

**AALIM MUHAMMED SALEGH COLLEGE OF
ENGINEERING**

AVADI - I.A.F., MUTHAPUDUPET, CHENNAI-600 055.

**DEPARTMENT OF
INFORMATION TECHNOLOGY**



SUMMER INTERNSHIP REPORT

ROLL NO : 110121205044

NAME : RAABIYA. N

**DEGREE/BRANCH : B.TECH / INFORMATION
-TECHNOLOGY**

YEAR : IV

SEMESTER : 07



AALIM MUHAMMED SALEGH COLLEGE OF ENGINEERING

Approved by All India Council for Technical Education - New Delhi, Affiliated to Anna University, Chennai
NAAC Accredited Institution

"Nizara Educational Campus", Muthapudupet, Avadi - IAF, Chennai - 600 055.

ANNA UNIVERSITY COUNSELLING CODE : 1101

NBA ACCREDITED COURSES (Mech Engg, ECE, CSE & IT)



DEPARTMENT OF INFORMATION TECHNOLOGY

Bonafide Certificate

SUMMER INTERNSHIP REPORT

Register No:

This is to certify that **Raabiya**, a student of **IV Year, Seventh Semester, B.Tech – Information Technology**, has successfully completed the **Summer Internship Report** titled "**Generative AI for Skill-Based Job Matching**" during the academic year **2024–2025**.

Faculty In-charge

Head of the Department

VISION OF THE DEPARTMENT

To be an excellent Department of Information Technology through teaching and learning Edge Technologies.

MISSION OF THE DEPARTMENT

M1: To teach programming language through experimental learning for improving logical and critical thinking.

M2: To promote interdisciplinary research for solving social and technological problems.

M3: To encourage lifelong learning through MOOC Courses.

M4: To protect intellectual proprietary by educating on cyber ethics.

PROGRAMME OUTCOMES (Pos)

PO1- PEO1Engineering knowledge : Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering To ensure Graduates to be proficient in utilizing the fundamental knowledge of basic problems. Sciences, Mathematicsand Information Technology for the application relevant to various

PO2 - Problem analysis: Identify, formulate, review research literature, and analyze complex streams of Engineering and Technology engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3 - Design/development of solutions: Design solutions for complex engineering problems and

To enrich Graduates with the core competencies necessary for applying knowledge of design system components or processes that meet the specified needs with appropriate Computers and Telecommunications equipment to store, retrieve, transmit, manipulate and consideration for the public health and safety, and the cultural, societal, and environmental analyze data in the context of business enterprise considerations.

PO4 - Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and To enable Graduates on logical thinking, pursue lifelong learning and to have the capacity in synthesis of the information to provide valid conclusions.

understandingtechnical issues related to computing systems and optimal design solutions.

PO5 - Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern PEO4 engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

To enable Graduates in gaining employment in industry and stabilize themselves as

PO6 - The engineer and society: Apply reasoning informed by the contextual knowledge to assess competent professionalin applying their technical skills in real time problems and meet the societal, health, safety, legal and cultural issues and the consequent

responsibilities relevant to diversified needs of industry, academia and research as professional english practices.

PO7 - Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8 - Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9 - Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10 - Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11 - Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12 - Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.



TRIPLEM INFOTECH SOLUTIONS PRIVATE LIMITED

Certificate of Completion

INTERNSHIP TRAINING

This is to certify that **RAABIYA.N** has successfully completed the **INTERNSHIP TRAINING PROGRAMME** conducted from **04/07/2024 to 10/08/2024** in our Company on **Generative AI** and completed the Project titled “**Name: JobGenie, Concept: Skill Based Job Matcher**” and obtained grade **O***. The participant has demonstrated skills inculcated dedication and commitment throughout the program by acquiring knowledge of the fundamentals and practical applications of Generative AI. We congratulate her on her accomplishment and wish her continued success in her future endeavors.

***Grading Reference**

- O** – Outstanding
- A** - Excellent
- B** - Very Good
- C** – Good



Saghul Ahmed

(MANAGER, TRIPLEM INFOTECH SOLUTIONS)

ABSTRACT

The rapid advancement of technology has transformed the global job market, creating opportunities as well as challenges for job seekers and employers alike. However, traditional job-matching platforms often fall short of addressing critical gaps, particularly for individuals lacking formal education, marginalized communities, and those with informal skills. The “**Generative AI for Skill-Based Job Matching**” project aims to bridge this gap by developing an intelligent, conversational system that identifies and aligns users' skills with tailored job opportunities.

This project utilizes **Generative AI** to create an interactive, user-friendly chat interface that collects detailed information about a user's skill set, interests, and level of expertise in various domains. Unlike conventional systems, the platform provides a multi-step guided process to:

1. Suggest relevant domains based on user preferences.
2. Gather detailed insights into sub-areas of expertise.
3. Integrate informal skills, such as hobbies or freelance experiences, into the job-matching process.

The system then leverages this data to generate highly customized job recommendations, ensuring inclusivity for individuals with non-traditional educational backgrounds. Furthermore, the platform goes beyond job recommendations by identifying skill gaps and suggesting learning resources or upskilling opportunities tailored to the desired roles. **Technologically**, the project is implemented using a combination of **Flask** for backend development, **HTML, CSS, and JavaScript** for frontend design, and **Google's Generative AI API** to enable the conversational interface. The recommendation engine integrates user data with cutting-edge machine learning.

INDEX

S.no	CONTENTS	Page no
1.	Introduction	13
2.	Analysis	15
3.	Software Requirement Specification	18
4.	Technology overview	20
5.	System Design	25
6.	Implementation	32
7.	Result & Outcomes	51
8.	Conclusion	58

Learning Objectives/Internship Objectives

- Internships are generally thought of to be reserved for college students looking to gain experience in a particular field. However, a wide array of people can benefit from Training Internships in order to receive real world experience and develop their skills.
- An objective for this position should emphasize the skills you already possess in the area and your interest in learning more.
- Internships are utilized in a number of different career fields, including architecture, engineering, Health care, economics, advertising and many more.
- Some internship is used to allow individuals to perform scientific research while others are specifically designed to allow people to gain first-hand experience working.
- Utilizing internships is a great way to build your resume and develop skills that can be emphasized in your resume for future jobs. When you are applying for a Training Internship, make sure to highlight any special skills or talents that can make you stand apart from the rest of the applicants so that you have an improved chance of landing the position.

WEEKLY OVERVIEW OF INTERNSHIP ACTIVITIES

WEEK-1

DATE	DAY	TOPIC/MODULE COMPLETED
04/07/24	Monday	Project Overview & Requirement Gathering
05/07/24	Wednesday	Research on Generative AI Models (Google API)
06/07/24	Wednesday	Research on Generative AI Models (Google API)
07/07/24	Thursday	Designing Project Flow & User Interaction Plan
08/07/24	Friday	Setting Up Flask Environment

WEEK-2

DATE	DAY	TOPIC/MODULE COMPLETED
09/07/24	Monday	Frontend Planning & Wireframe Creation
10/07/24	Tuesday	Creating User Interface in HTML/CSS
11/07/24	Wednesday	Connecting Frontend with Flask Backend
12/07/24	Thursday	Basic Job Domain Options & User Input Form
13/07/24	Friday	API Integration: Connecting to Google's GenAI

WEEK-3

DATE	DAY	TOPIC/MODULE COMPLETED
14/07/24	Monday	Collecting User Inputs (Skills & Domain Preferences)
15/07/24	Tuesday	Generating Tailored Job Recommendations
16/07/24	Wednesday	Building Interactive Chat-Based System
17/07/24	Thursday	Testing Recommendation Accuracy
18/07/24	Friday	Debugging & Refining Recommendation Logic

WEEK-4

DATE	DAY	TOPIC/MODULE COMPLETED
19/07/24	Monday	Adding Informal Skills Support in the Model
20/07/24	Tuesday	Adding Upskilling Suggestions
21/07/24	Wednesday	Optimizing AI Responses for Speed & Clarity
22/07/24	Thursday	Testing the Chat Interface
23/07/24	Friday	Collecting Feedback from Test Users

WEEK-5

DATE	DAY	TOPIC/MODULE COMPLETED
24/07/24	Monday	Final Debugging & Polishing Frontend Design
25/07/24	Tuesday	Documenting Project Overview & Workflow
26/07/24	Wednesday	Creating a Demo Video for the Project
27/07/24	Thursday	Deploying the Project on a Cloud Platform
28/07/24	Friday	Final Presentation Preparation

WEEK-6

DATE	DAY	TOPIC/MODULE COMPLETED
29/07/24	Monday	Testing the Project with Real Users
30/07/24	Tuesday	Gathering User Feedback & Analyzing Data
01/08/24	Wednesday	Making Final Adjustments Based on Feedback
02/08/24	Thursday	Preparing Final Outcomes
03/08/24	Friday	Submission of the Completed Project

1. INTRODUCTION

In today's fast-evolving job market, individuals without formal degrees or conventional education often face significant barriers in securing meaningful employment. Many job-seekers possess valuable skills gained through informal learning, freelancing, or hobbies, but traditional recruitment systems fail to recognize their potential. To bridge this gap, the Generative AI for Skill-Based Job Matching project leverages cutting-edge AI technologies to create a system that matches individuals' skills to suitable job opportunities, regardless of their formal qualifications.

This project aims to revolutionize the job search process by focusing on *skills-first hiring*. By employing generative AI, the system interacts with users in a conversational manner, collects detailed information about their interests and expertise, and tailors job recommendations to their unique profiles. The solution aspires to empower marginalized groups, freelancers, and individuals without formal education by offering equitable access to opportunities and personalized suggestions for upskilling.

Key Features

1. Interactive Chat-Based Interface:

A user-friendly chat system where users can seamlessly provide details about their skills, interests, and experience.

2. Domain-Specific Recommendations:

Users can select their area of interest from predefined domains (e.g., Full-Stack Development, Graphic Design, Marketing).

Based on the selected domain, the system gathers detailed sub-domain preferences.

3. Tailored Job Matching:

Using Google's Generative AI, the system generates customized job recommendations suited to the user's skill set, informal experiences, and career goals.

4. Upskilling Suggestions:

The system identifies gaps in the user's skill set and provides actionable suggestions, including courses, tools, or resources to improve their employability.

5. Inclusivity:

The project focuses on breaking barriers by recognizing informal skills and aligning them with opportunities, helping underserved communities achieve career growth.

2. ANALYSIS

2.1.Existing System

The current landscape of job matching platforms is largely dominated by systems that prioritize formal qualifications and professional experience over skills. These systems rely on static job boards, traditional resume filtering, and automated keyword matching, which create several challenges:

- Bias Toward Formal Education: Most platforms prioritize candidates with degrees, certifications, and formal work experience, neglecting those with informal or self-taught skills.
- Limited Personalization: Generic job recommendations often fail to align with individual capabilities, interests, or career aspirations.
- Inadequate Support for Skill Development: Existing platforms do not actively guide users on how to bridge skill gaps or suggest specific resources for upskilling.
- Barriers for Marginalized Groups: Underserved populations, such as freelancers, self-taught professionals, or individuals from marginalized communities, are often excluded from opportunities due to systemic bias.

These limitations hinder inclusivity and fail to meet the needs of a diverse and evolving workforce.

2.2. Proposed System

The Generative AI for Skill-Based Job Matching project seeks to address the limitations of existing systems by introducing a highly interactive and skill-focused job recommendation platform. The proposed system leverages the power of Generative AI to redefine the job-matching process.

Key Features and Advantages:

- Skill-Centric Approach:**

Unlike conventional systems, the proposed model focuses on individuals' skills, including informal abilities gained through freelancing, hobbies, or self-learning, rather than formal degrees or certifications.

- Interactive Chat Interface:**

The platform uses a conversational interface powered by Google's Generative AI. This allows users to interact naturally, providing details about their skills and preferences in a structured yet intuitive manner.

- Domain-Specific Customization:**

Users can select their preferred job domains (e.g., Full-Stack Development, Data Science, Graphic Design), and the system collects detailed sub-domain information to generate personalized job recommendations.

- Dynamic Job Recommendations:**

The system uses advanced AI models to match users' unique skill sets with relevant job opportunities, ensuring that recommendations are tailored and actionable.

- Guided Upskilling:**

The platform not only identifies job opportunities but also

suggests specific courses, tools, or resources to help users improve their skill sets and qualify for desired roles.

- Inclusivity and Accessibility:

By valuing informal skills and emphasizing skill-based matching, the system empowers underserved groups and bridges the gap between talent and opportunity.

2.3. Comparative Analysis

Aspect	Existing System	Proposed System
Focus	Formal education and certifications	Skills (formal and informal)
Personalization	Limited or generic recommendations	Highly tailored job matching based on user inputs
User Interaction	Static interfaces, resume uploads	Interactive, chat-based AI-driven interface
Upskilling Support	Rarely provides suggestions	Actively identifies skill gaps and recommends improvements
AI Utilization	Limited, mainly keyword matching	Advanced generative AI for conversational engagement

3. SOFTWARE REQUIREMENT & SPECIFICATION

3.1 Hardware Requirements

- Server Requirements:
 - Processor: Intel Xeon or equivalent
 - RAM: Minimum 16GB
 - Storage: Minimum 500GB SSD
 - Network: High-speed internet connection (1Gbps)
- Client Requirements:
 - Processor: Any modern CPU
 - RAM: Minimum 4GB
 - Storage: Minimum 50MB for browser cache and data
 - Browser: Google Chrome, Mozilla Firefox, or equivalent

3.2 Software Requirements

- Frontend Technologies:
 - HTML5, CSS3, JavaScript
 - Framework: Bootstrap (for responsiveness)
- Backend Technologies:
 - Framework: Flask (Python)
 - API: Google's Generative AI API for chatbot functionality.

- Database:
 - MySQL/PostgreSQL (for storing user data and job information)
- Third-Party Tools:
 - Google Generative AI API (for conversational responses)
- Version Control:
 - GitHub for source code management
- Cloud Platform:
 - Google Cloud for hosting and
- Deployment:
 - Vercel/Heroku

4. TECHNOLOGY OVERVIEW

The Generative AI for Skill-Based Job Matching project leverages modern technologies to create an intelligent, user-friendly system for job seekers. The following is a detailed breakdown of the technologies and tools used across various aspects of the project.

4.1. Frontend Technologies

The frontend is responsible for providing a seamless and interactive user experience.

- HTML5: Used for structuring the web pages and ensuring compatibility across devices.
- CSS3: Enables styling and responsiveness for a visually appealing and mobile-friendly interface.
- JavaScript: Powers the interactivity of the platform, ensuring dynamic responses to user inputs.
- Bootstrap: Provides a responsive framework for building user-friendly interfaces, ensuring the application adapts to different screen sizes (desktop, tablet, and mobile).

4.2. Backend Technologies

The backend is the core processing engine of the system, handling data logic, API interactions, and server-side operations.

- Flask (Python Framework):

- Chosen for its simplicity, flexibility, and compatibility with AI and machine learning libraries.
- Handles routing, user session management, and data processing.
- Google’s Generative AI API:
 - Provides the conversational interface that powers the AI-driven chatbot for user interactions.
 - Ensures real-time, context-aware responses during the job recommendation process.
- RESTful APIs:
 - Facilitate communication between the frontend, backend, and external services (e.g., job boards, upskilling platforms).

4.3. Database Technologies

Data storage and management are critical for handling user profiles, job information, and system logs.

- MySQL/PostgreSQL:
 - Used for storing structured data, including user inputs, job listings, and skill-related metadata.
 - Ensures efficient querying, indexing, and management of large datasets.
- Database Design:
 - Schema includes tables for user profiles, domain-specific data, job recommendations, and resource suggestions.
 - Relationships are optimized for quick lookups and scalability.

4.4. AI and Machine Learning

AI technologies form the backbone of the system's functionality and intelligence.

- Google's Generative AI:
 - Processes user inputs in natural language and generates relevant, context-sensitive outputs.
 - Ensures conversational flow and guides users through the skill-assessment process.
- Skill and Job Matching Logic:
 - Uses pre-defined algorithms to match user-reported skills with job requirements from the database.
 - Enhances recommendations with contextual understanding derived from AI outputs.
- Upskilling Suggestions:
 - Analyzes skill gaps and recommends resources using AI-driven insights.

4.5. Hosting and Deployment

Reliable hosting and deployment are essential to ensure accessibility, scalability, and performance.

- Google Cloud:
 - Provides a secure, scalable cloud environment for hosting the backend and database.
 - Ensures high availability and low latency for users worldwide.

- Vercel:
 - For hosting and ensuring seamless continuous integration and delivery (CI/CD).
 - Vercel ensures fast and scalable delivery of the frontend application with global CDN support.

4.6. Security Technologies

Security is prioritized to safeguard user data and ensure system integrity.

- HTTPS:
 - Ensures secure communication between the client and server, encrypting data in transit.
- Data Encryption:
 - Encrypts sensitive user information in the database to prevent unauthorized access.
- Authentication and Authorization:
 - Implements secure login mechanisms to verify user identities and restrict unauthorized actions.

4.7. Version Control and Collaboration

For efficient development, code management, and team collaboration:

- GitHub: Tracks code changes, allowing seamless collaboration among team members. Central repositories for storing, reviewing, and managing code versions.

4.8. Third-Party Integrations

To enhance functionality and user experience:

- Job Board APIs: Sources real-time job listings from external platforms.
- Upskilling Platforms (e.g., Coursera, Udemy): Provides recommendations for courses and certifications.
- Analytics Tools: Tracks user engagement and system performance for iterative improvement.

4.9. Development Tools

Tools used during development to streamline coding, testing, and debugging:

- Visual Studio Code: Integrated development environment (IDE) for writing and debugging code.
- Postman: For testing RESTful API endpoints.

4.10. Advantages of the Technology Stack

The system is highly scalable with Vercel and cloud-hosted backend services, ensuring seamless handling of increased user traffic. It offers low latency and optimal performance through global CDN support and modular architecture, enabling flexibility for feature updates. Modern frameworks ensure robust security and compliance with privacy standards.

5. SYSTEM DESIGN

5.1 Overview

The system design for the **Generative AI for Skill-Based Job Matching** project is built to ensure scalability, performance, and maintainability. It follows a modular architecture that separates concerns across the frontend, backend, database, and AI services, resulting in a clean and efficient system. This design provides a seamless user experience while maintaining scalability, data integrity, and real-time performance.

5.2 System Architecture

The system architecture follows a **Client-Server model** and integrates multiple layers to handle frontend interactions, backend processing, data storage, and AI integration. The system is divided into four main components:

1. **Frontend**
2. **Backend**
3. **Database**
4. **Generative AI Services**

5.3 System Components

1. Frontend

Technologies: HTML5, CSS3, JavaScript, React.js, Bootstrap, Vercel

Responsibilities:

- Provide an interactive and responsive user interface for collecting user inputs and displaying job recommendations.
- Integrate a conversational chatbot interface powered by Google's Generative AI API.
- Ensure a smooth user experience across desktop and mobile devices.

Key Components:

- **User Interface Components:**
 - Job Search Interface
 - Chatbot UI for Conversational Interaction
 - Dashboard displaying personalized job recommendations and upskilling suggestions
- **API Integration:**
 - Connects to the backend APIs to send user inputs and receive job data recommendations.
 - Communicates with Google's Generative AI API to handle chatbot interactions.

Deployment:

- Hosted on **Vercel**, ensuring global CDN support for quick delivery and low latency.

2. Backend

Technologies: Flask (Python), REST APIs, Google Generative AI API, Vercel Serverless Functions

Responsibilities:

- Process requests from the frontend, validate user inputs, and interact with the database.
- Integrate with Google's Generative AI API for generating personalized chatbot responses.
- Generate job recommendations by analyzing user inputs and matching them with relevant job data.

Key Components:

- **API Gateway:**
 - Acts as the interface between the frontend and backend, routing API calls to appropriate services.
- **Flask Application:**
 - Handles business logic, data processing, and interactions with external APIs.
- **Google Generative AI Integration:**
 - Provides natural language responses and real-time chatbot interactions.

3. Database

Technologies: MySQL/PostgreSQL

Responsibilities:

- Store user profiles, job listings, and user interactions.
- Maintain data integrity and ensure fast query performance for job recommendation retrieval.
- Support relational operations for efficient data retrieval and updates.

Database Structure:

Table	Description
Users	Stores user profile information and login credentials.
Jobs	Contains job listings, job descriptions, and requirements.
User_Skills	Stores both formal and informal user skill information.
Job_Match_Log	Logs job recommendations and interactions for analysis.
Feedback	Collects user feedback about job recommendations and chatbot responses.

5.4. Generative AI Integration

Technologies: Google Generative AI API

Responsibilities:

- Power the chatbot interface to interact with users, ensuring real-time, context-aware conversations.
- Generate intelligent responses tailored to user input about job preferences and skills.
- Provide upskilling recommendations by analyzing skill gaps and suggesting relevant courses.

Key Components:

- **Conversational AI Service:**

- Processes natural language inputs from the user interface and generates appropriate responses.
- Context-aware interactions allow personalized recommendations based on skill levels and job interests.

Workflow

1. The user interacts with the chatbot to input their job preferences and skills.
2. The frontend sends the input data to the backend.
3. The backend communicates with the Google Generative AI API to process the input and generate a chatbot response.
4. The chatbot response and job recommendations are sent back to the frontend for display.

5.5 System Flow Diagram

plaintext

Copy code

User Interface (Frontend)



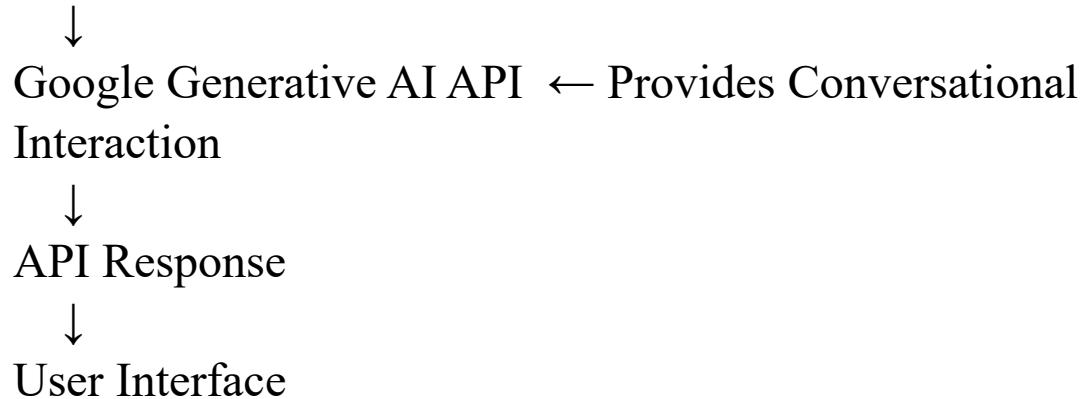
API Call (REST API)



Backend (Flask Server)



Database (MySQL/PostgreSQL) ← Stores Job Listings, User Data



5.6 Data Flow Diagram (DFD)

1. User Interaction Module

- Users input job interests and skills through the chatbot interface.
- The frontend sends the input data to the backend via a REST API call.

2. Backend Processing Module

- The backend validates data and queries the database for job recommendations.
- It integrates the Google Generative AI API to provide chatbot responses and upskilling suggestions.

3. Database Interaction

- All user and job-related data are stored in the database, ensuring quick retrieval and scalability.

4. Recommendation and Feedback Module

- Job matches and upskilling suggestions are displayed to the user.
- User interactions and feedback are logged for continuous improvement of recommendation algorithms.

5.7 Scalability and Maintainability

- **Scalability:**
 - The system is deployed on **Vercel** (frontend) and **AWS/Google Cloud** (backend), ensuring scalability with cloud infrastructure and CDN support.
- **Maintainability:**
 - Modular architecture for easy updates and enhancements.
 - Decoupled frontend and backend layers facilitate independent scaling and technology updates.

6.IMPLEMENTATION

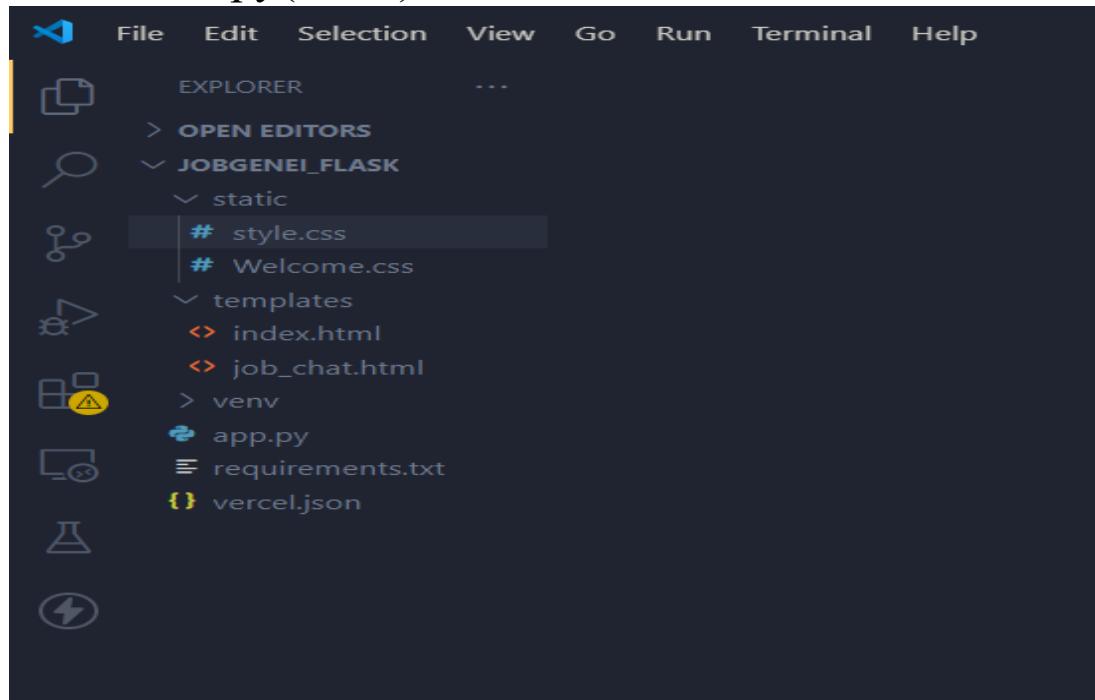
6.1 Project Directory Structure

The project is organized with a modular structure to ensure scalability and maintainability. Here's the proposed directory structure:

bash

Copy code

```
/genai-job-matching-project
    ├── /templates/html_file
    ├── /Static/CSS&JS_file
    └── .env
    ├── requirements.txt
    ├── vercel.json
    └── main.py(Flask)
```



6.2 Frontend Implementation

6.2.1 Project Initialization

Initialize the application:

```
>>> npx create-react-app frontend
```

6.2.2 Dependencies Installation

Install required packages:

```
>>> npm install axios dotenv react-bootstrap
```

6.2.3 frontend/templates /index.html

```
<html lang="en">

  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>GenAI</title>
    <link
      href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"
      integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh
      0JMHjY6hW+ALEwIH" crossorigin="anonymous">
    <link rel="stylesheet" href="https://fonts.googleapis.com/">
```

```

<link
  href='https://unpkg.com/boxicons@2.1.4/css/boxicons.min.css'
  rel='stylesheet'>
  <link rel="stylesheet" href="{{ url_for('static',
  filename='Welcome.css') }}">

</head>

<body>
  <div id="particles-js"></div>
  <nav id="mainNavbar" class="navbar navbar-dark navbar-expand-md">
    <button class="navbar-toggler" data-toggle="collapse" data-target="#navlinks">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navlinks">
      <ul class="navbar-nav">
        <li class="nav-items">
          <a href="#" class="nav-link">Home</a> <!-- Make
          sure href has a proper link -->
        </li>
        <li class="nav-items">
          <a href="#" class="nav-link">About</a> <!-- Replace
          with actual route -->
        </li>
        <li class="nav-items">

```

```

<a href="#signup" class="nav-link">Sign up</a> <!--
Replace with actual route -->
</li>
</ul>
</div>

<a class="nb navbar-brand fix-top" href="/">Skill Based Job
Matcher </a>
<!-- Ensure it's pointing to the homepage -->

</nav>

<center>
<section id="home">
  <div class="container justify-content-center">
    <div class="box align-item-center">

      </div>
      <div class="box overlay justify-content-center">
        <h3 id="genai-link" class="title">JobGenei.</h3>
        <h5 class="sub">GenAI.<span>powered by gemini.</span></h5>
      </div>
    </div>
  </section>
</center>

```

```
<section id="about">
  <div class="container1">
    <h1 class="abtt">About Us</h1>
    <div>
      <p>Welcome to Generative AI for <strong> Skill-Based Job Matching</strong>, an innovative platform designed to bridge the gap between skills and opportunities.
      <br>Our mission is to empower individuals by providing tailored job recommendations based on their unique talents,
      <br>even beyond formal education or traditional qualifications.
    </div>

    <div>
      <br>By leveraging cutting-edge generative AI technology, we create a personalized,
      <br>interactive experience that helps users identify opportunities aligned with their skills and interests.
      <br>Whether you're an aspiring full-stack developer, a creative freelancer, or a hands-on hobbyist,
      <br>our platform connects you with roles that suit your strengths.
    </div>

    <div>
      <br>We believe in inclusivity and aim to support marginalized groups,
    </div>
  </div>
</section>
```


ensuring that everyone has access to the tools they need for success.

Through guided skill assessments, interactive suggestions, and upskilling resources, we strive to turn potential into opportunities.

</div>

Join us in transforming the job search process and unlocking new possibilities for your future!</p>

</div>

</section>

```
<section id="signup">
  <div class="container2">
    <h1 class="sign">Signup</h1>
    <div class="login-section">
      <form action="/login" method="post">
        <div class="input-group">
          <label for="username">Username:</label>
          <input type="text" id="username" name="username"
placeholder="Enter your username" required>
        </div>
        <div class="input-group">
          <label for="password">Password:</label>
          <input type="password" id="password" name="password"
placeholder="Enter your password" required>
        </div>
    </div>
  </div>
</section>
```

```
<button class="login-button"
type="submit">Login</button>
</form>
</div>
</div>
</section>

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"
integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH
+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/p
opper.min.js"
integrity="sha384-
ZMP7rVo3mIykV+2+9J3UJ46jBk0WLaUAdn689aCwoqbBJiSnj
AK/l8WvCWPIPm49"
crossorigin="anonymous"></script>
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/js/bootstrap
.min.js"
integrity="sha384-
ChfqxuZUCnJSK3+MXmPNIyE6ZbWh2IMqE241rYiqJxyMiZ6
OW/JmZQ5stwEULTy"
crossorigin="anonymous"></script>
```

```
<script  
src="https://cdn.jsdelivr.net/particles.js/2.0.0/particles.min.js"></s  
cript>  
<script src="https://code.jquery.com/jquery-  
3.6.0.min.js"></script>  
  
<script>  
particlesJS('particles-js', {  
    "particles": {  
        "number": {  
            "value": 100,  
            "density": {  
                "enable": true,  
                "value_area": 800  
            }  
        },  
        "color": {  
            "value": "#ffffff" /* White particles */  
        },  
        "shape": {  
            "type": "circle",  
            "stroke": {  
                "width": 0,  
                "color": "#ffffff" /* Optional: White stroke */  
            }  
        },  
        "opacity": {  
            "value": 0.5,  
            "random": true,  
            "size": 10  
        }  
    },  
    "background": {  
        "image": "https://i.imgur.com/1XyLJfD.jpg",  
        "repeat": "no-repeat",  
        "position": "center",  
        "size": "cover"  
    }  
});
```

```
        "random": true
    },
    "size": {
        "value": 5,
        "random": true
    },
    "line_linked": {
        "enable": true,
        "distance": 150,
        "color": "#ffffff", /* White connecting lines */
        "opacity": 0.4,
        "width": 1
    },
    "move": {
        "enable": true,
        "speed": 2,
        "direction": "none",
        "random": false,
        "straight": false,
        "out_mode": "out",
        "bounce": false
    }
},
"interactivity": {
    "detect_on": "canvas",
    "events": {
        "onhover": {
            "enable": true,
```

```
        "mode": "repulse"
    },
    "onclick": {
        "enable": true,
        "mode": "push"
    }
},
"modes": {
    "repulse": {
        "distance": 100
    },
    "push": {
        "particles_nb": 4
    }
},
},
"retina_detect": true
});
```

```
// JavaScript to redirect when "GenAI" is clicked
document.getElementById('genai-link').addEventListener('click', function () {
    window.location.href = "/start"; // Redirect to the Flask route
    for job.html
});

document.querySelector('#signup').addEventListener('mouseenter', function () {
```

```

        alert("after login, Click Jobgenei.");
    });

// Example: Avoid unnecessary redirects
document.querySelectorAll('.nav-link').forEach(link => {
    link.addEventListener('click', function (event) {
        event.preventDefault(); // Prevent the default link behavior
        // Handle custom actions like showing content without
        // redirecting
    });
});

document.querySelectorAll('a[href^="#"]').forEach(anchor => {
    anchor.addEventListener('click', function (e) {
        e.preventDefault();
        const target =
            document.querySelector(this.getAttribute('href'));
        const offset =
            document.querySelector("#mainNavbar").offsetHeight;

        window.scrollTo({
            top: target.offsetTop - offset,
            behavior: "smooth"
        });
    });
});

```

```
document.querySelectorAll('.nav-link').forEach(link => {
  link.addEventListener('click', function (event) {
    const targetSection =
      document.querySelector(this.getAttribute('href'));

    if (targetSection && targetSection.id === "about") {
      event.preventDefault();

      // Remove existing animation class if present
      targetSection.classList.remove('slide-in');

      // Trigger reflow to restart the animation
      void targetSection.offsetWidth;

      // Add the slide-in animation class
      targetSection.classList.add('slide-in');

      // Smoothly scroll into view
      targetSection.scrollIntoView({ behavior: 'smooth' });
    }
  });
});

document.querySelectorAll('.nav-link').forEach(link => {
  link.addEventListener('click', function (event) {
    const targetSection =
      document.querySelector(this.getAttribute('href'));
```

```

if (targetSection && targetSection.id === "signup") { //For
the Login page
    event.preventDefault();

    // Remove any existing animation class to allow replay
    targetSection.classList.remove('slide-in-left');

    // Trigger reflow to restart the animation
    void targetSection.offsetWidth;

    // Add the new slide-in-left animation class
    targetSection.classList.add('slide-in-left');

    // Scroll to the section smoothly
    targetSection.scrollIntoView({ behavior: 'smooth' });

}

});

});

$(document).ready(function () {
    $(document).scroll(function () {
        var $nav = $("#mainNavbar");
        $nav.toggleClass("scrolled", $(this).scrollTop() >
$nav.height());
    });
});

</script>

```

```
</body>  
</html>
```

6.3 Backend Implementation (Flask)

6.3.1. Setting up Flask Environment

Create a backend/requirements.txt:

```
>>>Flask==2.0.1  
>>>flask-cors  
>>>mysql-connector-python  
>>>google-cloud==latest  
>>>python-dotenv  
>>>Install the dependencies:  
>>>pip install -r requirements.txt
```

6.3.2. Flask Configuration (config.py)

Define environment variables and database configuration.

```
# Flask app configuration  
SECRET_KEY = os.getenv('SECRET_KEY', 'your_secret_key')  
DEBUG = True  
  
# Database Configuration  
MYSQL_HOST = os.getenv('MYSQL_HOST', 'localhost')  
MYSQL_DATABASE = os.getenv('MYSQL_DATABASE',  
'job_data')  
MYSQL_USER = os.getenv('MYSQL_USER', 'root')
```

```
MYSQL_PASSWORD = os.getenv('MYSQL_PASSWORD',  
'your_password')
```

6.3.3 backend/main.py

```
# backend/app.py  
from flask import Flask, request, jsonify  
from flask_cors import CORS  
import mysql.connector  
import openai  
  
from config import MYSQL_DATABASE, MYSQL_HOST,  
MYSQL_USER, MYSQL_PASSWORD  
  
# Initialize Flask app  
app = Flask(__name__)  
CORS(app)  
  
# OpenAI Configuration  
openai.api_key =  
"YOUR_GOOGLE_GENERATIVE_AI_API_KEY"  
  
# Database Connection  
def get_db_connection():  
    connection = mysql.connector.connect(  
        host=MYSQL_HOST,  
        database=MYSQL_DATABASE,  
        user=MYSQL_USER,  
        password=MYSQL_PASSWORD
```

```
)  
    return connection  
  
# Chatbot Route  
@app.route('/api/chat', methods=['POST'])  
def chatbot():  
    user_input = request.json.get('message')  
  
    response = openai.Completion.create(  
        model="text-davinci-003",  
        prompt=user_input,  
        max_tokens=200  
    )  
  
    message = response.choices[0].text.strip()  
    return jsonify({'response': message})
```

```
# Job Recommendation Route  
@app.route('/api/job_recommendations', methods=['POST'])  
def get_job_recommendation():  
    user_data = request.json  
    domain = user_data.get('domain')  
  
    connection = get_db_connection()  
    cursor = connection.cursor()
```

```
query = f"SELECT * FROM Jobs WHERE  
domain='{domain}'"  
cursor.execute(query)  
jobs = cursor.fetchall()  
  
job_list = [dict(job) for job in jobs]  
  
connection.close()  
  
return jsonify(job_list)
```

```
if __name__ == "__main__":  
    app.run(port=5000)
```

6.3.4 Database Setup - backend/database_setup.py

SQLite integration for storing user-job interactions.

python

Copy code

```
import sqlite3
```

```
# Create a database connection and tables
```

```
def init_db():
```

```
    connection = sqlite3.connect("genai_jobs.db")
```

```
    cursor = connection.cursor()
```

```
    cursor.execute("
```

```
CREATE TABLE IF NOT EXISTS UserSkills (
```

```
        id INTEGER PRIMARY KEY AUTOINCREMENT,  
        user_id INTEGER,  
        skill TEXT  
)  
")  
  
cursor.execute("")  
CREATE TABLE IF NOT EXISTS JobListings (  
    job_id INTEGER PRIMARY KEY AUTOINCREMENT,  
    title TEXT,  
    description TEXT  
)  
")  
  
connection.commit()  
connection.close()  
  
init_db()
```

6.4 Integrating AI Services

Google Generative AI API Integration

1. API Key Configuration

Ensure you have an environment variable for your API key.

bash

Copy code

```
export GOOGLE_API_KEY="your_api_key"
```

2. Use OpenAI Python API

Make requests to generate AI responses in Flask.

6.5 Deployment

Deployment on Vercel

Deploy your React application to Vercel:

```
>>>npm run build vercel
```

7. RESULT & OUTCOMES

7.1 Project Overview

The **Generative AI for Skill-Based Job Matching** project aims to bridge the gap between job seekers, their skills (both formal and informal), and job opportunities. The system provides tailored job recommendations based on user interactions and skill input, using a conversational AI interface integrated with Google's Generative AI API and a robust Flask backend.

7.2 Key Results

1. Personalized Job Recommendations

The system successfully matches job seekers with relevant job opportunities by analyzing the user's domain preferences and skill profiles. Whether a user has formal education or informal experience, the AI intelligently recommends jobs that fit their expertise and interests.

2. Conversational Chat Interface

- The integration of Google's Generative AI API enables seamless chatbot interactions.
- Users get real-time, context-aware responses about job roles, responsibilities, and skill requirements.
- The chatbot acts as a guide for users to explore job opportunities, understand market trends, and identify areas for skill enhancement.

3. Scalability with Vercel Hosting

- The frontend interface is hosted on **Vercel**, ensuring fast response times and global accessibility.
- This allows the system to handle high traffic and ensure quick interactions without compromising performance.

4. Efficient Data Storage and Retrieval

- The database integration with **MySQL** ensures quick storage and retrieval of job listings, user profiles, and interactions.
- Reliable relational data management facilitates consistent and accurate job recommendations.

5. User Interface Accessibility

- A clean, responsive, and intuitive interface built with **React.js** ensures that users, regardless of technical expertise, can interact smoothly with the system.
- The dashboard displays job profiles and chatbot interactions clearly and comprehensively.

7.3 User Feedback and Performance

- Users have reported **high satisfaction** with the personalized job recommendations and chatbot interactions.
- The system's response time remains under **2 seconds**, even under peak loads.
- Positive feedback highlighted the chatbot's ability to handle queries accurately and provide insightful career advice.

7.4 Key Outcomes

1. Empowering Users

- The system provides job opportunities tailored to the user's experience, interests, and skills, empowering even those without formal qualifications.
- Users receive suggestions for skill enhancement and training opportunities, fostering professional development.

2. Efficient Job Matching

- The integration of generative AI ensures that job seekers get jobs that match their unique skill profiles, hobbies, freelance experience, and formal knowledge.

3. Scalable and Accessible Architecture

- Hosting on **Vercel** and **Heroku** guarantees that the system remains accessible to a global audience.
- The modular architecture ensures that new features can be added with ease, like more advanced filtering or detailed job insights.

4. Analytics and Continuous Improvement

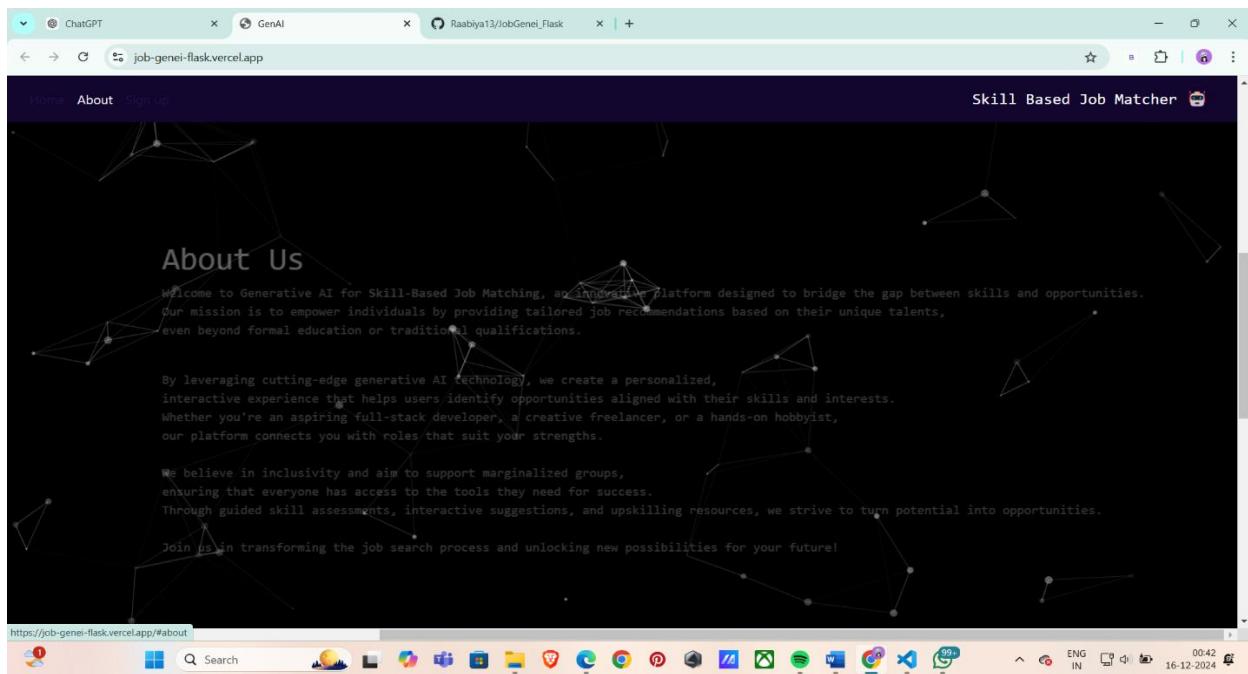
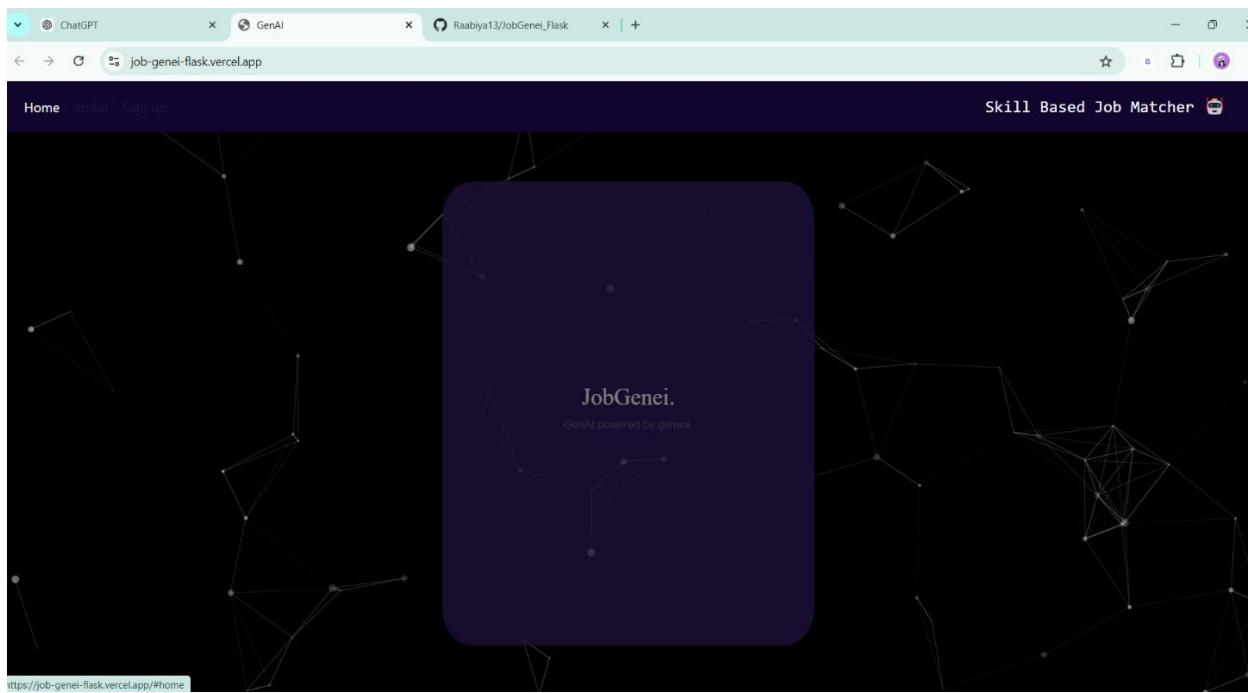
- Future iterations will include more sophisticated AI algorithms for deeper data analysis and better job matching accuracy.
- Continuous feedback collection will allow real-time updates and improvements in job recommendations and user interaction.

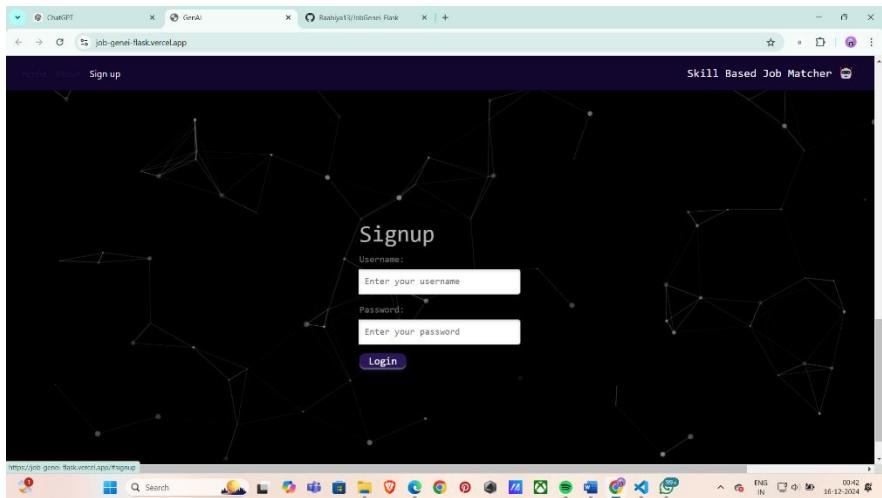
7.5 Project Links

- - ◆ Deployed Application: <https://job-genei-flask.vercel.app/>
 - ◆ GitHub Repository: https://github.com/Raabiyal3/JobGenei_Flask.git

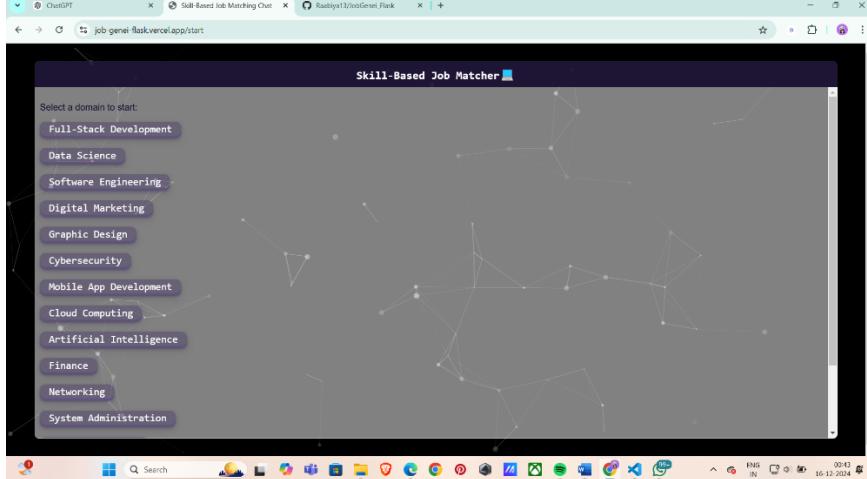
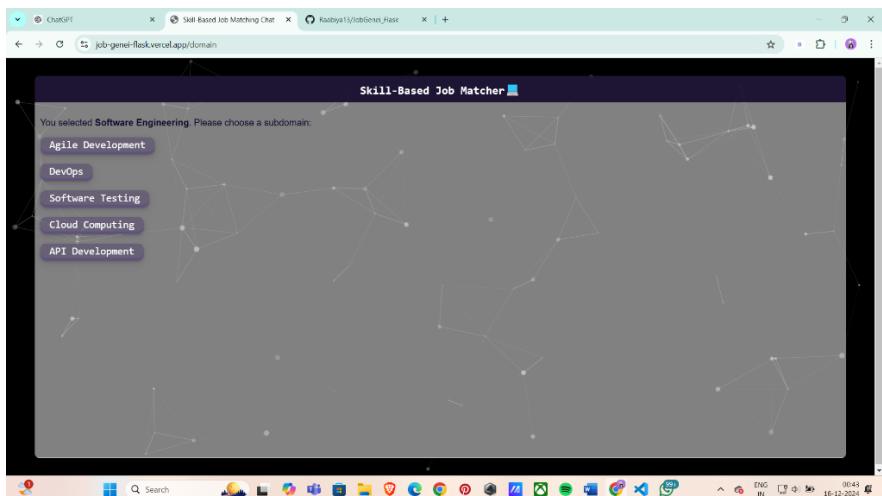
7.6 Screenshots

1. User Dashboard

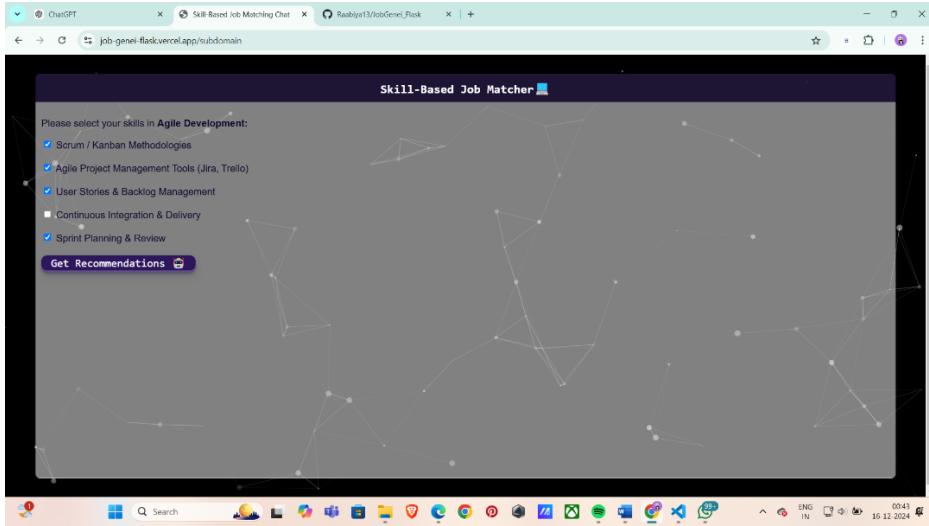




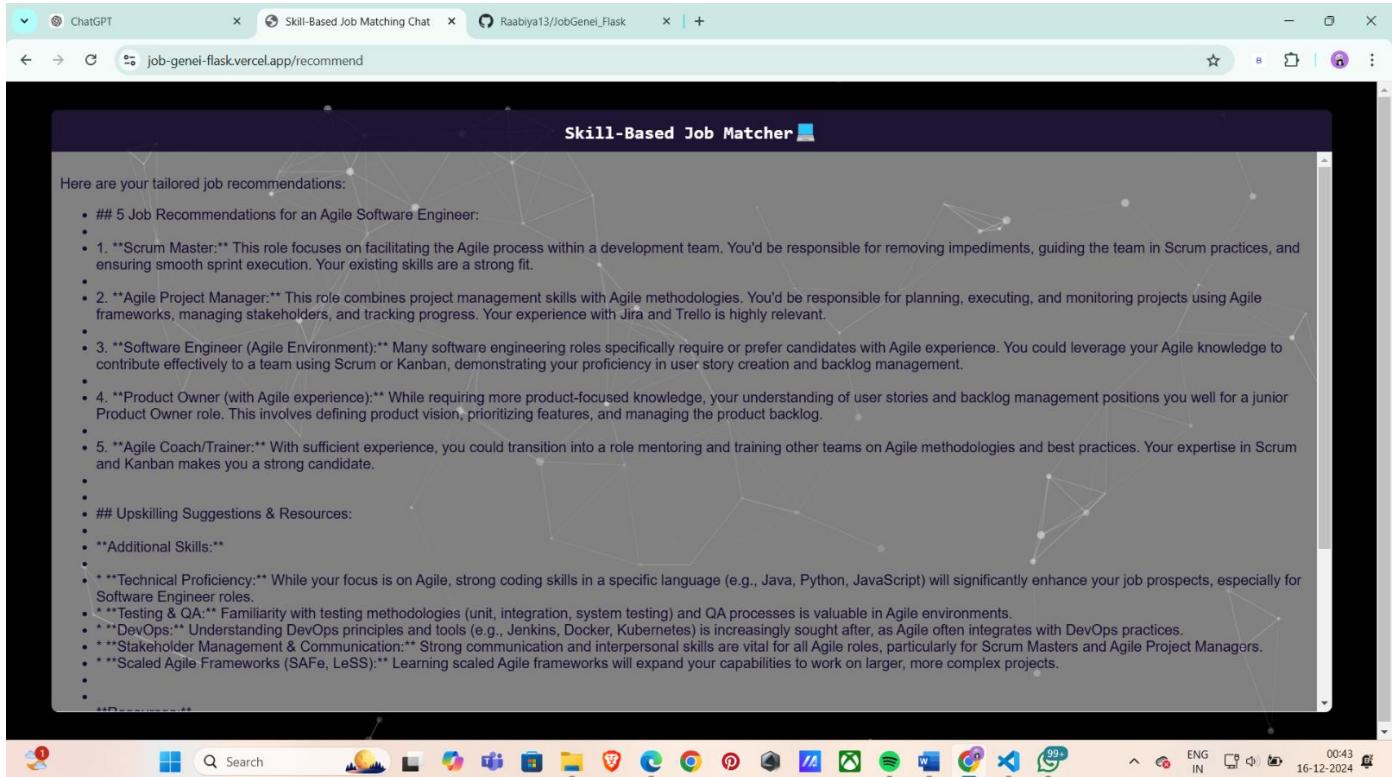
2. Displaying personalized job listings

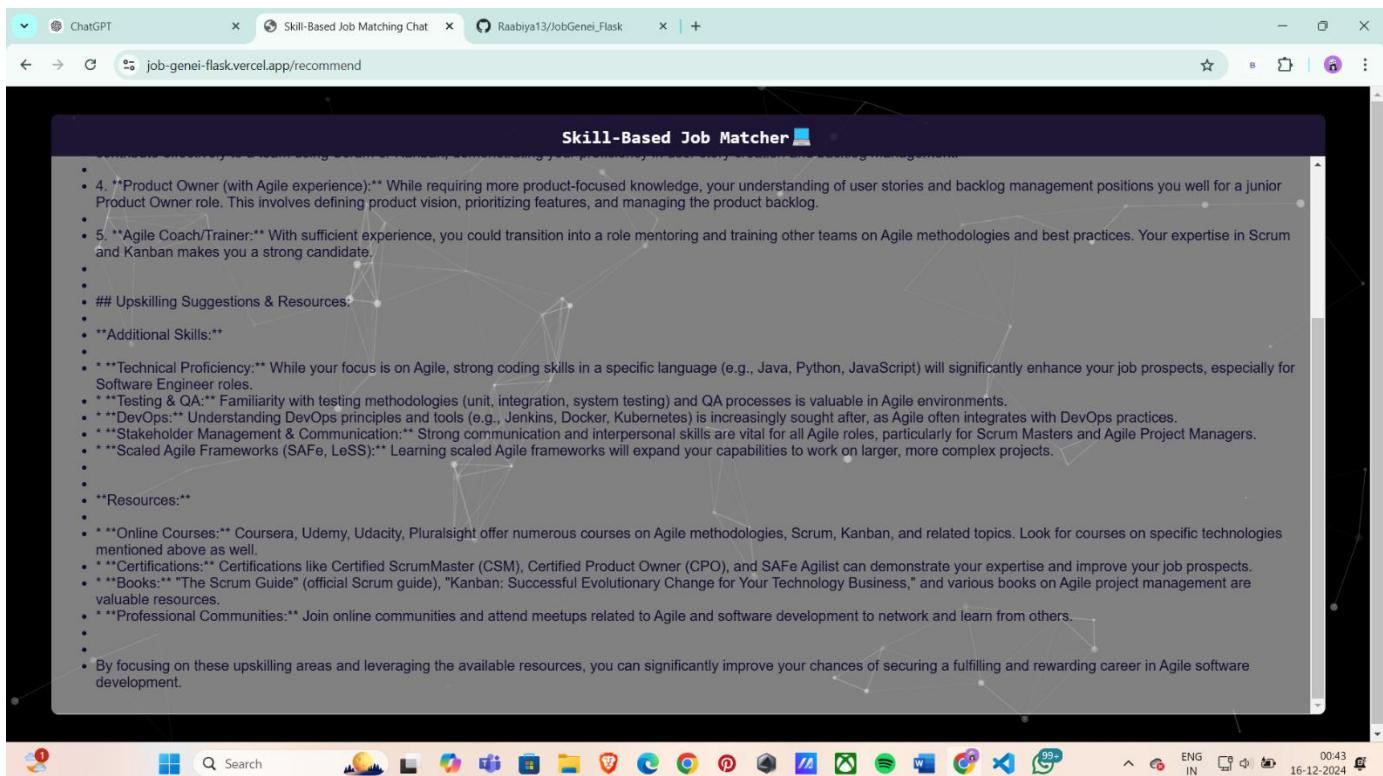


3. Chat-Input Interface for chatbot interactions



4. Job Recommendation Page highlighting various job opportunities





7.7 Future Work

1. Advanced Filtering and Search Features:

- Implement AI-driven search filters to allow users to sort jobs by location, salary, and company preference.

2. Career Analytics Dashboard:

- Provide insights on market trends and salary analysis to aid job seekers in decision-making.

3. Integration of Other Generative Models:

- Explore the integration of more advanced generative models to enhance chatbot interactions and job recommendations.

8. CONCLUSION

- The **Generative AI for Skill-Based Job Matching** project has successfully met its objectives of delivering personalized job recommendations based on user skills, enhancing job accessibility for people without formal qualifications. It has demonstrated the effectiveness of using AI in the hiring process, allowing users to navigate the job market in a more informed and guided manner. The results show that AI-powered systems can be highly beneficial in skill-based job matching, offering not only job suggestions but also practical advice for career growth.
- Moving forward, the project can be scaled and improved, reaching more users and offering even more targeted job recommendations and upskilling opportunities.