**AI-Based Career Path Predictor Using Machine Learning**

# Submitted by

Shaik Mohammed Junaid

B.Tech CSE.

Contact:

Ph :- +91 8790039170

Email :- junaidbuss18@gmail.com

# 1. Objective

The goal of this project is to build an AI model that predicts the best career path for a student based on academic scores, skills, and interest area using machine learning techniques.

# 2. Tools & Technologies Used

* Python
* Pandas, Scikit-learn
* Random Forest Classifier
* Streamlit (Web App)
* CSV File Handling
* VS Code / Jupyter Notebook

# 3. Dataset Used

* Records: 1600 (Clean and Balanced)
* Features: Interest Area, Skills, 10th %, 12th %, UG %, Career Path
* Dataset Name: career\_data\_large.csv

# 4. Model Architecture

* Data Cleaning and Skill Encoding
* Label Encoding for interest area and output
* Combined all features for training
* Used RandomForestClassifier for multi-class prediction

# 5. Web Application (UI)

A front-end was built using Streamlit where users enter marks, interest area, and skills. The app returns a predicted career path using the trained model.

# 6. Results & Accuracy

* The trained model achieved 100% accuracy on the cleaned dataset
* Realistic predictions were validated with test inputs
* Output was displayed through a user-friendly web interface

# 7. Learnings & Takeaways

* Learned end-to-end AI pipeline: data collection, preprocessing, model training, and UIdevelopment
* Improved confidence in AI tools and Python libraries
* Understood how machine learning can be applied to real-world problems

# 8. Conclusion

This project enhanced my understanding of AI development and gave me hands-on experience in solving a real-world problem using ML and Python.

**9. Key Python Code (Simplified)** # Load and preprocess data df = pd.read\_csv("career\_data\_large.csv")

df['Skills'] = df['Skills'].apply(lambda x: [skill.strip() for skill in x.split(',')]) mlb = MultiLabelBinarizer() skills\_encoded = pd.DataFrame(mlb.fit\_transform(df['Skills']), columns=mlb.classes\_) le\_interest = LabelEncoder() df['Interest\_Encoded'] = le\_interest.fit\_transform(df['Interest Area'])

X = pd.concat([df[['10th %', '12th %', 'UG %', 'Interest\_Encoded']], skills\_encoded], axis=1) le\_target = LabelEncoder() y = le\_target.fit\_transform(df['Career Path'])

# Train model model = RandomForestClassifier() model.fit(X, y)

# Predict def predict\_career(input\_data):

interest\_encoded = le\_interest.transform([input\_data['Interest Area']])[0] skill\_vector = [1 if skill in input\_data['Skills'] else 0 for skill in mlb.classes\_] features = [input\_data['10th %'], input\_data['12th %'], input\_data['UG %'], interest\_encoded] + skill\_vector return le\_target.inverse\_transform(model.predict([features]))[0]

# 10. Screenshots

Terminal Screenshot: ( Files in the Folder & Run Streamlit)

A screenshot of a computer

AI-generated content may be incorrect.

Before Entering the Data -User Interface (**Career Path Predictor**)

A screenshot of a computer

AI-generated content may be incorrect.

After Entering the Data -User Interface (**Career Path Predictor**) A screenshot of a computer

AI-generated content may be incorrect.