Product Name: CareerMatch- AI Powered Job Platform

Submitted by:

Team Name/Project Group: Team B

Team Members:

- Preeti Deora
- Pawan Kumar
- Tisha Gurjar
- Yuvraj Singh

Under the Supervision of:

Mentor/Supervisor: Mr. Shyam Bhushan

Institution/Organization: AIPlaneTech, Jodhpur, Raj., India

Company Name: AIPlaneTech Pvt. Ltd.

Submission Date: 02/08/2025

Declaration (Team)

We, the undersigned, hereby affirm that the product presented herein is the result of our team's dedicated efforts, original research, and collaborative development. This work reflects our independent contributions and has been completed with integrity, without any form of plagiarism or duplication.

Furthermore, we confirm that this product has not been previously submitted, published, or presented to any other institution, organization, or platform for academic, professional, or commercial purposes.

Team Members :

1. Preeti Deora
2. Pawan Kumar
3. Tisha Gurjar
4. Yuvraj Singh

Certificate

This is to certify that the team, comprising dedicated members, has successfully completed the project titled “Career Match” under the guidance of Mr. Shyam Bhushan Sir. The project demonstrates innovative solutions for job application processes, showcasing the team's technical expertise and commitment. This certification acknowledges their diligent efforts and successful execution at Aiplanetech Pvt. Ltd., Jodhpur, submitted on 2nd July 2025.

Shyam Bhushan

AIPlaneTech Pvt. Ltd., India

Table of Contents

Acknowledgement

We, the members of this team, extend our sincere gratitude to everyone who made this internship an enriching and memorable experience. This journey has been invaluable in applying our academic knowledge to real-world challenges and strengthening our skills in teamwork, problem-solving, and professional communication.

First, we are deeply grateful to **AIPLANETECH** for providing us with this esteemed opportunity. The hands-on exposure and guided mentorship have been fundamental to our professional growth.

Our heartfelt appreciation goes to our internship supervisor, **Mr. Shyam Bhushan**, for his continuous guidance and constructive feedback. His mentorship was crucial in helping us navigate the project's technical frameworks and professional standards.

We also thank the entire team at AIPLANETECH for their cooperation and encouragement, which fostered a positive and collaborative work environment. We would also like to acknowledge our academic institution, **MBM University**, for their support in bridging the gap between academia and industry.

Finally, we are proud of the collaborative spirit within our own team. Our collective effort and mutual respect were key to the successful completion of this product. We also thank our families and friends for their unwavering motivation.

This report reflects the learning and growth we experienced as a group, and we are proud to have had this opportunity.

Thank you.

Abstract

Career Match is an intelligent, AI-powered recruitment platform designed to revolutionize the hiring process by seamlessly connecting job seekers with suitable opportunities and empowering HR professionals to manage recruitment with unparalleled efficiency. Leveraging the advanced capabilities of Google Gemini and the robust, scalable infrastructure of Google Cloud Platform (GCP), the platform delivers intelligent resume parsing, automated job matching, and dynamic content generation in a secure, user-friendly web application.

Key components of Career Match include AI-powered resume analysis, role-specific dashboards for Applicants, HR Representatives, and Administrators, and automated tools for creating tailored resumes and cover letters. The platform integrates with standard communication protocols like SMTP for automated application submissions and notifications, offering a complete and intuitive experience.

With a focus on automation, efficiency, and user-centric design, Career Match enables job seekers to accelerate their search while allowing recruiters to source qualified talent more accurately and quickly. This report outlines the core architecture, technology stack, and user workflows of the platform, demonstrating how Career Match addresses the critical challenges of inefficiency and manual effort in today's recruitment landscape.

1. Purpose of Product

Career Match is an intelligent platform built to simplify and optimize the job search and recruitment process. It automates key tasks such as resume parsing, intelligent job matching, and communication to ensure the right talent finds the right opportunities with maximum efficiency. The product helps reduce manual effort for both job seekers and recruiters, enhances the accuracy of hiring, and provides a streamlined, modern user experience.

2. Product Features

Key features of Career Match include AI-powered resume parsing and profile creation , an intelligent job matching engine based on skills and preferences , and automated generation of job-specific resumes and cover letters. It offers distinct dashboards for Applicants, HR Representatives, and Administrators , AI-assisted job posting for recruiters , and direct application submission with automated email notifications. The platform features a responsive, user-friendly UI for seamless use across all devices.

3. Technology Used

The product is developed using

React with TypeScript for the frontend, ensuring a responsive and dynamic user interface. The backend is built on a

microservices architecture and connects to a **PostgreSQL** database for data management. AI capabilities are powered by the

Google Gemini API. The entire solution is hosted on

Google Cloud Platform (GCP), utilizing services like **Cloud Storage** for files and **Cloud SQL** for the database.

Firebase Authentication is used for secure user login and role management.

4. Business Architecture

Career Match supports key stakeholders, including

Job Seekers, HR Representatives/Recruiters, and Platform Administrators. The platform aligns with critical business functions like talent acquisition, recruitment process optimization, and application management. For recruiters, it enhances operational efficiency by automating repetitive tasks. For job seekers, it provides powerful tools to conduct a more effective and targeted job search, thereby improving the overall quality and speed of the hiring cycle.

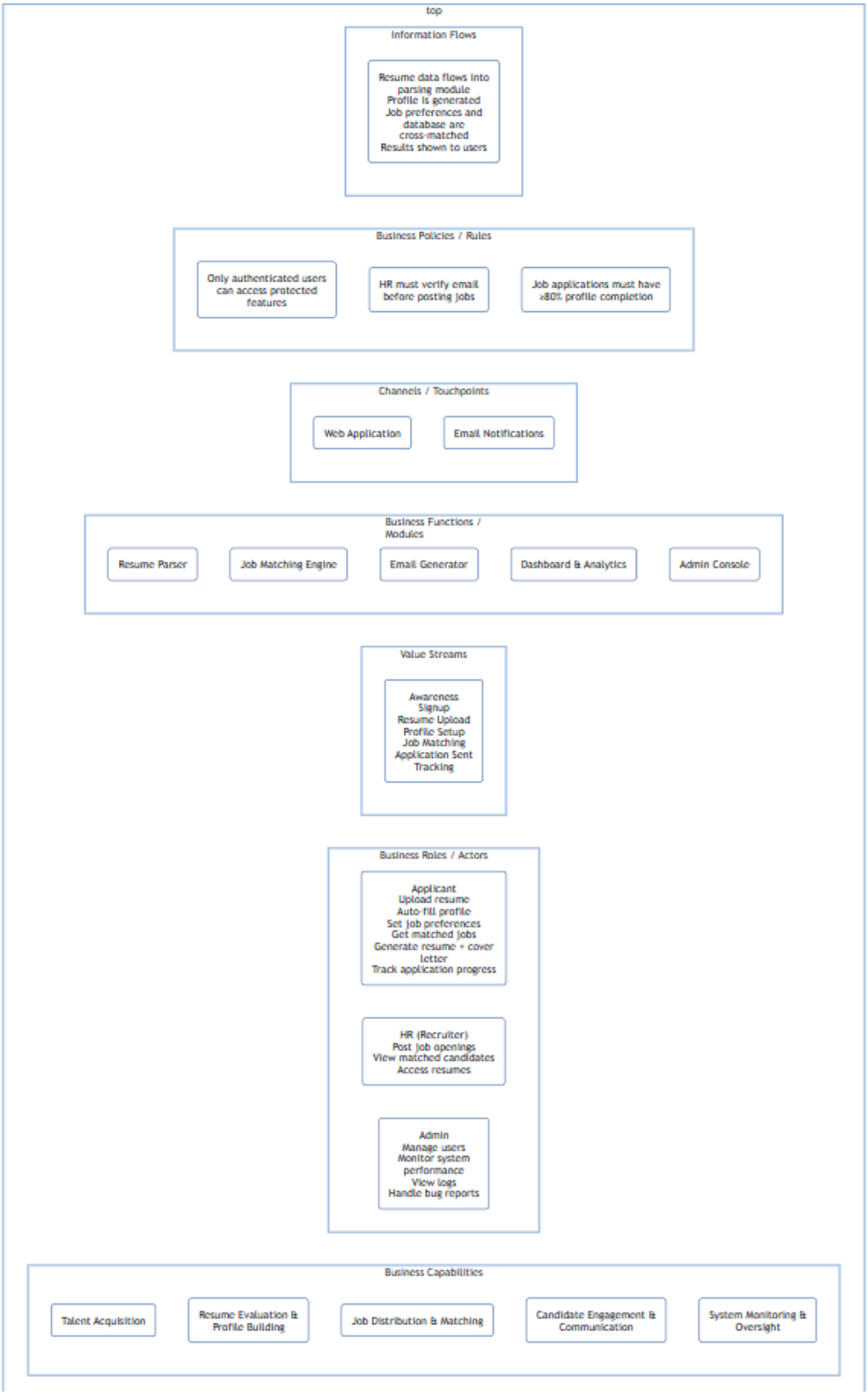


figure 1.1 : Business architecture of product

5. Solution Architecture

The architecture is

microservice-based, with decoupled frontend and backend components communicating via **RESTful APIs**. The

Google Gemini API functions as the core AI engine for parsing, matching, and content generation. Data is managed in a

PostgreSQL database for structured data and **GCP Cloud Storage** for resume files. The system is deployed as containerized services on GCP and secured with

Firebase Authentication and role-based access control to ensure data protection and integrity.

6. Process Workflows

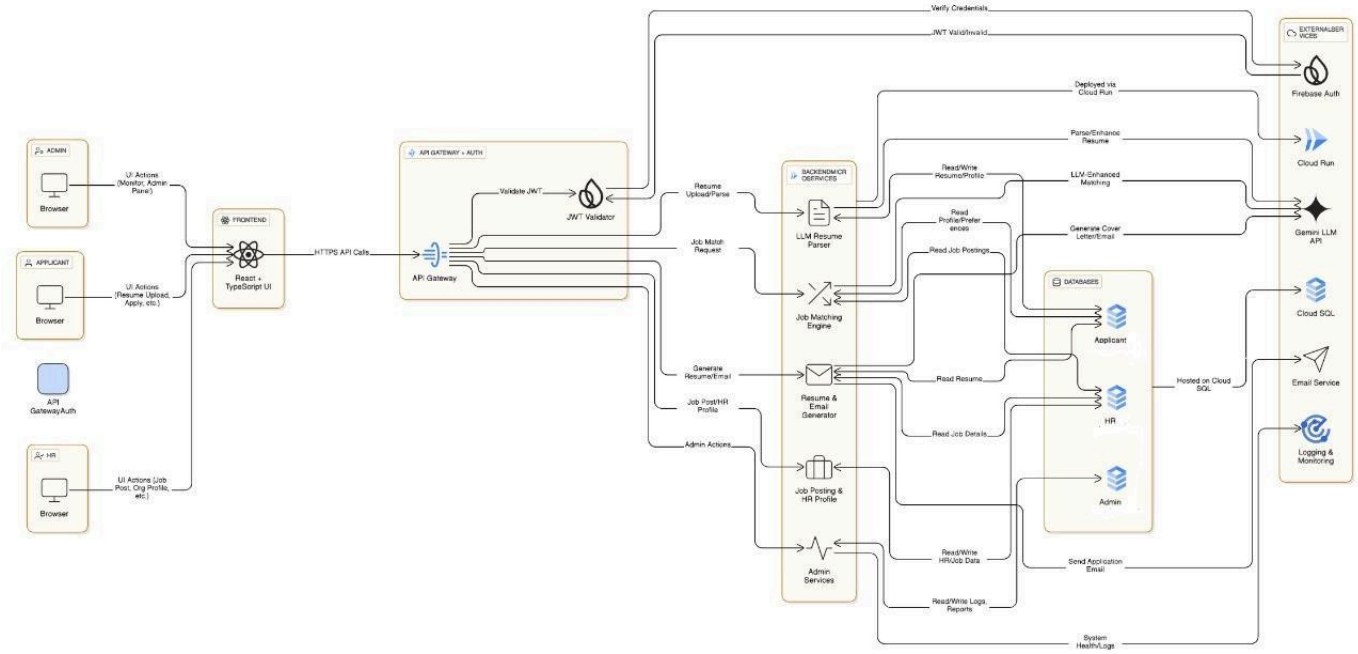


figure 1.2 : Workflow process of product

The job seeker workflow begins with registration and AI-powered profile creation via resume upload. This is followed by setting job preferences, receiving AI-matched job recommendations, and applying with auto-generated resumes and cover letters. The HR workflow involves posting jobs (manually or with AI assistance) , viewing a ranked list

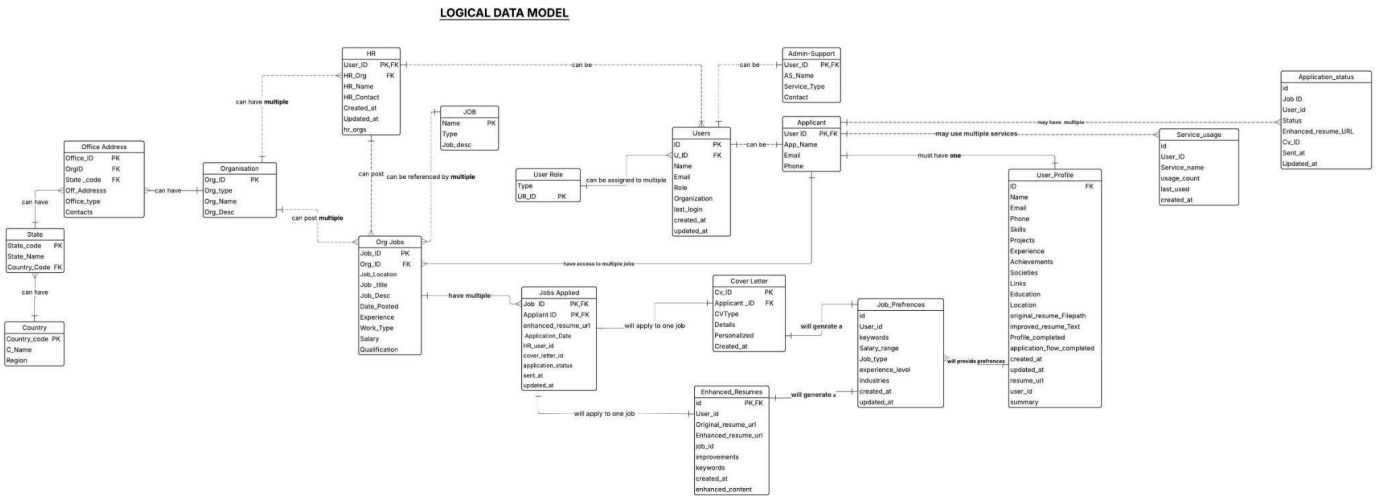
of matched candidates on a dashboard , reviewing profiles, and managing the application pipeline through to hiring.

7. AI Platform Services

Career Match leverages the

Google Gemini API as its central AI service. Its core AI use cases include:

- **AI-Powered Resume Parsing:** To extract structured data like work experience, skills, and education from uploaded documents.
- **Intelligent Job Matching:** To analyze user profiles and job descriptions and provide highly relevant matches based on skills, experience, and preferences.
- **Dynamic Content Generation:** To automatically create tailored, professional-quality resumes, cover letters, and application emails for specific job opportunities.



Introduction

1. Background

With the increasing complexity of the modern job market, traditional recruitment processes have become largely inefficient and time-consuming for both employers and candidates. Manual methods of screening resumes and matching applicants to roles are often slow and struggle to keep pace with the volume of applications. Career Match was conceptualized to address these challenges by leveraging automation, artificial intelligence, and modern web technologies to create a smarter and faster hiring ecosystem.

2. Why This Product is Needed

Job seekers often struggle to find relevant opportunities that match their unique skills and career goals, while recruiters are inundated with applications, making it difficult to identify the most qualified candidates efficiently. This disconnect creates a significant gap between talent and opportunity. Career Match fulfills the urgent need for an intelligent, automated platform that connects applicants with employers, streamlines the application process, and ensures the right talent finds the right opportunities with unparalleled efficiency.

3. Target Audience or Use-Case

Career Match is designed for three distinct user groups:

- **Job Seekers/Applicants:** Individuals looking to find and apply for suitable job opportunities using advanced tools to enhance their search.
- **HR Representatives/Recruiters:** Professionals seeking to find qualified candidates, manage the recruitment process, and track hiring metrics efficiently.
- **Platform Administrators:** System overseers responsible for user management, monitoring platform health, and accessing performance insights.

Team Structure & Roles

Member Names	Role	Responsibilities	Additional Responsibility
Preeti Deora	Product Lead and Full Stack Developer	-Define and document comprehensive user stories and product requirements, coordinating closely with the Scrum Master to plan and manage agile sprints. - Lead the overall product vision and planning, ensuring alignment between technical implementation and user needs. - Design and implement the user authentication and login system with support for multiple user roles (e.g., HR, Applicant). -Perform full-stack integration of all microservices with each other, the frontend interface, and the backend database system. -Assist in designing and managing the database schema and data models to ensure smooth data flow and scalability.	- Develop and integrate the automated email application system for applicants, improving user interaction and engagement. -Contribute to the development of frontend components on both the Applicant and HR sides of the platform. -Collaborate cross-functionally to ensure seamless coordination between backend, frontend, and deployment processes. - Handle client-side and server side error gracefully.

Pawan Kumar	Full-Stack Developer (HR Module) & Technical Writer	- Develop the end-to-end HR module, including the HR dashboard UI and corresponding backend APIs. - Implement the "Create Job Post" feature with both manual forms and prompt-based AI generation. - Build the system for HR users to view, manage, and track candidate applications. - Design the database schema and API endpoints for all HR-related functionalities.	- Lead the creation, compilation, and documentation of the final product report. - Write and execute test cases for the HR module. - Collaborate with the team to document API endpoints.
Yuvraj Singh	Scrum Master, Backend & DevOps Developer (AI Services)	- Develop the backend logic for the intelligent Job Matching engine. - Implement the AI services for the Resume Enhancer and Cover Letter Generator using the Google Gemini API. - Develop auto apply service for Applicant and connecting it with the enhanced resume and cover letter service. - Deploy backend microservices as containerized applications to **Google Cloud Run**. - Manage the cloud infrastructure and ensure service availability and performance.	- Implement and manage monitoring and logging for all deployed backend services to ensure high availability. - Manage github, jira and confluence. -Assigned User Stories to individuals.
Tisha Gurjar	Lead Frontend Developer (UI/UX & AI Features)	- Design and develop the frontend interface of the platform, focusing on clean, responsive, and user-friendly UI components. - Manage the overall UI/UX design, ensuring consistency, accessibility, and a	- Build and integrate a general-purpose chatbot to assist users with queries and provide information about the website.

		seamless experience across all user roles and pages. - Develop the Resume Upload and Profile Autofill features to streamline user onboarding and enhance usability. - Implement role-based access control (RBAC) on the frontend, determining how different user types (e.g., HR, Applicant) interact with the platform. -Ensure cross-browser compatibility and visual consistency across all components.	- Collaborate with backend and integration teams to align frontend behavior with data and service flows.

Product Overview

1. Product Architecture

Career Match is built on a modern, microservice-based architecture designed for scalability and reliability. The frontend, developed with **React and TypeScript**, communicates with the backend via **RESTful APIs**. Core functionalities like resume parsing, job matching, and content generation are powered by the **Google Gemini API**. The platform uses **PostgreSQL** for structured data storage, such as user profiles and application tracking. The entire solution is cloud-native, with services containerized using **Docker** and deployed on the **Google Cloud Platform (GCP)**, ensuring a secure and high-performance user experience.

2. Features / Modules

Career Match is composed of the following core modules:

- **AI Resume Parser:** Securely ingests and processes user-uploaded resumes (PDF, DOCX) to automatically extract key information and populate the applicant's profile.

Resume Parsing Workflow This diagram will illustrate the flow from a user uploading a resume file to GCP Cloud Storage, triggering the Google Gemini API for parsing, and populating the user's profile data in the PostgreSQL database.

- **Intelligent Job Matcher:** An AI-powered engine that analyzes applicant profiles, skills, and preferences to match them with the most relevant job postings available on the platform.

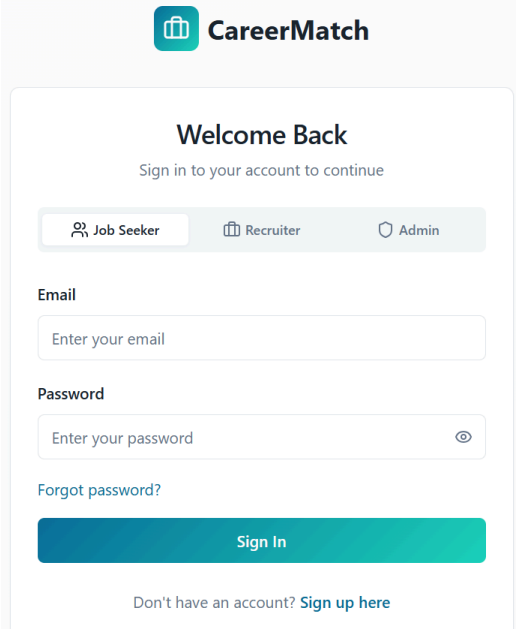
Job Matching Process This diagram will show how the matching engine queries the database for candidate profiles and active job postings, uses AI to calculate a relevance score, and delivers personalized job recommendations to the user's dashboard.

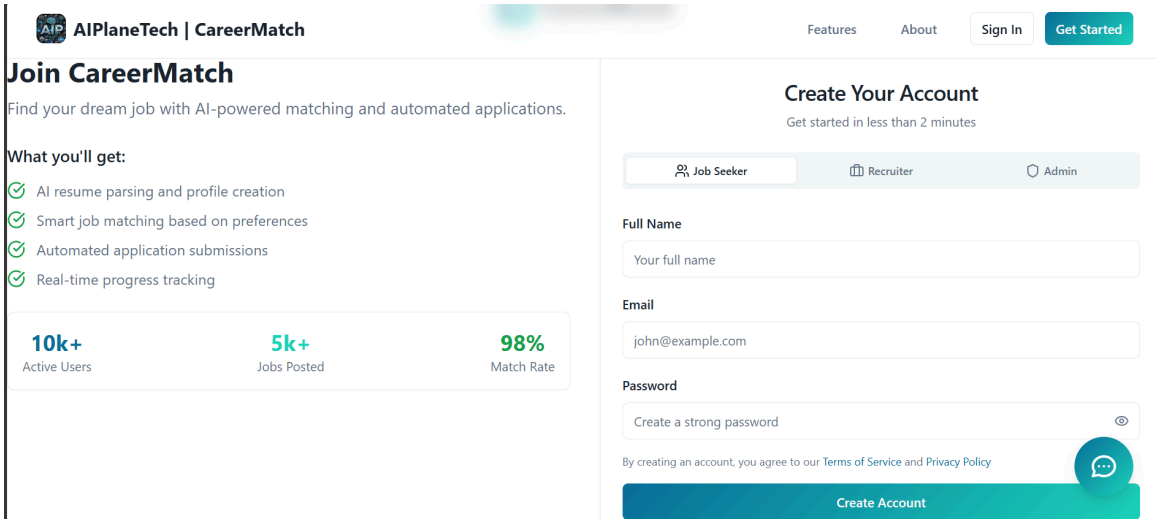
- **Automated Content Generator:** A suite of AI tools that creates tailored resumes and professional cover letters based on the requirements of a specific job description, helping applicants stand out.
- **Role-Based Dashboards:** Separate, intuitive interfaces for Applicants, HR Representatives, and Administrators, each displaying relevant metrics, tasks, and analytics.

- **Application Management Module:** Allows applicants to track the status of their submitted applications and enables recruiters to manage incoming candidates through their entire hiring pipeline.
- **User Management & Security:** Provides secure, role-based access control (RBAC) for all users, managed through **Firebase Authentication** to protect user data and privacy.

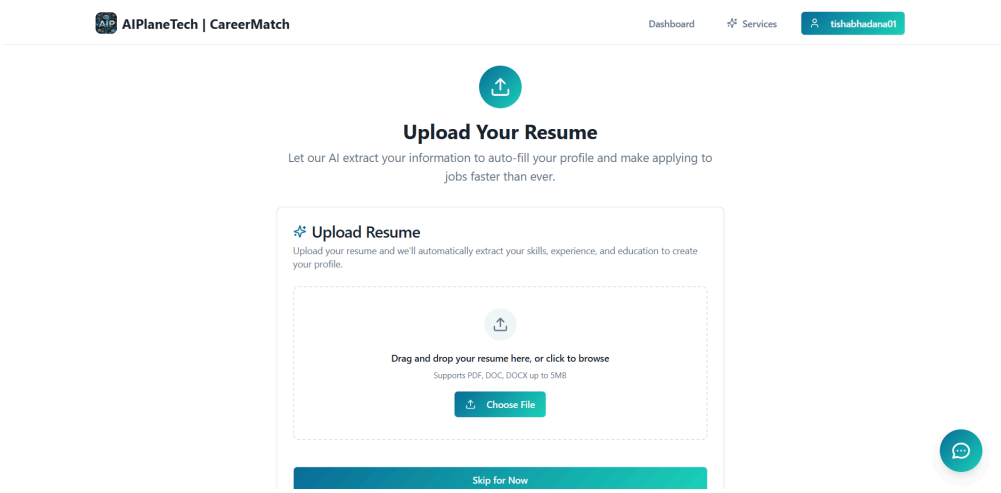
3. Screenshots or UI Wireframes

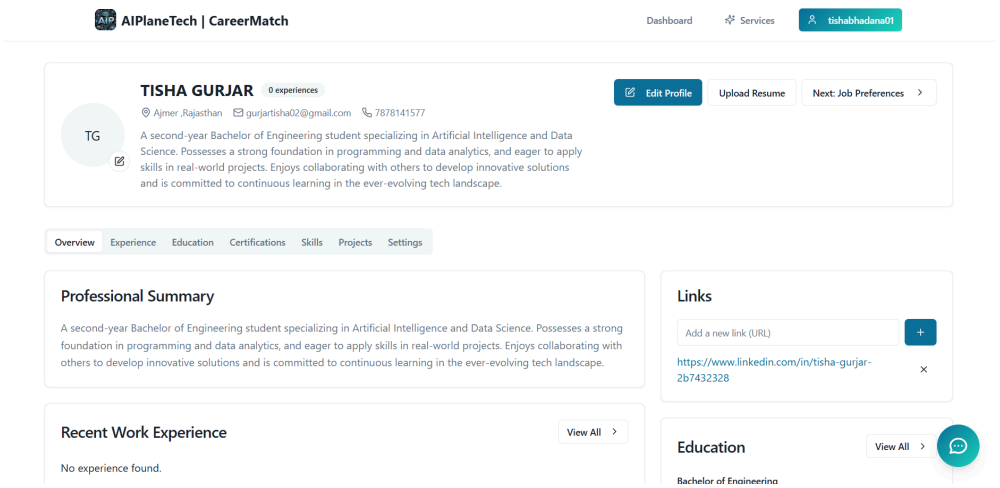
- **Login & Registration Page:** A clean and secure interface for users to sign up or log in based on their role (Applicant, HR, or Admin).

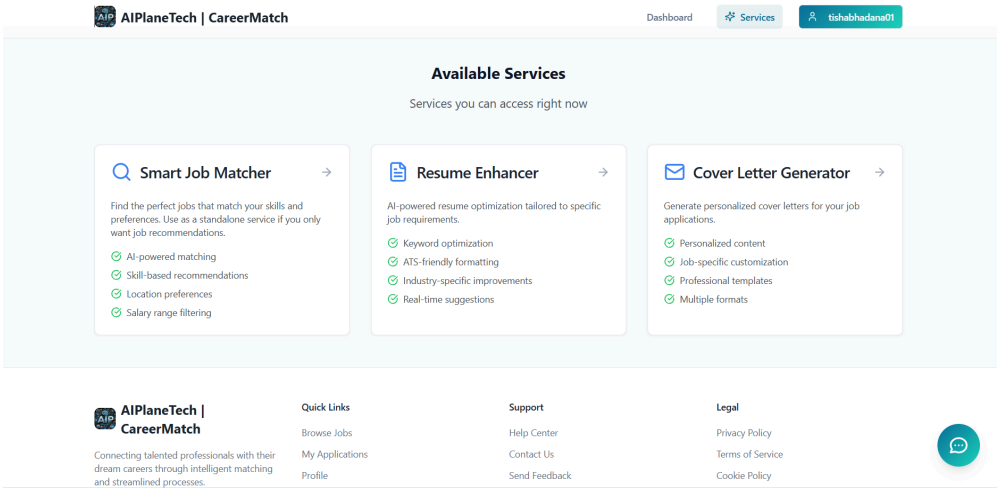




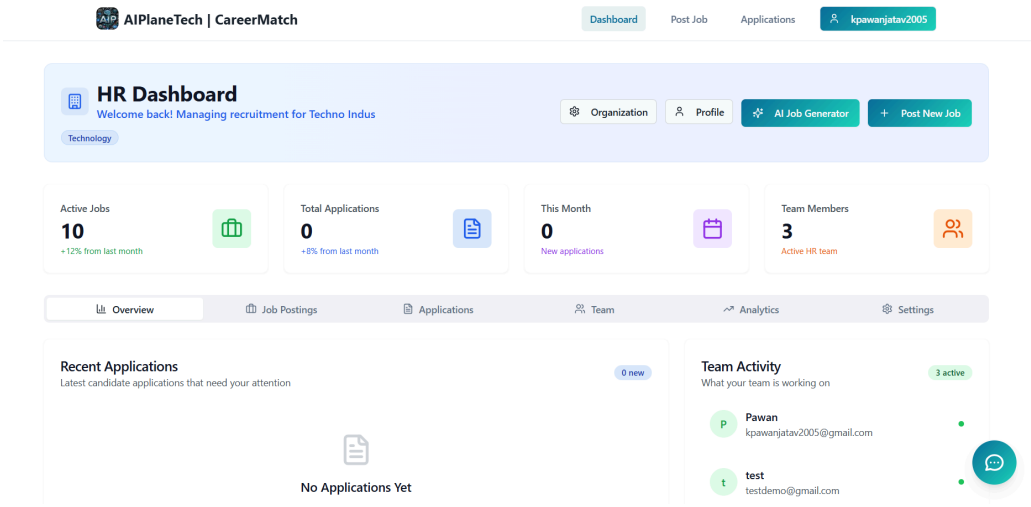
- **Applicant Dashboard:** A personalized view displaying key metrics like matched jobs and applications sent, along with lists of recommended jobs and application statuses.

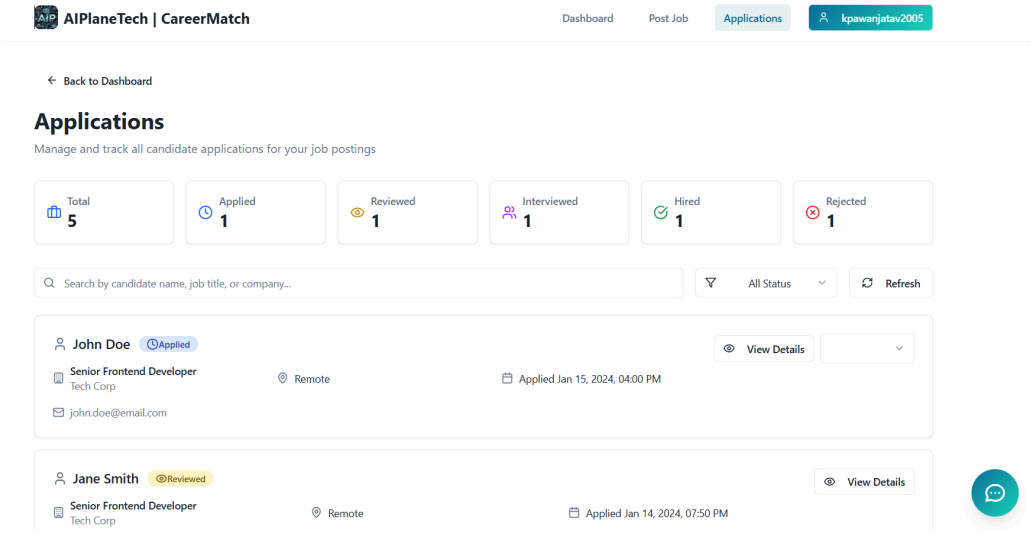


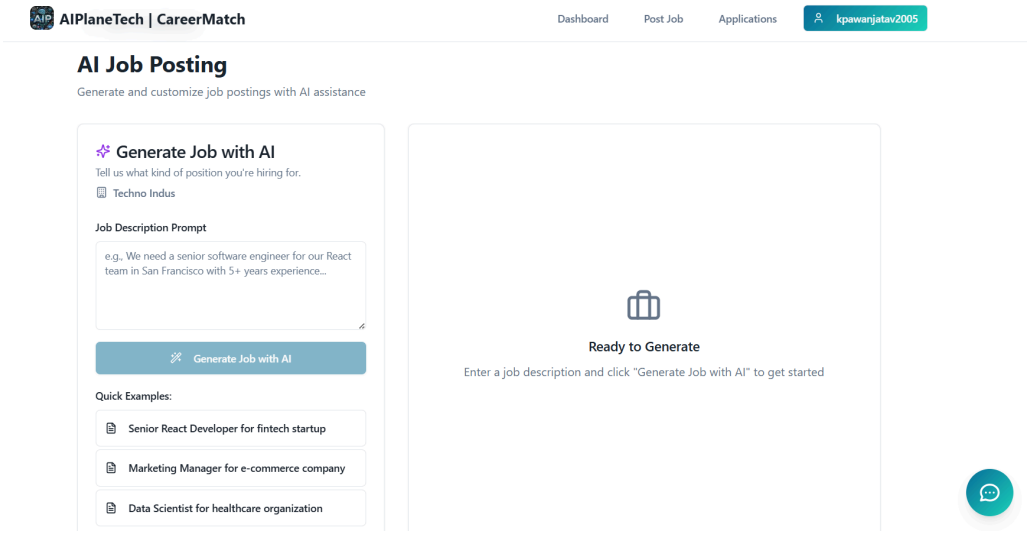




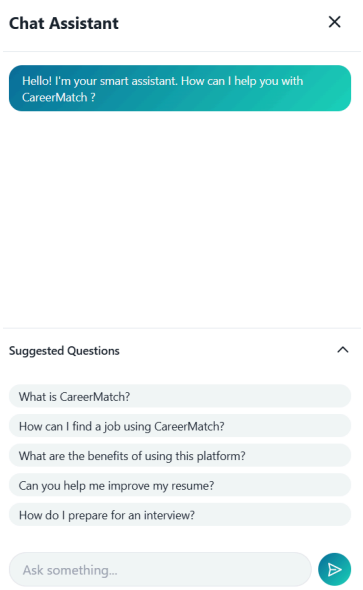
- **HR Dashboard:** A comprehensive interface for recruiters to manage job postings, view matched candidates, track application pipelines, and access hiring analytics.


AIPlaneTech | CareerMatch
Dashboard
Post Job
Applications
kpawanjatav2005
HR Dashboard
Welcome back! Managing recruitment for Techno Indus
Technology
Organization
Profile
AI Job Generator
Post New Job
Active Jobs
10
+12% from last month
Total Applications
0
+8% from last month
This Month
0
New applications
Team Members
3
Active HR team
Overview
Job Postings
Applications
Team
Analytics
Settings
Recent Applications
Latest candidate applications that need your attention
0 new
No Applications Yet
Team Activity
What your team is working on
3 active
Pawan
kpawanjatav2005@gmail.com
test
testdemo@gmail.com


AIPlaneTech | CareerMatch
Dashboard
Post Job
Applications
kpawanjatav2005
Back to Dashboard
Applications
Manage and track all candidate applications for your job postings
Total
5
Applied
1
Reviewed
1
Interviewed
1
Hired
1
Rejected
1
Search by candidate name, job title, or company...
All Status
Refresh
John Doe
Applied
Senior Frontend Developer
Tech Corp
Remote
Applied Jan 15, 2024, 04:00 PM
john.doe@email.com
View Details
Jane Smith
Reviewed
Senior Frontend Developer
Tech Corp
Remote
Applied Jan 14, 2024, 07:50 PM
View Details


AIPlaneTech | CareerMatch
Dashboard
Post Job
Applications
kpawanjatav2005
AI Job Posting
Generate and customize job postings with AI assistance
Generate Job with AI
Tell us what kind of position you're hiring for.
Techno Indus
Job Description Prompt
e.g., We need a senior software engineer for our React team in San Francisco with 5+ years experience...
Generate Job with AI
Quick Examples:
Senior React Developer for fintech startup
Marketing Manager for e-commerce company
Data Scientist for healthcare organization
Ready to Generate
Enter a job description and click "Generate Job with AI" to get started

- **Profile & Preferences Page:** Allows users to manage their personal details, professional experience, skills, and job search criteria to refine their matching results.



Tools & Technologies

Frontend Layer

- **React.js (with TypeScript):** Used to build a fast, component-driven, and type-safe user interface.
- **CSS3:** Provides modern styling, layouts, and responsive design to enhance the user experience.

Backend Layer

- **Microservices Architecture:** The backend is designed as a set of independent services to ensure scalability and maintainability.

- **RESTful APIs:** Enables seamless communication between the frontend client and the backend services.

Data Storage & Processing

- **PostgreSQL:** The primary relational database for storing all structured data, including user profiles, job postings, application data, and system logs.
- **Google Cloud Storage:** Used for securely storing all user-uploaded files, such as resumes, in various formats (PDF, DOCX).

Platform & Infrastructure

- **Google Cloud Platform (GCP):** The core cloud infrastructure for hosting the application, database, and storage, providing scalability and enterprise-grade security.
- **Docker:** Used to containerize each application component, ensuring consistent and reliable deployment across all environments.

AI & Authentication

- **Google Gemini API:** The core AI engine that powers resume parsing, semantic job matching, and the automated generation of application content.
- **Firebase Authentication:** Provides a secure and robust system for managing user authentication, sessions, and role-based access control.

Development Methodology

1. AI Framework & Platform

Career Match leverages a powerful and targeted AI framework centered on the **Google Gemini API**. The platform integrates AI into its core workflows:

- **AI-Powered Data Extraction:** Google Gemini is used to perform sophisticated parsing of unstructured resume documents, accurately identifying and extracting structured information like work experience, skills, education, and contact details.
- **Semantic Matching Engine:** The AI moves beyond simple keyword matching, using semantic understanding to analyze the context of both a candidate's profile

and a job description. This ensures more relevant and accurate matches between applicants and opportunities.

- **Natural Language Generation (NLG):** The platform uses Gemini's generative capabilities to create high-quality, context-aware content, such as professional resumes and cover letters tailored to specific job applications.

This AI architecture supports a real-time, intelligent, and highly automated recruitment experience for all users.

2. Agile

The Career Match platform was developed using the

Agile methodology to ensure flexibility and continuous improvement throughout the project lifecycle. This approach emphasized:

- **Iterative Development** with frequent delivery of functional components.
- **Continuous Feedback** loops with mentors and stakeholders to refine features.
- **Adaptive Planning** to respond to new requirements and technical challenges.

This allowed the team to rapidly build, test, and enhance core modules like the resume parser, job matching engine, and user dashboards.

3. Scrum

Within the Agile model, the team utilized the

Scrum framework to structure development work effectively.

- **Sprints:** Development was organized into 1–2 week sprints, with each sprint focused on delivering a specific, high-priority milestone (e.g., implementing the applicant dashboard or integrating the AI content generator).
- **Daily Stand-ups:** Brief daily meetings were held to sync progress, discuss challenges, and coordinate tasks.
- **Sprint Planning & Retrospectives:** These ceremonies ensured that tasks were well-defined before each sprint and that the team continuously improved its processes after each cycle.

The Scrum framework fostered transparency, accountability, and adaptability, which were crucial for the successful delivery of the platform.

4. Task Management Tools Used

To facilitate collaboration and track progress within the Agile/Scrum framework, the team used a modern toolset:

- **GitHub Projects:** For creating sprint boards, managing the product backlog, and tracking issues alongside the codebase.
- **Jira** – For visual sprint boards and backlog grooming.
- **VS Code:** As the primary integrated development environment (IDE) for coding, debugging, and managing containerized workflows with Docker.

Implementation Details

1. Deployment Process

The deployment of Career Match is designed for a containerized and cloud-native environment to ensure scalability, consistency, and security. All application services—including the frontend, backend microservices, and database—are managed as distinct components within the Google Cloud Platform (GCP).

The deployment steps include:

- **Backend Containerization:** Each microservice in the backend architecture (e.g., user management service, job posting service, application tracker) is individually packaged into a **Docker container**. This approach ensures that each service runs in a consistent and isolated environment from development to production.
- **Frontend Build:** The **React.js** frontend application is compiled into a static production build. This bundle of HTML, CSS, and JavaScript files is then deployed to **Google Cloud Storage** and served globally for fast, low-latency access.
- **Database Setup: PostgreSQL**, hosted on GCP's Cloud SQL, serves as the primary relational database. The database schema for storing user profiles, job postings, and application data is initialized and managed using **SQL migration scripts**.
- **AI Integration:** The connection to the **Google Gemini API** is configured securely using API keys. These keys are stored as environment variables and injected into the relevant backend containers at runtime to power resume parsing, job matching, and content generation.
- **Authentication Setup: Firebase Authentication** is configured within the GCP project. Both the frontend and backend are integrated with the Firebase SDK to manage secure user registration, login sessions, and role-based access control (RBAC) for Applicants, HR Representatives, and Administrators.

2. Modules

Career Match is built using a modular architecture to separate concerns, improve maintainability, and allow for independent scaling of its components.

Module	Description
Authentication	Manages secure user login, registration, and session tracking using Firebase. Implements role-based access for Applicants, HR, and Admins.
User Profile Management	Handles the creation, retrieval, and updating of profiles for all user roles. Includes logic for the AI-powered profile population from resumes.
AI Resume Parser	A dedicated backend module that uses the Google Gemini API to process uploaded resume files and extract structured data like experience, skills, and education.
Job Posting Module	Allows HR users to create, edit, view job listings. Features both a manual rich-text editor and an AI-assisted generation option.
Intelligent Job Matcher	The core AI engine that semantically compares candidate profiles against job requirements to calculate and display the most relevant matches.
Application & Content Generation	Enables applicants to submit applications for jobs and leverages the AI engine to generate tailored resumes and cover letters on demand.
Role-Based Dashboards	Provides three distinct frontend views (Applicant, HR, Admin) with tailored metrics, analytics, and task management functionalities for each role.
Database Layer	Utilizes **PostgreSQL** on Cloud SQL for persistent, structured data and **Google Cloud Storage** for storing resume files and other user-uploaded assets.
Frontend (TypeScript)	The complete, component-based user interface that enables all user interactions with the platform's features, including dashboards, forms, and settings.

Testing & Debugging

1. Test Scenarios by Modules

Comprehensive testing was performed across each module of the Career Match platform to ensure functionality, accuracy, and a robust user experience under various conditions. The major test scenarios included:

- **Authentication & User Management**
    - Login/logout functionality with valid and invalid credentials for all three roles (Applicant, HR, Admin).
    - Role-based access control verification to ensure users can only access their designated dashboards and features.
    - Successful user registration and email verification flow.
    - Session timeout and renewal testing.
- **AI Resume Parser Module**
    - File upload integrity and processing for supported formats (PDF, DOC, DOCX).
    - Accuracy of data extraction (experience, skills, education) from resumes with various layouts and structures.
    - Error handling for corrupted, password-protected, or unreadable resume files.
- **User Profile & Preferences Module**
    - Successful creation of profiles via both AI-powered parsing and manual entry.
    - Validation of editing, updating, and saving profile information.
    - Functionality of the job preferences setup (domain, location, salary) and its impact on matching.
- **Job Posting Module (HR)**
    - Functionality of creating, editing, and deleting job postings.

    - Validation of both manual job posting and the AI-assisted "Prompt-Based Job Posting" feature.

- **Intelligent Job Matcher**
    - Verification of matching accuracy based on skills, experience level, location, and qualifications.
    - Testing with profiles that are a perfect match, a partial match, and no match to ensure logical relevance scoring.
    - Ensuring new jobs that match user preferences appear correctly in the "Job Recommendations" section.
- **Application & Content Generation Module**
    - Successful generation of tailored resumes and cover letters for specific job applications.
    - Validation of the automated application email generation and submission process.
    - Verification that all sent applications are correctly logged in the applicant's dashboard.
- **Role-Based Dashboards**
    - Accuracy of all displayed metrics (e.g., total matched jobs, resumes generated).
    - Functionality of all interactive elements, including filters, download buttons, and navigation links.

2. Test Cases by Scenario Use Cases

Scenario	Test Case Description	Expected Outcome
New Applicant Onboarding	A new user registers as an "Applicant" and uploads a valid resume file.	The user account is created, an email verification is sent, and the user's profile is accurately auto-populated with the parsed data from the resume.

Scenario	Test Case Description	Expected Outcome
HR Posts a New Job	An HR user logs in, navigates to the job posting section, and uses the AI-assisted feature to create a new job listing.	The job post is created successfully, appears in the "Job Postings" tab on the HR dashboard, and becomes active for matching against candidates.
Applicant Job Matching	An applicant with a completed profile and defined job preferences logs into the platform.	The applicant dashboard correctly displays a list of relevant job opportunities found by the intelligent matching engine.
Applicant Applies for a Job	An applicant finds a suitable job and uses the automated application feature to apply.	The system generates a tailored resume and cover letter, and upon user approval, sends the application. The job status updates to "Applied" in the dashboard.
HR Reviews Candidate	An HR user receives an email notification for a new application and clicks to view the candidate's profile.	The HR dashboard displays the candidate's full profile, relevance score, and provides options to download the tailored application materials.
Role-Based Access Validation	A logged-in "Applicant" user attempts to directly access a URL for the HR dashboard.	The user is denied access and is redirected to their own applicant dashboard or a "permission denied" page.

3. Known Issues

While the platform is stable and functional, a few minor issues have been identified for future improvement.

Issue	Status	Remarks
Inconsistent Parsing for Non-Standard Resumes	Intermittent	While standard resume formats are parsed with high accuracy, highly stylized resumes with complex tables or multi-column layouts can sometimes result in minor data extraction errors. The AI model is being continuously fine-tuned.

Issue	Status	Remarks
AI-Generated Cover Letter Tone Can Be Generic	Minor	The AI-generated content is consistently professional but can occasionally lack a unique voice for highly creative roles. Future enhancements will include more user-configurable "tone" settings.
Dashboard Metrics Lag During Peak Load	Intermittent	During periods of very high user activity, there can be a slight delay (a few seconds) in the real-time refresh of dashboard statistics. Backend query optimization is planned for a future release.
Slower Initial Matching for Complex Profiles	Minor	The first-time job matching process for new users with exceptionally long and detailed work histories can sometimes take longer than average. Caching strategies are being explored to improve this initial load time.

Product Outcome

1. **Final product screenshots / videos(demo) :**

   Applicant demo **:**
   https://drive.google.com/file/d/1xi6fXnyeJIxGKSoEgpoYTZ9WOBbmkO9S/view?usp=sharing

   HR demo :
   https://drive.google.com/file/d/1CyX6ggTPEWKlulI9ipzpEHV4pjxCHFO5/view?usp=sharing

2. **Product links :**

   Git Hub repo link : https://github.com/Preeti-deora/Team-B/tree/deployment

   Product link : https://frontend-job-1071432896229.asia-south2.run.app/

Challenges Faced

Throughout the development of Career Match, the team navigated several technical, architectural, and user-experience challenges to deliver a robust and intuitive platform.

1. Technical Challenges

- **Resume Parsing Variability:** Resumes lack a standard format, which posed a significant challenge for the AI parser. Handling diverse layouts, custom sections, and inconsistent date formats required extensive prompt engineering and validation logic to ensure accurate data extraction.
- **AI Content Generation Quality:** Ensuring that the AI-generated resumes and cover letters were professional, coherent, and accurately reflected the candidate's unique experience—without sounding generic—required iterative tuning of the AI models and prompts.
- **Google Gemini API Reliability:** The platform's core AI functionalities are dependent on the Google Gemini API. Any latency or downtime from this third-party service could directly impact key user workflows like profile creation and application submission.

Resolution: Developed a robust data validation layer to review parsed data, implemented multi-shot prompting techniques to improve content quality, and added error handling and retry mechanisms for API calls.

2. Scalability

- **Real-time Job Matching Load:** As the number of users and job postings increases, running the intelligent matching algorithm for every new job against thousands of applicant profiles in real-time presents a significant computational load.
- **Concurrent AI API Calls:** A high volume of simultaneous users uploading resumes or generating content could lead to API rate-limiting issues or backend processing bottlenecks.

Resolution: Implemented an optimized job matching process that runs as an asynchronous background task. Introduced a queueing system to manage concurrent requests to the Gemini API, ensuring smooth processing without overwhelming the system.

3. Performance

- **Resume Upload and Parsing Speed:** The time taken to upload a resume, send it to the AI for parsing, and receive the structured data could lead to a noticeable delay in the user onboarding experience.
- **Real-time Dashboard Updates:** Ensuring that new job matches or application status updates appeared on user dashboards instantly required an efficient real-time communication mechanism between the server and the client.

Resolution: Optimized the resume preprocessing logic on the backend. Implemented progressive loading on the frontend and utilized efficient data-fetching patterns to ensure dashboards feel responsive and up-to-date.

4. Integration Challenges

- **GCP & Firebase Credential Management:** Securely managing API keys and service account credentials for Google Cloud Storage, Cloud SQL, and Firebase Authentication across development and production environments was a key security concern.
- **SMTP Integration Reliability:** Ensuring reliable delivery of automated emails for applications and notifications required careful configuration to avoid being flagged as spam and to handle potential delivery failures.

Resolution: Utilized .env files for local development and a secure secret management system within GCP for production credentials. Integrated a robust SMTP service with proper SPF/DKIM records and implemented logging for email delivery status.

5. Solution Architecture

- **Microservice Communication:** With a decoupled architecture, ensuring seamless and reliable communication between the user service, job posting service, and application service required a well-defined and consistent API contract.
- **Data Consistency:** Maintaining state synchronization between a user's master profile, the various AI-generated resume versions, and their application statuses required careful orchestration and a clear data flow strategy.

Resolution: Standardized API schemas using OpenAPI specifications. Implemented a reliable task-tracking system within the PostgreSQL database to manage the state of asynchronous jobs like resume parsing and application submissions.

6. Intuitiveness (Ease of Use Without Support)

- **Onboarding Flow Complexity:** Guiding new users through the multi-step process of registration, resume upload, AI-driven profile verification, and preference setting without causing confusion was a design challenge.
- **Explaining AI-Driven Matches:** Users needed to trust *why* a job was recommended. Visually communicating the connection between their profile and a job description in an intuitive way was difficult.

Resolution: Designed a step-by-step guided onboarding tour with tooltips and progress indicators. The job matching interface was enhanced to highlight the specific skills or experiences that resulted in a match, thereby increasing transparency and user trust.

Future Scope

As Career Match continues to grow, several key areas have been identified for future development to enhance its intelligence, usability, and value proposition for both job seekers and recruiters.

1. Potential Improvements

- **Enable the Admin role and its functionality.**
- **Enhanced job application features tailored for HR-posted listings**
- **Advanced Candidate Analytics:**
    - Integrate predictive analytics for recruiters to offer insights into a candidate's potential cultural fit or likelihood to succeed in a role based on historical hiring data.
- **Expanded Platform Integrations:**
    - Enable one-click profile import from professional networks like **LinkedIn**.

    - Build integrations with major **Applicant Tracking Systems (ATS)** to allow seamless application delivery directly into a company's existing hiring workflow.

- **Automated Interview Scheduling:**
    - Introduce an AI-powered scheduling assistant that can coordinate between the recruiter's and candidate's availability to automate the interview booking process.
- **Direct Communication Tools:**
    - Add in-platform real-time chat and video-calling features to allow recruiters and candidates to communicate directly and securely.
- **Localization and Multi-language Support:**
    - Adapt the user interface and AI content generation capabilities to support multiple languages and cater to different international job markets.

2. Enhancements

- **AI-Powered Career Coach:**
    - Enhance the AI functionality to act as a personal career coach for job seekers, suggesting skills to acquire, certifications to pursue, and potential career paths based on their profile.
- **Gamification of the Applicant Experience:**
    - Introduce gamified elements like points, badges, and profile strength meters to encourage users to fully complete their profiles and actively engage with the platform.
- **In-Depth Company Profiles:**
    - Allow companies to build rich, detailed profiles that showcase their work culture, values, benefits, and team members to attract better-aligned talent.
- **Mobile App Development:**
    - Create dedicated native mobile apps for iOS and Android to provide users with on-the-go access to job alerts, application tracking, and messages.
- **Predictive Job Market Insights:**

- Use aggregated and anonymized platform data to offer both recruiters and job seekers valuable insights into hiring trends, in-demand skills, and salary benchmarks for specific industries and roles.

Conclusion

The development of Career Match has been a transformative journey for our team, blending advanced concepts from artificial intelligence, cloud infrastructure, full-stack development, and user-centric design into a single, cohesive recruitment platform. This project provided us with invaluable hands-on experience in building a complex, AI-driven application from the ground up, designed to solve tangible, real-world challenges in the hiring industry.

Summary of Accomplishments

- **End-to-End Recruitment Platform Built:** We successfully designed, developed, and implemented a comprehensive platform that automates the entire recruitment lifecycle, from AI-powered resume parsing and intelligent job matching to streamlined application management.
- **Role-Based Architecture:** We created a sophisticated architecture with distinct and tailored dashboards and functionalities for three core user roles: Applicants, HR Representatives, and Administrators.
- **Google Gemini API Integration:** We successfully embedded the Google Gemini API as the core intelligence layer, powering resume analysis, semantic job matching, and the automated generation of professional-quality resumes and cover letters.
- **Cloud-Native & Secure:** We utilized Google Cloud Platform (GCP) services for scalable hosting, database management, and file storage. The platform incorporates secure authentication, session management, and role-based access control through Firebase Authentication.
- **Full-Stack Development:** We built a fully responsive and dynamic frontend using **React with TypeScript** and a robust backend based on a

microservices architecture with a **PostgreSQL** database, all containerized with **Docker** for consistent deployment.

- **Automated Workflows & Dashboards:** We implemented end-to-end automated workflows for the job application process and delivered real-time dashboards that provide actionable insights and key metrics for each user role.

What the Team Learned

- **End-to-End System Design:** We gained practical experience in designing and building a production-grade software system from initial concept to final deployment, covering all layers of the technology stack.
- **Applied AI for Business Solutions:** We learned how to effectively integrate a powerful large language model (Google Gemini) into a real-world application to solve concrete business problems, moving from theoretical AI concepts to practical implementation.
- **Agile & Scrum Practices:** We successfully adopted Agile principles and the Scrum framework, learning to work in sprints, manage tasks on project boards, and conduct effective sprint planning, demos, and retrospectives.
- **Modular & Microservice Architecture:** We came to appreciate the critical importance of a decoupled microservices architecture for building a scalable, maintainable, and resilient multi-user platform.
- **Solving Real-World Challenges:** We navigated and resolved complex issues inherent in software development, including third-party API reliability, data parsing inconsistencies, and cloud deployment complexities, learning to debug and document solutions systematically.
- **Collaboration & Communication:** We significantly improved our ability to work as a cohesive unit, coordinating tasks effectively, resolving technical disagreements constructively, and supporting each other throughout the project lifecycle.

This project not only resulted in the delivery of a high-impact product but also provided our team with a practical learning experience that extended far beyond standard coursework. Career Match stands as a testament to our collective vision, technical execution, and collaborative spirit.

References

1. Tools Used

- **Visual Studio Code (VS Code):** Primary code editor used for developing the Python backend, React frontend, and managing project files.
- **Git & GitHub:** For version control, code repository management, and team collaboration.
- **Postman:** Utilized for testing and validating the backend Flask API endpoints, such as /api/generate_job and /api/create_job.
- **Docker:** For containerizing application services to ensure consistent deployment environments.
- **Slack / Discord:** Primary channels for team communication, daily syncs, and technical discussions.

2. Frameworks & Platforms

- **React.js:** The core frontend framework used to build a responsive, component-based user interface.
- **Flask:** A lightweight web framework used to create the backend RESTful APIs that serve job data and interact with the AI model.
- **PostgreSQL:** The primary relational database used for storing all application data, including job postings and user information.
- **Google Cloud Platform (GCP):** The primary cloud provider for hosting the PostgreSQL database, as indicated by the connection URI in the .env file.
- **Firebase:** The platform used for managing user authentication and security tokens, as suggested by the authorization logic in code files.

3. Libraries & Packages

- **Axios:** The JavaScript library used in the React frontend to handle API requests to the backend services.
- **Flask-CORS:** A Flask extension used to handle Cross-Origin Resource Sharing (CORS) for the API.

- **SQLAlchemy:** A Python SQL toolkit and Object-Relational Mapper used for database interactions with PostgreSQL.
- **psycopg2-binary:** The PostgreSQL database adapter for Python, enabling the connection between the Flask application and the database.
- **google-generativeai:** The official Python SDK for interacting with the Google Gemini API to generate job descriptions.
- **python-dotenv:** A Python library used to manage environment variables for API keys and database URIs from a .env file.
- **Lucide React:** The icon library used throughout the frontend to provide clean and intuitive visual elements.
- **React Router:** The standard library for handling routing and navigation within the React single-page application.

4. Research Papers / Technical References

- **Google Gemini API Documentation:** Official documentation for the generative AI model used to create the job post details.
- **Flask Documentation:** Official documentation for the Flask web framework.
- **React Official Docs:** The official documentation and guides for the React.js library.
- **SQLAlchemy Documentation:** The official documentation for the SQLAlchemy library.
- **PostgreSQL Documentation:** The official documentation for the PostgreSQL database system.
- **MDN Web Docs:** A key resource for web standards and best practices for HTML, CSS, and JavaScript/TypeScript.