# Major/Minor Project 2012-13 • Student Project Suggestion

We will soon be sorting out supervisors and project topics for the Major/Minor projects (CS39440, CC39440, CS39540, CS39620). Lecturers and other members of staff will make a list of project suggestions, which will be released as soon as possible.

We also allow students to suggest project topics. If you would like to suggest a topic for your project, please complete this form and return it to Neil Taylor ([nst@aber.ac.uk](mailto:nst@aber.ac.uk)). He will identify possible project supervisors that you can talk to about the project.

*Making a project suggestion does not commit you to the project, or guarantee that we think that the project is suitable for the module. It is a starting point for a conversation about your idea.*

For other questions about the Major/Minor Project, please see the website at: <http://www.aber.ac.uk/~dcswww/Dept/Teaching/CourseNotes/current/CS39440/>

This form is for any student taking the modules CS39440, CC39440, CS39540, CS39620.

## Student Details

*Please enter the following details.*

|  |  |
| --- | --- |
| Name | Emanuil Evgeniev Tolev |
| Email | eet9@aber.ac.uk |
| Degree Scheme | G401 (Computer Science with Industrial Placement Year) |

## Suggested Project Title/Topic

*Enter a brief (up to 100 characters) title or topic area.*

Sharing and mapping out funding opportunities for science

## Project Description

A web application which presents information about current and past scientific funding opportunities in an easy-to-comprehend way.

The pain point is that said opportunities are buried deep within (usually) not-very-well designed websites and, worse, are scattered all over online and offline sources. For example, JISC are very nice and would actually fund something like \*this idea\*, I’ve been told, but undergraduates like me do NOT KNOW they exist; also, a commercial company might offer funding at a conference or other event, but might not actually know how to reach more scientists. **This web. app. would aim to simplify presenting and finding scientific funding.**

With that last sentence in mind, we have two types of audience to think about:

I. submitters – people who know about a scientific funding opportunity and want to tell us about it.

II. searchers – people who DON’T know where to get money for their science, be it an MRes candidate or a senior lecturer looking for new grants for visual processing

The general design (architecture) would probably follow this pattern:

1. Input -> 2. Storage -> 3. Presentation / visualisation

#1: Input. Gathering information about scientific funding opportunities. Here we are concerned with audience type I – submitters. Examples would include grant calls, challenges, statements of interest. Sources for these would be institutions like JISC, the 7 UK funding councils, the European Commission (e.g. the FP series of research programmes), commercial companies.

I thought that gathering might happen in one of three ways:

a) “scholar” approach – one or more people study scientific funding opportunities and preserve the results of their search as they go along (e.g. a student like myself researches some MRes funding opportunities, but instead of keeping the results to themselves, they post them as they find them and bingo, many more people can now reuse that knowledge)

b) “crowdsourcing” approach – ask funded MSc, MRes, PhD, RA or tenured researchers where they got / are getting their research money from. Here the focus would be on a great interface and making the most out of fewer pieces of data (as these kinds of people wouldn’t have much time to spare)

c) “automated understanding” approach – crawling or hand-screening grant call documents (HTML, PDF, anything modern) and policy documents and text mining them to build a meaningful / relatively precise model of the information contained within. Automating the understanding of difficult to find and comprehend documents is beyond my time constraints, but will provide for a nice “suggested future work” section. Web app. should be designed with extendibility in mind.

**“Input” will be extendable.** I will take care of a) and maybe b) if there’s time, but somebody else will be able to come along and add c) as an input method later!

#2: storage. Either a conventional SQL DB or (more likely) a search index like Apache Solr or (personal preference ->) elasticsearch. elasticsearch is basically a search engine server – you throw data at it and it indexes all of it. It can then fulfill full-text exact or fuzzy queries extremely quickly (e.g. search within 20 million non-trivial JSON records with several fields for 3.5 seconds, return JSON). **Storage will be fast and intelligent.** Implementing fuzzy full-text search on top of a conventional SQL DB is worth 2 dissertations of its own! There are also advantages to getting ready JSON out of the storage (it simplifies coding up API endpoints or data visualisations using Javascript libraries).

#3: Presentation. Say we’ve got 500 grants in the search index. Now, our audience type II, searchers, wants some scientific funding. We need to be able to analyse and present the data we have in a clear, simple and useful way. This is, of course, impossible to do if we just have one approach to presenting the data (e.g. dump it out in a listing on an HTML page). Therefore we need to consider different angles of presenting the data – different Points of View. One person might want to know about funding for speech recognition and synthesis, but another one might want all grants below £20k (probably not scientists, but journalists and open data enthusiasts like me will). They might want to make infographics out of it. I would personally implement a good, browseable and searchable list of all funding opportunities and a few basic visualisations (like <http://wheredoesmymoneygo.org/bubbletree-map.html> - the software which generates this pretty good presentation on the fly is Free Software).

**The web app. should be extendable in the sense of presentation and visualization. Besides that, an API should be provided to those wishing to mash up or visualise the data in their own way.**

Taking the conclusions of the three sections above, we end up with a web application that at the very least: stores submitted information about scientific funding opportunities and present it back to the users on a web page, while stressing good UX (browse/search). Both submissions and access will have API endpoints in addition to the web interface to facilitate even more opportunities for sharing the information. The design will allow or evolve to allow for sufficient flexibility so that most major funding opportunities can not only be listed out, but summarised – a “map” of scientific funding.

The project will be **coded using** [front-end] HTML5/CSS3/Javascript (+jQuery), [storage] indexing engine like elasticsearch and [back-end] Python (Flask or Django) and licenced under the **MIT licence (or under no licence**, copyhearted instead of copyrighted). For the answer to “why these”, see the “Any previous investigations” section. The project will use other pieces of software but a significant amount of non-trivial code will need to be authored.

I am open to suggestions and of course, rejection. Any of the above could be changed in substance or scope as advised.

## Relevance of the project to your degree scheme

*Your project needs to relate to your degree scheme. Some schemes have more flexibility than others, e.g. G400 Computer Science, but there is still plenty of scope to choose a project that will be interesting and suitably challenging for your scheme. Briefly explain the particular relevance of this idea to your scheme and/or your future work plans.*

A web application with a non-trivial portion to process data submissions and present the data fits nicely within the remit of G401, a version of G400. Further upgrade of my web development skills would be very welcome as most of my CV consists of making web applications and just a few offline projects. Potential web development projects (freelance, or with a company, might also benefit from this). Also see “Work with a company” section.

## Work with a company

*Does this project relate to work with a company, e.g. your Industrial Year Employer? If so, what is the company and what is your relation to them? What discussions have you had with the company? What discussions have you had with them about the Intellectual Property Rights in the outputs of the project?*

Cottage Labs (<http://cottagelabs.com/about>) is a Limited Liability Partnership of several developers who work on a freelance basis within the Higher Education software sector (e.g. research information management). They are also very keen participants in the Open Knowledge / Science movement which aims to make access to science free for the reader & more (<http://okfn.org/about/#our-vision>). I met them during my Industrial Year and developed an idea they had in my free time using similar technology to the one I’ve proposed for this project and they invited me to do further work with them. Their work is usually based on something which has proven itself as a piece of software or technology and is relatively easy to fit within the requirements of academic institutions or JISC – new ideas might be very good and become easily fundable, but getting them to a working prototype is time spent not earning the daily freelancer bread (they do do innovative work, obviously – usually within current projects). This project would be a way to try to contribute an unproven idea to the Open Knowledge movement and see if it might be taken up by people who provide scientific funding at some point.

## Any previous investigations

*Have you investigated any aspects of this project so far? For example, have you been reading about the topic or making some prototypes? If you have, please describe what you have done.*

Used a very similar set of technologies (Python, front-end web dev incl. jQuery) and got acquainted with the flexibility that elasticsearch offers as a data storage back-end during prev. project: <https://github.com/CottageLabs/idfind/>. I had a look at the Django framework for Python web apps (IDFind project was done using the Flask framework) and would rather use that – simply to enhance my skillset a little bit (freelancing can be a bit risky so I’d rather diversify).

Talked to other Cottage Labs developers about it, the idea was generally perceived to be a worthwhile one if done well (i.e. proposed outcome of sharing funding info and mapping out funding is useful to science stakeholders).

The usual initial Google searches to see if anybody has already done this (e.g. “mapping out funding for science”). There \*are\* lists of grants on the web, but these are usually provided by the actual institution doing the funding (e.g. latest iteration of the European Research Framework) – I couldn’t find anything which federated information from multiple sources of funding together.

- END -