

Final Year Project Proposal - Web Service for Project Collaborators

Ravi Woods - BEng EIE - rbw14@ic.ac.uk

The project aims to be similar to an online dating profile, but instead is used for finding collaborators in personal projects. The user journey would go as follows:

- User initially creates an account and uploads their CV. The system finds their skillset from this.
- They then specify what projects they want to be involved in (e.g: Role, Length of Time, Location)
- The system then recommends people, or projects. These would have percentage matches.
- In addition, the user can browse all projects, and can create their own project listing.
- These project listings would need the role/skills of collaborators.

Background

The project was discussed with Jing Sheng Pang, from ICAH. They have always had the problem of matching collaborators in the past, with proposed solutions ranging from an email system, to a dedicated online forum. However, user takeup was always the issue and, in the end, they have always needed to manage and match individuals themselves. He welcomed using ICAH as the problem space, with a view for ICAH to use my final solution in October 2017.

Proposed Method

The first challenge would be to create a user interface for the system. Prototypes would be created in HTML using a Web Framework (such as JQuery Mobile or Bootstrap). This section would need knowledge from User-centred Information Systems (EE1–12).

I would then create a basic matching algorithm, which would have a weighting based on each characteristics (for example distance between collaborators), generating a percentage match. This section would need knowledge from Algorithms and Complexity (EE2–10C), and Object-oriented Software Engineering (EE2–12).

Then, I would create a Machine Learning system, in Python, which would help improve the matching algorithm over time. I would create training data from the initial basic matching algorithm. This section would need knowledge of from Machine Learning (EE3–23)

From here, I would use Flask to interface between the HTML frontend and the Python backend. In addition, I would use a DBaaS, such as Firebase - both to store data, and password information. This section would need knowledge from Advanced Databases (CO312).

Timetable

By 24th March

Confirmed dynamic interfaces for the system, in HTML.

25th March to 1st May

Literature Review, especially with regards to the Machine Learning algorithm

1st to 5th May

Creation of the basic matching algorithm.

8th to 17th May

Implementation and testing of the Machine Learning algorithm.

18th to 23rd May

Implementation of the Database backend, along with secure user login.

24th May to 2nd June

Interfacing of the HTML front-end and the Python backend using Flask, creating a full implementation of the system.

5th to 9th June

Buffer time, in case of delays.

12th to 29th June

Production of the final report and poster

Practical Details

For supervision, I have spoken with Mrs Esther Perea, who is happy to supervise me if the project goes ahead.

For pricing, the only necessary cost would be hosting, both of the website and of the database. The site hosting would be [£4 per month](#) . The database hosting would use [Firebase's free plan](#). So, this would be around £8.