MINI PROJECT REPORT

On

"KnowYourSelf" – A Personality Prediction System

Submitted By

Mr. Amaan Ranapurwala **Mr. Harsh Maroo**

Mr. Yash Mishra **Mr. Aniruddha Kate**

Submitted in partial fulfillment of the requirements
Of
Mini Project in third year of Bachelor of Engineering

Guided by-
Ms. Titiksha Bhagat

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

S.B. JAIN INSTITUTE OF TECHNOLOGY, MANAGEMENT & RESEARCH, NAGPUR.

2021-2022

Institute Vision:

Emerge as a leading Institute for developing competent and creative Professionals.

Institute Mission:

- Providing Quality Infrastructure and experienced faculty for academic excellence.
- Inculcating skills, knowledge and opportunities for competency and creativity.
- Aligning with Industries for knowledge sharing, research and development.

Department Vision:

To become a center for quality education in the field of computer science & engineering and to create competent professionals.

Department Mission:

To provide academic ambience and latest software tools to prepare competent Software Engineers with strong theoretical and practical knowledge.

To foster professionalism and strong work ethics in students for the betterment of Society

To provide adequate infrastructure as well as experienced & skilled faculty members.

To encourage the spirit of entrepreneurship and adaptability in our students in view of the ever-changing scenario of the Software Industry.

Program Educational Objectives (PEO's)

Have analytical, design and implementation skills, to innovate, design and develop software products and systems.

Have strong work ethics and professionalism, reflected through communication skills, leadership, teamwork and sense of responsibility towards the society.

Be successful professionals through lifelong learning with allied objectives of higher education or research.

S.B. JAIN INSTITUTE OF TECHNOLOGY MANAGEMENT AND RESEARCH, NAGPUR

(An Autonomous Institute, Affiliated to RTMNU, Nagpur)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SESSION 2021-2022

CERTIFICATE

This is to certify that the Mini project titled **"KnowYourself - A Personality Prediction System"** is a bonafide work of **Mr. Amaan Ranapurwala, Mr. Harsh Maroo, Mr. Yash Mishra, Mr. Aniruddha Kate,** carried out for the partial fulfillment of the requirement of Mini Project in third year of Bachelor of Engineering in **Computer Science & Engineering, Rashtrasant Tukadoji Maharaj Nagpur University, Nagpur.**

Ms. Titiksha Bhagat
Assistant Professor
(Project Guide)

Mr. Animesh Tayal
Head of Department

Dr. S. L. Badjate
Principal

External Examiner

Mr/Mrs/Ms/Dr. _____

Designation: _____

Institution: _____

Date: _____

DECLARATION

We, hereby declare that the Mini Project titled **"*KnowYourSelf – A Personality Prediction System*"** submitted herein has been carried out by us in the Department of Computer Science & Engineering of S. B. Jain Institute of Technology Management and Research, Nagpur under the guidance of **Ms. Titiksha Bhagat**. The work is original and has not been submitted earlier as a whole or in part for the award of any degree / diploma at this or any other Institution / University.

.

Mr. Amaan Ranapurwala _____

Mr. Harsh Maroo _____

Mr. Yash Mishra _____

Mr. Aniruddha Kate _____

Date: - / /

ACKNOWLEDGEMENT

We would like to express deep sense of gratitude to our Project Guide, **Ms. Titiksha Bhagat**, **Department of Computer Science & Engineering**, for being the cornerstone of our project. It was her incessant motivation and guidance during periods of doubts and uncertainties that has helped us to carry on with this project.

We would like to thank **Mr. Animesh Tayal**, **Head of Department**, **Computer Science & Engineering** for providing necessary guidance, support, motivation and inspiration without which this project would not have been possible.

We would like to extend our special thanks to **Dr. S. L. Badjate**, **Principal** of **S.B. Jain Institute of Technology, Management & Research** for his encouragement and best wishes.

We would like to extend our sincere thanks to the **Management** of **S.B. Jain Institute of Technology, Management & Research** for providing all the necessary infrastructure and laboratory facilities.

We also like to acknowledge the help extended by the **faculty members and non-teaching staff** of Computer Science & Engineering Department for successful completion of our project.

Last but not the least, special thanks to our family members, friends and colleagues for their continuous support.

ABSTRACT

The majority of people build their CVs and personality assessments via third-party portals. The problem with such websites is that they charge a fee, and there is no guarantee that the personality results gained are accurate. To address this issue, we created software that would help candidates build their CVs by asking them a few questions and identifying their personality. Five characteristics of different individuals commonly known as big five characteristics namely, openness, neuroticism, conscientiousness, agreeableness and extraversion are stored in a dataset and used for training. Based on this training, the personality of individuals is predicted using data mining concepts. Before testing the dataset, it is pre-processed using different data mining concepts like handling missing values, data discretization, normalisation etc. This pre-processed data can then be used to classify/predict user personality based on past classifications. The system analyses user characteristics and behaviours. System then predicts new user personality based on personality data stored by classification of previous user data. Model used to predict test dataset is "Logistic Regression" because Logistic regression is an effective model to predict output class labels for dependent categorical data.

Key Words— Personality, Machine Learning, Logistic Regression, Ocean Model.

INDEX

LIST OF FIGURES

Chapter No 1

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1. PROJECT BACKGROUND

The Big Five Personality Traits model is based on findings from several independent researchers, and it dates back to the late 1950s. But the model as we know it now began to take shape in the 1990s. Lewis Goldberg, a researcher at the Oregon Research Institute, is credited with naming the model "The Big Five." It is now considered to be an accurate and respected personality scale, which is routinely used by businesses and in psychological research.

The Big Five Personality Traits Model measures five key dimensions of people's personalities:

Openness: sometimes called "Intellect" or "Imagination," this measures your level of creativity, and your desire for knowledge and new experiences.

Conscientiousness: this looks at the level of care that you take in your life and work. If you score highly in conscientiousness, you'll likely be organized and thorough, and know how to make plans and follow them through. If you score low, you'll likely be lax and disorganized.

Extraversion/Introversion: this dimension measures your level of sociability. Are you outgoing or quiet, for instance? Do you draw energy from a crowd, or do you find it difficult to work and communicate with other people?

Agreeableness: this dimension measures how well you get on with other people. Are you considerate, helpful and willing to compromise? Or do you tend to put your needs before others'?

Natural Reactions: sometimes called "Emotional Stability" or "Neuroticism," this measure emotional reactions. Do you react negatively or calmly to bad news? Do you worry obsessively about small details, or are you relaxed in stressful situations?

1.2. PROBLEM STATEMENT

The majority of people build their CVs and personality assessments via third-party portals. The problem with such websites is that they charge a fee, and there is no guarantee that the personality results gained are accurate.

1.3. PROPOSED WORK

With the availability of high-dimensional and fine-grained data about human behavior, it becomes too handy to research and observe human behavior. Using mobile sensing studies, data collected from our day-to-day activities have drastically altered how psychologists perform research and undertake personality assessments.

Machine learning models are à boon to researchers and are used to learn highly complex relationships and evaluate their generalizability and robustness using the resampling method. It has the potential to transform research and assessment in personality psychology Algorithms can handle vast datasets, including thousands of attributes, without succumbing to collinearity issues. Moreover, ML algorithms are highly efficient in recognizing patterns in datasets that humans cannot even perceive. The use of these ML models can lead to better, more objective, and automated personality assessments.

People interact and express their likes, thoughts, feelings, and opinions on social media, capturing their personality traits. Machine Learning models have been actively using such a wide range of data to predict individuals Big Five (OCEAN) personality traits.

Various supervised machine learning algorithms like Naïve Bayes and Support Vector Machines are widely used among industries to predict personality traits. Moreover, recently, researchers have started to apply unsupervised learning methods to identify other psychological constructs in digital data.

Objectives

- To give early disease prediction mechanism.
- To get instant results of the risk factor of health.
- To provide convenience of a system software to check results any time.
- To provide accurate results using the ML models.

1.4. TECHNOLOGICAL BASE

This Project can be implemented by using various technologies like-

Python

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum from 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL). Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a battery included" language due to its comprehensive standard library.

Features:

- Easy-to-learn − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- Easy-to-read − Python code is more clearly defined and visible to the eyes.
- Easy-to-maintain − Python's source code is fairly easy-to-maintaining.
- Scalable − Python provides a better structure and support for large programs than shell scripting.

JAVA

Java is a high-level, class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is a general-purpose programming language intended to let programmers *write once, run anywhere* (WORA), meaning that compiled Java code can run on all platforms that support Java without the need to recompile. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The syntax of Java is similar to C and C++, but has fewer low-level facilities than either of them. The Java runtime provides dynamic capabilities (such as reflection and runtime code modification) that are typically not available in traditional compiled languages. As of 2019, Java was one of the most popular programming languages in use according to GitHub, particularly for client–server web applications, with a reported 9 million developers.

Java was originally developed by James Gosling at Sun Microsystems and released in May 1995 as a core component of Sun Microsystems' Java platform. The original and reference implementation Java compilers, virtual machines, and class libraries were originally released by Sun under proprietary licenses. As of May 2007, in compliance with the specifications of the Java Community Process, Sun had relicensed most of its Java technologies under the GPL-2.0-only license. Oracle offers its own HotSpot Java Virtual Machine, however the official reference implementation is the OpenJDK JVM which is free open-source software and used by most developers and is the default JVM for almost all Linux distributions.

As of March 2022, Java 18 is the latest version, while Java 17, 11 and 8 are the current long-term support (LTS) versions. Oracle released the last zero-cost public update for the legacy version Java 8 LTS in January 2019 for commercial use, although it will otherwise still support Java 8 with public updates for personal use indefinitely. Other vendors have begun to offer zero-cost builds of OpenJDK 8 and 11 that are still receiving security and other upgrades.

Oracle (and others) highly recommend uninstalling outdated and unsupported versions of Java, due to unresolved security issues in older versions. Oracle advises its users to immediately transition to a supported version, such as one of the LTS versions (8, 11, 17).

Chapter No 2

LITERATURE SURVEY

CHAPTER 2

LITERATURE SURVEY

2.1. RELATED WORKS

- **A Demetriou, L. Kyriakides, and C. Avraamidou, The missing link in the relations between intelligenceand personality** - The corporate world today does not focus just on the skills a potential employee possesses but also their personality. Personality is what helps one be successful in professional as well as personal life. Hence, the recruiter must be aware of the personality traits a person has. With an exponential increase in job seekers but a decrease in the number of jobs, it is difficult to manually shortlist the best fit candidate for a suitable job by looking at the CV. This paper attempts to examine different machine learning approaches for efficiently predicting personality through CV analysis using Natural Language Processing (NLP) techniques as well. Results show that the Random Forest algorithm achieved better accuracy when compared to other algorithms such as kNN, Logistic Regression, SVM and Naive Bayes.
- **M. Kalghatgi, M Ramannavar, and Dr. N. S. Sidnal, Neural Network approach to personality prediction based on the Big-Five Model** - The use of social networking websites has increased since last few decades. The social networking sites such as Twitter, Facebook, LinkedIn and YouTube are sources of human interaction, where the users are allowed to create and share their activities, thoughts and opinions pertaining to different subjects. These sites have not only connected large user populations but have also captured massive information associated with their daily interactions in the form of Big Data. This data provides unprecedented information about human behavior and social interactions. It makes it possible to understand who the users are, what their interests are and what they need. This information is vital for businesses to target potential consumers or seek customer opinions in the event of diversification as a business strategy. Thus, this work analyzes social media data to predict significant personality traits, i.e. qualities or characteristics specific to an individual, using the Big Five Model. The predictions can be applied to various domains like business intelligence, marketing, sociology and psychology. A parallelism between an individual's personality traits and his/her linguistic information is explored for analytics. Keywords— Big Five model, Hadoop, Personality, Multi-label classification, Neural Networks, Social Media.

- **A..Robey, K. Shukla, K. Agarwal, K. Joshi, Professor S. Joshi "Personality prediction system through CV Analysis** - Human Resource Management is apparently supported by and provided with more opportunities by the development of Job Characteristics Model (JCM) which in turn is based on the concept of modern job design. Fortunately, the development in modern information system, digital technologies, the universal access of electronic technology and internet led to the inclination of the global Human Resource Management development and make the system more applicable. Following the trend, the proposed system tries to design a plan to integrate Job Characteristics Model into E-HR system to search for a new model of efficient operation on Human Resource Management in the Internet Age. In this project, we present a set of techniques that makes the whole recruitment process more effective and efficient. We have implemented a system that ranks the candidates based on weight-age policy as well as an aptitude test. Today there is a growing interest in the personality traits of a candidate by the organization to better examine and understand the candidate's response to similar circumstances. Therefore, the system conducts a personality prediction test to determine the personality traits of the candidate. Finally, it presents the results of the candidates to the recruiter who evaluates the top candidates and shortlists the candidate.
- **J. Zubeda, M. Shaheen, G. Narsayya Godavari, and S. Naseem "Resume Ranking using NLP and Machine Learning** - Using NLP(Natural Language Processing) and ML(Machine Learning) to rank the resumes according to the given constraint, this intelligent system ranks the resume of any format according to the given constraints or the following requirement provided by the client company. We will basically take the bulk of input resume from the client company and that client company will also provide the requirement and the constraints according to which the resume should be ranked by our system. Beside the information provide by the resume we are going to read the candidates social profiles (like LinkedIn, Github etc) which will give us the more genuine information about that candidate.

Chapter No 3

METHODOLOGY / PROPOSED SOLUTION

CHAPTER 3

METHODOLOGY / PROPOSED SOLUTION

3.1. PROPOSED SOLUTION

The majority of people use third-party portals to create their CVs and personality evaluations. However, the problem with such portals is that they charge for them, and there is no guarantee that the personality results obtained from them are correct.

3.2. SOLUTION

The proposed system would provide a more efficient way to evaluate the candidates. The system will list the experience and key skills needed for a specific job role. The model would allow the HR department to easily create a shortlist of candidates based on the qualifications needed for the job role.

3.3. WORKING OF THE SYSTEM

The proposed system will be developed as a web application. The admin will first need to login with proper credentials after which they can add aptitude and personality test questions and can also modify them. For each aptitude question, a correct answer is stored in the database. Next, the candidate will have to register him/herself with all the details and will also have to fill their own CV details into the system. The candidate will have to give an aptitude test after which scores are stored in databases. Next the candidate will have to appear for a personality test. In this test various situations will be encountered by the candidate during which their personality will be judged on factors like Openness, Conscientiousness, Extraversion, Acceptability, and Neuroticism. Each question has the fix set of choices varying from strongly agree to strongly disagree.

3.4. SYSTEM ARCHITECTURE

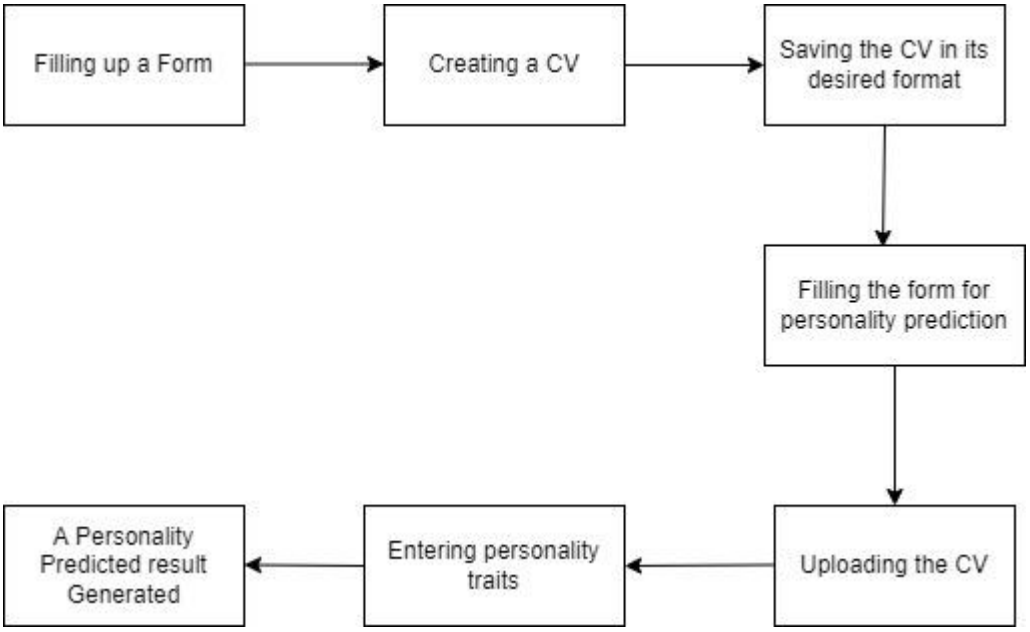


Fig 3.1 System Architecture

The primary focus of the system software is to provide user an interface to get an instant result of Personality Prediction. The user just has to provide their information and the model will generate the results.

The above architecture describes the complete view of the software and its usage in the perfect manner. It also shows the working of all the actors.

3.5. FLOW CHART

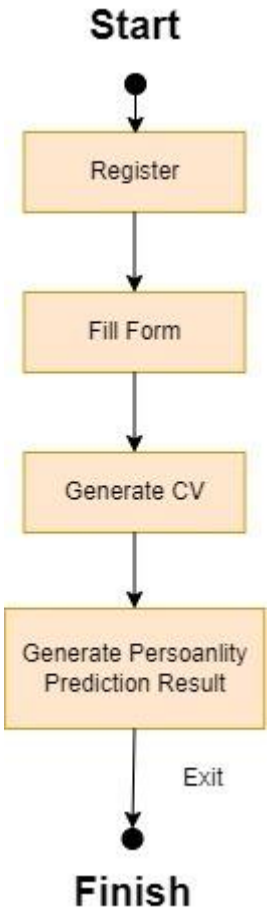


Fig 3.2 Flow chart

Above we have shown the flow of our System software in the project. In this, the user just has to provide their data and the model will generate the results of user.

Chapter No 4

TOOLS/PLATFORM

CHAPTER 4

TOOLS/PLATFORM

4.1 SOFTWARE REQUIREMENT

a) **TECHNOLOGY:** Python, Java, R

b) **IDE:** Visual Studio, Jupyter Notebook, NetBeans.

c) **LIBRARIES:** Pandas, Matplotlib, Numpy, Tkinter, Functools, Pyresparser, Sklearn:

d) **DATABASE:** MS-Excel

e) **OPERATING SYSTEM:** Windows 7 (Preferred)

f) **DESIGNING TOOLS:** Draw.io

g) **TESTING TOOL:** MS-EXCEL

a) TECHNOLOGY:

- **Python:**

  Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. Like Perl, Python source code is also available under the GNU General Public License (GPL). Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a battery included language due to its comprehensive standard library.

b) IDE:

- **VSCode:**

  Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

- **NetBeans:**

  NetBeans is an integrated development environment (IDE) for Java. NetBeans allows applications to be developed from a set of modular software components called modules. NetBeans runs on Windows, macOS, Linux and Solaris. In addition to Java development, it has extensions for other languages like PHP, C, C++, HTML5 and JavaScript. Applications based on NetBeans, including the NetBeans IDE, can be extended by third party developers.

- **Jupyter Notebook:**

  Jupyter Notebook (formerly IPython Notebooks) is a web-based interactive computational environment for creating notebook documents.

  A Jupyter Notebook document is a browser-based REPL containing an ordered list of input/output cells which can contain code, text (using Markdown), mathematics, plots and rich media. Underneath the interface, a notebook is a JSON document, following a versioned schema, usually ending with the ". ipynb" extension.

  Jupyter notebooks are built upon a number of popular open-source libraries:

  - IPython
  - ZeroMQ
  - Tornado
  - jQuery
  - Bootstrap (front-end framework)
  - MathJax

c) Libraries used:

- **Pandas:**

  Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license. The name is derived from the term "panel data", an econometrics term for data sets that include observations over multiple time periods for the same individuals. Its name is a play on the phrase "Python data analysis" itself. Wes McKinney started building what would become pandas at AQR Capital while he was a researcher there from 2007 to 2010.

- **Matplotlib:**

  Matplotlib is one of the most popular Python packages used for data

visualization. It is a cross-platform library for making 2D plots from data in arrays. Matplotlib is written in Python and makes use of NumPy, the numerical mathematics extension of Python. It provides an object-oriented API that helps in embedding plots in applications using Python GUI toolkits such as PyQt, Tkinter. It can beused in Python and IPython shells, Jupyter notebook and web application servers also.

- **Scikit-Learn:**

  Scikit-learn is probably the most useful library for machine learning in Python.The sklearn library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction.Please note that sklearn is used to build machine learning models. It should not be used for reading the data, manipulating and summarizing it. There are better libraries for that (e.g. NumPy, Pandas etc.)

- **NumPy:**

  NumPyis a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. The ancestor of NumPy, Numeric, was originally created by Jim Hugunin with contributions from several other developers. In 2005, Travis Oliphant created NumPy by incorporating features of the competing Numarray into Numeric, with extensive modifications. NumPy is open-source software and has many contributors. NumPy is a NumFOCUS fiscally sponsored project.

- **Tkinter:**

  Tkinter is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit, and is Python's de facto standard GUI. Tkinter is included with standard GNU/Linux, Microsoft Windows and macOS installs of Python.

  The name Tkinter comes from Tk interface. Tkinter was written by Steen Lumholt and Guido van Rossum, then later revised by Fredrik Lundh.

  Tkinter is free software released under a Python license.

- **Functools:**

  The functools module is for higher-order functions: functions that act on or return other functions. In general, any callable object can be treated as a function for the purposes of this module.

d) OPERATING SYSTEM:

Any Operating System can handle the work easily and Efficiently.

e) DESIGNING TOOLS:

- **Draw.io**

  Draw.io is completely free online diagram editor built around Google Drive (TM) that enables you to create flowcharts, UML, entity relation, networkdiagrams, mockups and more.

- **Features**:
  - Collaboration Tools
  - Data Import/Export
  - Drag & Drop Interface
  - Third Party Integration

4.2 HARDWARE REQUIREMENT

a) Processor – dual core processor

b) RAM – 512mb ram

c) **Operating System -** No specific OS required

Chapter No 5

DESIGN & IMPLEMENTATION

CHAPTER 5

DESIGN & IMPLEMENTATION

5.1. SYSTEM DESIGN

5.1.1. USE CASE DIAGRAM

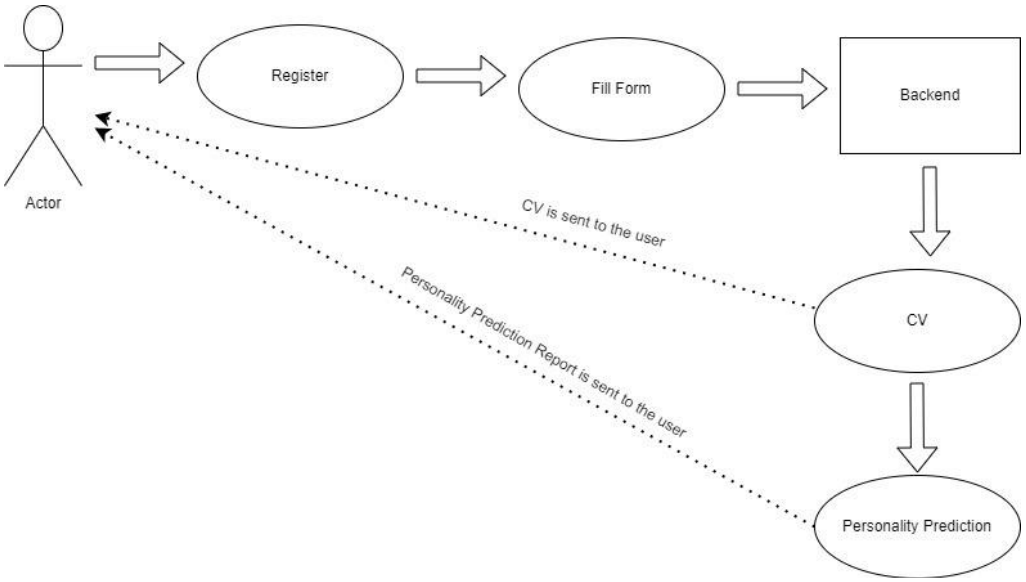


Fig 5.1 Use case diagram

The above use case diagram simply depicts the task which a user can perform through our application.

These tasks are as follows:

- User can register and login to the application.
- User can fill the form which will generate CV
- User can view the result.

5.1.2. CLASS DIAGRAM

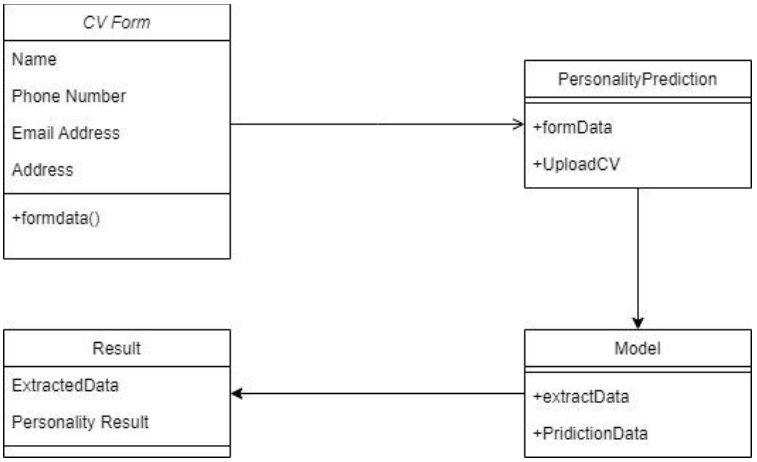


Fig 5.2 Class Diagram

In the above class diagram, the whole working of our software is represented in a certain manner. The CV form class is responsible for registration and validation of login info as well as sending data provided by the user and retrieve the final results. The CV Form class is responsible for the systematically entering data from the user. The Model class will generate the results based on the input.

5.1.3. SEQUENCE DIAGRAM

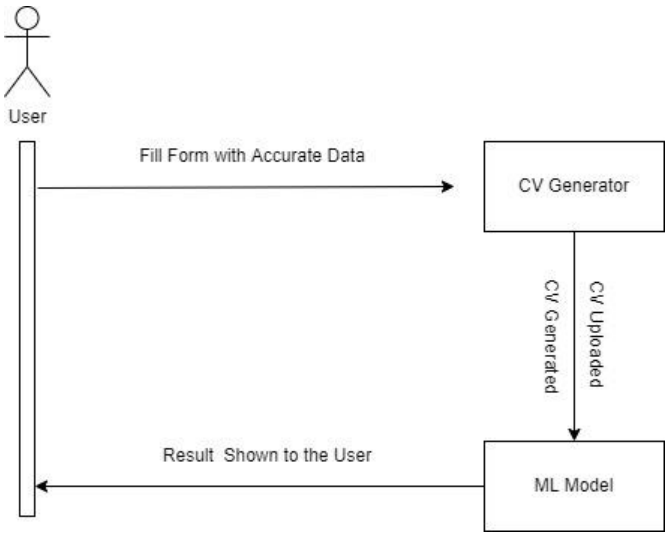


Fig 5.3 Sequence Diagram

5.2. IMPLEMENTED MODULES

We have completed the Dataset Gathering and Processing, Model training. We also completed the GUI Designing implementation using Java.In this module, we have completed designing the user interface of CV form and Personality Prediction Form

Dataset Gathering and Pre-processing: We have used Kaggle dataset for training of our model. The dataset consists of approx. 10000 different test cases for each type of disease. A dataset is collected and preprocessed accordingly for training and testing purpose of the model.

Model Training: The dataset is fed to the model for training and the test cases are run to create the output. The test cases are then verified and the accuracy is checked and improved.

GUI Design and Deployment: We have developed the application using Java to create our CV generation part and then we have used the Tkinter Library for Developing the UI of the Personality Prediction form.

5.3. SAMPLE CODE

Python program for creating Model:

```
import os
from turtle import bgcolor
import pandas as pd
import numpy as np
from tkinter import *
from tkinter import filedialog
import tkinter.font as font
from functools import partial
from pyresparser import ResumeParser
from sklearn import datasets, linear_model
#import doctest
class train_model:

    def train(self):
        data =pd.read_csv('training_dataset.csv')
        array = data.values

        for i in range(len(array)):
            if array[i][0]=="Male":
                array[i][0]=1
            else:
                array[i][0]=0

        df=pd.DataFrame(array)

        maindf =df[[0,1,2,3,4,5,6]]
        mainarray=maindf.values

        temp=df[7]
        train_y =temp.values

        self.mul_lr =
linear_model.LogisticRegression(multi_class='multinomial', solver='newton-
```

```
cg',max_iter =1000)
        self.mul_lr.fit(mainarray, train_y)

    def test(self, test_data):
        try:
            test_predict=list()
            for i in test_data:
                test_predict.append(int(i))
            y_pred = self.mul_lr.predict([test_predict])
            return y_pred
        except: print("All Factors For Finding Personality Not Entered!")

def check_type(data):
    if type(data)==str or type(data)==str:
        return str(data).title()
    if type(data)==list or type(data)==tuple:
        str_list=""
        for i,item in enumerate(data):
            str_list+=item+", "
        return str_list
    else:   return str(data)

def prediction_result(top, aplcnt_name, cv_path, personality_values):
    "after applying a job"
    top.withdraw()
    applicant_data={"Candidate Name":aplcnt_name.get(),  "CV
Location":cv_path}

    age = personality_values[1]

    print("\n############## Candidate Entered Data #############\n")
    print(applicant_data, personality_values)

    personality = model.test(personality_values)
    print("\n############## Predicted Personality #############\n")
    print(personality)
    data = ResumeParser(cv_path).get_extracted_data()
```

```
    try:
        del data['name']
        if len(data['mobile_number'])<10:
            del data['mobile_number']
    except:
        pass

    print("\n############## Resume Parsed Data #############\n")

    for key in data.keys():
        if data[key] is not None:
            print('{} : {}'.format(key,data[key]))

    result=Tk()
  #  result.geometry('700x550')
    result.overrideredirect(False)
    result.geometry("{0}x{1}+0+0".format(result.winfo_screenwidth(),
result.winfo_screenheight()))
    result.configure()
    result.title("Predicted Personality")

    #Title
    titleFont = font.Font(family='Arial', size=40, weight='bold')
    Label(result, text="Result - Personality Prediction", font=titleFont,
pady=10, anchor=CENTER).pack(fill=BOTH)

    Label(result, text = str('{} : {}'.format("Name:",
aplcnt_name.get())).title(), anchor='w').pack(fill=BOTH)
    Label(result, text = str('{} : {}'.format("Age:", age)),
anchor='w').pack(fill=BOTH)
    for key in data.keys():
        if data[key] is not None:
            Label(result, text = str('{} :
{}'.format(check_type(key.title()),check_type(data[key]))), anchor='w',
width=60).pack(fill=BOTH)
    print(personality)
    Label(result, text = str("perdicted personality: "+personality).title(),
anchor='w').pack(fill=BOTH)
```

```
    quitBtn = Button(result, text="Exit", command =lambda:
result.destroy()).pack()

    terms_mean = """
# Openness:
    People who like to learn new things and enjoy new experiences usually
score high in openness. Openness includes traits like being insightful and
imaginative and having a wide variety of interests.

# Conscientiousness:
    People that have a high degree of conscientiousness are reliable and
prompt. Traits include being organised, methodic, and thorough.

# Extraversion:
    Extraversion traits include being; energetic, talkative, and assertive
(sometime seen as outspoken by Introverts). Extraverts get their energy and
drive from others, while introverts are self-driven get their drive from
within themselves.

# Agreeableness:
    As it perhaps sounds, these individuals are warm, friendly, compassionate
and cooperative and traits include being kind, affectionate, and sympathetic.
In contrast, people with lower levels of agreeableness may be more distant.

# Neuroticism:
    Neuroticism or Emotional Stability relates to degree of negative
emotions. People that score high on neuroticism often experience emotional
instability and negative emotions. Characteristics typically include being
moody and tense.
"""

    Label(result, text = terms_mean, anchor='w',
justify=LEFT).pack(fill=BOTH)

    result.mainloop()
```

```
def perdict_person():
    """Predict Personality"""

    # Closing The Previous Window
    root.withdraw()

    # Creating new window
    top = Toplevel()
    top.geometry('700x500')
    top.configure()
    top.title("Apply For A Job")

    #Title
    titleFont = font.Font(family='Helvetica', size=20, weight='bold')
    lab=Label(top, text="Personality Prediction",font=titleFont,
pady=10).pack()

    #Job_Form
    job_list=('Select Job', '101-Developer at TTC', '102-Chef at Taj', '103-
Professor at MIT')
    job = StringVar(top)
    job.set(job_list[0])

    l1=Label(top, text="Applicant Name").place(x=70, y=130)
    l2=Label(top, text="Age").place(x=70, y=160)
    l3=Label(top, text="Gender").place(x=70, y=190)
    l4=Label(top, text="Upload Resume").place(x=70, y=220)
    l5=Label(top, text="Enjoy New Experience or thing(Openness)").place(x=70,
y=250)
    l6=Label(top, text="How Offen You Feel
Negativity(Neuroticism)").place(x=70, y=280)
    l7=Label(top, text="Wishing to do one's work well and
thoroughly(Conscientiousness)").place(x=70, y=310)
    l8=Label(top, text="How much would you like work with your
peers(Agreeableness)").place(x=70, y=340)
    l9=Label(top, text="How outgoing and social interaction you
like(Extraversion)").place(x=70, y=370)
```

```
    sName=Entry(top)
    sName.place(x=450, y=130, width=160)
    age=Entry(top)
    age.place(x=450, y=160, width=160)
    gender = IntVar()
    R1 = Radiobutton(top, text="Male", variable=gender, value=1, padx=7)
    R1.place(x=450, y=190)
    R2 = Radiobutton(top, text="Female", variable=gender, value=0, padx=3)
    R2.place(x=540, y=190)
    cv=Button(top, text="Select File", command=lambda:  OpenFile(cv))
    cv.place(x=450, y=220, width=160)
    openness=Entry(top)
    openness.insert(0,'1-10')
    openness.place(x=450, y=250, width=160)
    neuroticism=Entry(top)
    neuroticism.insert(0,'1-10')
    neuroticism.place(x=450, y=280, width=160)
    conscientiousness=Entry(top)
    conscientiousness.insert(0,'1-10')
    conscientiousness.place(x=450, y=310, width=160)
    agreeableness=Entry(top)
    agreeableness.insert(0,'1-10')
    agreeableness.place(x=450, y=340, width=160)
    extraversion=Entry(top)
    extraversion.insert(0,'1-10')
    extraversion.place(x=450, y=370, width=160)

    submitBtn=Button(top, padx=2, pady=0, text="Submit", bd=0,font=(12),
foreground='white', bg='red')
    submitBtn.config(command=lambda:
prediction_result(top,sName,loc,(gender.get(),age.get(),openness.get(),neurot
icism.get(),conscientiousness.get(),agreeableness.get(),extraversion.get())))
    submitBtn.place(x=350, y=400, width=200)



    top.mainloop()

def OpenFile(b4):
```

```
    global loc;
    name =
filedialog.askopenfilename(initialdir="C:/Users/Amaan/Downloads/MicroProject/
KnowYourself-A Personality Prediction System/KnowYourself-A Personality
Prediction System",
                           filetypes
=(("Document","*.docx*"),("PDF","*.pdf*"),('All files', '*')),
                           title = "Choose a file."
                           )
    try:
        filename=os.path.basename(name)
        loc=name
    except:
        filename=name
        loc=name
    b4.config(text=filename)
    return


if __name__ == "__main__":
    model = train_model()
    model.train()

    root = Tk()
    root.geometry('700x500')
    root.configure()
```

Chapter No 6

TESTING, RESULTS & DISCUSSION

CHAPTER 6

TESTING, RESULTS & DISCUSSION

6.1. TESTING

6.1.1 TYPES OF TESTING

- **Manual Testing**

Manual testing includes testing a software manually, i.e., without using any automated tool or any script. In this type, the tester takes over the role of an end-user and tests the software to identify any unexpected behavior or bug. There are different stages for manual testing such as unit testing, integration testing, system testing, and user acceptance testing.

Testers use test plans, test cases, or test scenarios to test a software to ensure the completeness of testing. Manual testing also includes exploratory testing, as testers explore the software to identify errors in it.

Following are the testing techniques that are performed manually during the test life cycle:

- Acceptance Testing
- White Box Testing
- Black Box Testing
- Unit Testing
- System Testing
- Integration Testing

- **Automation Testing**

Automation testing, which is also known as Test Automation, is when the tester writes scripts and uses software to test the product. This process involves automation of a manual process. Automation Testing is used to re-run the test scenarios that were performed manually, quickly, and repeatedly.

Apart from regression testing, automation testing is also used to test the application from load, performance, and stress point of view.

Test Automation should be used by considering the following aspects of a software:

- Large and critical projects
- Projects that require testing the same areas frequently
- Requirements not changing frequently
- Accessing the application for load and performance with many virtual users
- Stable software with respect to manual testing
- Availability of time

6.1.2 LEVELS OF TESTING

There are four levels of testing: Unit, Integration, System and Acceptance.

1. **Unit Testing:** A level of the software testing process where individual units/components of a software/system are tested. The purpose is to validate that each unit of the software performs as designed.

2. **Integration Testing:** A level of the software testing process where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units
   .
3. **System Testing:** A level of the software testing process where a complete, integrated system/software is tested. The purpose of this test is to evaluate the system's compliance with the specified requirements.

4. **Acceptance Testing:** A level of the software testing process where a system is tested for acceptability. The purpose of this test is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery**.**

6.1.3 TESTING REPORT

PROJECT:	KnowYourSelf
MODULE:	CV Generator Form & Personality Prediction
PREPARED BY:	Amaan Ranapurwala
	Harsh Maroo
	Yash Mishra
	Aniruddha Kate

SR.NO.	TEST CASE ID	TEST OBJECTIVE	STEPS	DATA	PREREQUISITE	EXPECTED RESULT	ACTUAL RESULT	STATUS
1	TC_CV_FORM_FILL	To check whether the data is inserted properly or not	1. open application 2.Enter required details	Name , Contact no , Gender, etc		after successful insertion of data , the CV should be generated	It is working properly	Pass
2	TC_CV_DATA	To check whether the data entered in the CV form is saved or not	1. click the Generate CV button		There must be a proper format for generating CV	Data should be properly saved and a CV should be generated in a word document	Data saved and cv generated	Pass
3	TC_CV_FORMAT	To check whether the CV generated is in proper format or not	1.click on generate CV		Proper format required for generation of CV	A CV should be generated in a proper format	CV generated in proper format	Pass
4	TC_PREDICTION_VALUES	To check whether the values are correctly taken for personality	1. choose appropriate option for the parameters	values from 1-10 for the 5 parameters		According to the values the personality should be predicted	Personality predicted according to the values.	Pass
5	TC_DATA_EXTRACTION_FROM_CV	To check whether the data is extracted from the generated CV or not	1. Click on predict personality			Predicted personality should be according to the parameters and the data from the CV	Personality is predicted according to the parameters and CV	Pass
6	TC_PERSONALITY_PREDICTION	To check whether the Personality is predicted or not	1.Enter data into CV generation form 2.Generate CV 3.Give values to parameters from 1-10 4.Click Predict Personality			Personality should be predicted according to the supplied bdata	Personality is predicted according to the data	Pass

6.2. RESULTS AND DISCUSSION

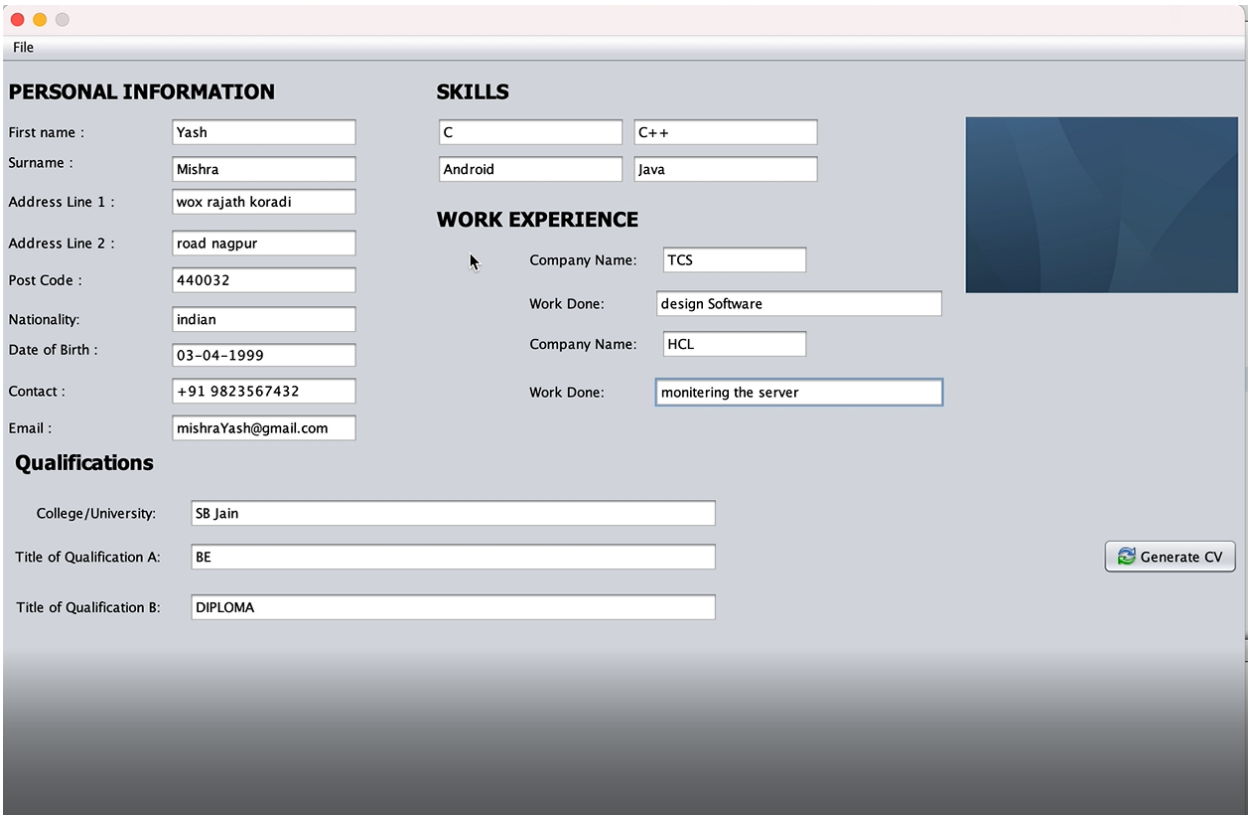


Fig 6.1 CV Form

The above figure shows that the user has entered the required information in the CV generator Form.

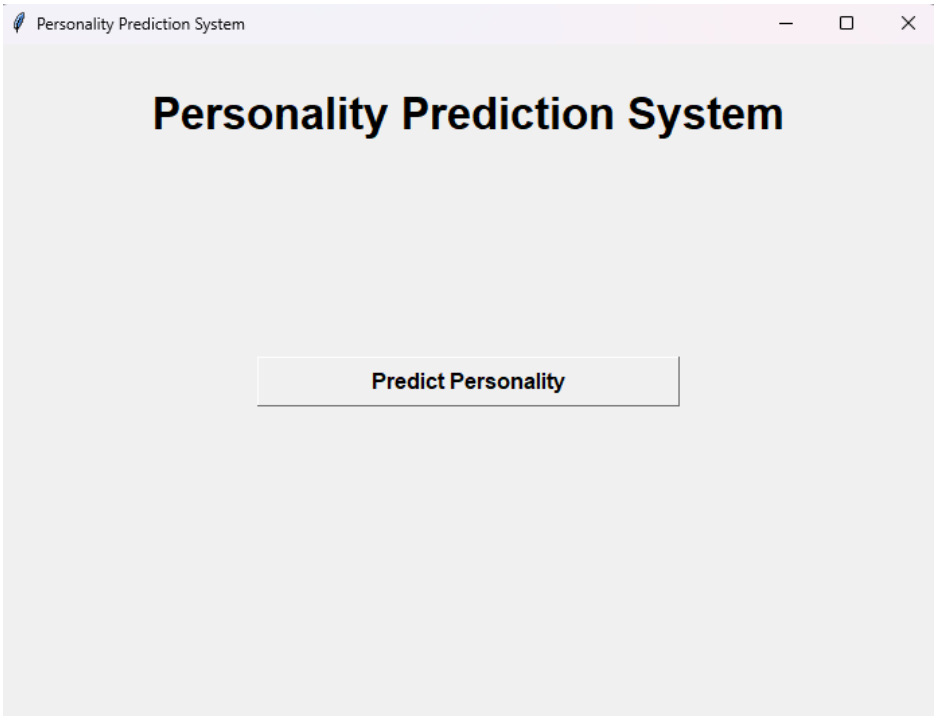


Fig 6.2 Personality Prediction System

This the home page for the Personality Prediction System

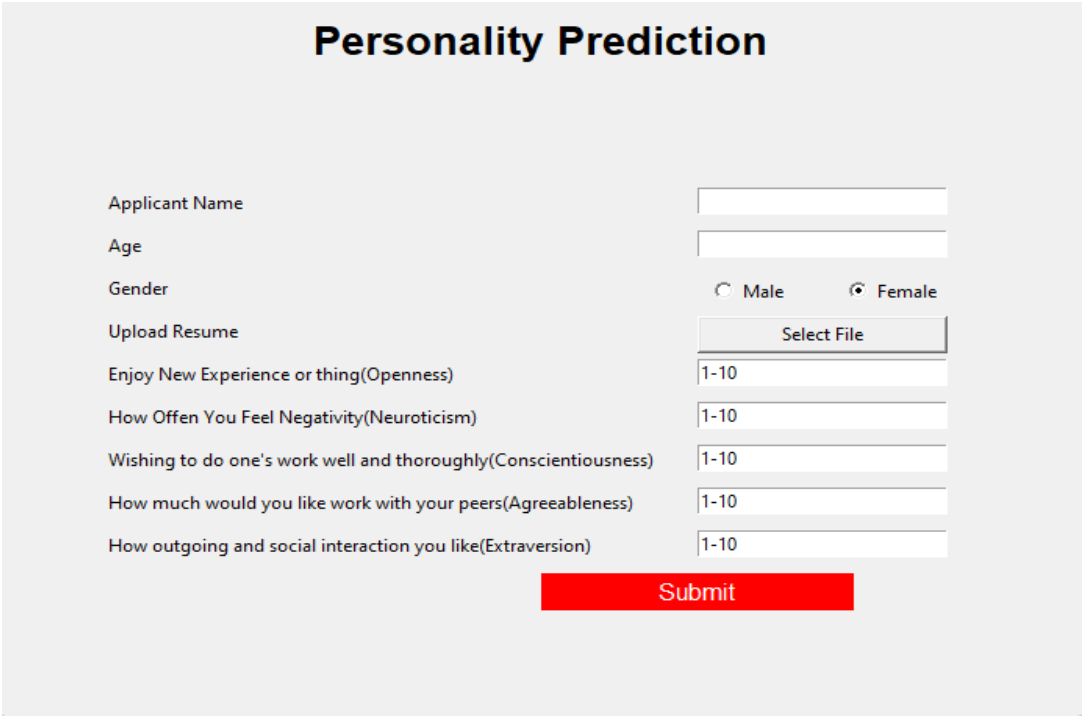


Fig 6.3 Prediction Form

Here the user will enter the data according the parameters and upload the generated Resume

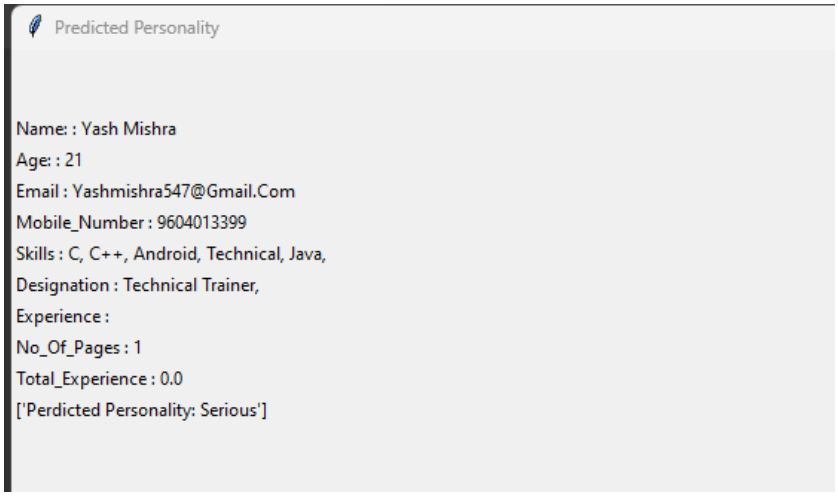


Fig 6.4 Result

This is the Result page, containing the CV information as well as the Predicted Personality

Chapter No 7

ADVANTAGES AND APPLICATIONS

CHAPTER 7

ADVANTAGES AND APPLICATIONS

7.1. ADVANTAGES

- Individuals receive an instant resume.
- It is handy for consumers and reduces the time spent thinking and generating resumes.
- Simpler than the majority of third-party applications on the internet.
- The resumes generated here are more accurate than those seen on most other websites.
- The CVs that are created are free of charge.

7.2. APPLICATIONS

- This system can be used by anybody in order to generate their CV and determine their personality.
- With an accuracy of 85.81 percent, it may be used to predict a person's personality.
- Used to match the appropriate candidate to the suitable job based on his personality and qualifications.
- It can also be used it to match marital profiles.

Chapter No 8

CONCLUSION & FURTHER SCOPE

CHAPTER 8

CONCLUSION & FURTHER SCOPE

8.1. CONCLUSION

We have successfully developed our project " KnowYourSelf - A Personality Prediction System " which is able to generate a CV and predict personality according to the personality parameters as well as the CV generated. We have applied **engineering knowledge** to **analyze the difficulties** in our algorithm and developed a user-friendly AI in both the forms . We will also refer other research papers and articles which are available and try to make our project more efficient and accurate in the future. We have used Python, Java and R for Machine Learning During this project we understood the importance of **team work , communication, work planning** , which will surely help us in future.

8.2. FURTHER SCOPE

- The proposed system can be used by various companies in order to streamline the recruitment process by considering the personality of potential candidates.
- Our algorithm is now limited to working on a resume that we have created in our format, but in the future, it will be able to function on any resume in any format.
- Our project now extracts specific keywords from CVs to obtain information, but in the future, our programme will be capable of extracting data from CVs.
- Future work can also be done to increase the efficiency and performance of the proposed system in order to predict personality using CV analysis more accurately.

REFERENCES

PAPERS

- A Demetriou, L. Kyriakides, and C. Avraamidou, “The missing link in the relations between intelligenceand personality” in Journal of Research in Personality,vol. 37 issue 6, December 2003, pp 547- 581.

- M. Kalghatgi, M Ramannavar, and Dr. N. S. Sidnal, “Neural Network approach to personality prediction based on the Big-Five Model” in IJIRAE, vol2 issue 8, August 2015, pp 56-63.

- A..Robey, K. Shukla, K. Agarwal, K. Joshi, Professor S. Joshi “Personality prediction system through CV Analysis, in IRJET vol 6, issue 02, February2019.

- J. Zubeda, M. Shaheen, G. Narsayya Godavari, and S. Naseem “Resume Ranking using NLP and Machine Learning”, unpublished

- Md Tanzim Reza, and Md. Sakib Zaman, “Analyzing CV/Resume using natural language processing and machine learning”, unpublished.

WEBSITES

1. https://towardsdatascience.com/

2. https://spacy.io/

3. https://www.kaggle.com/

APPENDIX I

PLAGIARISM REPORT

100%

Plagiarism	Unique
0%	100%

Extra Word Count Statistics

Syllables	398	Average Sentence Length (word)	17.2
Sentences	11	Syllables Per Word(s)	2.1
Unique Word(s)	115 (61%)	Paragraph(s)	3
Average Word Length (characters)	5.9	Difficult Word(s)	100 (53%)

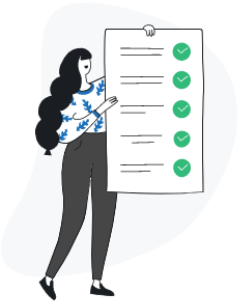
Reading Time 1 min Speak Time 2 mins

The majority of people build their CVs and personality assessments via third-party portals. The problem with such websites is that they charge a fee, and there is no guarantee that the personality results gained are accurate.

To address this issue, we created software that would help candidates build their CVs by asking them a few questions and identifying their personality. Five characteristics of different individuals commonly known as big five characteristics namely, openness, neuroticism, conscientiousness, agreeableness and extraversion are stored in a dataset and used for training. Based on this training, the personality of individuals are predicted using data mining concepts. Before testing the dataset, it is pre-processed using different data mining concepts like handling missing values, data discretization, normalisation etc. This pre-processed data can then be used to classify/predict user personality based on past classifications. The system analyses user characteristics and behaviours. System then predicts new user personality based on personality data stored by classification of previous user data.
Model used to predict test dataset is "Logistic Regression" because Logistic regression is an effective model to predict output class labels for dependent categorical data.

Text To Speech

Total Words: 185 Total Characters: 1303



Congratulation!

No Plagiarism Found

APPENDIX II

Instructional Manual

On

"KnowYourSelf" – A Personality Prediction System

Submitted By

Mr. Amaan Ranapurwala **Mr. Harsh Maroo**

Mr. Yash Mishra **Mr. Aniruddha Kate**

Submitted in partial fulfillment of the requirements
Of
Mini Project in third year of Bachelor of Engineering

Guided by-
Ms. Titiksha Bhagat

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

S.B. JAIN INSTITUTE OF TECHNOLOGY, MANAGEMENT & RESEARCH, NAGPUR.

2021-2022

ABSTRACT

The majority of people build their CVs and personality assessments via third-party portals. The problem with such websites is that they charge a fee, and there is no guarantee that the personality results gained are accurate. To address this issue, we created software that would help candidates build their CVs by asking them a few questions and identifying their personality. Five characteristics of different individuals commonly known as big five characteristics namely, openness, neuroticism, conscientiousness, agreeableness and extraversion are stored in a dataset and used for training. Based on this training, the personality of individuals are predicted using data mining concepts. Before testing the dataset, it is pre-processed using different data mining concepts like handling missing values, data discretization, normalisation etc. This pre-processed data can then be used to classify/predict user personality based on past classifications. The system analyses user characteristics and behaviours. System then predicts new user personality based on personality data stored by classification of previous user data. Model used to predict test dataset is "Logistic Regression" because Logistic regression is an effective model to predict output class labels for dependent categorical data.

Key Words— Personality, Machine Learning, Logistic Regression, Ocean Model.

SOFTWARE REQUIREMENT

a) **TECHNOLOGY:** Python, Java, R

b) **IDE:** Visual Studio, Jupyter Notebook, NetBeans.

c) **LIBRARIES:** Pandas, Matplotlib, Numpy, Tkinter, Functools, Pyresparser, Sklearn:

d) **DATABASE:** MS-Excel

e) **OPERATING SYSTEM:** Windows 7 (Preferred)

f) **DESIGNING TOOLS:** Draw.io

g) **TESTING TOOL:** MS-EXCEL

f) TECHNOLOGY:

- **Python:**

  Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. Like Perl, Python source code is also available under the GNU General Public License (GPL). Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a battery included language due to its comprehensive standard library.

g) IDE:

- **VSCode:**

  Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

- **NetBeans:** NetBeans is an integrated development environment (IDE) for Java. NetBeans allows applications to be developed from a set of modular software components called modules. NetBeans runs on Windows, macOS, Linux and Solaris. In addition to Java development, it has extensions for other languages like PHP, C, C++, HTML5,[3] and JavaScript. Applications based on NetBeans, including the NetBeans IDE, can be extended by third party developers.
- **Jupyter Notebook:** Jupyter Notebook (formerly IPython Notebooks) is a web-based interactive computational environment for creating notebook documents.

  A Jupyter Notebook document is a browser-based REPL containing an ordered list of input/output cells which can contain code, text (using Markdown), mathematics, plots and rich media. Underneath the interface, a notebook is a JSON document, following a versioned schema, usually ending with the ".ipynb" extension.

  Jupyter notebooks are built upon a number of popular open-source libraries:
  - IPython
  - ZeroMQ
  - Tornado
  - jQuery
  - Bootstrap (front-end framework)
  - MathJax

h) Libraries used:

- **Pandas:** pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license. The name is derived from the term "panel data", an econometrics term for data sets that include observations over multiple time periods for the same individuals. Its name is a play on the phrase "Python data analysis" itself. Wes McKinney started building what would become pandas at AQR Capital while he was a researcher there from 2007 to 2010.
- **Matplotlib :**

  Matplotlib is one of the most popular Python packages used for data visualization. It is a cross-platform library for making 2D plots from data in arrays. Matplotlib is written in Python and makes use of NumPy, the numerical mathematics extension of Python. It provides an object-oriented API that helps in embedding plots in applications using Python GUI toolkits such as PyQt, WxPythonotTkinter. It can be used in Python and IPython shells, Jupyter notebook and web application servers also.

- **Scikit-Learn:**

  Scikit-learn is probably the most useful library for machine learning in Python. The sklearn library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction. Please note that sklearn is used to build machine learning models. It should not be used for reading the data, manipulating and summarizing it. There are better libraries for that (e.g. NumPy, Pandas etc.)

- **NumPy:** is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. The ancestor of NumPy, Numeric, was originally created by Jim Hugunin with contributions from several other developers. In 2005, Travis Oliphant created NumPy by incorporating features of the competing Numarray into Numeric, with extensive modifications. NumPy is open-source software and has many contributors. NumPy is a NumFOCUS fiscally sponsored project.

- **Tkinter:** Tkinter is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit, and is Python's de facto standard GUI. Tkinter is included with standard GNU/Linux, Microsoft Windows and macOS installs of Python.

  The name Tkinter comes from Tk interface. Tkinter was written by Steen Lumholt and Guido van Rossum, then later revised by Fredrik Lundh.

  Tkinter is free software released under a Python license.

- **Functools:** The functools module is for higher-order functions: functions that act on or return other functions. In general, any callable object can be treated as a function for the purposes of this module.

i) OPERATING SYSTEM:

Any Operating System can handle the work easily and Efficiently.

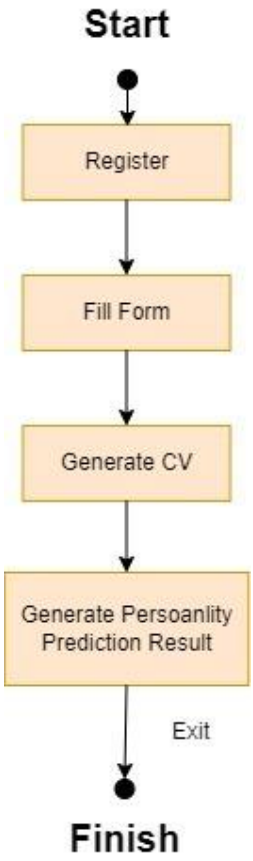
j) DESIGNING TOOLS:

- **Draw.io**

  Draw.io is completely free online diagram editor built around Google Drive(TM) that enables you to create flowcharts, UML, entity relation, network diagrams, mockups and more.

- **Features**:
    - Collaboration Tools
    - Data Import/Export
    - Drag & Drop Interface
    - Third Party Integration

Flow Chart



1. Steps to Run the Project

- Install the software
- Fill the CV form with suitable details
- Click on generate CV
- CV in a word document will be generated
- Click on Predict Personality
- Enter the details in Personality Prediction window
- Click Next
- Personality Prediction result would be generated.

2. Future Scope

- The proposed system can be used by various companies in order to streamline the recruitment process by considering the personality of potential candidates.
- Our algorithm is now limited to working on a resume that we have created in our format, but in the future, it will be able to function on any resume in any format.
- Our project now extracts specific keywords from CVs to obtain information, but in the future, our programme will be capable of extracting data from CVs.
- Future work can also be done to increase the efficiency and performance of the proposed system in order to predict personality using CV analysis more accurately.
  .

3. Limitations

- Our algorithm is now limited to working on a resume that we have created in our format, but in the future, it will be able to function on any resume in any format.
- Our project now extracts specific keywords from CVs to obtain information, but in the future, our programme will be capable of extracting data from CVs.