# Project Proposal

## Project Title

**Job Matching Recommender System API**

## Product Brief

Job Matching Recommender System is an innovative API service designed to address the challenges of our Coop students in finding jobs on the CSM platform, and to improve the efficiency of students in finding suitable positions. By leveraging advanced knowledge graph technology and graph algorithms, this system can deeply analyses the content of students' CVs, intelligently match them with existing job data, and recommend jobs that best match their skills and experience.

## Motivation

The project is motivated by two main goals to tackle distinct challenges faced by Coop students in the job market.

1. The first goal is to improve the current CSM platforms, which primarily use basic keyword searches that often don't align job titles with their actual roles and responsibilities. This misalignment makes it challenging for students to find jobs that truly suit their skills and interests. By analyzing students' CVs directly, the project intends to enhance the job matching process, ensuring students relate to positions that genuinely match their qualifications.
2. The second goal is to assist students in navigating their career paths. Many students are unsure about their future careers and have difficulty identifying suitable job opportunities. The project plans to use the information in students' CVs to offer personalized career guidance, matching their skills and experiences with relevant job openings. This approach aims not only to help students find fitting jobs but also to encourage them to define and pursue their career goals more confidently.

## Solution Workflow

1. Data Collection and Processing: Collect resume and job descriptions, clean and preprocess them to form keyword lists.  
2. Knowledge Graph Construction: Build a knowledge graph using Neo4jAura, containing job id, prefer program, skill nodes and their relationships.  
3.Match Algorithm Development: Apply community detection algorithms to divide job communities, and node similarity algorithm to find best match job list.  
4. Backend Logic and API Development:  
 - Develop a RESTful API that allows users to submit skills and other relevant information and returns a list of recommended jobs.  
 - Implement interaction with the Neo4j database and the core algorithm of the recommendation logic.  
5. System Deployment:  
 - Deploy the API service on a cloud platform to ensure high availability and scalability.  
 - Set up monitoring and logging to ensure the stable operation of the system.

## Technology Stack

- Database: Neo4j Aura, CSM Replica Database, Google Cloud Storage  
- Backend Development: Python (Flask/FastAPI)  
- Information Extraction: HuggingFace, Llama2, Spacy/or similar NLP library, PyMuPDF/or similar pdf read library  
- API Documentation: OpenAPI

## Milestones and Timeline

- Week 1: Data preparation and knowledge graph construction   
- Week 2: API development and recommendation algorithm implementation  
- Week 3: System testing, documentation, and deployment

## Expected Outcomes

- A knowledge graph-based job recommendation API that can quickly return a list of matching jobs based on user-provided information.  
- API documentation and developer guides.  
- System Demo and operation guidance documentation.

## Conclusion

This project, by developing a knowledge graph-based job recommendation API, provides job seekers with a more intelligent and efficient job matching service, helping to improve the accuracy of job searches and user satisfaction.