

# **Infosys Springboard Virtual Internship**

**“Edu2Job Predicting Job Roles  
from Educational Background”**

Intelligent Career Prediction & Visualization System

**Name:** Thanuja Pinninti

**Date:** 30-December-2025

# **Index**

- **Introduction**
- **Objective**
- **Workflow**
- **Code Implementation**
- **Explanation of Code**
- **Output Screenshots**
- **Conclusion**

# 01.Introduction

In the modern digital era, career planning has become increasingly complex due to the rapid evolution of technology, emergence of new job roles, and changing industry requirements. Students and early professionals often face significant challenges in identifying suitable career paths that align with their educational background, academic performance, and skill sets. Despite having access to vast online resources, most individuals still rely on generic career advice, peer influence, or limited counseling services, which may not accurately reflect real-world job market trends or individual capabilities.

Traditional career guidance systems are largely manual, static, and subjective. They fail to leverage data-driven insights and do not provide personalized recommendations. As a result, many students either choose unsuitable career paths or remain unaware of alternative job roles that match their qualifications. This gap between education and employment outcomes highlights the need for an intelligent, automated, and scalable career recommendation system.

The **EDU2JOB – Intelligent Career Prediction System** is designed to address this challenge by utilizing **Machine Learning, data analytics, and visualization techniques** to predict suitable job roles based on a user's educational profile. The system collects structured academic data such as degree, specialization, CGPA, certifications, and skills, processes this data through preprocessing pipelines, and applies trained machine learning models to generate accurate job role predictions. Instead of providing a single rigid recommendation, the system suggests multiple ranked job roles along with confidence scores, offering users flexibility and better decision-making support.

In addition to prediction, EDU2JOB emphasizes **explainability and transparency** through visualizations and career insights. Interactive charts and comparative analysis help users understand how their profile relates to real career trends and similar profiles. This approach builds trust in the system and empowers users to explore alternative career pathways. The inclusion of user feedback further enhances the system by enabling continuous improvement of prediction quality.

The project also incorporates a comprehensive **admin dashboard**, allowing administrators to manage datasets, retrain models, review prediction logs, and analyze feedback. This feature reflects real-world machine learning system practices, where models evolve over time as data changes. By integrating frontend interfaces, backend APIs, machine learning models, visualization tools, and admin controls, EDU2JOB represents a complete end-to-end intelligent application.

Overall, this project demonstrates how data-driven techniques can be effectively applied to career guidance, bridging the gap between education and employment while providing scalable, reliable, and personalized career recommendations.

## 02.Objective

The primary objective of the **EDU2JOB – Intelligent Career Prediction System** is to develop a data-driven platform that predicts suitable job roles for users based on their educational background, academic performance, and skill set using machine learning techniques. The system aims to assist students and early professionals in making informed career decisions by providing personalized, accurate, and explainable job recommendations.

### 2.1 Primary Objectives

- To design and implement a machine learning-based system that predicts suitable job roles from user education data such as degree, specialization, CGPA, certifications, and skills.
- To recommend the **top 3–5 job roles** instead of a single outcome, along with confidence scores, enabling flexible career exploration.
- To bridge the gap between academic qualifications and industry job requirements through intelligent data analysis.
- To automate career guidance using real-world data patterns rather than manual or generic counseling approaches.

### 2.2 Functional Objectives

- To build a secure user authentication and profile management system for storing personal and academic details.
- To collect, validate, preprocess, and transform user input data into a machine-learning-ready format.
- To integrate trained machine learning models with backend APIs for real-time job role prediction.
- To store user prediction history and allow users to review past recommendations.
- To implement a feedback mechanism where users can rate the relevance of predictions.

### **2.3 Visualization & Insight Objectives**

- To provide visual representations such as bar charts and pie charts that show education-to-job role trends.
- To display career insights that explain why a particular job role is recommended.
- To compare user profiles with similar profiles to suggest alternative career pathways.
- To improve transparency and trust in predictions through explainable visual outputs.

### **2.4 Administrative Objectives**

- To develop an admin dashboard for managing datasets and monitoring system activity.
- To allow administrators to upload new datasets and retrain the machine learning model.
- To maintain logs of user activity, predictions, and feedback for evaluation and improvement.
- To ensure role-based access control between users and administrators.

### **2.5 Technical Objectives**

- To design a scalable and modular system architecture using frontend, backend, database, and ML components.
- To follow industry best practices in API design, security, and machine learning lifecycle management.
- To ensure system reliability, maintainability, and future extensibility.

## 03.Workflow

The **EDU2JOB – Intelligent Career Prediction System** follows a structured, end-to-end workflow that integrates user interaction, data processing, machine learning prediction, visualization, and administrative management. The workflow ensures secure data handling, accurate predictions, and continuous system improvement.

### **Step 1: User Registration and Login**

- Users register by providing basic details such as username, email, and password.
- Secure authentication mechanisms are used to protect user data.
- After successful login, users are redirected to their personalized career dashboard.
- Role-based access control distinguishes between **User** and **Admin** functionalities.

### **Step 2: Profile Creation and Education Data Input**

- Users enter their academic and professional details, including:
  - Degree and specialization
  - CGPA
  - Skills and certifications
- Frontend validation ensures completeness and correctness of inputs.
- Data is sent securely to the backend server via REST APIs.

### **Step 3: Data Preprocessing**

- The backend receives raw user input and prepares it for machine learning prediction.
- Preprocessing includes:
  - Handling missing values
  - Encoding categorical variables (degree, specialization, skills)
  - Normalizing numerical features such as CGPA

- This step ensures consistency between training data and prediction data.

#### **Step 4: Job Role Prediction**

- The preprocessed data is passed to the trained machine learning model.
- The model analyzes the input and predicts suitable job roles.
- Multiple job roles are ranked based on probability scores.
- The top recommendations are selected and sent to the frontend.

#### **Step 5: Result Display and Career Insights**

- Predicted job roles are displayed in a user-friendly dashboard.
- Each recommendation includes:
  - Job role name
  - Brief role description
  - Confidence score
- Career insights explain how the user's profile matches the recommended roles.

#### **Step 6: Visualization of Career Data**

- Interactive visualizations are generated using Chart.js.
- Visuals include:
  - Degree vs Job Role bar charts
  - Job domain distribution pie charts
- These graphs help users understand career trends and alternatives.

#### **Step 7: Feedback and Prediction History**

- Users can provide feedback on prediction relevance.
- Feedback data is stored in the database.
- Prediction history is saved for user reference and analysis.

### **Step 8: Admin Dashboard Operations**

- Admin logs into a secure admin dashboard.
- Admin can:
  - Upload new datasets
  - Retrain the ML model
  - View prediction logs
  - Analyze user feedback
- Admin actions ensure system accuracy and relevance over time.

### **Step 9: Model Update and Continuous Improvement**

- New data and feedback are analyzed.
- ML model is retrained when required.
- Updated model replaces the old one without affecting users.
- System performance improves continuously.



## 04.Code Implementation

The implementation of the **EDU2JOB – Intelligent Career Prediction System** follows a modular and scalable coding approach. The complete source code of the project is maintained in a version-controlled GitHub repository to ensure transparency, maintainability, and ease of future enhancement.

### 4.1 Technology Stack

- **Frontend:** HTML, CSS, JavaScript, Chart.js
- **Backend:** Python (Flask Framework)
- **Machine Learning:** Python, pandas, scikit-learn, joblib
- **Database:** MySQL
- **Version Control:** Git & GitHub

### 4.2 Project Structure

The project is structured into multiple layers to maintain separation of concerns:

- |  |               |
|--|---------------|
| • <b>Frontend</b>  | <b>Layer:</b> |
| Handles user interfaces, form validation, dashboards, and visualizations.                      |               |
| • <b>Backend</b>   | <b>Layer:</b> |
| Implements REST APIs for authentication, prediction, visualization data, and admin operations. |               |
| • <b>Machine Learning</b>  | <b>Layer:</b> |
| Responsible for model training, prediction, evaluation, and retraining.                        |               |
| • <b>Database</b>  | <b>Layer:</b> |
| Stores user data, prediction history, feedback, and logs.                                      |               |

### 4.3 Backend Implementation

The backend is implemented using the **Flask framework**. It exposes RESTful APIs that handle:

- User authentication and authorization
- Collection and validation of education data

- Job role prediction requests
- Retrieval of visualization data
- Admin operations such as dataset upload and model retraining

The backend communicates with the machine learning model using serialized model files (.pkl) loaded at runtime. Secure routing and role-based access control ensure that only authorized users can access sensitive operations.

#### **4.4 Machine Learning Implementation**

The machine learning component is implemented using **scikit-learn**. The process includes:

- Dataset loading and preprocessing
- Feature encoding and normalization
- Model training and evaluation
- Model serialization using joblib
- Real-time prediction through backend APIs

The trained model is integrated seamlessly into the backend, enabling efficient and accurate job role predictions.

#### **4.5 Frontend Implementation**

The frontend is developed using standard web technologies. It provides:

- Interactive forms for user input
- Dynamic dashboards for displaying predictions
- Visualizations using Chart.js
- Admin interfaces for dataset and model management

The frontend communicates with the backend through secure API calls and displays responses in a user-friendly manner.

#### **4.6 Database Implementation**

A relational database (MySQL) is used to store:

- User credentials and profiles
- Education and skill details
- Prediction history
- User feedback
- Admin logs

This structured storage enables easy retrieval, analysis, and scalability.

#### 4.7 Source Code Availability

The complete implementation of this project, including frontend, backend, machine learning models, and database scripts, is available on GitHub at the following link:

**GitHub Repository:** *[Insert your GitHub repository link here]*

The repository contains well-documented code, folder structure, and setup instructions to facilitate understanding and reuse.

#### 4.8 Code Quality and Best Practices

- Modular coding structure
- Meaningful variable and function naming
- Error handling and validation
- Secure authentication practices
- Version control using Git

**My project link:** <https://github.com/ThanujaPinninti/edu2job>

## 05.Explanation of Code

The **EDU2JOB – Intelligent Career Prediction System** is implemented using a modular architecture that separates frontend logic, backend processing, machine learning operations, and database management. Each module performs a specific function, making the system scalable, maintainable, and easy to understand.

### 5.1 Frontend Code Explanation

The frontend code is responsible for user interaction, data collection, and result presentation.

#### Key Functionalities

- User registration and login forms
- Education and skill input forms
- Career prediction dashboard
- Visualization dashboards
- Admin panel interface

#### Implementation Details

- **HTML** defines the structure of pages such as login, dashboard, prediction, and admin views.
- **CSS** ensures responsive design, layout alignment, and user-friendly styling.
- **JavaScript** handles:
  - Form validation
  - API requests to the backend
  - Dynamic content updates
  - Rendering of charts using Chart.js

#### Visualization Logic

- Chart.js is used to generate bar charts and pie charts.
- Data received from backend APIs is parsed and displayed dynamically.

- Charts update automatically based on user input and prediction results.

## **5.2 Backend Code Explanation**

The backend is developed using the **Flask framework** and acts as the core processing layer of the system.

### **Routing and API Design**

- RESTful APIs are defined for:
  - User authentication
  - Job role prediction
  - Visualization data retrieval
  - Feedback submission
  - Admin operations
- Each API performs validation before processing requests.

### **Authentication Logic**

- Secure login mechanisms are implemented using session/JWT management.
- Role-based access control ensures separation between user and admin functionalities.

### **Prediction Workflow**

1. Backend receives user education data from frontend.
2. Data is validated and preprocessed.
3. Trained ML model is loaded from serialized files.
4. Prediction results are generated and ranked.
5. Results are sent back to frontend as JSON responses.

## **5.3 Machine Learning Code Explanation**

The machine learning component handles training, evaluation, and prediction.

### **Data Handling**

- Dataset is loaded using pandas.

- Missing values are handled appropriately.
- Categorical features are encoded.
- Numerical values such as CGPA are normalized.

### **Model Training**

- A classification algorithm (Random Forest) is used.
- Dataset is split into training and testing sets.
- Model performance is evaluated using accuracy and F1-score.

### **Model Deployment**

- The trained model is serialized using joblib.
- Backend loads the model at runtime for real-time prediction.
- Admin-triggered retraining updates the model without affecting users.

## **5.4 Database Code Explanation**

The database layer manages all persistent data.

### **Database Operations**

- User registration and authentication data storage
- Storage of education and skill details
- Prediction history logging
- Feedback collection
- Admin activity logs

### **Database Connectivity**

- Secure connections are established using configuration files.
- SQL queries are optimized to reduce redundancy.
- Data integrity is maintained through relational constraints.

## **5.5 Admin Module Code Explanation**

The admin module ensures system control and continuous improvement.

### **Admin Functionalities**

- Dataset upload handling
- Model retraining triggers
- Viewing prediction logs
- Reviewing user feedback

### **Implementation Logic**

- Admin routes are protected using authentication checks.
- Uploaded datasets are validated before processing.
- Retraining scripts execute model training pipelines and update model files.

### **5.6 Logging and Feedback Mechanism**

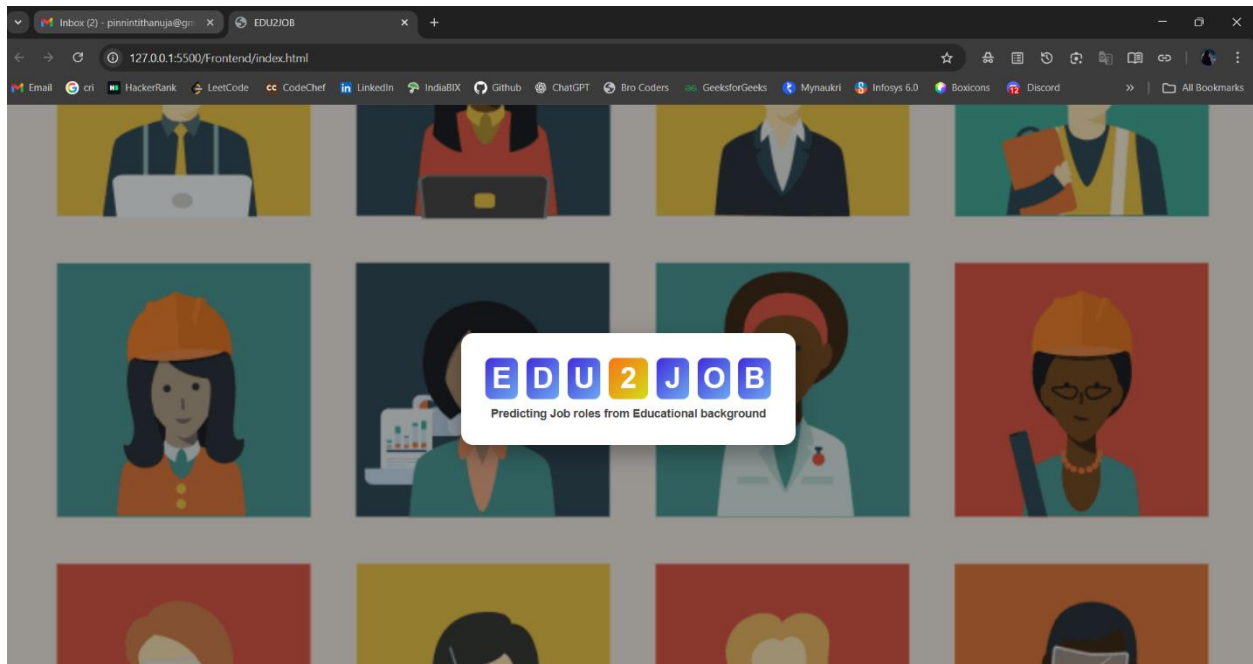
- Prediction logs store:
  - Timestamp
  - User ID
  - Input features
  - Predicted job roles
- Feedback data helps evaluate model effectiveness.
- Logs are used for auditing and performance analysis.

### **5.7 Error Handling and Security**

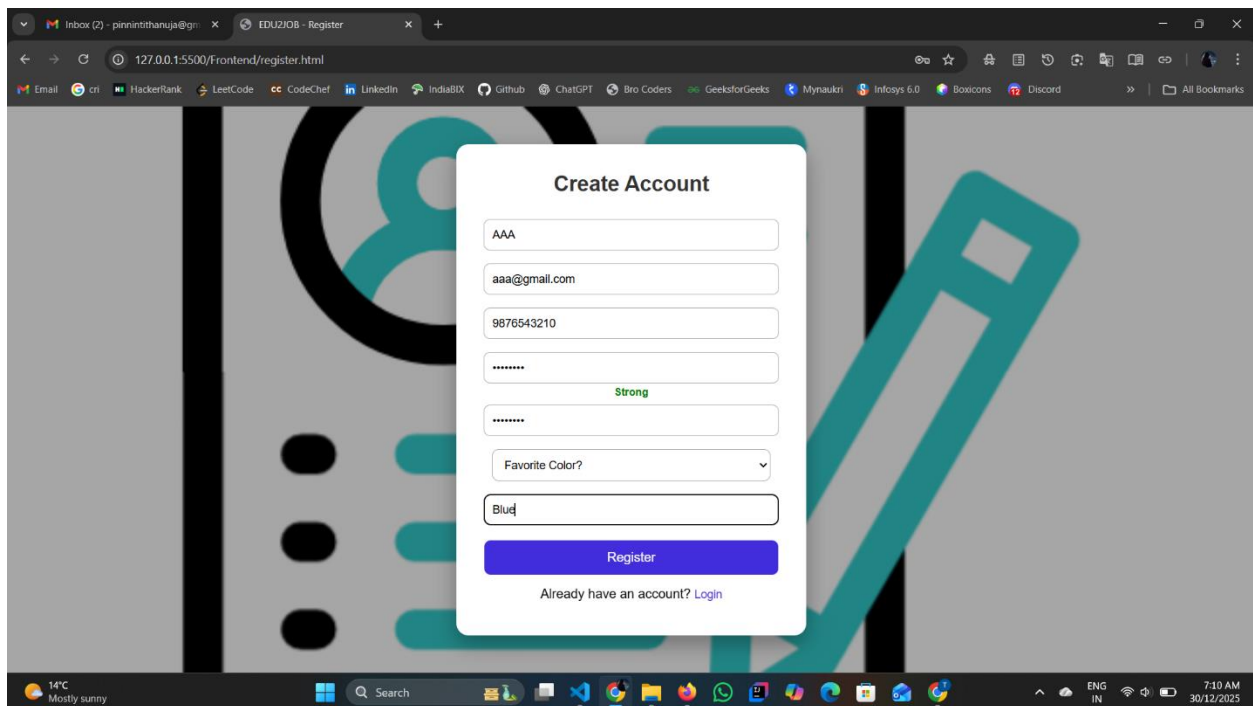
- Input validation prevents invalid data submissions.
- Exception handling ensures system stability.
- Sensitive data such as passwords are encrypted.
- Unauthorized access is restricted.

## 06.Output Screenshots

### 1. EDU2JOB – Open Page

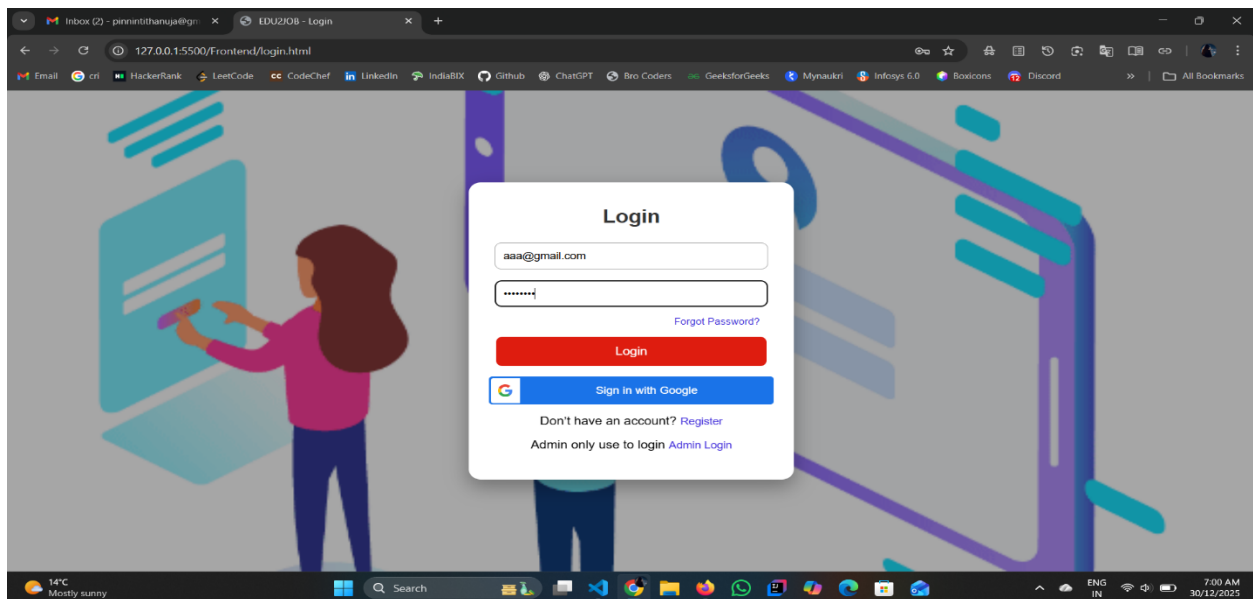


### 2. Register Page

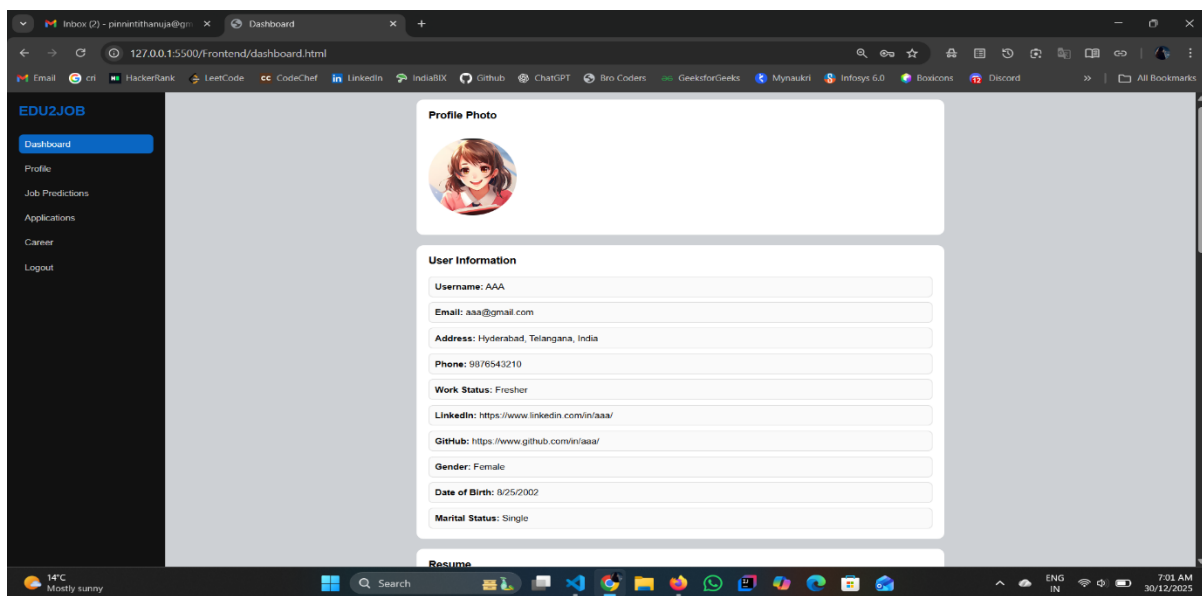


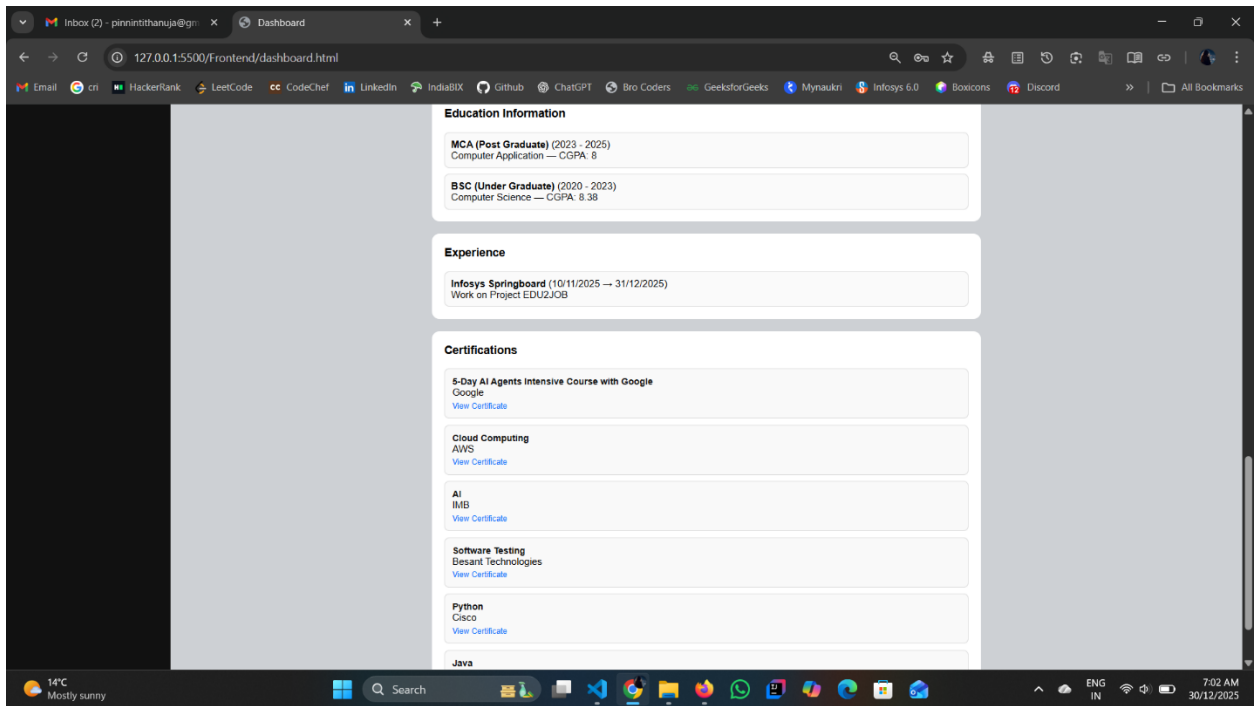
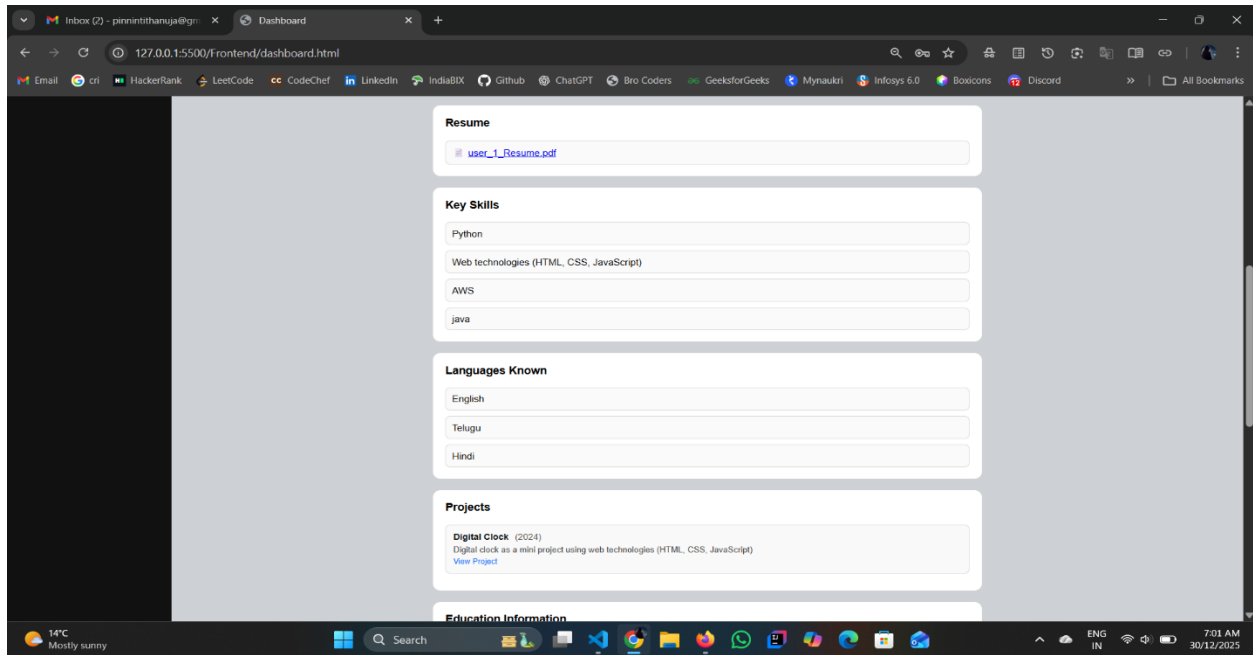


### 3. Login Page

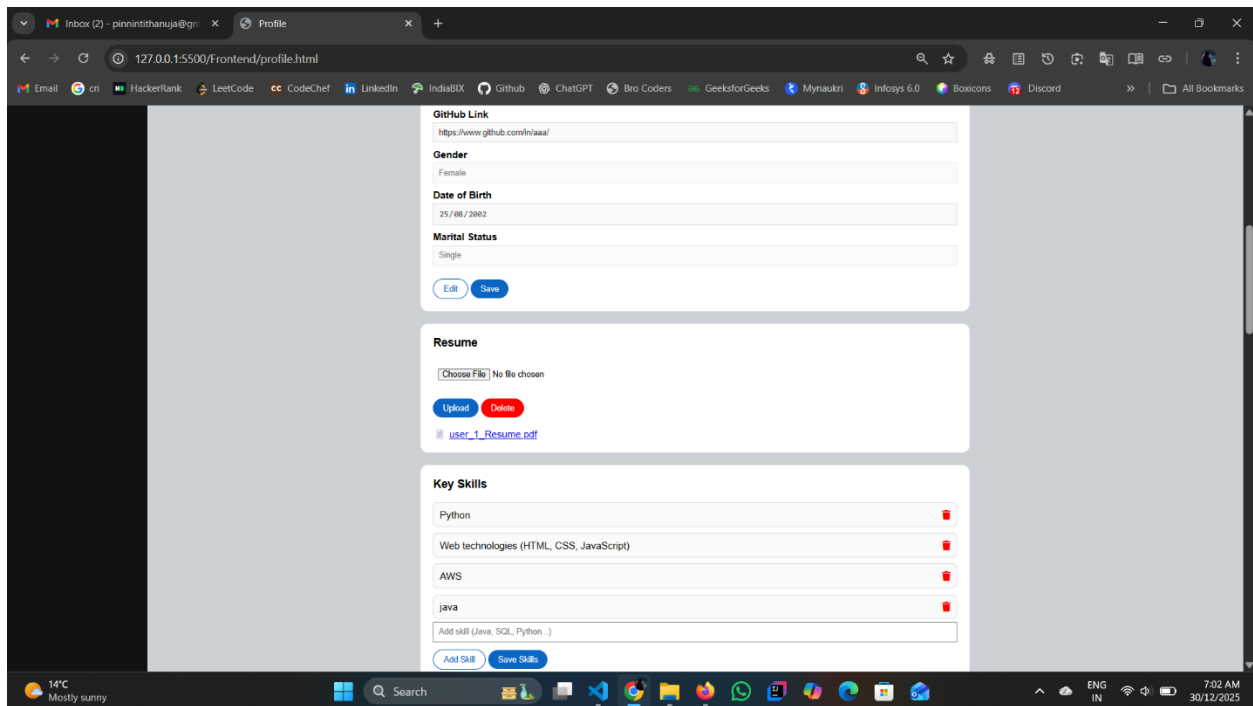
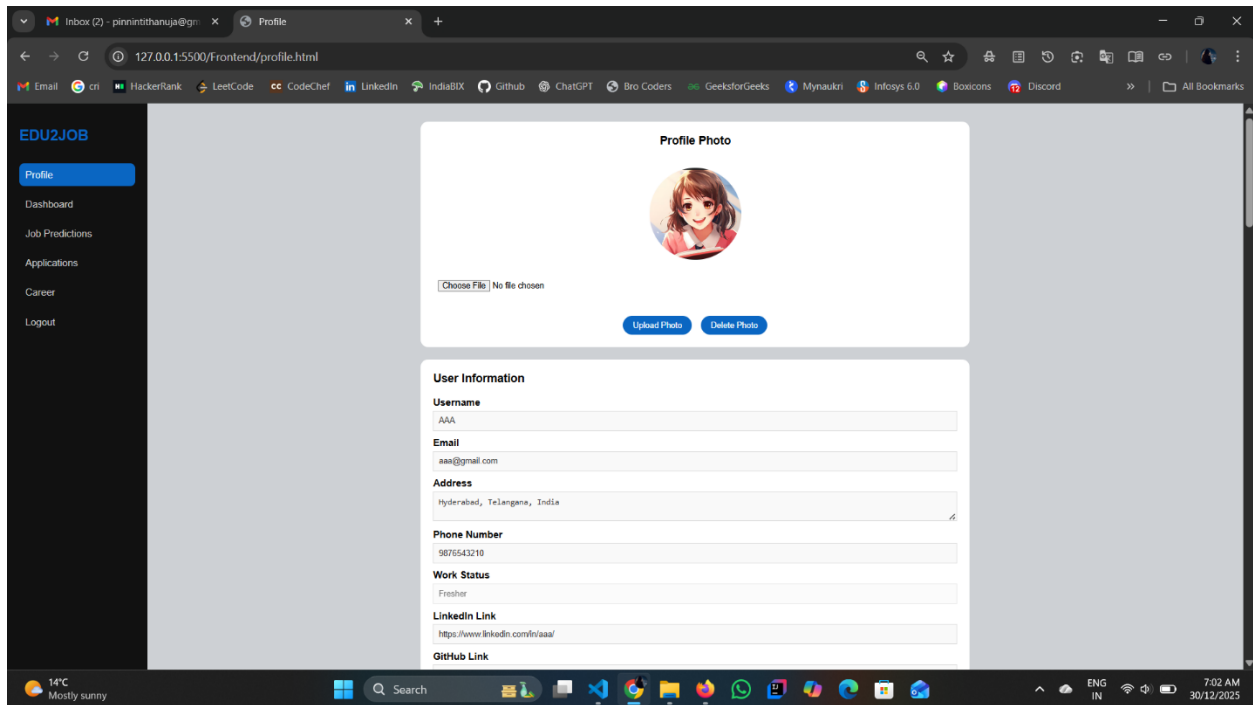


### 4. Dashboard





## 5. Profile Page



Inbox (2) - pinnintithanuja@gmail.com

Profile

127.0.0.1:5500/Frontend/profile.html

EmailcricHackerRankLeetCodeCodeChefLinkedInIndiaBIXGithubChatGPTBro CodersGeeksforGeeksMynaukriInfosys 6.0BosiconsDiscord

Languages Known

English

Telugu

Hindi

Add language (English, Telugu)

Add LanguageSave

Projects

Digital Clock (2024)  
Digital clock as a mini project using web technologies (HTML, CSS, JavaScript)  
[View Project](#)

Project Name

Year

About Project

Project Link

Add ProjectSave

Education Information

MCA (Post Graduate)  
(2023 - 2025)  
Computer Application — CGPA: 8

BSC (Under Graduate)  
(2020 - 2023)  
Computer Science — CGPA: 8.38

14°C  
Mostly sunny

Search

ENG  
IN

7:02 AM  
30/12/2025

Inbox (2) - pinnintithanuja@gmail.com

Profile

127.0.0.1:5500/Frontend/profile.html

EmailcricHackerRankLeetCodeCodeChefLinkedInIndiaBIXGithubChatGPTBro CodersGeeksforGeeksMynaukriInfosys 6.0BosiconsDiscord

Degree

Select Degree

Specialization  
Computer Science

CGPA  
5 - 10

Start Year  
2019

End Year  
2023

Add EducationSave

Experience

Infosys Springboard  
(10/11/2025 → 31/12/2025)  
Work on Project EDU2JOB

Company Name

Start Date  
dd/mm/yyyy

End Date  
dd/mm/yyyy

Description

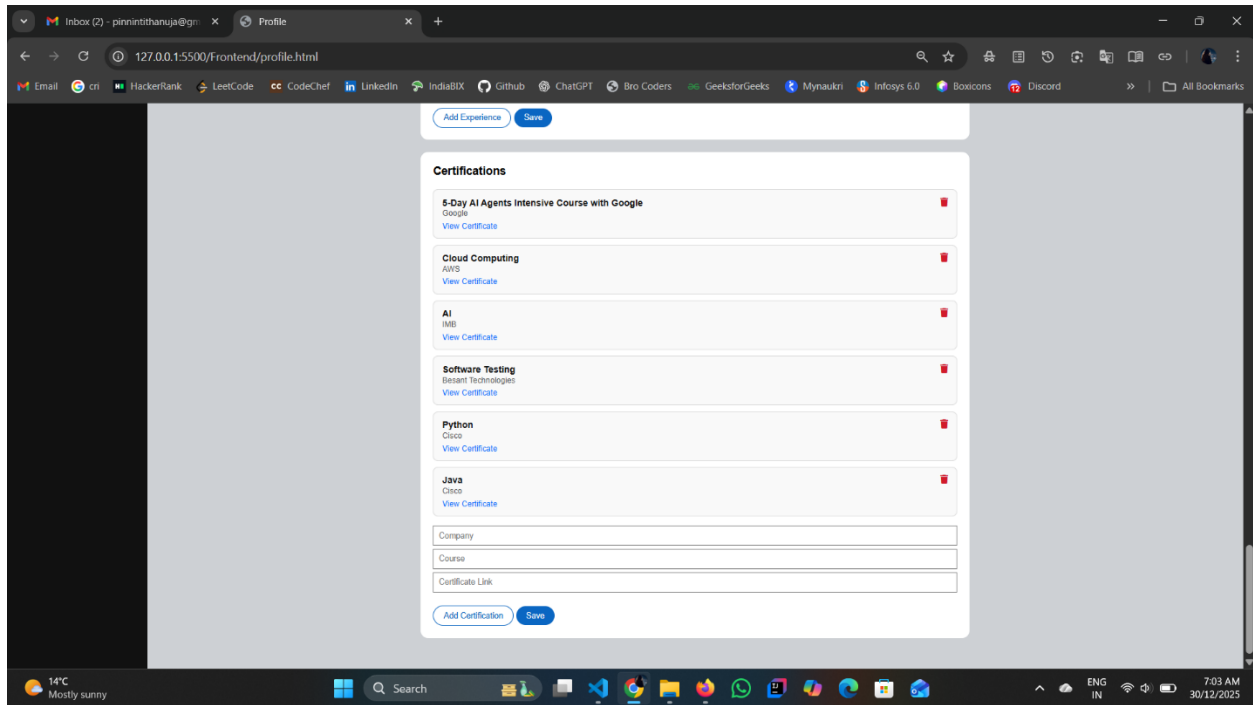
Add ExperienceSave

14°C  
Mostly sunny

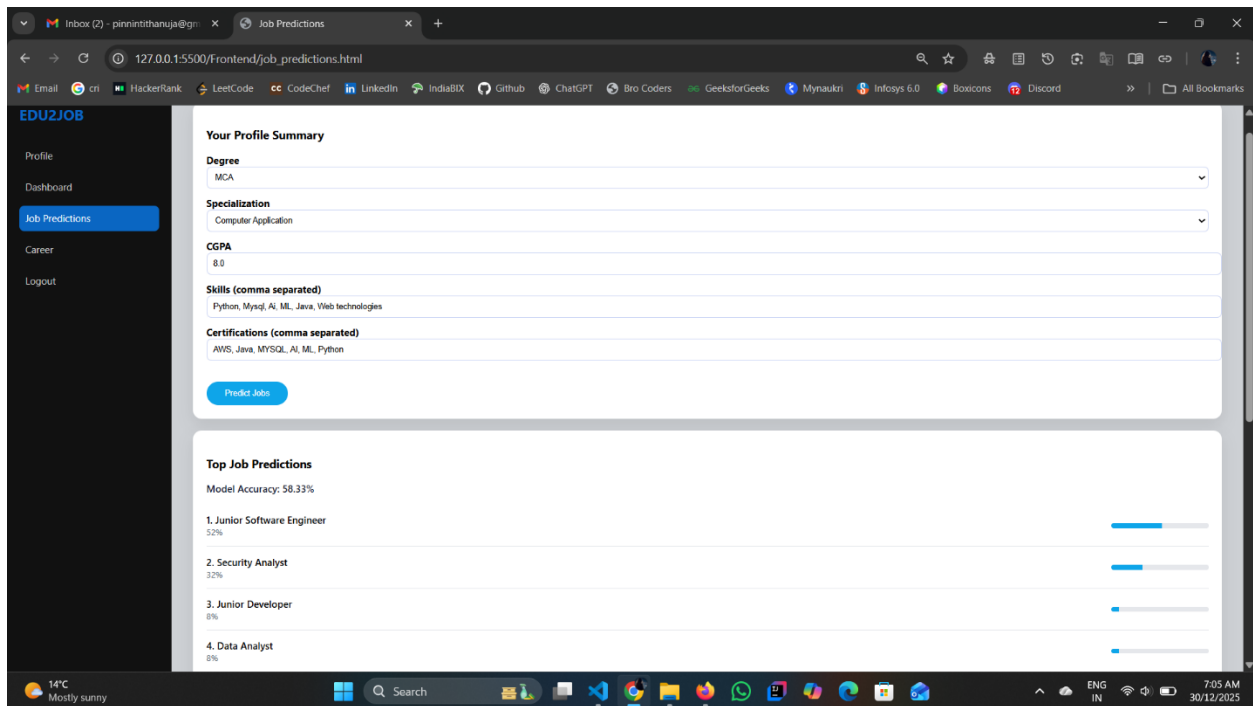
Search

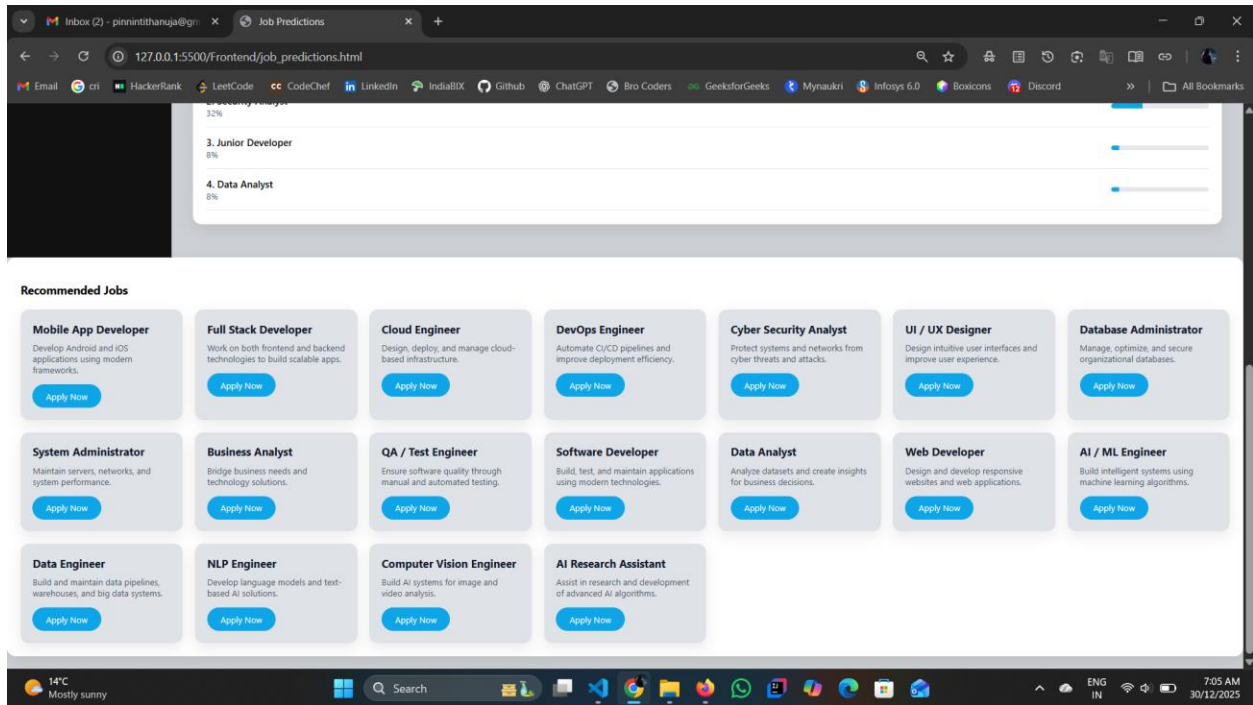
ENG  
IN

7:02 AM  
30/12/2025

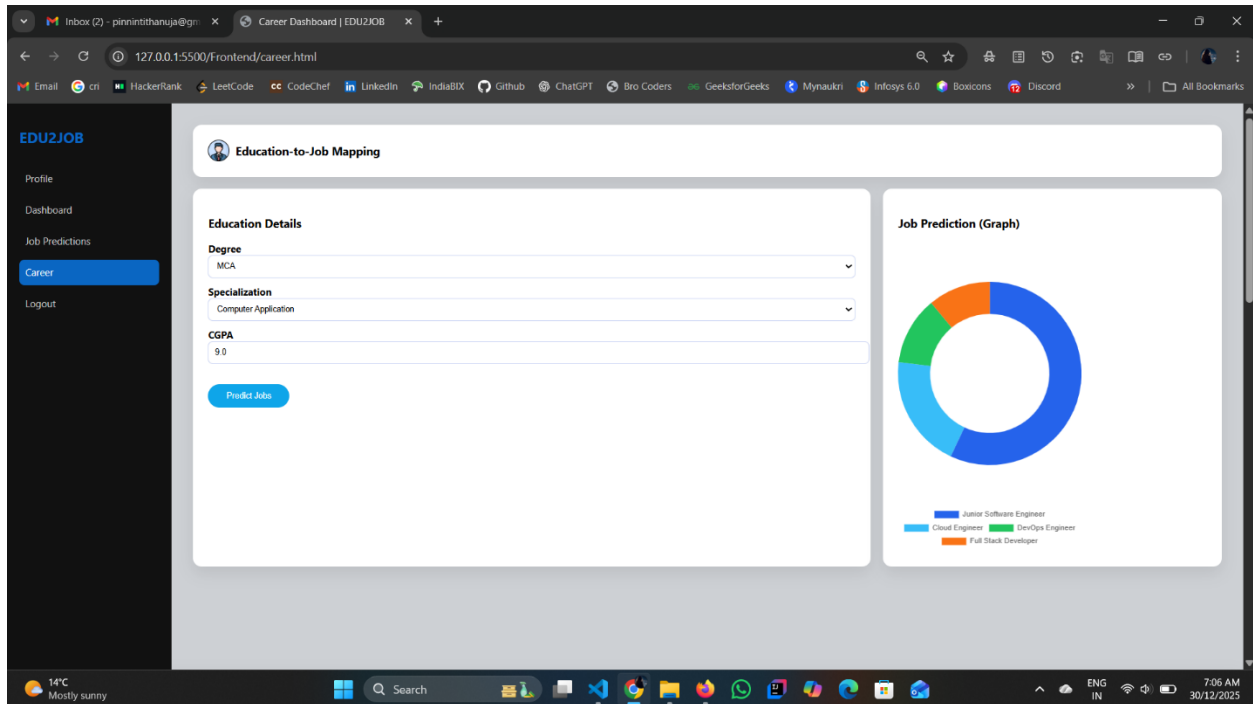


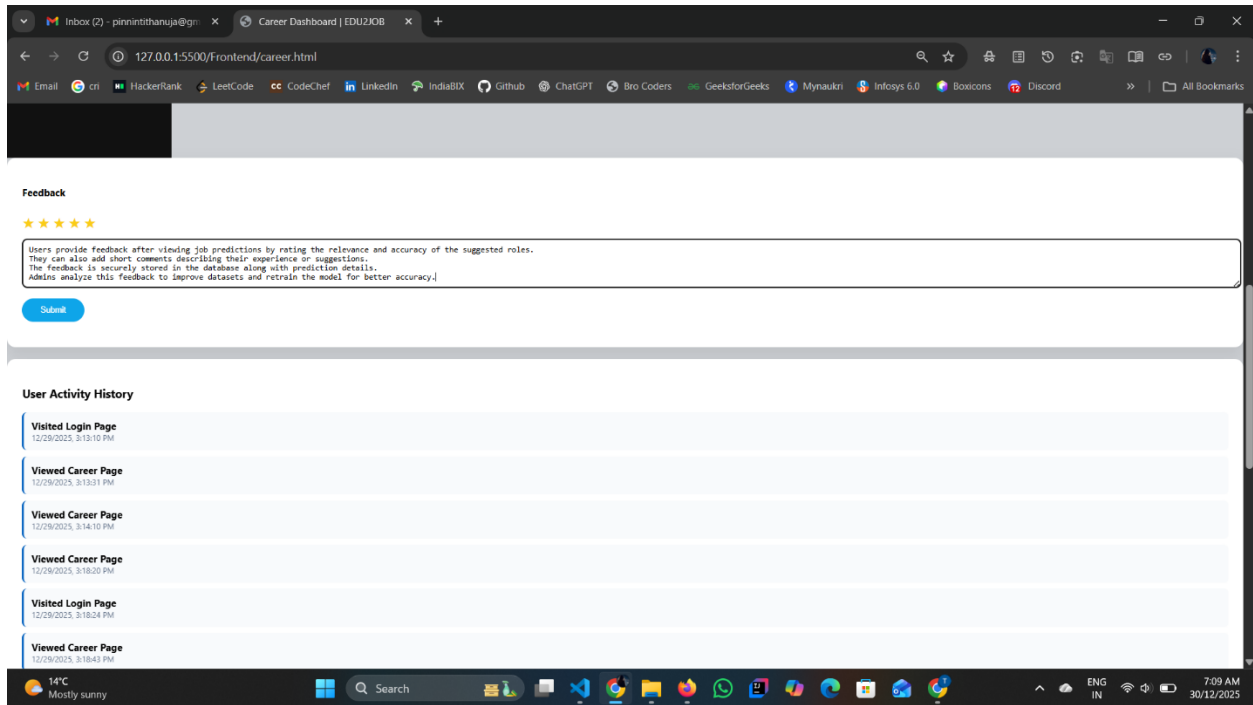
## 6. Job Prediction



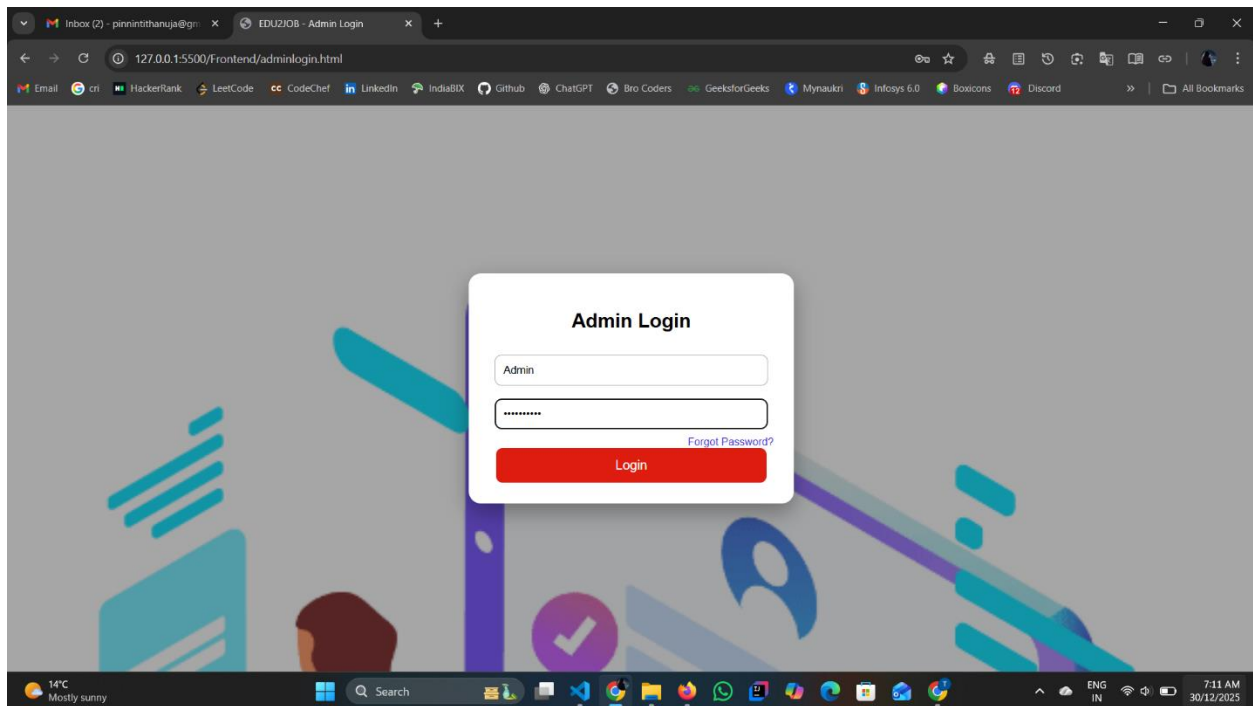


## 7. Career page





## 8. Admin Login page



## 9. Dashboard- Admin

EDU/fwzJOB

Dashboard

Prediction Logs

Upload Dataset


ML Model

Visualizations


Feedback

Logout


Admin Control Panel




**User Prediction Logs**  
View all job predictions made by users with timestamps




**Upload Dataset**  
Upload or update training dataset (CSV format)



**Retrain ML Model**  
Retrain job prediction model using latest data



**Career Visualizations**  
View graphs and analytics of career trends



**Review Feedback**  
Read feedback and suggestions from users

## 10. Predication logs

EDU2JOB

Dashboard

Prediction Logs

Upload Dataset

ML Model

Visualizations

Feedback

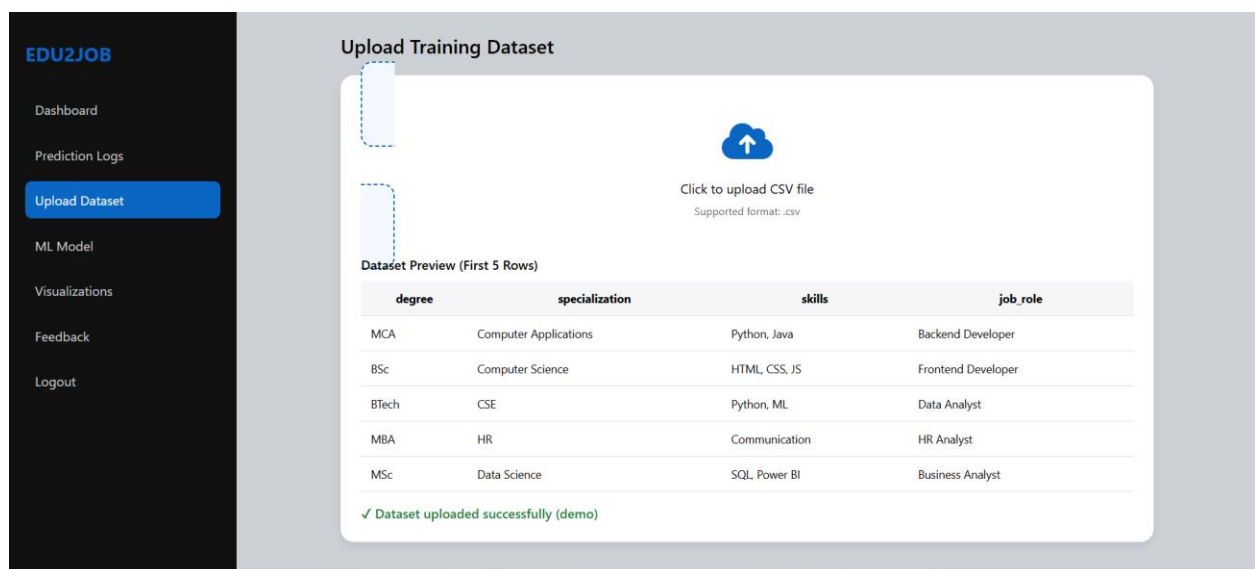
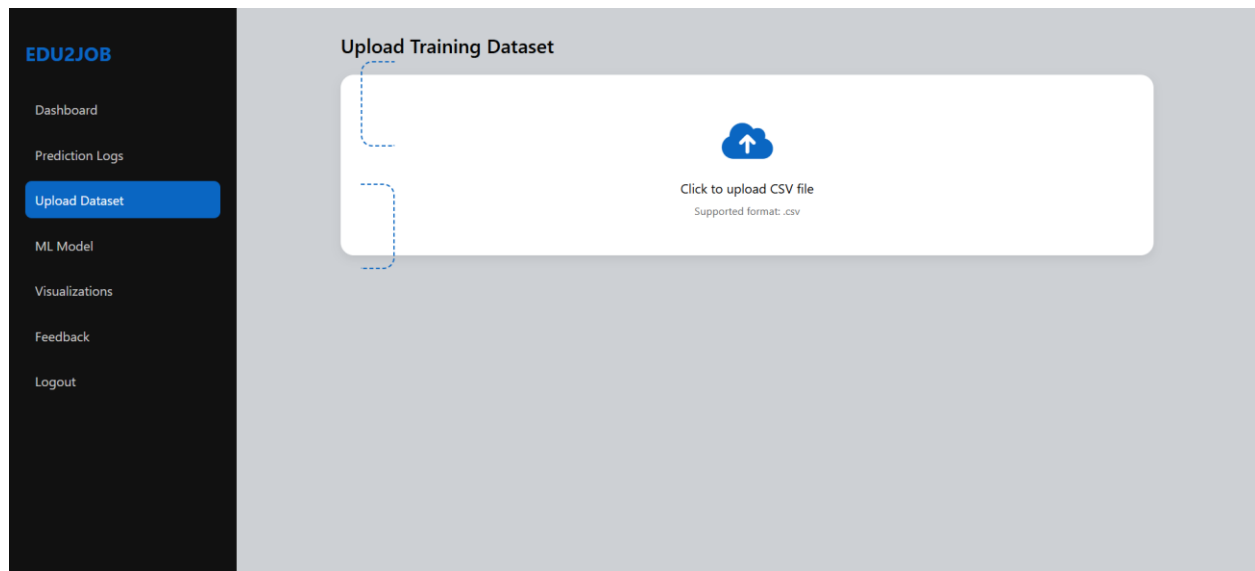
Logout

User Prediction Logs

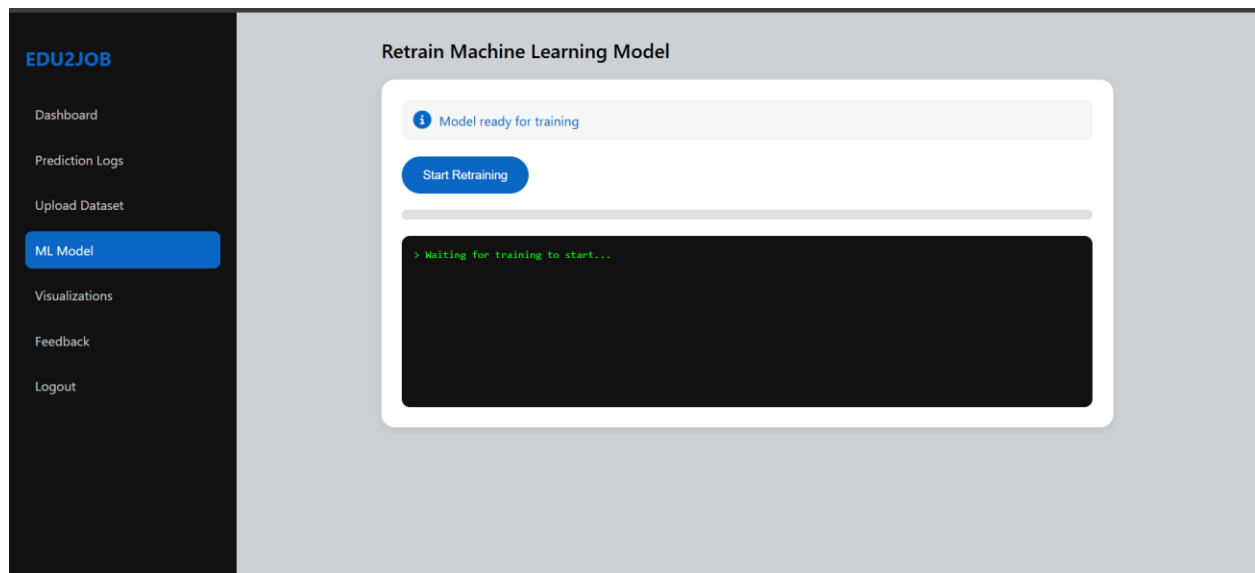
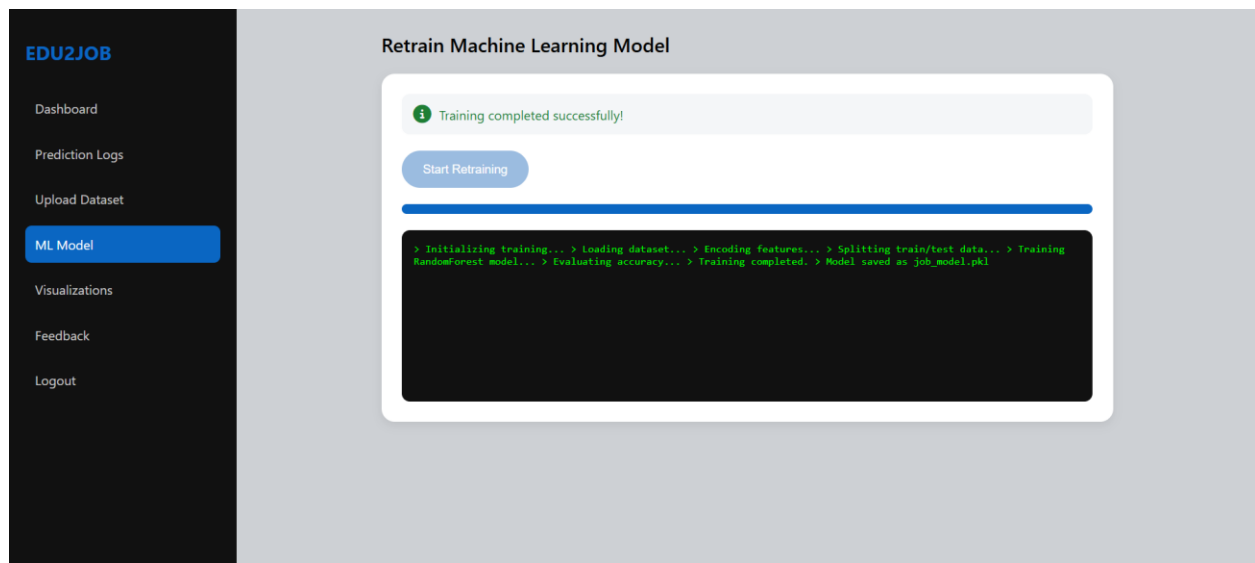
User	Degree	Skills	Predicted Job	Date & Time
AAA	MCA	Python, Java, SQL	Backend Developer	2025-12-30 10:45 AM
Thanuja	BSc Computer Science	HTML, CSS, JavaScript	Frontend Developer	2025-12-29 03:20 PM
Ravi	BTech	Python, Machine Learning	Data Analyst	2025-12-28 11:10 AM
Sneha	MSc Data Science	SQL, Power BI	Business Analyst	2025-12-27 09:40 AM



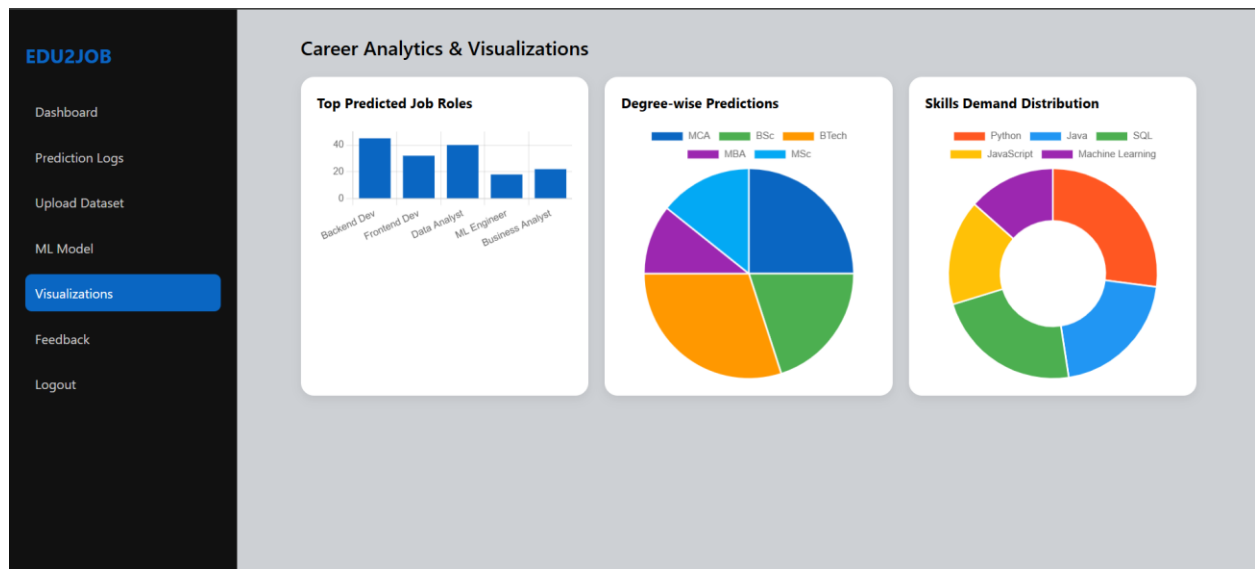
## 11. Upload Dataset



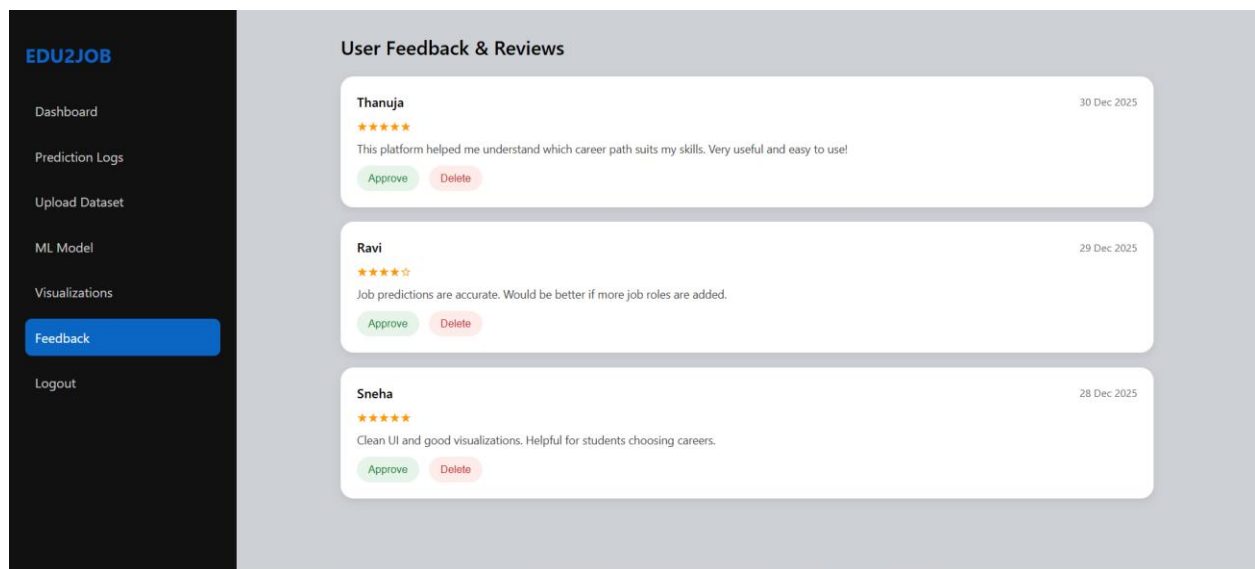
## 12. ML Model



## 13. Data Visualization



## 14. Feedback



## 07.Conclusion

The **EDU2JOB – Intelligent Career Prediction System** successfully demonstrates the effective integration of machine learning techniques with modern web technologies to address the critical problem of career guidance. The project provides a data-driven approach for predicting suitable job roles based on a user's educational background, academic performance, skills, and certifications, thereby reducing reliance on traditional, manual, and generic career counseling methods.

Through a structured workflow involving user authentication, data collection, preprocessing, prediction, visualization, and feedback, the system delivers accurate and personalized career recommendations. The inclusion of multiple ranked job roles with confidence scores allows users to explore various career possibilities rather than being restricted to a single outcome. Visual representations such as charts and career insights further enhance transparency and help users understand the reasoning behind the predictions.

The implementation of an **admin dashboard** adds significant value to the system by enabling dataset management, model retraining, feedback analysis, and activity monitoring. This feature aligns the project with real-world machine learning lifecycle practices, where continuous improvement and adaptability are essential. The modular architecture of the system ensures scalability, maintainability, and ease of future enhancements.

Overall, the EDU2JOB system achieves its objectives of providing intelligent, explainable, and user-friendly career recommendations. The project not only showcases practical machine learning deployment but also demonstrates how technology can bridge the gap between education and employment. This work serves as a strong foundation for future advancements such as real-time job market integration, skill-gap analysis, and personalized learning recommendations, making it highly relevant for academic evaluation and industry applications.

## 08.Future Scope

The **EDU2JOB – Intelligent Career Prediction System** provides a strong foundation for data-driven career guidance; however, it can be further enhanced and extended in several ways to improve accuracy, usability, and real-world applicability.

### 1. Integration with Real-Time Job Market Data

Future versions of the system can integrate real-time job market data from job portals and professional platforms. This would enable dynamic updates of job roles, skill requirements, and industry demand, making predictions more relevant and up-to-date.

### 2. Skill Gap Analysis and Learning Recommendations

The system can be extended to analyze the gap between a user's current skills and the skills required for recommended job roles. Based on this analysis, personalized learning paths, courses, and certifications can be suggested to improve employability.

### 3. Advanced Machine Learning Models

More advanced models such as deep learning or ensemble techniques can be implemented to improve prediction accuracy. Natural Language Processing (NLP) can also be used to analyze resumes and project descriptions for richer feature extraction.

### 4. Personalized Career Roadmaps

Future enhancements can include step-by-step career roadmaps showing required skills, certifications, internships, and experience levels needed to achieve specific job roles.

### 5. Mobile Application Development

A mobile application version of the system can be developed to increase accessibility and user engagement. This would allow users to receive career guidance anytime and anywhere.

### 6. Multilingual Support

Adding support for multiple languages would make the platform accessible to a wider user base, particularly in diverse educational environments.

### 7. Employer and Recruiter Integration

The platform can be expanded to include recruiter and employer dashboards, enabling companies to view candidate trends and connect with potential talent based on predicted career paths.

### 8. Continuous Model Learning

Future implementations can enable automated retraining pipelines where the model learns continuously from new data and feedback, further improving system performance.