OLLSCOIL TEICNEOLAÍOCHTA
BHAILE ÁTHA CLIATH
DUBLIN
TECHNOLOGICAL
UNIVERSITY DUBLIN

Final Year Project Proposal

TU856

AI Career Advisor

Aileen Coliban
C22304291

School of Computer Science
TU Dublin – City Campus

05/10/2025

Table of Contents

Contents

Declaration

I hereby declare that the work described in this dissertation is, except where otherwise stated, entirely my own work and has not been submitted as an exercise for a degree at this or any other university.

Signed:

*Aileen Coliban*____________________

Aileen Coliban

07/10/2025

Project Title

AI Career Advisor: AI-Driven Screening and Career Recommendation System

Summary

My project, AI Career Advisor, is an AI-driven web application that supports both job seekers and recruiters.

For jobseekers, the idea is that they can upload their CV and the system will analyse the CV with Natural Language Processing to extract key information such as key skills, education and work experience. It will then compare that to job descriptions from the job market and give them feedback, like what roles they might fit into, what skills they're missing and what areas they could improve in.

On the recruiter side of the applications, they will be able to upload job postings for the system to learn from and use as recommendations and also upload multiple CVs to be analysed. The AI will automatically summarize and visualise key information from each CV which are skills, experience, fit score, etc. in a dashboard with an option to shortlist that application.

The goal is to make recruitment and job seeking more transparent, ethical and data driven, by giving users useful insights into decision making.

The final deliverable will be a working prototype with a functional AI backend, a clean accessible UI and a working dashboard for data visualisation.

Background (and References)

When researching for this project, I looked into how companies like Eightfold AI handle CV screening and candidate recommendations for recruiters. Their approach to using AI for talent matching really stood out to me, especially how they extract and structure applicants' data from uploaded CVs. I also looked at HiringBranch, which uses interactive dashboards to help recruiters make decisions efficiently based on candidate performance and skill fit.

From a user perspective, I wanted to understand how AI could be used not just to help recruiters, but also help job seekers identify where they stand in the job market since it can be ambiguous and stressful for the most part. I came across several previous projects that explored AI dashboards, recommendation systems, and chatbot advisors, which gave me inspiration for the overall structure and user flow.

During my internship, I also worked on an AI CV Scanner project that focused on being a prompt generator where a Hiring Manager / Recruiter can upload multiple applicants CV where they can query information like "which applicant is more experienced", "which applicant is best for the job Requisition: SuccessFactors Consultant" etc. This helped me understand the real challenges recruiters face in shortlisting candidates and making data-driven decisions. That experience made me want to take the concept further and try make an application that could benefit both sides.

Proposed Approach

These are the main areas to my approach to this project:

1. Research
   I will start by doing in depth research on how existing AI screening tools like Eightfold and HiringBranch operate, focusing on their use of NLP and dashboards. I will also review past student projects on recommendation systems, dashboards, and chatbots to identify features or approaches I can build on while looking at different articles online on how it would be best to approach this project.

2. Requirements Gathering
   During the requirement gathering stage I want to define the core functional requirements for both job seekers and recruiters. This includes features like CV upload, job posting upload, skill extraction, similarity matching, and recruiter dashboard. I would also like to reach out to classmates who are actively applying to jobs and I would want to reach out to colleagues from my internship on what features they would think would be useful for future use.

3. Analysis and Design
   I will design the overall architecture and database schema for the system. The backend will use FastAPI for API endpoints, with FAISS as the vector store and SQLite or PostgreSQL for structure data. The frontend will be simple at first for testing purposes and later expanded to include dashboards for recruiters. I will also plan out how to integrate the NLP model to extract data from CVs and job descriptions, and how those embeddings will be compared.

4. Implementation
   Development will begin with setting up the backend, NLP logic, a simple frontend and connecting them first. I will use a python module to read CVs and extract text, then generate embeddings using OpenAI or Hugging Face models. The FAISS vector database will store these embeddings and perform similarity searches. After that, I will work on improving the UI, adding dashboards, making the different portals work, and refining the design to follow Universal Design standards.

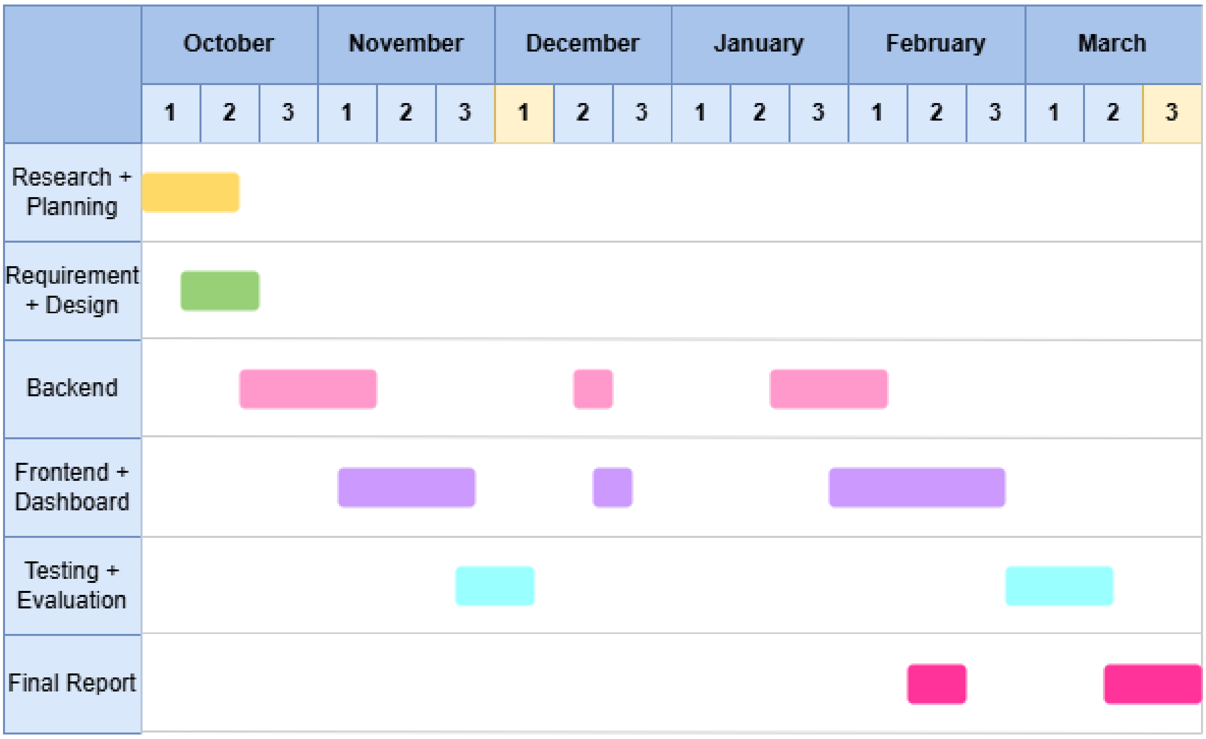
5. Testing and Evaluation
   Testing will be split into two stages or periods, one for the prototype and final testing later in the year. Although I will be following an Agile approach where testing happens throughout every stage of development, these two dedicated testing periods will focus specifically on analysing and improving the model and systems

overall performance. I will check the accuracy of the AI's skill extraction and similarity matching using sample data. I will also be testing the recruiter dashboard to display correct information. For evaluation, I will use different metrics such as precision, recall and F1-score to assess the model, while also gathering user feedback to measure usability and accessibility.

Deliverables

The final deliverables will include a fully working web application with separate portals for users and recruiters. It will have an NLP model that extracts key information from CVs and job descriptions, supported by a FAISS vector database for accurate similarity matching. The recruiter portal will have a dashboard that displays candidate summaries, match scores, and insights in a clear and concise format. The front-end will be designed based on the Universal Design principles. A full final report including technical documentation and research will be done alongside a presentation outlining the project.

Project Schedule



The month is split up in thirds in this gantt chart, the first third of the month, middle and end of the month.

The first two thirds of October will be focused on finalising the project proposal, background research and confirming which technologies should be used which will be the research and planning. Alongside research and planning, I will be working on the requirements and design of the system which will entail creating an architecture diagram, database schema and mock-ups for the user and recruiter portals.

With backend and frontend developments the work will be interlinked together so I can see everything in the backend is working properly. So, I will start off by making a simple frontend to test functionality of the backend, then I will make it more detailed once the project picks up steam.

The yellow highlighted thirds indicate important dates for example, the first third of December will indicate that a working prototype and the interim report will have to be completed by then.

With Testing and evaluation, there will be two periods. The first period will be testing and evaluating to make sure the prototype is working correctly and the second period will be testing and making sure the final product is up to standard.

Finally, there is time sectioned for the final report documentation and completing the presentation alongside the report.

Technical Requirements

The project will be developed using Python, FastAPI, and FAISS for the backend. The NLP model will be based on OpenAI or Hugging Face embeddings, and PyMuPDF will be used for reading PDF CVs. The frontend will be built with Streamlit, HTML, CSS and JavaScript. The database will use SQLite or PostgreSQL for scalability. All development will be done on my local machine with GitHub. For deployment, Render or Heroku will possibly be used.

Conclusion

Summary of the project idea

The AI Career Advisor project aims to use AI and NLP to make the hiring process fairer and more transparent. It helps job seekers see where they stand in the market while giving recruiters a faster, data-driven way to review candidates. The project will incorporate a portal system, a side for job-seekers and another side for recruiters. The recruiters will have access to a dashboard of potential applicants. The job-seekers will have access to their metrics, where to improve their metrics and recommendations for different jobs based on their CV.

References

A Guide to Artificial Intelligence Resume Screening. (n.d.). Retrieved from HiringBranch: https://www.hiringbranch.com/blog/artificial-intelligence-resume-screening

Agentic AI Talent Platform for Smarter Hiring. (n.d.). Retrieved from SeekOut: https://www.seekout.com/

AI Career Coach. (n.d.). Retrieved from Sapia: https://sapia.ai/phai/

AI talent acquisition & recruiting platform . (n.d.). Retrieved from Eightfold: https://eightfold.ai/

COACH: AI-powered Career Coach . (n.d.). Retrieved from COACH: https://www.aicareercoach.org/

Gomes, P. (n.d.). *Generative AI Project: CV Analyzer*. Retrieved from Medium: ttps://patotricks15.medium.com/generative-ai-project-python-langchain-cv-analyzer-c56f07c7f1f0

Sonawale, Y. (n.d.). *How to Build an LLM-Based Resume Analyzer*. Retrieved from mercity: https://www.mercity.ai/blog-post/build-an-llm-based-resume-analyzer?utm_source=chatgpt.com

Appendix A: First Project Review

Title: NLPurchase – eCommerce Chatbot

Student: Stephanie Finn

Description (brief):
This project was based on creating an AI chatbot that could help users browse and buy products directly through Facebook Messenger. It used Wit.ai to process natural language so it could understand user queries and respond like a real assistant. It has a backend built in Node.js with a MongoDB database that handled all the products, conversations, and recommendations. The bot could suggest products, keep track of preferences and guide users through the shopping process in chat form.

What is complex in this project:
The complexity was connecting multiple systems together such as the NLP model, database and Facebook's APIs, requiring managing real time data and keeping responses fast and accurate. Natural language can be unpredictable alongside getting the chatbot to properly understand human input and respond correctly. Designing intent recognition and entity extraction tool a lot of fine tuning, making it the main complexity of this project.

What technical architecture was used:
The technical architecture consisted of Node.js for the backend, MongoDB for storing data, Wit.ai handled all the NLP implementation, and the Facebook Messenger APIs so users could chat to it directly.

Explain key strengths and weaknesses of this project, as you see it.
One of the key strengths of this project was the use of NLP to create a natural conversational environment.
However, the chatbot depended on predefined intents, which means it wasn't as flexible or adaptive as newer NLP models. It was also limited to Facebook Messenger.
Analysing this specific projects strengths and weaknesses, it is relevant to my project because it shows how NLP can be used to interpret user input and give meaningful, human like response. I liked how it handled user interaction and guided people through making decisions.

Appendix B: Second Project Review

Title: Auto-Efficiency – Machine Learning for Predictive Analysis and Recommender System

Student: Glory Pierce Eguare

Description (brief):
This project was based on building a machine learning model that predicts a car's fuel efficiency and recommends similar vehicles based on user preferences. The recommender system compared car specs like horsepower, engine size, and weight to find the most similar vehicles for the user. It combined predictive analytics with recommendation logic and included a dashboard where users could view the model's predications and recommendations.

What is complex in this project:
The complexity is within the training and optimising the prediction model and then connecting it to a user facing dashboard that updates dynamically. Managing the data between the ML model, backend API, and frontend dashboard required good system design and consistent testing.

What technical architecture was used:
It had a Flask backend, a TensorFlow/Keras model for prediction, and a React/Ionic frontend for the dashboard. The project followed an Agile development process.

Explain key strengths and weaknesses of this project, as you see it.
The key strength was how it combined machine learning with an interactive dashboard to make the data easy to understand. The connection between the predictive model, backend and frontend worked well. It was also well structured and modular, which made it easy to follow and potentially scale for bigger demands. One weakness was the the dataset might have limited how well the model could generalise its predictions.
This project is close to what I want to do with the AI Career Advisor. I liked how it linked an AI model to a live dashboard and turned predictions into something users could easily interpret. My system will do something similar, extracting and matching data using AI and visualising it clearly for recruiters and job seekers.

Appendix C: Prompts Used with ChatGPT

Prompt 1:
I am student doing Computer Science. I am in my final year starting to do my final year project and I need some ideas. Here are the modules I have done throughout the 4 years of my degree:
Year 1: Algorithm and Design and Problem Operating Systems 1 Computer Architecture and Technology Microprocessor Systems Data Exploration Mathematics 1 Communications for Computer Tech Web Development Information Technology Fundamentals Program Design Programming in C
Year 2: Algorithms and Data Structures Human Computer Interaction Data Communications Legal and Professional Issues Mathematics 2 Web Development 2 Software Engineering 1 Software Engineering 2 Databases 1 Object Oriented Programming in Java and Python Operating Systems 2
Year 3: Mobile Software Development Intro to Artificial Intelligence Cloud Computing Client Serving Programming Databases 2 Software Engineering 3 Year 4: Advanced Security 1 Advanced Databases Machine Learning Forensics
I have also completed 2 internships, one in optum as a data scientist where I made an application that uses predictive analytics and machine learning to create a loan calculator and the second one I was at SAP CoE as a business processes consultant, where I completed a couple of projects stemming from and AI CV scanner which allows recruiters to input pdfs of applicants and query them through an AI and compare CVs, to a Gamify Networking application that allows interns to network freely. Based on the information I have given to you above what ideas would be good for a final year project.

Prompt 2:
What are some references and tools I can research for my AI Career Advisor project?

Prompt 3:
What would be a suitable tech stack?

Prompt 4:
For the background I focused on looking at how Eightfold does their CV screening for recruiters, I also looked at hiringbranch for examples of how dashboards are used to visualise the data during screening so decision making can be quick, I am looking at previous final year projects from the years before what type of projects should I focus on for reference?

Prompt 5:
What would my project schedule look like so I can have a working prototype by December 1st and a complete project by march?

Prompt 6:
What ways can I test an AI model?

Appendix D: