A PROJECT REPORT ON

SkillsBridge – An AI- Powered Job Readiness Platform

SUBMITTED TOWARDS THE

PARTIAL FULFILLMENT OF THE REQUIREMENTS OF

BACHELOR OF ENGINEERING (Computer Engineering)
BY

Omkar Ghodekar	Roll No: BECO2526B067
Utkarsh Salunke	Roll No: BECO2526B048
Raj Wagh	Roll No: BECO2526B008
Sandhya Dhaygude	Roll No: BECO2526B046

Under The Guidance of

Dr. Mrs. Madhuri A. Potey

DEPARTMENT OF COMPUTER ENGINEERING

D. Y. Patil College of Engineering, Akurdi, Pune-44

SAVITRIBAI PHULE PUNE UNIVERSITY

2025-26

D.Y. Patil College of Engineering, Akurdi, Pune-44

DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE

This is to certify that the Project Entitled

“SkillsBridge – An AI- Powered Job Readiness Platform”

Submitted by

Omkar Ghodekar	Roll No: BECO2526B067
Utkarsh Salunke	Roll No: BECO2526B048
Raj Wagh	Roll No: BECO2526B008
Sandhya Dhaygude	Roll No: BECO2526B046

is a bona-fide work carried out by students under the supervision of Dr. Mrs. Madhuri A. Potey and it is submitted towards the partial fulfillment of the requirement of Bachelor of Engineering (Computer Engineering).

Dr. Mrs. Madhuri A. Potey
H.O.D & Internal Guide
Dept. of Computer Engg.
.

Dr. Mrs. P. Malathi
Principal
DYPCOE, Akurdi, Pune 44

Signature of Internal Examiner

Signature of External Examiner

PROJECT APPROVAL SHEET

A Project on

SkillsBridge – An AI- Powered Job Readiness Platform

Is successfully completed by

Omkar Ghodekar	Roll No : BECO2526B067
Utkarsh Salunke	Roll No : BECO2526B048
Raj Wagh	Roll No : BECO2526B008
Sandhya Dhaygude	Roll No : BECO2526B046

DEPARTMENT OF COMPUTER ENGINEERING

D. Y. Patil College of Engineering, Akurdi, Pune

SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE

2024-2025

Dr. Mrs. Madhuri A. Potey
H.O.D & Internal Guide
Dept. of Computer Engg.

Abstract

The rapid evolution of industry demands has widened the gap between academic learning and employer expectations, leading to challenges for students and fresh graduates in securing suitable job opportunities. To address this issue, SkillsBridge is proposed as an AI-powered job readiness and interview preparation platform that provides end-to-end career support. The system extracts technical and soft skills from user resumes using NLP-based approaches such as SBERT and TF-IDF, and maps these skills against real-time job descriptions sourced through automated web scraping. Based on relevance ranking, users receive personalized job recommendations.

SkillsBridge further integrates an AI-driven mock interview module using the Ollama–Mistral model to generate domain-specific interview questions, while the Gemini model evaluates user responses based on accuracy, communication, and confidence. The platform generates a structured and personalized learning path to help users improve skill gaps identified during evaluation. By automating resume analysis, job-role matching, interview assessment, and upskilling recommendations, SkillsBridge aims to create a comprehensive, scalable, and intelligent ecosystem that enhances job readiness and supports candidates in achieving successful career outcomes.

Acknowledgment

*It gives us great pleasure in presenting the preliminary project report on '**SkillsBridge – An AI-Powered Job Readiness Platform**'*

I would like to take this opportunity to thank my internal guide ***Dr. Mrs. Madhuri A. Potey*** *for giving me all the help and guidance I needed. I am really grateful to them for their kind support. Their valuable suggestions were very helpful.*

I am also grateful to ***Dr. Mrs. M. A. Potey****, Head of Computer Engineering Department D. Y. Patil College of Engineering, Akurdi, Pune for her indispensable support, suggestions.*

In the end our special thanks to ***Mrs. Supriya Sathe*** *for providing various resources such as laboratory with all needed software platforms, continuous Internet connection, for our project.*

Omkar Ghodekar
Utkarsh Salunke
Raj Wagh
Sandhya Dhaygude
(B.E. Computer Engg.)

Contents

1. Synopsis 1

1.1 **Project Title 1**

1.2 **Project Option 1**

1.3 **Internal Guide 2**

1.4 **Sponsorship and External Guide 2**

1.5 **Technical Keywords (As per ACM Keywords) 2**

1.6 **Problem Statement 2**

1.7 **Abstract 2**

1.8 **Goals and Objectives 3**

1.9 **Relevant mathematics associated with the Project 3**

1.10 **Names of Conferences / Journals where papers can be published 5**

1.11 **Review of Conference/Journal Papers supporting Project idea 5**

1.12 **Plan of Project Execution 6**

2. Technical Keywords 8

2.1 **Area of Project 8**

2.2 **Technical Keywords 8**

3. Introduction 9

3.1 **Project Idea 9**

3.2 **Motivation of the Project 9**

3.3 **Literature Survey 10**

4. Problem Definition and scope 12

4.1 **Problem Statement 12**

4.1.1 Goals and objectives 12

4.1.2 Statement of scope 13

4.2 **Major Constraints 13**

4.3 **Methodologies of Problem solving and efficiency issues 14**

4.4 **Outcome 15**

4.5 **Applications 15**

4.6 **Hardware Resources Required 16**

4.7 **Software Resources Required 17**

5. Project Plan **19**

5.1 **Project Estimates** **19**

5.1.1 Reconciled Estimates 19

5.1.2 Project Resources 20

5.2 **Risk Management w.r.t. NP Hard analysis** **21**

5.2.1 Risk Identification 21

5.2.2 Risk Analysis 22

5.2.3 Overview of Risk Mitigation, Monitoring, Management 24

5.3 **Project Schedule** **24**

5.3.1 Project task set 28

5.3.2 Task network 31

5.3.3 Timeline Chart 32

5.4 **Team Organization** **32**

5.4.1 Team structure 32

5.4.2 Management reporting and communication 32

6. Software requirement specification **33**

6.1 **Introduction** **33**

6.1.1 Purpose and Scope of Document 34

6.1.2 Overview of responsibilities of Developer 34

6.2 **Usage Scenario** **35**

6.2.1 User profiles 35

6.2.2 Use-cases 36

6.2.3 Use Case View 37

6.3 **Data Model and Description** **38**

6.3.1 Data Description 38

6.3.2 Data objects and Relationships 39

6.4 **Functional Model and Description** **40**

6.4.1 Data Flow Diagram 40

6.4.2 Activity Diagram: 41

6.4.3 Non Functional Requirements: 42

6.4.4 State Diagram: 43

7. Detailed Design Document using Appendix A and B **44**

7.1 **Introduction** **44**

7.2 **Architectural Design** **42**

7.3 **Data design (using Appendices A and B)** **42**

7.3.1 Internal software data structure 42

7.3.2 Global data structure 43

7.3.3 Temporary data structure 43

7.3.4 Database description 43

7.4 **Component Design 44**

7.4.1 Component Diagram 44

8. Project Implementation 45

8.1 **Introduction 45**

8.2 **Tools and Technologies Used 45**

8.3 **Methodologies/Algorithm Details 46**

8.3.1 Algorithm 1/Pseudo Code 46

8.3.2 Algorithm 2/Pseudo Code 46

8.4 **Verification and Validation for Acceptance 47**

9. Conclusion and future scope 50

10. Software Testing

10.1 **Type of Test Cases Used ....**

10.2 **Test Cases ....**

11. Bibliography 53

List of Figures

Figure No	Figure Name	Page No.
1.	5.1 Task Network	31
2.	5.2 Timeline Chart	32
3.	6.1 Use Case Diagram	37
4.	6.2 Level 0 Data Flow Diagram	40
5.	6.3 Level 1 Data Flow Diagram	41
6.	6.4 Activity Diagram	42
7.	6.5 State Diagram	43
8.	8.1 Architecture Diagram	46
9.	8.2 Class Diagram	48
10.	Plagiarism Report	73

List of Tables

Table No	Table Name	Page No.
1.	1.1 Plan of Project Execution	14
2.	3.1 Literature Survey	17- 20
3.	4.1 Hardware Resources Required	25
4.	5.1 Risk Table	31
5.	5.2 Risk Management Plan	32
6.	5.3 Risk Probability Definitions	34
7.	5.3 Risk A	35
8.	5.4 Risk B	35
9.	5.5 Risk C	36
10.	5.6 Task Assigned	36
11.	5.7 Team Structure	41
12.	6.1 Use Cases	45
13.	9.1 Test Cases and Expected Output	60
14.	IDEA Matrix	67

List of Tables

1 Synopsis

1.1 Project Title

SkillsBridge – An AI-Powered Job Readiness Platform.

1.2 Project Options

1.2.1 Project Title: Car Price Prediction Using Machine Learning

1.2.1.1 Type: Self-Sponsored.

1.2.1.2 Description: This project proposes the development of a predictive system that estimates the selling price of used cars based on features such as brand, model, mileage, engine capacity, and age. Machine learning algorithms like Linear Regression, Random Forest, and Gradient Boosting would be used to build an accurate price prediction model. The system can assist buyers and sellers in making informed decisions by providing a data-driven estimation.

1.2.1.3 Key Technologies: Python, Machine Learning, Scikit-Learn, Pandas, NumPy, Matplotlib.

1.2.2 Project Title: Heart Disease Prediction Using Machine Learning

1.2.2.1 Type: Self-Sponsored.

1.2.2.2 Description : This project aims to create a machine learning–based model that predicts the likelihood of heart disease using clinical parameters such as age, blood pressure, cholesterol levels, heart rate, and ECG patterns. The system would utilize classification algorithms to analyze patient data and identify potential health risks. It supports early detection and can assist healthcare providers in preventive diagnosis.

1.2.2.3 Key Technologies :Python, Machine Learning, Logistic Regression,Random Forest, Data Preprocessing, Healthcare Datasets.

1.3 Internal Guide

Internal Guide Name – Dr. Mrs. Madhuri A. Potey

1.4 Sponsorship and External Guide

Not Applicable

1.5 Technical Keywords (As per ACM Keywords)

- Artificial Intelligence
- Natural Language Processing (NLP)
- Machine Learning
- Resume Parsing and Skill Extraction
- Semantic Similarity Scoring
- Skill Gap Detection
- Mock Interview Simulation
- Career Recommendation System
- User Profiling and Personalization
- AI-Based Evaluation and Feedback

1.6 Problem Statement

Students and job seekers often lack clarity about required industry skills, suitable job roles, and effective interview preparation. Traditional methods provide limited personalization, leading to poor job readiness. There is a need for an AI-driven platform that analyzes skills, matches relevant jobs, conducts mock interviews, and offers targeted improvement guidance.

1.7 Abstract

The rapid evolution of industry demands has widened the gap between academic learning and employer expectations, leading to challenges for students and fresh graduates in securing suitable job opportunities. To address this issue, SkillsBridge is proposed as an AI-powered job readiness and interview preparation platform that provides end-to-end career support. The system extracts technical and soft skills from user resumes using NLP-based approaches such as SBERT and TF-IDF, and maps these skills against real-time job descriptions sourced through automated web scraping. Based on relevance ranking, users receive personalized job recommendations.

SkillsBridge further integrates an AI-driven mock interview module using the Ollama–Mistral model to generate domain-specific interview questions, while the Gemini model evaluates user responses based on

accuracy, communication, and confidence. The platform generates a structured and personalized learning path to help users improve skill gaps identified during evaluation. By automating resume analysis, job-role matching, interview assessment, and upskilling recommendations, SkillsBridge aims to create a comprehensive, scalable, and intelligent ecosystem that enhances job readiness and supports candidates in achieving successful career outcomes.

1.8 Goals and Objectives

1. **Research AI-Driven Job Readiness Techniques:**
   Explore existing technologies used for resume analysis, job-role mapping, and AI-powered interview simulation. Study current NLP and machine learning methods used for skill extraction, semantic similarity, and candidate evaluation.
2. **Develop and Test an Intelligent Job Readiness System:**
   Design and implement an AI-based system capable of extracting skills, matching job descriptions, generating mock interviews, and evaluating user responses. Continuously refine system performance using accuracy metrics, relevance scores, and user feedback.
3. **Evaluate Ethical, Fairness, and User-Centric Concerns:**
   Assess fairness, transparency, and reliability of AI-generated evaluations. Address concerns related to bias, data privacy, secure handling of resumes, and ensuring that recommendations and interview assessments remain objective and user-friendly.

1.9 Relevant Mathematics Associated with the Project

1. **Vector Space Modeling & Similarity Measures :**
   User skills and job descriptions are represented as high-dimensional vectors. Similarity between these vectors is calculated to determine job relevance using methods such as
   - **Cosine Similarity:**

$$\text{Cosine Similarity} = \frac{V_s \cdot V_j}{\parallel V_s \parallel \parallel V_j \parallel}$$

- **Euclidean Distance:**

$$\mathrm{d(Vs\,,Vj)} \;=\; \sqrt{\sum_{i=1}^{n}(\mathrm{Vsi} \;-\mathrm{Vji})2}$$

2. **TF-IDF Weighting for Keyword Importance:**

Resume and job description text is processed using Term Frequency–Inverse Document Frequency (TF-IDF) to quantify keyword importance:

$$\text{TF-IDF}(t,d) = TF(t,d) \times \log\left(\frac{N}{DF(t)}\right)$$

3. **Semantic Similarity through Sentence Embeddings:**

SBERT encodes sentences into dense vector representations. Mathematical operations such as dot-product and mean pooling are used to generate meaningful embeddings representing candidate profiles and job descriptions.

4. **Skill Gap Analysis using Set Theory :**

Skill-gap computation is performed using set operations:

$$\text{Gap} = \boldsymbol{J}_{\textbf{skills}} - \boldsymbol{U}_{\textbf{skills}}$$

Where:

- J_{skills}= Required skills for the job
- U_{skills}= Extracted skills from the resume

This identifies the missing skills that must be recommended in the learning path.

5. **Scoring and Normalization Techniques :**

Performance metrics derived from evaluation (accuracy, confidence, communication clarity) are normalized for consistency:

- Min-Max Scaling:

$$\boldsymbol{x'} = \frac{\boldsymbol{x} - \boldsymbol{x}_{\mathbf{min}}}{\boldsymbol{x}_{\mathbf{max}} - \boldsymbol{x}_{\mathbf{min}}}$$

6. **Ranking Algorithms for Job Prioritization:**

Jobs are sorted using a weighted relevance function:

$$\boldsymbol{R = w_1 S_{match} + w_2 E_{exp} + w_3 K_{alignment}}$$

7. **Interview Evaluation Metrics :**

   Gemini-generated evaluation uses statistical measures such as:

   - Mean and variance of response quality
   - Confidence score mapping
   - Weighted grading functions for communication and accuracy

1.10 Names of Conferences / Journals Where Papers Can Be Published

Conferences

1. **IEEE International Conference on Machine Learning and Applications (ICMLA))**

   Suitable for research involving machine learning-driven job role matching, skill extraction, and employment analytics.

2. **ACM International Conference on Intelligent User Interfaces (IUI)**

   Relevant due to the platform's AI-driven interview interface, personalized recommendations, and interactive learning environment..

3. **IEEE International Conference on Big Data (BigData)**

   Well-suited for SkillsBridge's use of large-scale job datasets, resume corpora, and data-driven relevance ranking techniques.

4. **International Conference on Natural Language Processing (ICON)**

   Pertinent for contributions related to NLP models such as SBERT, TF-IDF-based keyword extraction, and semantic matching.

5. **International Conference on Artificial Intelligence and Education (ICAIED)**

   Ideal for work focusing on AI-powered educational pathways, upskilling analytics, and personalized learning systems.

Journals

1. **IEEE Transactions on Learning Technologies (TLT)**

   A strong fit for research on AI-based learning path generation, automated evaluations, and adaptive skill development.

2. **Journal of Artificial Intelligence Research (JAIR)**

   Suitable for technical publications involving multi-model AI integration, semantic similarity modeling, and intelligent recommendation systems.

3. **ACM Transactions on Intelligent Systems and Technology (TIST)**
   Relevant for advanced AI algorithms, job-role matching techniques, and AI-driven interview simulation methodologies.
4. **International Journal of Human–Computer Interaction (IJHCI)**
   An appropriate venue for studies on user experience, AI avatar interviews, and interaction-based performance evaluation.
5. **Elsevier Expert Systems With Applications**
   Suitable for practical applications of AI in job-readiness systems, decision support, and automated career guidance tools.

1.11 Plan Of Execution

Table 1.1: Plan of Project Execution

Task	**Start Week**	**End Week**	**Duration**
Project Planning	Week 1	Week 4	4 Weeks
UI/UX Design	Week 20	Week 24	4 Weeks
Backend Development	Week 28	Week 32	4 Weeks
Frontend Development	Week 22	Week 26	4 Weeks
Algorithm Optimization	Week 18	Week 24	6 Weeks
Machine Learning Model Development	Week 16	Week 20	4 Weeks
Testing and Quality Assurance	Week 28	Week 32	4 Weeks
User Feedback and Iteration	Week 30	Week 34	4 Weeks

Task	Start Week	End Week	Duration
Final Deployment and Monitoring	Week 32	Week 36	4 Weeks
Documentation and Reporting	Week 36	Week 40	4 Weeks

2 Technical Keywords

2.1 Area of Project

- AI in Education and Employability
- Cognitive Computing and Human Skill Evaluation
- Recommender Systems and Intelligent Tutoring
- Natural Language Understanding for Resume and Job Data
- Generative AI for Interactive Assessments

2.2 Technical Keywords

1. Artificial Intelligence (AI)
2. Natural Language Processing (NLP)
3. Machine Learning (ML)
4. Semantic Similarity Analysis
5. TF-IDF Weighting
6. Skill Gap Detection
7. Mock Interview Simulation
8. AI-Based Evaluation and Feedback
9. Career Recommendation System
10. User Profiling and Personal

3 Introduction

3.1 Project Idea

The proposed project, **SkillsBridge**, is an AI-powered job readiness platform designed to assist students and job seekers in identifying suitable roles, improving required skills, and preparing effectively for interviews. The system extracts skills from resumes using NLP techniques and matches them with real-time job descriptions to provide relevant opportunities. An AI-driven mock interview module generates job-specific questions and evaluates user responses to measure technical accuracy, communication skills, and confidence. Based on this assessment, the system creates a personalized learning path to address skill gaps. The platform aims to offer a structured, intelligent, and user-centric solution that enhances employability by enabling users to understand industry expectations, improve performance, and become job-ready.

3.2 Motivation of the Project

The motivation for this project arises from the appearance of some difficulties and issues in achieving fairness and integrity in assessment tools used in online coding assessments. Automated testing in digital learning platforms and recruitment processes has accelerated the need for such situations, which also permits cheating or manipulation. Traditional methods of identifying cheating, particularly code review manually, are very time- consuming and not effective with large-scale assessments. The presence of sophisticated code plagiarism and manipulation tools calls for a reliable system that can be automatic in ascertaining the fact of validity and fairness in any coding assessment that has been made. This project, therefore, seeks to give greater prestige to online assessments while following fair judgments that may eventually boost the dependence of more people on digital education and hiring processes as well.

3.1.3 Literature Survey

Table 3.1: Literature Survey

Sr. No	Title and Author	Conference / Journal Name & Publication Year	Topic Reviewed/ Algorithms or Methodology used	Advantages and Disadvantages
1	Guan, Zhihao, et al. "JobFormer: skill-aware job recommendation with semantic-enhanced transformer."	ACM Transactions on Knowledge Discovery from Data 19.1 (2024): 1-20.	AI-powered platform for learning and placement using personalized learning pathways and skill tracking.	**Advantages**: Personalized learning, integrated placement support. **Disadvantages**: High implementation complexity, dependent on accurate profiling.
2	AI knows you: Prediction of extroversion trait - Naz, A. et al.	IEEE Access, 2024	Deep learning model (CNN) for personality trait prediction using social media/textual data.	**Advantages**: Predicts personality traits effectively. **Disadvantages**: Privacy concerns, needs large labeled data.
3	Contribution of Job Readiness Application - Nurmalasari, R.	2021 7th ICEEIE, IEEE	Survey-based analysis using regression on job readiness factors and competency mapping.	**Advantages**: Provides empirical insight into education-employment gap. **Disadvantages**: Limited to certain demographics or regions.
4	Prediction of Student Job Readiness - Gemilang, H. et al.	2024 ICoDSA, IEEE	MLP and XGBoost to predict job readiness from student performance metrics.	**Advantages**: High prediction accuracy with ensemble models. **Disadvantages**: Data quality sensitive, model interpretability low.
5	Enhancing Career Readiness - Jiranantanagorn, P.	2023 CSDE, IEEE	Online platform for competency assessment and matching with workplace preferences.	**Advantages**: Facilitates self-assessment and employer matching. **Disadvantages**: Requires regular platform updates and employer participation.
6	Collaborative Mock Interview Platform - Ahmad, M.M. et al.	2024 OTCON, IEEE	Web-based mock interview with collaborative tools, AI feedback.	**Advantages**: Real-time collaboration, useful for preparation. **Disadvantages**: Quality dependent on AI feedback system.
7	Virtual Job Interview Simulator - Rao, G.S. et al.	2025 ICOEI, IEEE	NLP-based virtual interviewer using real-time text and audio processing.	**Advantages**: Simulates real interviews, improves confidence. **Disadvantages**: NLP limitations may affect accuracy.

Sr. No	Title and Author	Conference / Journal Name & Publication Year	Topic Reviewed/ Algorithms or Methodology used	Advantages and Disadvantages
8	Intelligent Resume Recommendation - Mishra, A. et al.	2024 ICCCNT, IEEE	NLP-based resume parser and recommender with feedback system.	**Advantages**: Provides personalized resume improvements. **Disadvantages**: May overlook creative/unique formats.
9	AI-Powered Mock Interview Platform - Sharma, T. et al.	2025 ICSSAS, IEEE	Combines Computer Vision, NLP, and GenAI to simulate interviews and evaluate responses.	**Advantages**: Multi-modal analysis of interview performance. **Disadvantages**: High resource consumption.
10	AI-Powered Resume Builder - Jha, R. et al.	2025 ICDT, IEEE	AI-based resume template recommender and content enhancer.	**Advantages**: Boosts application quality with intelligent suggestions. **Disadvantages**: May generate generic suggestions.

4 Problem Definition and Scope

4.1 Problem Statement

Students and job seekers often lack clarity about required industry skills, suitable job roles, and effective interview preparation. Traditional methods provide limited personalization, leading to poor job readiness. There is a need for an AI-driven platform that analyzes skills, matches relevant jobs, conducts mock interviews, and offers targeted improvement guidance.

4.1.1 Goals and objectives

4.1.1.1 Develop an AI-Driven Resume Analysis System:

Create a module that extracts technical and soft skills from resumes using NLP techniques such as SBERT and TF-IDF.

4.1.1.2 Implement Intelligent Job Role Matching :

Design a matching system that compares user skills with real-time job descriptions and ranks roles using semantic similarity algorithms

4.1.1.3 Provide AI- Powered Mock Interview Simulation :

Generate domain-specific interview questions and conduct interactive mock interviews using the Ollama–Mistral model

4.1.1.4 Evaluate User Performance Objectively :

Use Gemini-based evaluation to assess accuracy, communication clarity, and confidence, followed by detailed feedback.

4.1.1.5 Generate Personalized Learning Paths :

Identify skill gaps and recommend targeted upskilling through structured, AI-generated learning roadmaps.

Input and Output

Input: User resume, job description selection, and interview responses.

Output: Extracted skills, matched job roles, interview evaluation results, and a personalized learning path indicating required improvements.

4.1.2 Statement of Scope

Scope of the Software:

Input Size: User resumes, job descriptions, and interview responses of varying lengths and formats, ranging from short single-page resumes to detailed multi-page profiles and diverse interview inputs.

Constraints on Input: Inputs must be in acceptable formats such as PDF, DOCX, or plain text for resumes, and standard text formats for interview responses. Job descriptions must follow structured or semi-structured text formats to enable accurate NLP-based extraction.

Input Validation: Code inputs are validated for proper format, syntax, and adherence to project guidelines before proceeding to plagiarism and originality checks.

Input Dependency: The effectiveness of the plagiarism detection system depends heavily on the code quality and compliance with coding standards. Poorly written or obfuscated code may require additional processing to ensure accurate detection.

I/O State Diagram: Intake of code submissions, processing through originality and plagiarism detection algorithms, and outputting a classification result ("Fair" or "Cheating") with associated scores.

4.2 Major Constraints

Input Quality: The accuracy of skill extraction and job-role matching depends heavily on the clarity, structure, and formatting of user resumes. Resumes with inconsistent layouts, missing keywords, or scanned/low-quality files may reduce NLP performance and lead to incomplete skill detection.

Dataset Limitations: Job matching and relevance scoring rely on the quality and diversity of job descriptions obtained through web scraping. Limited or outdated datasets may affect the accuracy of recommendations and reduce the system’s ability to align user skills with current industry requirements.

Real-Time Processing Constraints: Generating mock interview questions and evaluating responses using large language models such as Mistral and Gemini may introduce latency. Real-time interaction with AI interviewers requires significant computational resources and may slow down during high usage periods.

Continuous Model Updates: The system requires frequent updates to incorporate new skills, emerging job roles, and evolving industry trends. Without continuous improvements, the recommendation accuracy and interview simulation quality may decline over time.

Scalability Issues: As the number of users increases, the system must handle large volumes of resumes, job descriptions, and interview sessions. This growth may introduce computational overhead, increased storage requirements, and additional load on NLP and AI models.

Ethical and Privacy Considerations: Processing personal resumes and interview responses raises concerns regarding data privacy and user consent. AI-based evaluations must avoid bias and ensure fairness so that no user is disadvantaged due to demographic, linguistic, or stylistic differences.

4.3 Methodologies of Problem Solving and Efficiency Issues

1. Problem Solving Methodologies:

- **NLP-Based Skill Extraction**:

  The system uses advanced Natural Language Processing techniques such as SBERT embeddings and TF-IDF weighting to extract technical and soft skills from user resumes. These methods help identify key competencies, keywords, and experience patterns essential for accurate job-role mapping.

- **Semantic Similarity and Job Matching:**

  Resume vectors are compared with job description vectors using cosine similarity and relevance ranking algorithms. This ensures that recommended job roles closely align with the user's existing skill set and desired career path.

- **AI-Powered Interview Simulation:**

  Mock interviews are generated using the Ollama–Mistral language model, which creates domain-specific questions and simulates real interview scenarios. User responses are evaluated by Gemini to assess accuracy, clarity, and communication effectiveness.

- **Personalized Learning Path Generation:**

  Identified skill gaps are processed through rule-based and ML-driven recommendations to create tailored learning paths. These paths guide users toward the exact topics and skills needed for improvement.

2. Efficiency Issues:

High Computational Load for NLP Models:

Skill extraction, semantic similarity computation, and AI-based interview generation require significant processing power. Running large models like Mistral and Gemini may cause delays during peak usage.

Handling Large Job Datasets:

Processing and ranking a high volume of job descriptions can increase computation time, especially when dealing with multiple scraped sources such as LinkedIn, Naukri, and Internshala.

Real-Time Interaction Constraints:

Providing real-time AI interview responses, instant evaluation, and immediate feedback may be challenging due to response generation time, model load, and network latency.

Adaptability to Evolving Job Market Trends:

As job roles, skill requirements, and hiring standards evolve, the system must continuously update its datasets and models. Without frequent updates, the accuracy of job matching and learning recommendations may decline.

Dependence on Resume Quality:

Poorly structured or incomplete resumes may reduce the effectiveness of skill extraction algorithms, requiring additional preprocessing steps that increase processing time.

4.4 Outcome

The system will offer an AI-driven platform designed to boost job readiness by streamlining resume reviews, identifying key skills, and accurately matching users to suitable job roles. It will also simulate realistic mock interviews and provide clear, unbiased feedback powered by advanced AI models. With personalized learning paths, users will receive focused guidance to strengthen areas where they need improvement. Overall, this solution aims to make career preparation more fair, efficient, and dependable helping students and job seekers gain confidence, meet industry standards, and increase their chances of employment.

4.5 Applications

Educational Institutions:

Colleges and universities can use the platform to help students strengthen their job readiness through automated resume reviews, skill-gap identification, and guided interview practice.

Placement Cells:

Placement teams can assess how prepared students are, suggest suitable job roles, and monitor their progress with AI-generated interview insights and readiness scores.

Corporate Recruitment:

Companies can rely on the system to evaluate candidate skills, run initial AI-powered interviews, and shortlist applicants based on how well they fit the job requirements.

Career Development Centers:

The platform offers personalized learning paths and tailored skill recommendations, allowing career counselors to support students and job seekers more effectively.

Online Training & Upskilling Platforms:

It can integrate with existing e-learning systems to deliver targeted training modules based on identified skill gaps, boosting learner engagement and performance.

Job Portals:

Job platforms can enhance their services by adding smart job-role matching, automated resume screening, and AI-driven interview assessments

4.6 Hardware Resources Required

Table 4.1: Hardware Resources Required

Sr. No.	Parameter	Minimum Requirement	Justification
1	CPU Speed	8-core (3.0 GHz or higher)	A multi-core CPU is needed for the horsepower powering the intense computations, especially when doing model inferences and token generation. More cores and higher speeds improve total performance and response time.
2	TPU	Google Colab - 15 GB	A latest TPU is required for parallel processing and manipulation of deep learning operations. Increased Tensor cores boost model execution speed.
3	RAM	8 GB	At least 8 GB RAM is required to accommodate processes and system operations. It helps handle large datasets, run parallel tasks, and avoid memory bottlenecks.

Sr. No.	Parameter	Minimum Requirement	Justification
4	Storage	512 GB SSD (NVMe preferred)	SSD ensures faster read/write of large model weights, datasets, and logs compared to traditional storage.

4.7

Software Resources Required

1. **Operating System: Windows 10+**

Windows 10 or newer is recommended as the preferred operating system because it works well with most modern development tools, frameworks, and libraries. It provides a stable and efficient environment for running Python-based machine learning models, along with web applications built using technologies like Node.js, ReactJS, and MongoDB.

2. **IDE: VS Code / PyCharm**

- Visual Studio Code (VS Code) is recommended as the primary Integrated Development Environment because it offers strong support for Python, Node.js, and ReactJS. Its built-in debugging tools, Git integration, and large extension marketplace make it a great fit for machine learning, NLP tasks, and full-stack web development.
- PyCharm can be used as an alternative for Python-focused work. It provides powerful features like smart code suggestions, environment and package management, and advanced debugging options, making it well-suited for developing and experimenting with machine learning models.
- Jupyter Notebooks are ideal for interactive Python development, offering an easy way to prototype, test, and refine machine learning workflows through a clear, step-by-step and visual interface.

3. **Programming Languages & Frameworks:**

- Node.js:

  Node.js is used as the backend runtime environment to manage server-side operations, handle user authentication, process job-matching requests, and facilitate communication between the frontend and AI modules. It supports the creation of RESTful APIs and enables efficient handling of asynchronous tasks such as resume parsing, job scraping, and interview scheduling.

- ReactJS:

  ReactJS is utilized for the front-end interface, allowing users to upload resumes, browse matched job roles, attempt mock interviews, and access personalized learning paths. Its component-based structure enables the development of an interactive, dynamic, and responsive user experience essential for a career-readiness platform.

- MongoDB:

  MongoDB is used as the main database for managing user details, extracted skills, job descriptions, interview responses, evaluation reports, and progress data. Its flexible, document-based structure makes it ideal for storing unstructured NLP outputs, large collections of job-related information, and user records that change frequently.

5 Project Plan

5.1 Project Estimates

Reconciled estimates are the final, polished numbers that come together after the team talks through options, reviews resources, and thinks about possible risks. They give a realistic and well-balanced picture of the project's overall cost and timeline.

5.1.1 Reconciled Estimates

Reconciled estimates represent the finalized, polished numbers created after the team reviews feedback, available resources, and any possible risks. These estimates provide a practical and well-rounded picture of the project's total cost and timeline. The cost estimate includes the following components:

5.1.1.1 Cost Estimates

- Cloud Services:

  Most of the platform's operations rely on free-tier services like Firebase and third-party APIs for financial news or external data. At this stage, these services do not generate any direct costs.

- Training/Fine-Tuning AI Models:

  Some expenses are expected for training and fine-tuning machine learning and NLP models that power features such as resume analysis, job matching, and interview evaluation. These tasks may require GPU-based computing resources.

- Hosting Costs:

  After the models are trained, they need to be deployed on dependable cloud services to keep the system scalable and accessible. This may involve minimal hosting expenses.

- Total Cost Estimate:

  The total estimated cost includes cloud usage, model training, and hosting. Because free tiers are used whenever possible, the overall project cost stays relatively low.

5.1.1.2 Time Estimates

The project is organized into clear phases, and each phase has an estimated timeline based on its complexity and workload. Estimated project duration: 8–9 months.

- Project Planning and Research: 2 months.
- Development: 4-5 months.
- Testing and Quality Assurance: 1 month.
- Deployment: 1 month.
- Feedback and Iteration: 1 month.

Total Time Estimate: 8-9 months.

5.1.2 Project Resources

The project relies on a balanced mix of people, hardware, and software to ensure smooth development and deployment. A four-member team handles all major responsibilities.

1. People

- **Project Manager**: Manages timelines, keeps the project on track, and coordinates across teams.
- **Designer:** Builds UI/UX layouts and visual designs for all modules.
- **Software Developer**: Works on both the frontend and backend, designs the system architecture, integrates APIs, and manages database operations.
- **Tester**: Checks accuracy, performance, and the overall user experience.

2. Hardware

- **CPU**: 8-core (3.0 GHz or higher)
- **RAM**: 16 GB to manage large data processing.
- **Storage**: 512 GB SSD (provides fast read/write speeds and smooth model loading)

3. Software

- **Operating System**: Windows 10 or newer
- **IDE**: VS Code
- **Programming Language**: NodeJS, NextJS, MongoDB, Python 3.9 or higher, Ollama

5.2 Risk Management (Based on NP-Hard Analysis)

5.2.1 Risk Identification

Risk identification for the SkillsBridge platform focuses on understanding the factors that could impact the system's performance, timeline, or quality. By reviewing the project goals, technical design, AI model dependencies, and stakeholder input, several technical, managerial, and operational risks were identified. The following key questions were examined to identify potential risks:

1. Managerial Commitment
   Question: Are senior managers, mentors, and project stakeholders committed to supporting the AI-driven development process?
   Risk: Weak commitment may slow decision-making, delay approvals, or limit access to important resources such as APIs, model infrastructure, and scraping tools.

2. User Involvement
   Question: Are end-users (students, job seekers, and placement officers) actively providing feedback?
   Risk: Low user involvement may lead to inaccurate skill suggestions, irrelevant interview questions, or low overall adoption of the platform.

3. Requirements Clarity
   Question: Do developers and stakeholders clearly understand the entire workflow—from resume parsing and job matching to mock interviews and personalized learning paths?
   Risk: Misunderstood requirements may lead to incorrect model integration, missing features, or rework in NLP pipelines and evaluation modules.

4. Customer Participation in Requirement Gathering
   Question: Are stakeholders actively helping define AI model behavior, job scraping sources, and evaluation criteria?
   Risk: Poor participation can result in incomplete job-matching rules, unclear success metrics, and low-quality learning paths.

5. User Expectations
   Question: Are user expectations realistic regarding AI capabilities like skill extraction accuracy, job matching reliability, and avatar-based interviews?
   Risk: Unrealistic expectations may cause dissatisfaction if the AI does not perform perfectly, even when the platform is functioning as intended.

6. Team Skillset
   Question: Does the team have expertise in NLP, AI model integration (Ollama, Mistral, Gemini), web scraping, and full-stack development?
   Risk: Skill gaps could slow progress, produce inaccurate model outputs, or reduce the system's stability and performance.

7. Project Stability
   Question: Are core requirements—AI models, scraping sources, and evaluation criteria—stable throughout development?
   Risk: Frequent changes, especially involving model updates or API adjustments, may disrupt progress and create inconsistencies.

8. Team Size
   Question: Is the team size sufficient to handle frontend (Next.js), backend (Python), AI components, job scraping, and testing?
   Risk: A small or overloaded team may struggle to manage multiple tasks at once, causing delays or incomplete features.

9. Consensus on Requirements
   Question: Do all stakeholders agree on accuracy targets, UI flow, job ranking logic, and evaluation standards?
   Risk: A lack of consensus may lead to reworking workflows, AI outputs, or backend logic, resulting in major delays.

5.2.2 Risk Analysis

Table 5.1: Risk Table

ID	Risk Description	Probability	Schedule Impact	Quality Impact	Overall Impact
1	Unclear commitment from stakeholders regarding AI model integration and infrastructure.	Low	High	High	High
2	Low end-user (students/job seekers) engagement causing poor adoption.	Medium	High	Medium	Medium
3	Incomplete understanding of AI workflow requirements by the development team.	Medium	High	High	High

ID	Risk Description	Probability	Schedule Impact	Quality Impact	Overall Impact
4	Insufficient stakeholder involvement during requirement gathering for job matching and evaluation modules.	Medium	High	High	High
5	Unrealistic expectations from users about AI accuracy and avatar interview capabilities.	Low	Medium	Medium	Medium
6	Skill gaps in NLP, scraping, and AI model integration within the engineering team.	Medium	High	High	High
7	Frequent updates to AI models, APIs, or job scraping sources affecting system stability.	High	High	Medium	High
8	Insufficient development team size to manage frontend, backend, AI modules, and testing.	Medium	Medium	High	Medium
9	Stakeholder disagreements on evaluation criteria, scoring logic, or system outputs.	Medium	Medium	High	Medium

Table 5.2: Risk Probability Definitions

Probability	Value	Description
High	Probability of occurrence is >75%	Indicates a significant risk highly likely to occur in the SkillsBridge system, especially due to dynamic AI model updates, job scraping changes, or complex integrations. These risks can severely impact project timelines and system reliability.
Medium	Probability of occurrence is 26–75%	Indicates a moderate risk that may impact the platform—for example, NLP extraction inaccuracies, inconsistent user engagement, or delays in AI model responses. These require continuous monitoring and mitigation.
Low	Probability of occurrence is <25%	Indicates a minimal risk that is unlikely to affect development, such as complete system failure or long-term model unavailability. May not need immediate action but should still be documented.

5.2.3 Overview of Risk Mitigation, Monitoring, Management

The risk management plan for the SkillsBridge – AI-Powered Job Readiness Platform focuses on proactively identifying, reducing, and continually tracking risks that could affect the system's performance, AI accuracy, user adoption, or development schedule. Because the platform relies heavily on evolving AI models, real-time job scraping, NLP pipelines, and user-driven workflows, ensuring system reliability and data quality is essential.

To maintain consistent performance, SkillsBridge prioritizes regular model updates, prompt tuning, dataset improvements, and verification of job sources. Ongoing monitoring of scraping modules (LinkedIn, Naukri, Internshala), resume-parsing accuracy, and AI-generated interview feedback helps protect the system's relevance, accuracy, and overall reliability.

5.3 Project Schedule

1. Project Planning & Research — Weeks 1–8

Deliverables: finalized system requirements, workflow diagrams, NLP model selection (SBERT/TF-IDF), job-scraping strategy, and database schema.

Milestone: approved requirement specifications and finalized architectural plan.

2. Frontend Development — Weeks 9–11

Deliverables: Next.js interfaces for login, resume upload, user dashboard, job-match screen, and mock-interview UI.

Milestone: functional frontend prototype connected to dummy data.

3. Skill Extraction & Job Scraping Module — Weeks 12–15

Deliverables: resume text-processing pipeline, skill extraction using SBERT + TF-IDF, job scraping + relevance ranking engine.

Milestone: working skill extractor and job-matching output for sample resumes.

4. Question Generation & Mock Interview Model — Weeks 16–19

Deliverables: Ollama-Mistral powered question generation, job-specific interview question sets, mock-interview chatbot/voice module.

Milestone: functional mock-interview model capable of generating unique, role-specific questions.

5. Backend Integration — Weeks 20–22

Deliverables: unified backend API (Python), integration of all models/modules, Clerk authentication linkage, MongoDB connectivity, Supabase storage mapping.

Milestone: complete API workflow from resume upload → job matching → mock interview.

6. Testing & Deployment — Weeks 23–24

Deliverables: functional testing, performance testing, job-match accuracy validation, deployment of backend & frontend on cloud.

Milestone: stable deployed beta version accessible to users.

7. Feedback & Iteration — Weeks 25–29

Deliverables: improvements to interview evaluation, refinement in relevance ranking, UI adjustments, bug fixes based on testing feedback.

Milestone: improved release candidate ready for documentation review.

8. Documentation & Final Presentation — Weeks 30–32

Deliverables: finalized project report, system documentation, API references, presentation slides, and final demonstration package.

Milestone: complete, submission-ready project handover.

Table 5.3: Risk Management Plan

ID	Risk Description	Mitigation Strategy	Monitoring Approach	Management Plan
1	Inaccurate or insufficient data for resume parsing, skill extraction, or job-matching models.	Collect diverse resumes, job descriptions, and user samples to ensure balanced AI training.	Regularly review extraction accuracy and identify underrepresented patterns.	Expand datasets, perform data augmentation, and retrain AI pipelines periodically.
2	High false-positive or false-negative rates in skill extraction, job relevance ranking, or interview evaluation.	Use ensemble NLP techniques and multiple ranking models for improved accuracy.	Track accuracy, precision/recall, and incorrect recommendations.	Refine prompts, adjust thresholds, and retrain models with corrected samples.

ID	Risk Description	Mitigation Strategy	Monitoring Approach	Management Plan
3	Difficulty handling diverse job descriptions or evolving market skill requirements.	Integrate advanced semantic matching models and update scraping sources.	Monitor relevance percentage against user selections and feedback.	Continuously update algorithms and retrain using fresh job–market datasets.
4	Poor user experience or lack of trust in AI-generated interview questions or feedback.	Provide clear explanations for feedback, question logic, and evaluation parameters.	Collect user ratings and subjective feedback after each interview session.	Improve transparency and refine evaluation rules based on user feedback.
5	High computational cost for real-time NLP, scraping, and AI interview generation.	Optimize algorithms and cache recurring job patterns and parsed results.	Monitor system CPU/GPU usage and response time.	Scale using cloud compute services when workload increases.
6	Privacy concerns related to uploading resumes, interview recordings, or voice data.	Implement encryption during upload, processing, and storage.	Audit security logs and compliance regularly.	Enforce strict privacy policies, anonymize data, and maintain user consent mechanisms.
7	Misclassification of user skills or incorrect evaluation of interview responses.	Define standardized evaluation rubrics and enforce consistent scoring logic.	Compare AI evaluation with expert manual reviews to assess deviation.	Tune evaluation models and update scoring parameters for higher accuracy.
8	Inadequate testing across various job roles, domains, or skill categories.	Train and test models using diverse job domains and edge cases.	Track performance across sectors (IT, management, design, etc.).	Regularly retrain using domain-specific datasets for broader accuracy.
9	Legal and ethical concerns related to scraping job data or using AI for evaluation.	Ensure compliance with job-site terms, AI ethics guidelines, and data protection laws.	Monitor updates in compliance requirements and ethical standards.	Consult legal advisors and ensure continuous alignment with regulations.

Table 5.4: Risk A

Risk ID	**1**
Risk Description	Unclear stakeholder commitment towards AI model integration and system requirements.
Category	Management
Source	Project Scope & Stakeholder Communication
Probability	Low
Impact	High
Response	Mitigate
Strategy	Schedule regular sync-up meetings with supervisors and stakeholders to ensure alignment on requirements and progress.
Risk Status	Occurred

Table 5.5: Risk B

Risk ID	**2**
Risk Description	AI models (Ollama, Mistral, Gemini) unable to properly communicate due to format mismatch or API failures.
Category	Technical
Source	Development Phase
Probability	Medium
Impact	High
Response	Mitigate
Strategy	Implement strict integration protocols and standardized data formats for smooth model-to-model communication.
Risk Status	Identified

Table 5.6: Risk C

Risk ID	**3**
Risk Description	Insufficient computational resources for running NLP, scraping, and AI interview generation.
Category	Resource Management
Source	Backend Processing & Infrastructure
Probability	Medium
Impact	Medium
Response	Accept
Strategy	Use cloud-based compute scaling and pre-scheduled resource allocation during heavy workloads.
Risk Status	Identified

Table 5.7: Task Assigned

Task ID	Task Description	Duration	Assigned To	Status
Task 1	Initial Requirement & Planning	2 weeks	Omkar Ghodekar	Completed
Task 2	AI & Model Integration (Ollama, Mistral, Gemini)	8 weeks	Team Member 1	In Progress
Task 3	Frontend (Next.js) & Backend (Python) Development	8 weeks	Team Member 2	Not Started
Task 4	Testing, Debugging & Deployment	5 weeks	Team Member 3	Not Started
Task 5	Documentation & Review	3 weeks	Team Member 4	Not Started

5.3.1 Description of Tasks:

Task 1: Initial Phases

- Requirement Gathering:

Collect and document all functional and technical requirements from stakeholders, including students, job seekers, and training partners. Identify expectations for resume analysis, job matching, mock interviews, AI-based evaluations, and learning-path generation. Clearly define data flow, performance needs, AI model usage, and expected system outcomes to support smooth development and deployment.

- System Architecture Design:

Design a complete system architecture outlining key components such as the NLP skill-extraction pipeline, job-scraping engine, AI model orchestration (Ollama + Gemini), frontend-backend communication, and database structure. Ensure the architecture is scalable, supports real-time AI evaluations, and keeps a clean separation between UI, model inference, and asynchronous processing.

Task 2: Data & Model Development

- Dataset Collection & Preprocessing:
  1) Gather and prepare datasets needed for skill extraction and job recommendations, including sample resumes, job descriptions, and skill ontologies.

2) Preprocess live job-scraping data by cleaning text, removing noise, standardizing job requirements, and extracting domain-specific skill sets.
3) When job data is incomplete or inconsistent, create synthetic or standardized entries to keep job roles uniform and support accurate similarity scoring.

- Model Development (SBERT, TF-IDF, Mistral, Gemini):
  1) Use SBERT and TF-IDF to extract meaningful skills, keywords, and experience indicators from user resumes.
  2) Develop relevance-ranking logic using similarity scoring to match candidate profiles with scraped job descriptions.
  3) Configure the Mistral model through Ollama to generate job-specific technical and HR interview questions.
  4) Integrate the Gemini model to evaluate user responses and generate personalized learning paths based on strengths and weaknesses.

- Model Training & Validation:

  1) Fine-tune and test the skill-extraction pipeline across different resume styles and domains to improve accuracy.
  2) Validate job-relevance scoring using sample job descriptions and compare expected and generated ranking results.
  3) Continuously test the quality of mock interview questions and evaluation accuracy using trial responses to ensure realistic and reliable feedback.

Task 3: Web-Based Application Development

- User Interface Development:
  1) Build a modern, intuitive, and responsive UI using Next.js where users can upload resumes, explore job matches, take mock interviews, and view AI feedback.
  2) Create dashboards for tracking progress, storing interview history, visualizing readiness scores, and providing personalized learning roadmaps

- Backend Integration:
  1) Integrate NLP modules, the scraping engine, the Ollama API, and the Gemini evaluation model into the backend to automatically process resumes, match jobs, generate interview questions, and score responses.

2) Develop secure backend endpoints for storing interview data, maintaining user profiles, and retrieving recommendations from MongoDB and Supabase.
3) Ensure smooth real-time interaction between the AI avatar interviewer and backend logic to deliver realistic interview simulations

Task 4: Testing and Deployment

- Testing & Quality Assurance:
    1) Carry out unit, integration, and end-to-end testing to ensure smooth communication between the frontend (Next.js) and backend (Python), especially in modules such as resume processing, job-relevance ranking, interview-question generation, and AI-based evaluations.
    2) Verify the performance of SBERT, TF-IDF, Mistral, and Gemini modules across different resume formats, job roles, interview responses, and user scenarios to maintain consistent accuracy and reduce errors in skill extraction, job matching, and feedback generation.
    Conduct usability testing to confirm that users experience a smooth process—from uploading their resumes to receiving personalized learning paths.

.

- Deployment & Integration:
    1) Deploy the platform on a scalable cloud environment (e.g., Vercel for the frontend and AWS/Render for the backend) to support real-time resume parsing, job scraping, AI inference, and mock-interview evaluation.
    2) Ensure the deployment setup provides low latency, high performance, and secure handling of sensitive data such as resumes, interview recordings, and user profiles.

Task 5: Documentation & Review

- Documentation & Report Writing:
    1) Create detailed documentation covering all stages of development, including system architecture, module workflows, AI model configurations (SBERT + TF-IDF for skill extraction, Mistral for interview generation, Gemini for evaluation and learning-path creation), and the integration process between backend APIs and frontend components.
    2) Document the technical specifications for each module—resume-processing pipeline, job-scraping engine, interview generator, evaluation logic, and database schemas. Include explanations of preprocessing steps, similarity-scoring techniques, AI model interactions, and the methods used to evaluate candidate performance.
- Final Project Review & Closure:
    1) Prepare the final report summarizing the system's functionality, development challenges, testing outcomes, and key insights.
    2) Offer suggestions for future enhancements such as adding more job portals, improving avatar realism, integrating voice-emotion detection, or supporting multi-round interview simulations.

3) These recommendations will help guide future development phases for SkillsBridge

5.3.2 Task network

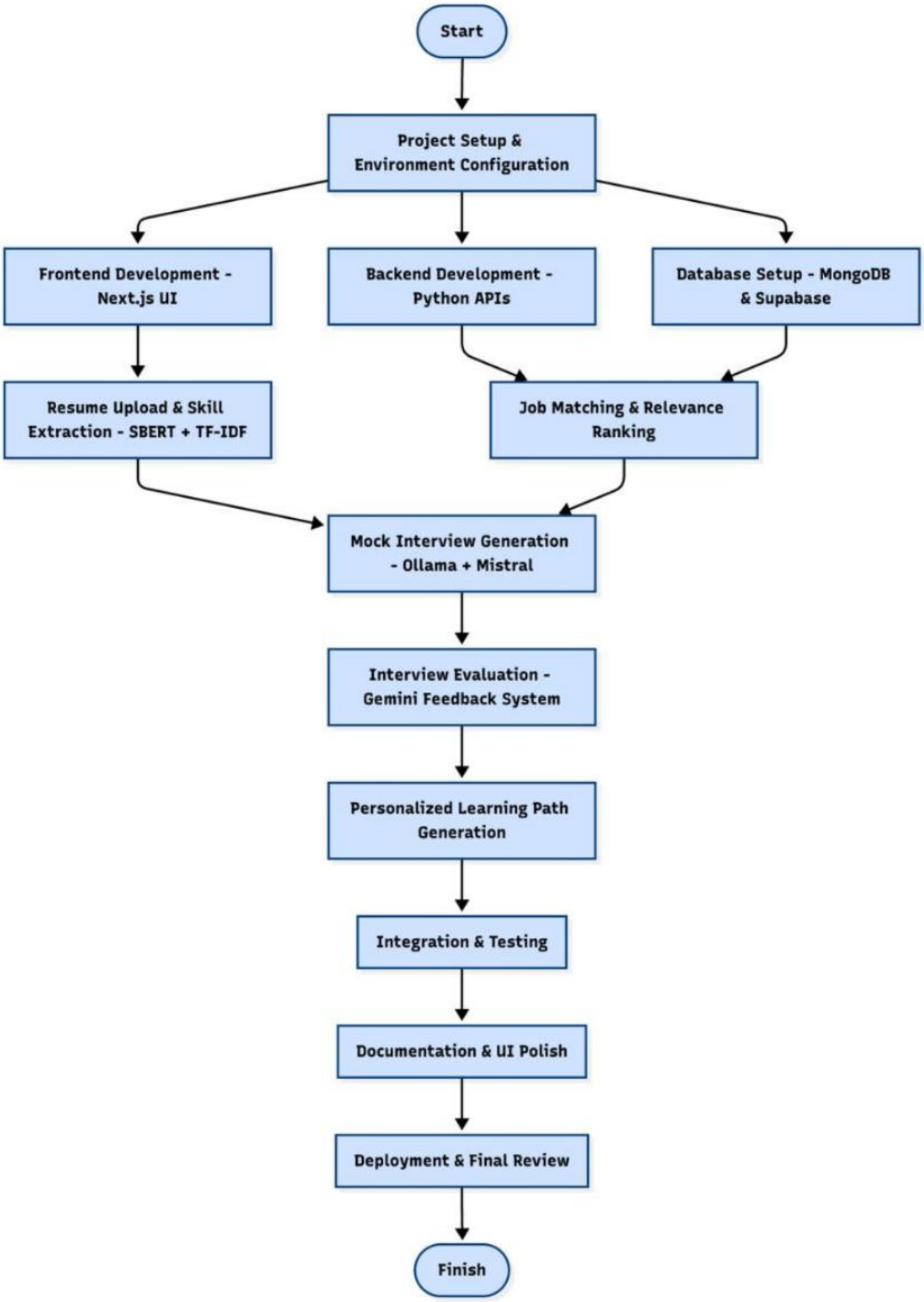
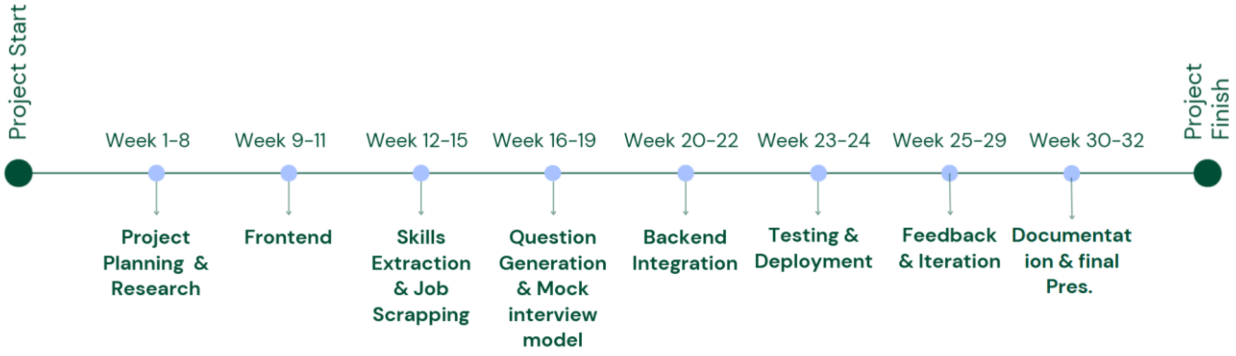

Start
Project Setup & Environment Configuration
Frontend Development - Next.js UI
Backend Development - Python APIs
Database Setup - MongoDB & Supabase
Resume Upload & Skill Extraction - SBERT + TF-IDF
Job Matching & Relevance Ranking
Mock Interview Generation - Ollama + Mistral
Interview Evaluation - Gemini Feedback System
Personalized Learning Path Generation
Integration & Testing
Documentation & UI Polish
Deployment & Final Review
Finish

Fig 5.1. Task Network

5.3.3 Timeline Chart



F

ig 5.2. Timeline chart

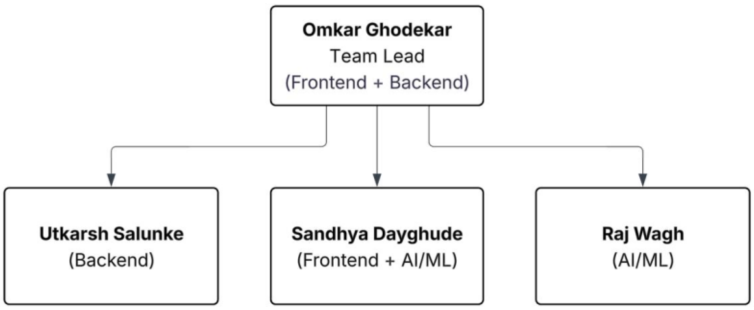
5.3.4 Team Organization

We plan to split the work equally amongst the four of us. We assigned each task a due date. Individual results are ultimately combined into a single result.

6.6.1 Organization of the team

The project's team structure is determined. Roles are established.

Table 5.8: Team Structure



5.3.4.1 Management, Reporting and Communication

A dedicated team of four developers effectively manages the progress of the **Integrated Assessment of Programming Skills using Originality Check** project through the following strategies:

- **Inter-Team Communications:**

  Virtual platforms (e.g., Slack, Microsoft Teams) are used to maintain continuous communication, allowing developers to collaborate seamlessly, share updates, and quickly resolve any issues that arise.

- **Mentorship Guidance:**

  Monthly mentor meetings are scheduled to receive expert advice and guidance on technical challenges, project direction, and solutions to potential bottlenecks. These mentors help the team navigate complex decisions, ensuring the project stays on track.

6 Software Requirement Specification (SRS)

6.1 Introduction

6.1.1 Purpose and Scope of Document

This Software Requirements Specification (SRS) outlines the complete set of requirements for the *SkillsBridge: AI-Powered Job Readiness and Career Development Platform*. Its purpose is to provide a detailed and shared understanding of what the system is expected to do, how it should operate, and the performance standards it must meet. By acting as a central reference point, this document helps ensure that everyone involved—developers, designers, stakeholders, and end users—has a consistent view of the system's goals and limitations.

The SRS describes the system's key objectives, including both functional and non-functional requirements, expected user interactions, workflow processes, and major design considerations. It also captures core capabilities such as resume evaluation, job-role matching, AI-driven mock interviews, performance assessment, and personalized learning recommendations. With a clearly defined set of requirements, this document supports smooth communication across the project team and guides the development process so the final product effectively improves job readiness in a structured, intelligent, and user-focused way

6.1.2 Overview of Responsibilities of Developer

In the *SkillsBridge: AI-Powered Job Readiness and Career Development Platform*, developers are responsible for several key tasks:

- **Requirement Analysis:**
  Developers work closey with stakeholders—such as project mentors, domain experts, and potential users—to 1.75
- **System Design:**

  Design a scalable and modular architecture that supports core functionalities such as resume parsing, NLP-based skill extraction, job description processing, interview question generation, and evaluation workflows. This includes planning database structures, API layers, UI components, and ensuring seamless integration between frontend, backend, and AI models.

- **Model Development:**

  Develop and optimize AI and NLP models required for the platform, including SBERT-based skill extraction, TF-IDF keyword analysis, semantic similarity scoring, and AI-driven interview generation using Mistral. Implement evaluation models using Gemini to assess user responses on accuracy, clarity, and confidence. Ensure high precision and reliability to minimize incorrect skill detection and inaccurate performance assessments.

- **Application Development:**

  Build an intuitive and user-friendly application interface that allows users to upload resumes, view extracted skills, explore matched job roles, participate in mock interviews, and review evaluation feedback. Ensure seamless backend integration to process inputs, generate AI-driven responses, and deliver personalized learning paths in real time.

- **Testing:**

  Perform rigorous testing across all system components, including unit tests for NLP modules, integration tests for backend–frontend communication, and end-to-end evaluations for resume analysis, job matching, and interview simulation. Validate that all functionalities operate accurately and reliably under different user scenarios.

- **Documentation**:

  Prepare comprehensive documentation covering system architecture, functional workflows, API specifications, and guidelines for installation, usage, and maintenance. Develop user manuals for students and administrators to ensure effective utilization of platform features and support future enhancements.

- **Maintenance and Support:**

  Provide ongoing maintenance to address bugs, improve system performance, and ensure compatibility with evolving AI models and job market trends. Implement regular updates to maintain accuracy in skill extraction, job matching, and evaluation processes while supporting users with timely technical assistance.

6.2 Usage Scenario

This section provides an overview of common usage scenarios, illustrating how different users will interact with the **SkillsBridge: AI-Powered Job Readiness and Career Development Platform** and highlighting its core functionalities. These scenarios demonstrate typical user journeys, system behavior, and the role of AI-driven features in enhancing job readiness through resume analysis, job-role matching, mock interview participation, and personalized learning recommendations.

Table 6.1: Use Cases

Sr. No.	Use Case	Description	Actors	Assumptions
1	User Logs into the Platform	The user navigates to the login page, enters valid credentials, and successfully accesses the dashboard.	Student, Instructor	The user has a valid registered account and remembers their login credentials.
2	Student Submits a Coding Assessment	The student accesses an assigned coding assessment and solves the problem using the integrated coding interface with optional voice input support.	Student	The assessment is active, and the coding environment is fully functional.
3	Instructor Reviews Submission	The instructor evaluates the student's submitted solution, provides comments, feedback, and assigns grades.	Instructor	The instructor is logged into the system and has access to the student's submissions.
4	Dynamic Question Generation	The system automatically generates coding questions dynamically to evaluate the student's conceptual understanding.	System, Student	The system has access to the student's previously submitted code or performance data.
5	Code Similarity Check	The system performs similarity analysis by comparing the student's submitted code against a real-time database to ensure code originality.	System, Student	The similarity-checking tool (e.g., STRANGE) is integrated and functioning correctly.

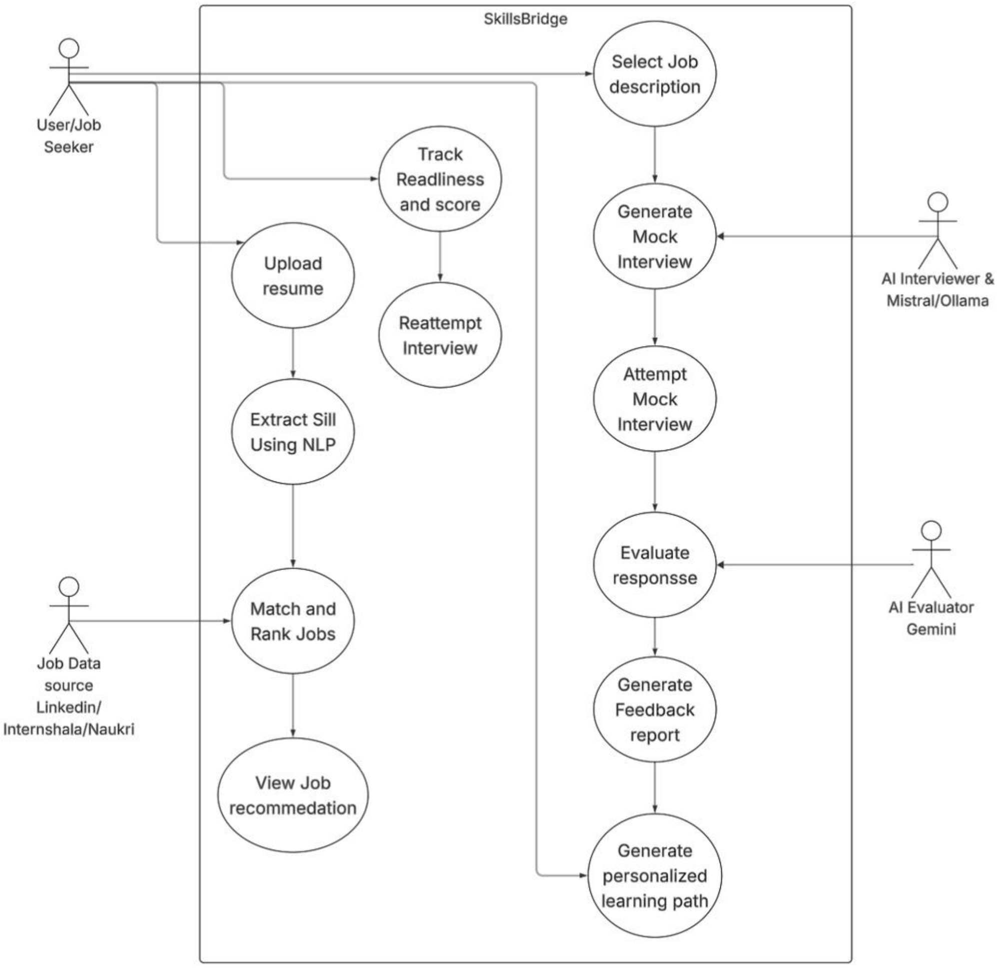


Fig 6.1. Use case diagram

6.2.1 Use Case View

SkillBridge System is at the center, representing the core functionalities of the platform. Each use case is represented as an oval and is connected to the relevant actors, depicted as stick figures.

• End User (Student / Job Seeker) can initiate the following use cases :

- **Upload Resume:**
  Users upload their resume to enable NLP-based skill extraction and analysis.
- **View Extracted Skills:**
  Users can see the technical and soft skills identified from their resume.
- **Explore Job Matches:**
  Users view ranked job recommendations based on semantic similarity between their skills and real-time job descriptions.
- **Attempt Mock Interview:**
  Users participate in an AI-driven mock interview generated by the Mistral model.

- **View Interview Evaluation:**
  Users receive AI-generated feedback on their interview performance based on accuracy, communication ability, and confidence.
- **Access Learning Path:**
  Users view a personalized skill-development roadmap designed to address identified gaps.
- **Provide Feedback:**
  Users give feedback on job matching accuracy, interview relevance, and overall system usability.

6.3 Data Model and Description

6.3.1 Data Description

The data handled by the SkillsBridge System includes various forms of user input, AI-generated outputs, evaluation metrics, and interaction logs. These data objects are essential for ensuring accurate skill extraction, job-role matching, mock interview processing, and personalized learning path generation. The key data elements are described below:

1. User Profile:

Contains user-specific information such as name, email, education details, experience, and authentication data. This enables personalized recommendations and progress tracking.

2. Resume Data:

Raw resume files and extracted textual content obtained through NLP processing. This includes identified skills, projects, experience summaries, and relevant keywords derived from SBERT and TF-IDF models.

3. Job Description Data:

Details of job roles collected from external job portals, including required skills, experience levels, job responsibilities, and semantic vectors used for matching.

4. Interview Responses:

User responses during AI-driven mock interviews, stored as text transcripts or audio-to-text conversion outputs. This data is analysed by evaluation models for accuracy, communication, and confidence.

5. Evaluation Report:

AI-generated assessment of the user’s interview performance, including scores, qualitative feedback, improvement suggestions, and readiness indicators.

6. Learning Path Data:

Personalized skill-development recommendations derived from performance evaluations and skill-gap analysis. Includes suggested topics, difficulty levels, and progression timelines.

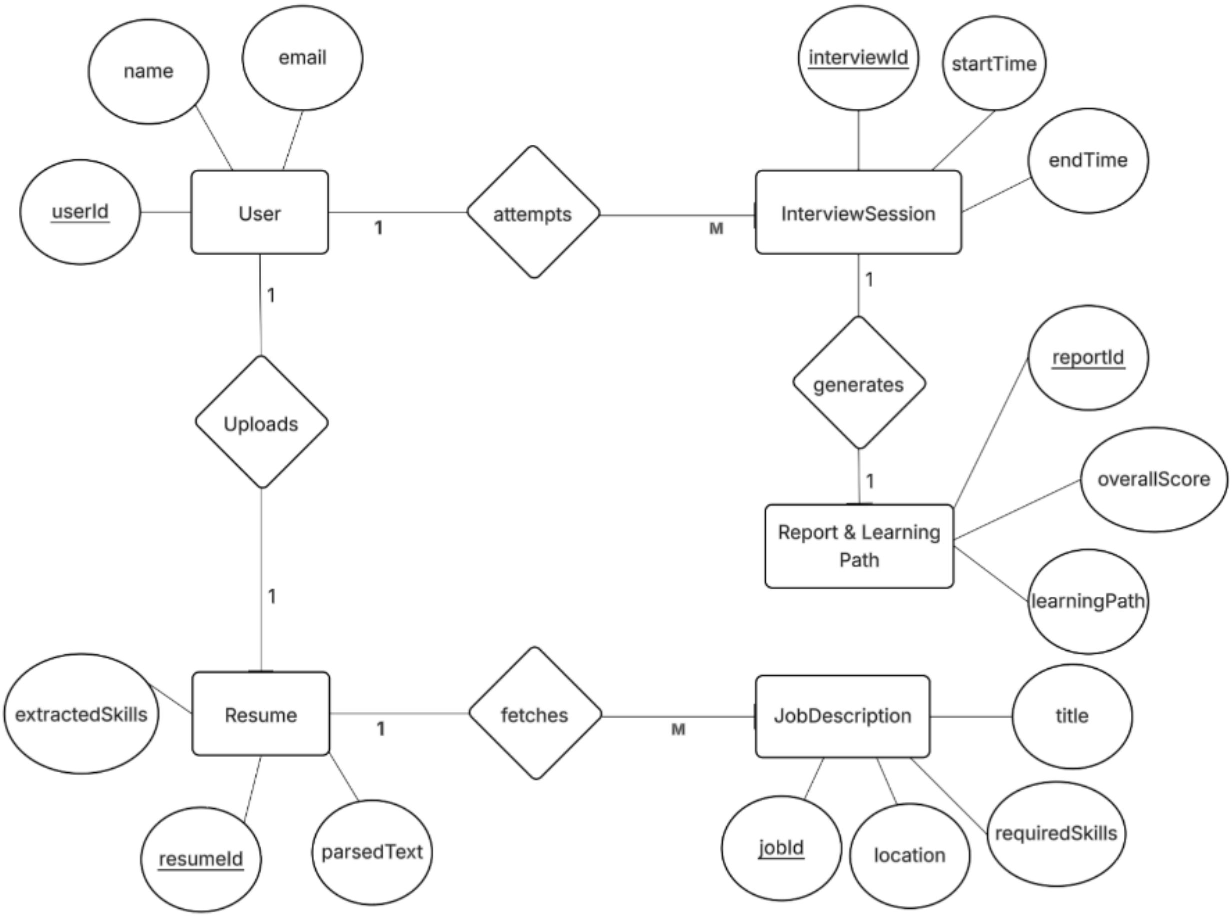
7. User Feedback:

Feedback provided by users regarding job match accuracy, interview relevance, system usability, and evaluation fairness. This data helps improve platform performance.

8. System Logs:

Log files capturing server actions, API calls, errors, model processing times, and system events. These logs are essential for debugging, performance monitoring, and maintaining platform reliability.

6.3.2 Data objects and Relationships



6.4 Functional Model and Description

6.4.1 Data Flow Diagram

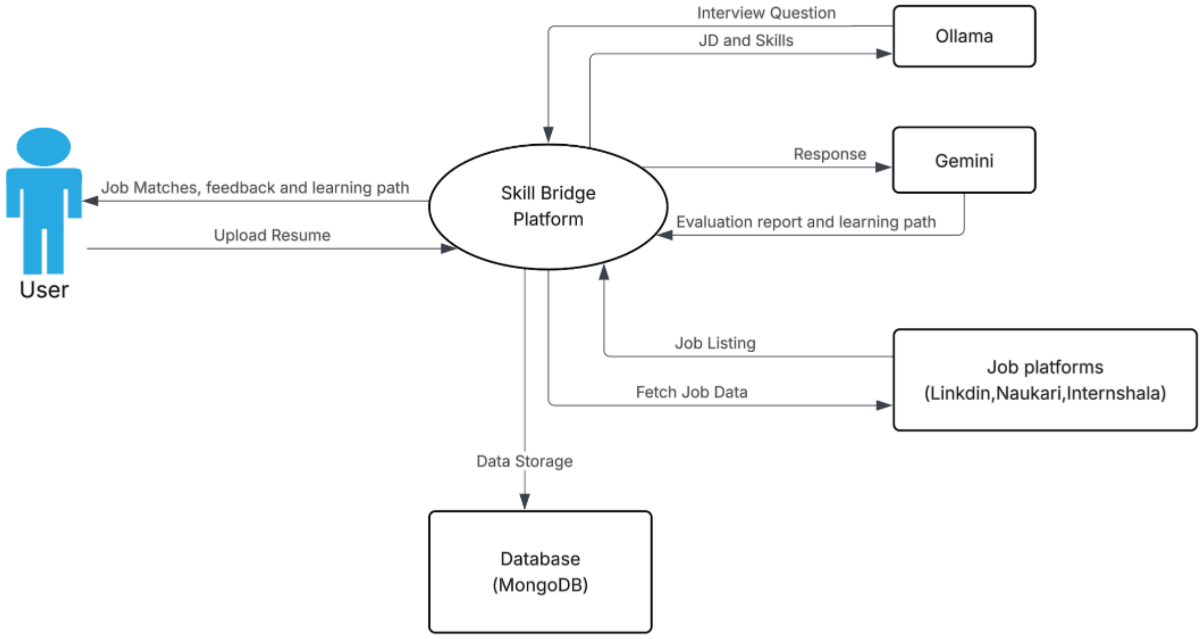


Fig 6.2. Level 0 Data Flow Diagram

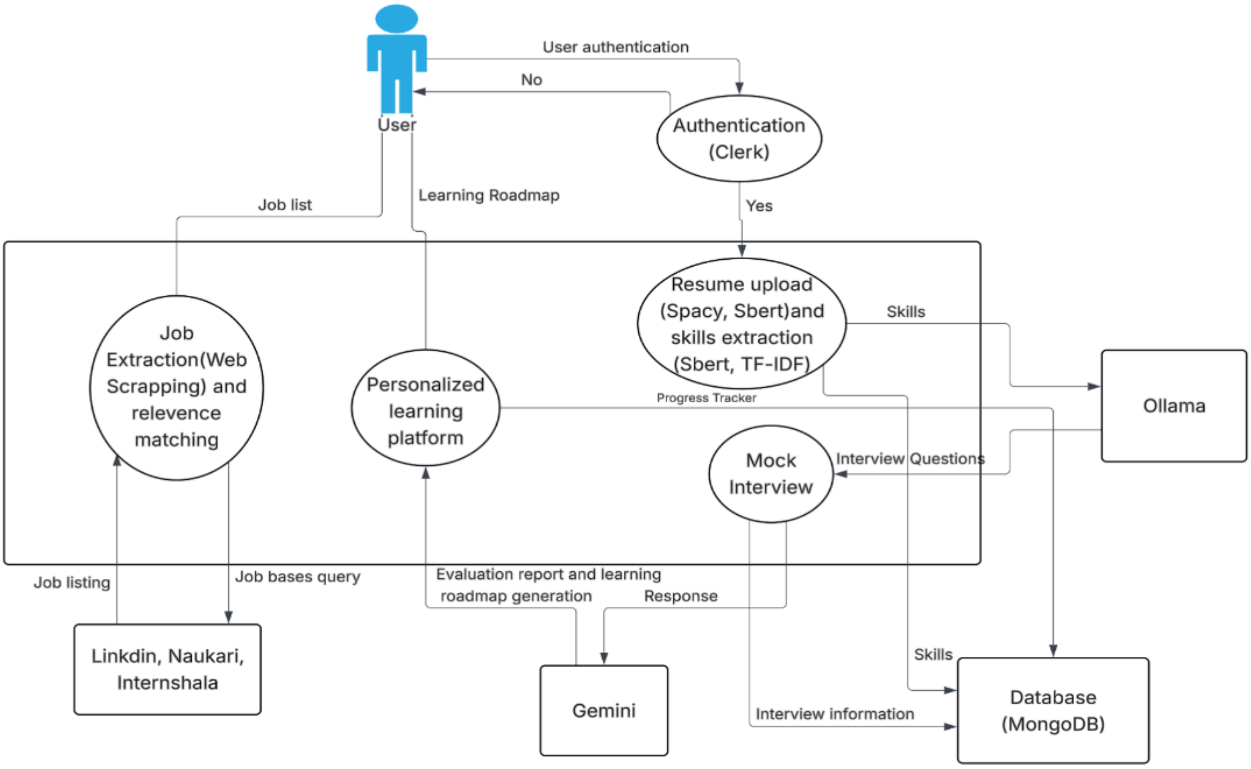

User
User authentication
No
Authentication (Clerk)
Job list
Learning Roadmap
Yes
Resume upload (Spacy, Sbert)and skills extraction (Sbert, TF-IDF)
Skills
Job Extraction(Web Scrapping) and relevence matching
Personalized learning platform
Progress Tracker
Ollama
Mock Interview
Interview Questions
Job listing
Job bases query
Evaluation report and learning roadmap generation
Response
Skills
Linkdin, Naukari, Internshala
Gemini
Interview information
Database (MongoDB)

Fig 6.3. Level 1 Data Flow Diagram

6.4.2 Activity Diagram:

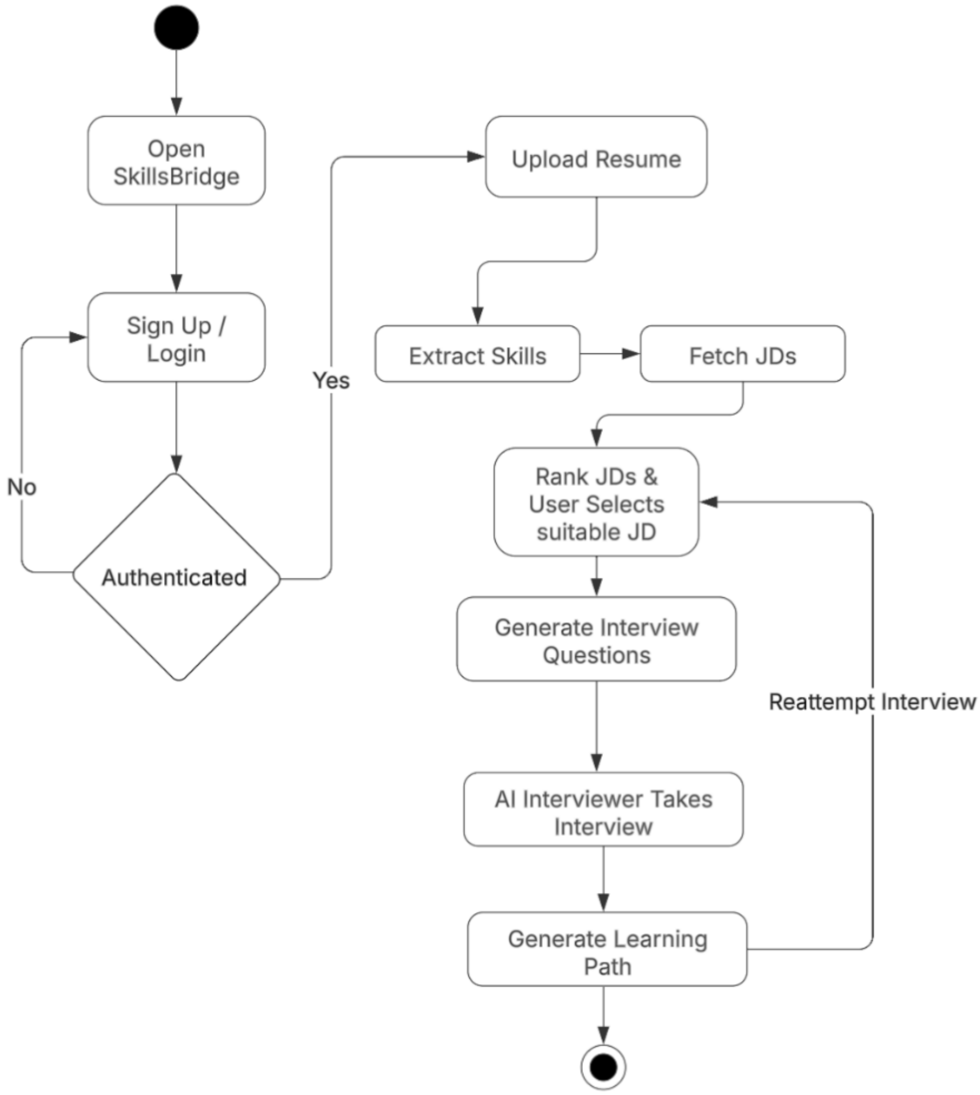


Fig 6.4. Activity Diagram

6.4.3 Non-Functional Requirements

❖ **Interface Requirements**

- The SkillsBridge platform should provide an intuitive and user-friendly interface, enabling users to upload resumes, view extracted skills, explore job matches, and participate in mock interviews with ease.

- The system must be fully responsive and accessible across various devices, including desktops, tablets, and mobile phones.

- Clear error messages should be displayed for unsupported file formats, incomplete inputs, or server-side issues to guide users effectively.

- The interface should offer smooth navigation between modules such as resume analysis, job recommendations, interview simulations, and learning paths.

❖ Performance Requirements

- The system must process resume uploads and generate skill extraction results within a short time frame, ideally within a few seconds for at least 90% of requests.

- Job-role matching and recommendation generation should be optimized to ensure efficient retrieval even when handling large job datasets.

- AI-driven mock interviews and evaluations should respond with minimal latency to maintain a realistic and interactive user experience.

- The platform must support a high volume of concurrent users (minimum 100–200 simultaneously) without degrading performance.

- System uptime should be maintained at **99.5% or higher** to ensure continuous accessibility for students and job seekers.

6.4.4 State Diagram

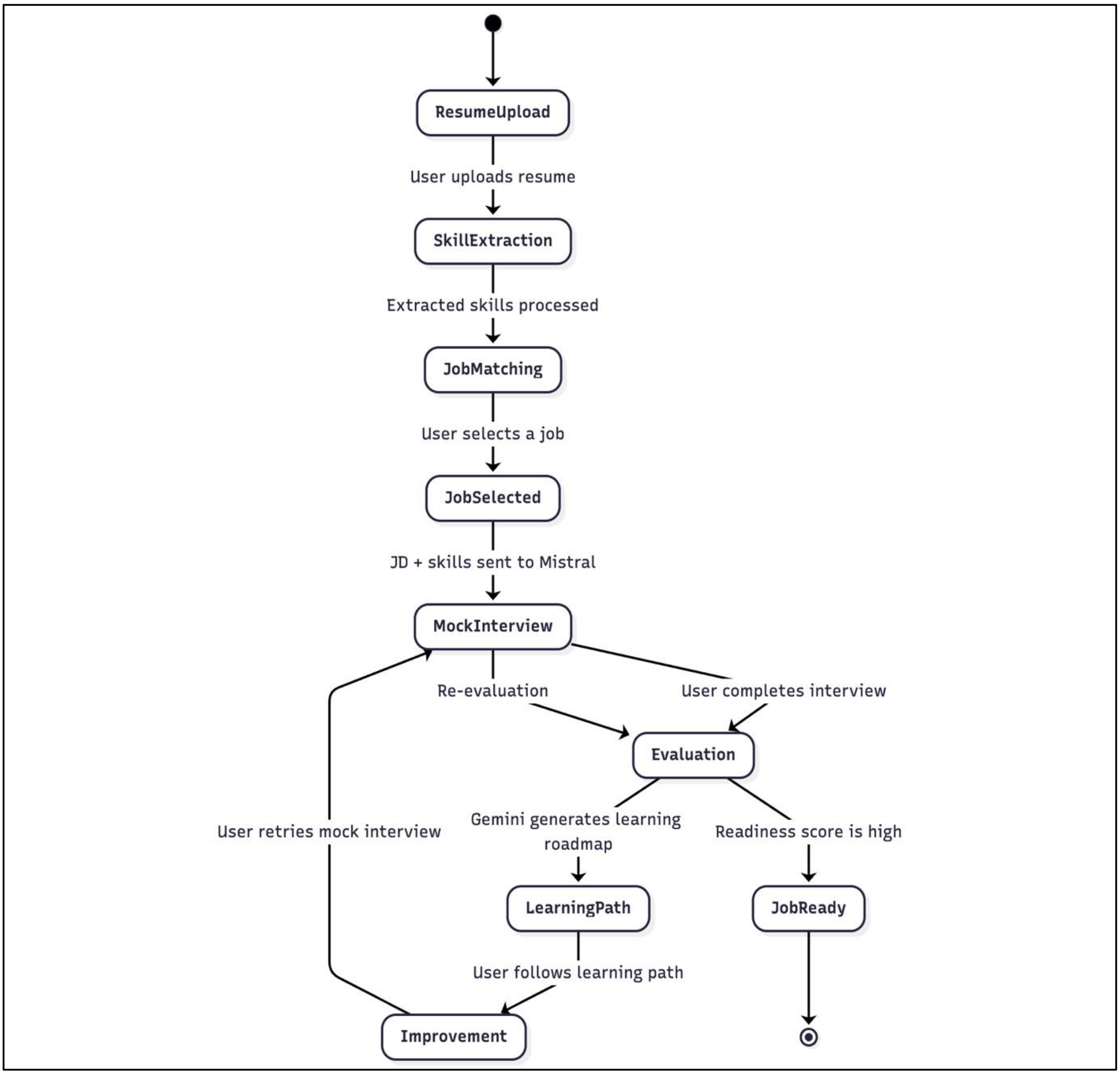


Fig. 6.5. State Diagram

7 Detailed Design Document using Appendix A and B

7.1 Introduction

This document outlines the system design used to develop an AI-driven job readiness platform capable of analyzing user resumes, matching relevant job roles, conducting AI-powered mock interviews, and generating personalized learning paths. The **SkillBridge Platform** processes user-submitted resumes, job descriptions, and interview responses using advanced NLP models and evaluation algorithms to deliver meaningful insights and improvement recommendations.

This section introduces the overall design framework, establishing the foundation for subsequent details regarding the system architecture, workflow, and data handling processes. It describes how various components—including resume extraction modules, job-matching engines, mock interview generators, and evaluation models—interact within the system. Additional visual references are provided in **Appendix A (Data Flow Diagrams)** and **Appendix B (ER Diagram)** to support a clearer understanding of the platform's internal processes and data structures.

7.2 Architectural Design

The architectural design provides a high-level overview of the **SkillBridge Platform**, an AI-powered system designed to extract user skills, match suitable job roles, conduct mock interviews, and generate personalized learning paths. The architecture is organized into three major components: **User**, **AI Processing Engine**, and **System Services Layer**. These components work together to manage the complete workflow—from resume upload to job-role recommendations and interview evaluation.

1. **User Interface (UI)**:

   - The UI facilitates all user interactions, including resume upload, job role selection, mock interview participation, and viewing evaluation reports.

   - It is developed using ReactJS and supports multi-device accessibility (desktop, tablet, mobile), ensuring a seamless and responsive user experience.

   - The interface integrates visual dashboards to display extracted skills, job matches, performance scores, and personalized learning paths.

- Detailed UI interactions are described in **Appendix A (Data Flow Diagrams)**.

2. Processing Engine (AI + NLP Modules):

- This module handles all AI-driven tasks, including parsing resumes, extracting skills, processing job descriptions, and generating semantic embeddings.
- It performs preprocessing steps such as text extraction, tokenization, normalization, and vectorization using models like **SBERT** and **TF-IDF**.
- The engine also generates interview questions using the **Ollama–Mistral** model and evaluates user responses through the **Gemini** evaluation framework.
- It ensures that inputs are formatted and processed correctly before being analyzed by downstream matching and evaluation models.

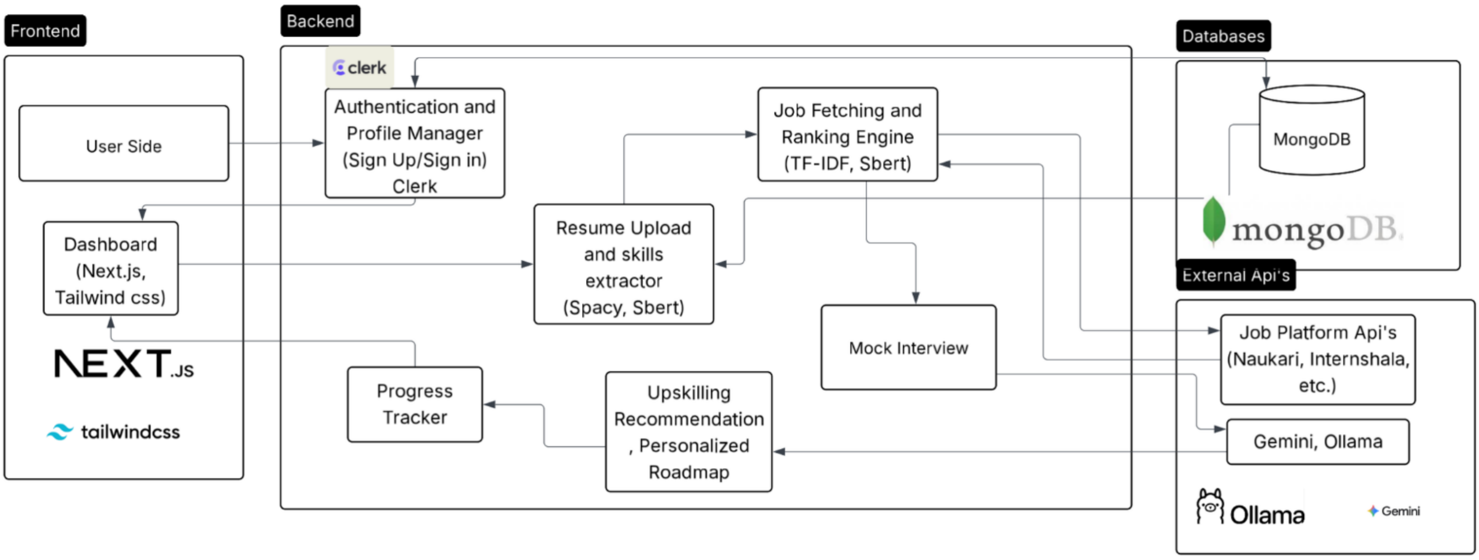

Frontend
User Side
Dashboard (Next.js, Tailwind css)
NEXT.JS
tailwindcss
Backend
clerk
Authentication and Profile Manager (Sign Up/Sign in) Clerk
Job Fetching and Ranking Engine (TF-IDF, Sbert)
Resume Upload and skills extractor (Spacy, Sbert)
Mock Interview
Progress Tracker
Upskilling Recommendation , Personalized Roadmap
Databases
MongoDB
mongoDB.
External Api's
Job Platform Api's (Naukari, Internshala, etc.)
Gemini, Ollama
Ollama
Gemini

Fig 7.1. Architecture Diagram

7.3 Data Design (Using Appendices A and B)

The SkillsBridge platform relies on a well-structured data model that ensures efficient skill extraction, interview generation, user evaluation, and job-role matching. The data design incorporates internal software structures, global shared structures, temporary session-based structures, and a comprehensive database schema.

Appendices A and B contain detailed ER diagrams and data dictionaries corresponding to the elements described in this section.

7.3.1 Internal Software Data Structure

Internal data structures are used within individual components (frontend or backend) during execution. These are not stored permanently and are used for processing NLP outputs, job matching, and interview evaluation.

a) Resume Processing Structure: Used during skill extraction.

```
ResumeText {
   raw_text: string
   cleaned_text: string
   tokens: list<string>
   tfidf_features: vector<float>
   sbert_embeddings: vector<float>
}
```

b) Job Matching Structure: Used to compute relevance scores.

```
JobMatchObject {
   job_id: string
   extracted_keywords: list<string>
   skill_similarity_score: float
   experience_match: float
   relevance_score: float
}
```

c) Interview Question Structure: Generated via Mistral model.

```
InterviewQuestion {
   question_id: string
   question_text: string
   difficulty_level: string
   question_type: string   // technical, behavioral, scenario-based
}
```

7.4 Component Design

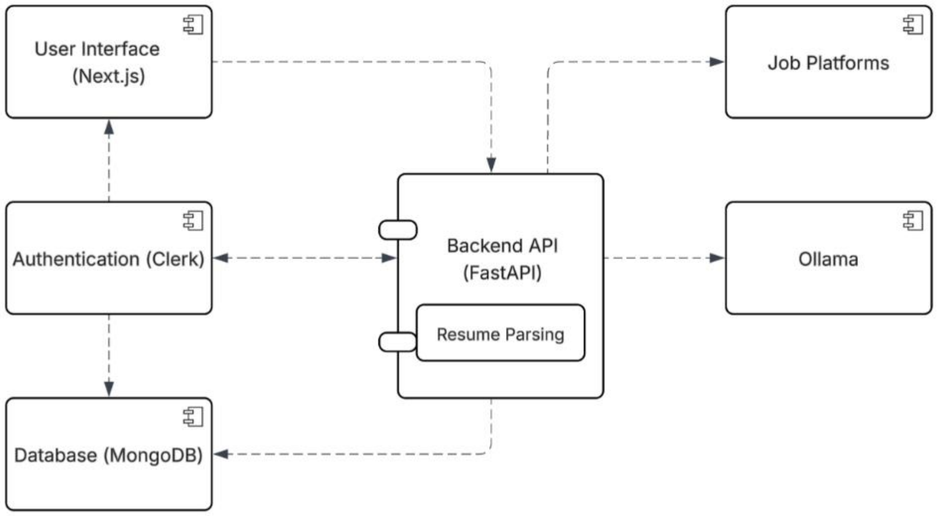


Fig 7.2 Component Diagram

7.4.1 Class Diagram

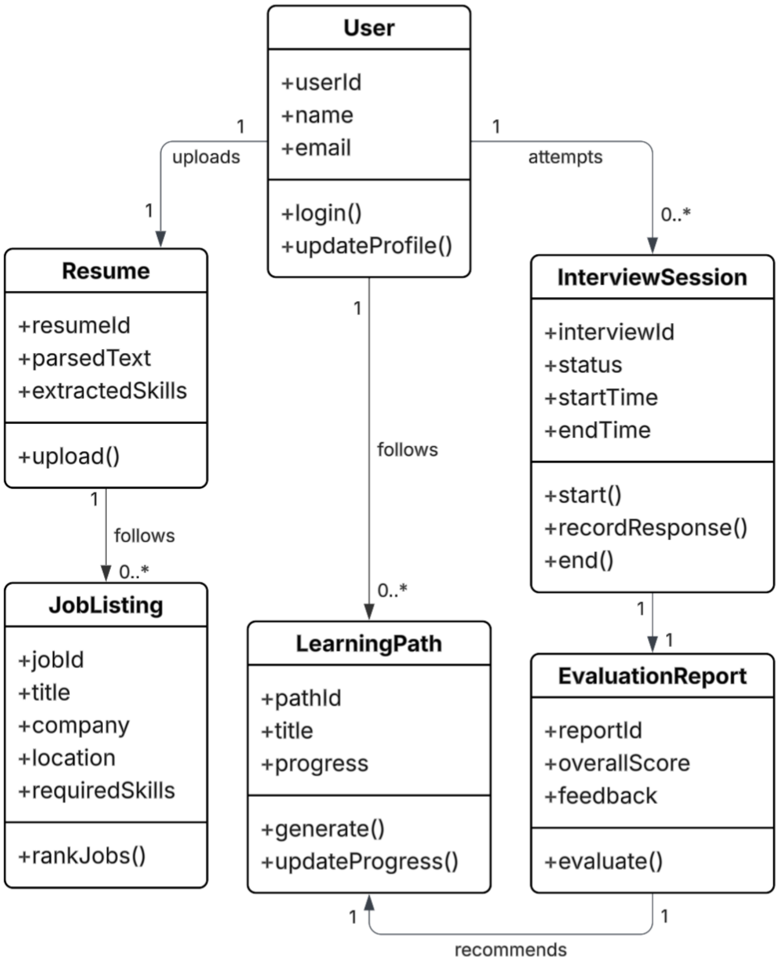


Fig 7.3 Class Diagram

8 Project Implementation

8.1 Introduction

The implementation of the **SkillsBridge – AI-Powered Job Readiness Platform** integrates advanced AI models, NLP pipelines, and scalable web technologies to deliver an end-to-end job-preparation ecosystem. The platform ensures accurate skill extraction, real-time job-role matching, AI-generated mock interviews, and automated performance evaluation.

This section presents the tools, technologies, machine learning models, and implementation methodologies used to develop SkillsBridge. It also explains how each module was designed, integrated, and optimized to ensure reliability, scalability, and seamless user experience.

8.2 Tools and Technologies Used

The implementation of this project relies on a variety of tools and technologies to ensure functionality, scalability, and ease of use. These include:

- Frontend: Next.js for building a fast, responsive, SEO-optimized interface.Tailwind CSS / ShadCN UI for modern, clean UI components. WebRTC for real-time voice/video interactions with AI avatar.
- Backend: Python (FastAPI / Flask) for: NLP pipelines, Resume parsing, Skill extraction, Job relevance scoring, Communication with AI models (Ollama & Gemini)
- Databases & Storage MongoDB – Stores user profiles, skills, interview history, resumes metadata. Supabase Storage – Stores uploaded resumes and audio/video interview recordings.
- AI & Machine Learning SBERT + TF-IDF – Skill extraction and semantic similarity. Ollama Mistral – Role-based interview question generation. Google Gemini – Performance evaluation & personalized learning path generation. Python ML stack – NumPy, Pandas, Scikit-learn for preprocessing and scoring.
- Job Scraping & Integration: BeautifulSoup – For scraping job descriptions from LinkedIn, Naukri, Internshala.

8.3 Methodologies/Algorithm Details

To deliver accurate skill extraction, job matching, and interview evaluation, SkillsBridge employs multiple algorithms involving NLP processing, semantic similarity, dynamic question creation, and AI-based scoring.

8.3.1 Algorithm / Pseudo Code – Skill Extraction (SBERT + TF-IDF)

```
● Upload resume (PDF/Docx)
● Convert resume to text
● Preprocess text:
    • Remove stopwords
    • Lemmatize
    • Normalize formatting
● Extract potential skills using TF-IDF scoring
● Generate embeddings using SBERT
    • For each skill in predefined skill dictionary:
        Compute cosine similarity with extracted resume embeddings
● If similarity > threshold:
    • Add skill to final_skill_set
● Return final_skill_set
● Stop
```

8.3.2 Algorithm / Pseudo Code – Job Relevance Ranking

This algorithm matches user skills with job descriptions and ranks jobs by suitability.

Pseudo Code:

```
● Start
● Fetch job descriptions from platforms
● Extract keywords and required skills from each JD
● For each job:
        •    Compute skill_match_score using cosine similarity (SBERT)
        •    Compute experience_match_score
        •    Compute keyword_relevance_score
● Final_relevance = 0.5 * skill_match + 0.3 * keyword_relevance + 0.2 * experience_match
● Rank jobs by Final_relevance (descending)
● Return Top-N ranked jobs
● Stop
```

8.3.3 Algorithm – AI Question Generation (Ollama Mistral)

This algorithm generates interview questions dynamically based on job role and user skills.

Pseudo Code:

```
● Start
● Input:
        • Selected job description (JD)
        • Extracted user skills
● Combine JD + skills into structured context
●
● Send context to Mistral model:
        • "Generate 8–10 interview questions related to this role,
        • including technical, behavioral, and scenario-based questions."
● Receive question_list
● Store questions for interview session
● Stop
```

8.3.4 Algorithm – Interview Evaluation (Gemini)

Gemini evaluates user responses on multiple parameters.

Pseudo Code:

```
● Start
● For each user response:
        •    Send audio/text to Gemini model
● Gemini returns:
        • Technical Accuracy Score
        • Communication Score
        • Confidence Score
        • Behavior Score
        • Overall Readiness Level
● Aggregate scores to generate final feedback report
● Generate personalized learning roadmap based on weak areas
● Stop
```

9 Conclusion and Future Scope

Conclusion

In conclusion, the SkillsBridge platform enhances the efficiency, fairness, and reliability of job readiness preparation by integrating advanced AI-driven skill analysis, job-role matching, and mock interview evaluation. By automating resume processing, generating relevant interview scenarios, and delivering personalized learning paths, the system ensures that users are assessed and guided based on their actual competencies and improvement potential.

The platform supports candidates in understanding industry expectations, identifying their strengths and weaknesses, and continuously improving through targeted upskilling recommendations. By maintaining ethical standards, data privacy, and unbiased evaluation, SkillsBridge provides a robust and scalable solution for students, job seekers, and institutions. This comprehensive approach not only promotes confidence and preparedness in candidates but also adapts to evolving industry trends, making it a valuable tool for career development and employability enhancement.

Future Scope

The SkillBridge platform can be expanded with several advanced features to enhance its effectiveness, scalability, and value for students, institutions, and organizations. Potential future enhancements include:

1. Integration with Learning Management Systems (LMS): SkillsBridge can be integrated with widely used LMS platforms (e.g., Moodle, Canvas, Blackboard) to enable seamless access to learning paths, progress tracking, and mock interview sessions. This would allow educational institutions to incorporate job readiness preparation directly into their existing academic infrastructure.

2. Multi-Modal Candidate Evaluation: Future versions may incorporate multimodal analytics such as voice tone analysis, facial expression detection, and body-language cues during mock interviews to provide deeper insights into communication skills, confidence levels, and behavioural readiness.

3. Adaptive Recommendation Models :

Implementing reinforcement learning or adaptive algorithms can allow the system to refine job-role matching, interview questions, and learning path suggestions based on user behavior and continual feedback. This would improve accuracy and personalization over time.

4. Cross-Platform and Mobile App Support : Extending the platform to dedicated mobile applications on Android and iOS would offer users on-the-go resume uploads, instant job-role alerts, interview practice, and micro-learning modules. This would significantly improve accessibility and user engagement.

5. Industry Partnership Ecosystem : Collaborating with companies, recruiters, and job portals could enable direct job applications, recruiter dashboards, and real-time industry-driven interview simulations. This would enhance SkillsBridge's relevance and create a full employment ecosystem.

6. Ethical AI Framework and Compliance : Developing a comprehensive ethical framework to address AI fairness, bias prevention, and data privacy will ensure responsible use of user data. Adhering to regulations like GDPR and upcoming AI governance standards will increase system trustworthiness and transparency.

10 Bibliography

1. ReactJS Documentation – https://reactjs.org
2. Node.js Documentation – https://nodejs.org
3. MongoDB Manual – https://www.mongodb.com/docs/manual/
4. Scikit-learn – https://scikit-learn.org
5. TF-IDF Algorithm – Wikipedia
6. Cosine Similarity – Stanford NLP

References

[1] Guan, Zhihao, et al. "JobFormer: skill-aware job recommendation with semantic-enhanced transformer." *ACM Transactions on Knowledge Discovery from Data* 19.1 (2024): 1-20.

[2] Naz, A., Khan, H.U., Alesawi, S., Abouola, O.I., Daud, A. and Ramzan, M., 2024. AI knows you: deep learning model for prediction of extroversion personality trait. *IEEE Access*.

[3] Nurmalasari, R., 2021, October. Contribution of Job Readiness Application and Student Competency Towards Job Readiness and Job Opportunities in Industry. In *2021 7th International Conference on Electrical, Electronics and Information Engineering (ICEEIE)* (pp. 1-4). IEEE.

[4] Gemilang, H. and Lhaksmana, K.M., 2024, July. Prediction of Student Job Readiness Using MLP and XGBoost Method. In *2024 International Conference on Data Science and Its Applications (ICoDSA)* (pp. 248-253). IEEE.

[5] Jiranantanagorn, P., 2023, December. Enhancing Career Readiness: An Online Platform for Competency Assessment and Workplace Selection in Cooperative and Work-Integrated Education for Thai University Students. In *2023 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)* (pp. 1-6). IEEE.

[6] Ahmad, M.M., Srivastava, P., Bharti, V., Shamsi, M.F., Jain, A. and Vishnoi, A., 2024, June. Collaborative Mock Interview Platform. In *2024 OPJU International Technology Conference (OTCON) on Smart Computing for Innovation and Advancement in Industry 4.0* (pp. 1-5). IEEE.

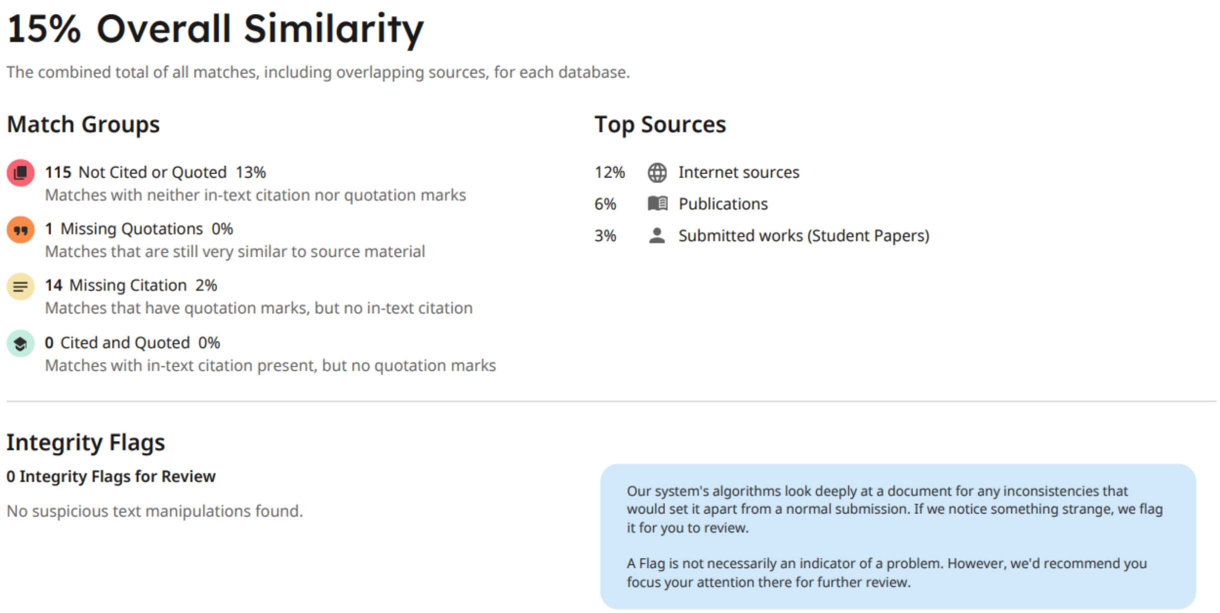
[7] Rao, G.S., Jaiganesh, M. and Parida, P.K., 2025, April. AI Powered Virtual Job Interview Simulator Using Natural Language Processing. In *2025 8th International Conference on Trends in Electronics and Informatics (ICOEI)* (pp. 1413-1421). IEEE.

[8] Mishra, A., Singh, S. and Jisha, R.C., 2024, June. AI-Powered Model for Intelligent Resume Recommendation and Feedback. In *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)* (pp. 1-6). IEEE.

[9] Sharma, T., Singh, A., Singh, S. and Gupta, G., 2025, June. AI-Powered Mock Interview Platform using Computer Vision, Natural Language Processing and Generative AI. In *2025 3rd International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS)* (pp. 1258-1263). IEEE.

[10] Jha, R., Paliwal, G. and Jha, B.K., 2025, March. AI-Powered Resume Builder: Enhancing Job Applications with Artificial Intelligence. In *2025 3rd International Conference on Disruptive Technologies (ICDT)* (pp. 1655-1660). IEEE.

Plagiarism Report



15% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Match Groups

- **115** Not Cited or Quoted 13%
  Matches with neither in-text citation nor quotation marks
- **1** Missing Quotations 0%
  Matches that are still very similar to source material
- **14** Missing Citation 2%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

12%	Internet sources
6%	Publications
3%	Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Fig. Plagiarism Report

Information of Project Group Members

Name: Omkar Santosh Ghodekar

1. Date of Birth: 14/08/2003
2. Gender: Male
3. Permanent Address: Khed, Pune.
4. E-Mail: omkarghodekar140803@gmail.com
5. Mobile/Contact No.: +917350978721
6. Placement Details: rtCamp
7. Paper Published: In Progress

Information of Project Group Members

Name: Utkarsh Sanjay Salunke

1. Date of Birth: 20/01/2004
2. Gender: Male
3. Permanent Address: A/P Lakshmi nest row bunglows-6, professor colony, Hanuman wadi, near racca's lawns, Nashik
4. E-Mail : utkarshssalunke20@gmail.com
5. Mobile/Contact No.: +918767250339
6. Placement Details: HSBC
7. Paper Published: In Progress

Information of Project Group Members

Name: Raj Mukesh Wagh

1. Date of Birth: 14/07/2004
2. Gender: Male
3. Permanent Address: A/P Vadner Bhairav, Tal-Chandwad, Dist-Pune
4. E-Mail: rajwagh1474@gmail.com
5. Mobile/Contact No.: +919529607875
6. Placement Details: In progress
7. Paper Published: In progress

Information of Project Group Members

Name: Sandhya Anil Dhaygude

1. Date of Birth: 15/04/2004
2. Gender: Female
3. Permanent Address: A/P Piliv, Tal- Malshiras, Dist- Solapur
4. E-Mail: dhaygudesandy0404@gmail.com
5. Mobile/Contact No.: +919834019857
6. Placement Details: In progress
7. Paper Published: In progress