1"placeholder="Age">

<input type="number"id="sen2"name="sen2"placeholder="Gender M(0),F(0)"

<input type="number"id="sen3"name="sen3"placeholder="Stream
CS(0),IT(1),ECE(2),Mech(3),EEE(4),Civil(5)">

<input type="number"id="sen4"name="sen4"placeholder="Internships">

<input type="number"id="sen5"name="sen5"placeholder="CGPA">

<input type="number"id="sen6"name="sen6"placeholder="Number of backlogs"><input
type="submit"value="submit">

</form>

</section>

3.Secondpage.html:

<section id="hero" class="d-flex flex-column justifycontent-center">

<h1>The Prediction is : {{y}}</h1>

<h3> 0 represents Not-placed </h3>

<h3> 1 represents Placed<h2>

</section>**4.Project.py:**

from flask import Flask, render_template , request app=Flask(name)

import pickle

import joblib

model=pickle.load(open("placement123.pkl",'rb'))

ct=joblib.load('placement') @app.route('/')

def hello():

returnrender_template("index.html") @app.route('/guest'

, methods =["post"]) def y_predict():

x_test=[[(yo) for yo in request.form.values()]] prediction

=model.predict(x_test)

```
prediction = prediction[0]


return render_template("secondpage.html",y=prediction)          app.run(debug=True)
```