

Sharing and mapping out scholarly funding opportunities

Report Name	Progress Report
Author (User Id)	Emanuil Tolev (eet9)
Supervisor (User Id)	Reyer Zwiggelaar (rrz)
Module	CS39440
Date	November 18, 2012
Revision	1.0
Status	Release
Word Count	4266

Contents

1	Project Summary	2
1.1	Scholarly funding opportunities	2
1.1.1	Funders	2
1.1.2	The value of Openness in scholarship and elsewhere	2
1.1.3	Scholars	3
1.2	The project	3
1.3	Existing works	3
2	Current Progress	4
2.1	Related literature and technologies	4
2.1.1	The faceted search concept	4
2.1.2	The data representation format	4
2.2	Related projects and technologies	5
2.2.1	Human users	5
2.2.2	Search	5
2.2.3	Exposing the data to other software projects	5
2.2.4	Visualisation technologies	6
2.2.5	Working with the data	6
2.3	Understanding the technologies involved	6
2.3.1	Back-end / front-end integration	6
2.3.2	Visualisation	6
2.3.3	New functionality	7
2.4	System design	7
2.5	Customer interaction - ensuring value	7
2.5.1	Potential audience	7
2.5.2	Focusing on certain audience groups	8
2.5.3	Project constraints due to audience characteristics	8
2.5.4	Audience involvement due to audience characteristics	8
2.5.5	Measuring success	9
2.6	Potential issues and risks	9
2.6.1	Potential legal issues	9
2.6.2	Technical issues	9
2.7	Planning progress	9
2.7.1	Project management tools	9
3	Planning	10
3.1	Development methodology	10
3.1.1	Considered methodologies	10
3.1.2	Chosen methodology	10
3.2	Current project timeline	12
3.2.1	Time budget breakdown	13
3.3	Demonstrating value	13
3.3.1	Mid-project demonstration	14
3.3.2	Final demonstration	14
	Annotated Bibliography	15

1 Project Summary

1.1 Scholarly funding opportunities

This project deals with an important bit of the infrastructure of scholarship - namely, the money it relies upon. "Scholarship" in this case (and throughout this document) refers to the human endeavours in the fields of the Humanities (e.g. arts), Social Sciences (e.g. law) and Science (e.g. biology, computer science).

The problem is that scholarly funding has a very peculiar system which has evolved over time from a series of ad-hoc changes to the way money is distributed in society. There is no easy way to track money - where it gets allocated to, what fields (or even topics) within scholarship get funded more than others. This has resulted in several problems:

1.1.1 Funders

There is no easy way for people who wish to fund scholarly endeavours to advertise their funding to a lot of scholars at once, getting a higher chance of the "perfect" researcher/problem fit.

Certain ad-hoc channels have evolved around this problem - for example, scholars interested in a funder's sphere can subscribe to their e-mail list. There are problems with this - the most obvious one is scholars having to subscribe to all the funders they might be interested in. While this sort of works for national funding bodies like the UK Research Councils, there are many more private funders - various non-profit and/or Non-Governmental Organisations and even private companies. A cross-disciplinary funding opportunity would have to be advertised across multiple channels, which would take time/effort on the part of funders therefore making such opportunities costlier to set up.

It also fails fantastically when it comes to the globalisation of scholarship. There are only two ways to get to know about funds coming from the other side of the world:

1. directly: be subscribed to that particular funder's news (the number of funders has not been estimated globally to the best of the author's knowledge, not to mention the huge variety of forms which news feeds take - from RSS through e-mail to just publishing on a website somewhere).
2. indirectly: hear about the opportunity from others. "Others" could mean colleagues - research development officers, other scholars - but these people have to be directly or indirectly related to the funding source. The funder will only reach a very small subset of the world's scholars directly, however, and this is crucial for a network model.

The other currently possible meaning of "Others" is commercial companies who collect scholarly funding information, package it into databases with a front-end and sell it to research institutions. While the author is not against making a profit out of information, there has been overwhelming evidence in the past two decades that commercially exploiting information *by virtue of hoarding it and then restricting access to it* is not a good strategy for anybody but the hoarder.

1.1.2 The value of Openness in scholarship and elsewhere

The Free Software, Open Source and more recently, Open Knowledge (which Open Access is a part of) movements have all demonstrated that the value of information can be greatly multiplied simply by having more people access it, reuse it and even be creative with it (e.g. visualise or summarise). The Open Knowledge Foundation argues this [26]. Seminal works like "The Cathedral and the Bazaar" [13] have also argued this.

Governments seem to have grasped the benefits as well. The UK Government has mandated that all research funded with public money (which is a lot in the UK - almost everything through the seven Research Councils) should be published as Open Access by 2014 [18]. It has also funded initiatives such as Open UK governmental data in the form of data.gov.uk [43] and the recent Finch report [2] [45].

Private organisations are not far behind and may in fact be leading the way - one of the largest funders of science in the UK and around the world, the Wellcome Trust, has had a progressive and strict Open Access policy since 2006 [9].

1.1.3 Scholars

Openness is recognised as beneficial - and many scholars are supposed to be able to benefit from having access to all this scholarship for free instead of having the libraries pay extortionate amounts to academic publishers. The very essence of scholarship is building on what has been discovered before.

However, while previous knowledge is a core requirement of science, sustenance is a very core human requirement, and in the modern world, this often translates to salaries *and* (sometimes “or”) grant money, at least for scholars. Why should *this* information be commercially exploited for the benefit of a few corporations when it could be used to better (globally) connect the minds to the money, to put it bluntly?

There is another point - information about funding opportunities may be available but may be too generic or hidden beyond layers and layers of website navigation, reflecting the mental model of the funding organisation instead of the mental model of the researcher (RCUK are a case in point [32]). Thus, the goal of this project from the perspective of scholars can be summarised as “bringing it together”.

1.2 The project

This project is about making an open-source web application named “FundFind” which lets stakeholders (just the scholars at first) share information about funding opportunities under open terms (the Open Definition defines “open” well [23]).

Follow-up work may increase this scope to include funding organisations or governments - they will already be able to submit funding information directly to a centrally hosted instance of the software produced by this project, but they will probably have special requirements related to this.

The project does include functionality to allow scholars to access the information, and allow developers / journalists / analysts and other stakeholders to analyse and mash up the information.

1.3 Existing works

There is no single piece or combination of pieces of software which enables involved stakeholders to do what this project would allow them to do upon completion, to the best of the author’s knowledge.

However, this project fits well within a current framework of Open Knowledge-related projects, a nice selection of which are hosted and developed by the Open Knowledge Foundation [25] [24].

This project will also require data to be useful. While one of its core aims is to enable crowdsourcing of scholarly funding information, it should also try to make use and “digest” existing information. This could mean looking at current funding opportunities [31] [1] [4] [12] or at loading information about past opportunities, such as what the Australian National Health and Medical Research Council has funded over the past 23 years [22]. Of course, part

of the point of this project is that it is not easy to get such information in a nice universally readable (e.g. machine-readable) format. Research about useful data would be an optional objective subject to good feedback on the core software output and enough budget (of time).

2 Current Progress

2.1 Related literature and technologies

2.1.1 The faceted search concept

Faceted search [55] was chosen as a core data consumption paradigm - the design of both the human-accessible HTML interface and a machine-friendly API would follow this concept.

“Facets” can be thought of as the various “columns” of a dataset if it was flattened out into a CSV file or a SQL database table. Each record in the dataset has a some value in each column, and blanks are allowed. The analysis / aggregation of the unique values in a column is called a “facet”. A (simplified) example of what the data for this project might look like can be seen in Table 1.

Opportunity name	Subject	Country	Funder
Shadow manipulation in automatic understanding of 3D representations of real-world video footage	Computer Science	UK	EPSRC
Say hello! A global qualitative study of greeting phrases	Psychology	USA	APA
Fast-Fourier and the Three Musketeers	Computer Science	UK	BBC

Table 1: Example dataset for the purpose of illustrating what facets are

If we choose to construct facets from the Subject and Country fields, they would look like the ones illustrated by Table 2.

Facet name	Value	Number of items
Subject	Computer Science	2
	Psychology	1
Country	UK	2
	USA	1

Table 2: Example facets

Faceted search is often used for semi-structured data when producing an exhaustive classification is difficult since the data available in different records varies and there are many ways to classify it. One example which is very relevant to this project is library catalogues, where the artefacts the user could be looking for vary from journal articles through manuscripts to books.

This approach has been observed in action by the author and found to be quite powerful when dealing with large datasets of complex data.

2.1.2 The data representation format

JSON has been chosen as the internal data format, as well as the data format which the API will serve (XML may be added to the API at the end of the project subject to satisfactory coding progress). There are three main reasons for this choice:

1. Interoperability. While each technology chosen for this project supports a variety of formats, the main format of choice of each one is JSON. The datastore stores JSON, the Javascript visualisation libraries use JSON, the back-end code which connects to the datastore is written in Python which handles JSON very well. Furthermore, a variety of Open Knowledge-related projects are still trying to stabilise their main functionality and will thus only support JSON - this project tries to work within the established infrastructure.
2. Simplicity and suitability for large semi-structured datasets (i.e. the reason for #1). It is *easily* human-readable [44] and has proven itself as an almost perfect mapping to the Python programming language's "dictionary" and "list" data types as well as standing for "Javascript Object Notation". These have, in turn, proven to be flexible, powerful and fast to use in order to make up more complex structures. The author has personally observed a multitude of Open Data projects use it for this reason.
3. Following on from #2, Python and Javascript are the main programming languages used by this project.

2.2 Related projects and technologies

This project is all about putting some data into a system and then exposing it to the world as well as presenting analysis of the data such as graphical visualisations. The potential users are both humans and machines (other software). Previous work in Computer Science has resulted in certain "best practice" guidelines for the interaction with both types of users.

2.2.1 Human users

Usability and accessibility are important principles to observe when crafting web pages meant for people. To this end, the Twitter Bootstrap [42] and the jQuery [51] web user interface libraries for changes package a great amount of user experience research into easily reusable components. They are used by Facetview and will be used extensively by this project for all its web pages.

2.2.2 Search

This project places emphasis on search and finding information - a powerful search back-end needs to be combined with an easy-to-use front-end to achieve this.

- elasticsearch [48] as its data store
- code from another Open Knowledge project as an example of how to use elasticsearch with Python (IDFind [11]). The author is currently the lead developer of IDFind.
- Facetview [8] will be used on the project's search / catalogue page. It exposes the data store's powerful search and data analysis (faceting) capabilities to the end user.

2.2.3 Exposing the data to other software projects

The IDFind project mentioned as the basis of the back-end has a JSON RESTful API, albeit not a very good one. IDFind's code will provide the basic setup for getting a JSON API working and will be enhanced to fully conform to RESTful principles and return data in a sensible structure. Flask [49] is the Python web framework which enables IDFind's API abilities.

2.2.4 Visualisation technologies

Currently, the scope of this project includes one graphical visualisation in order to explore the issues around visualising scholarly funding data and recording any difficulties encountered so that future work can start on a more informed ground.

Related projects are the D3 [47] Javascript visualisation library and the bubbletree [27] library which powers the main WhereDoesMyMoneyGo [29] UK government spending visualisation page.

Google Maps [16] is being considered if the project visualisation is decided to be a map. OpenStreetMap [54] provides an alternative with an API if this does not work out, however. The author has also observed abstract maps of the world (i.e. simple contours, not satellite imagery) in a few Open Knowledge projects. These visualise data nicely but further work is needed in order to establish the name of the project which enables this sort of visualisation.

2.2.5 Working with the data

Some consideration also needs to be given to tools which enable work with semistructured, large datasets (beyond spreadsheet-viewing tools such as LibreOffice Calc [39]) since this seems to be the norm for data related to academia (and generally the norm for comprehensive Open Data, in the author's experience).

OpenRefine [52] (used to be known as Google Refine [17]) is a flexible tool for analysing and making en-masse changes to semi-structured data. The author has experience in using it on a dataset with 15'000 items and 67 columns.

Recline.js [28] is, apparently, a library which allows you to build web applications which manipulate data. This could be useful if the project's scope was changed to include a way to explore and work with the data it contains (and could still be useful as the basis for the funding opportunity submission page, but this needs further investigation).

2.3 Understanding the technologies involved

2.3.1 Back-end / front-end integration

§2.2 lists IDFind as an example of how the back-end technologies could be integrated together with front-end ones such as jQuery and Bootstrap. However, IDFind does not make use of the Facetview project which exposes the powerful search capabilities of elasticsearch to the end user.

The code for this project has been created as a copy of the IDFind codebase and will now need to start using Facetview for its catalogue/search page. This single integration will actually fulfill the core requirements of the project (data collection and exposure). A lot of the prerequisites for starting the implementation of a RESTful API would have also been fulfilled. Work on this integration has started.

2.3.2 Visualisation

Further work will need to be done in order to understand how the visualisation libraries enumerated in §2.2 are to be used with this project's data. This will involve setting up the libraries on their own with a little bit of test data (working example stage), then using that stand-alone example on the main project dataset (visualisation test stage) and finally, integrating the visualisation into the main codebase (final stage).

2.3.3 New functionality

Finally, some of the user stories feature requirements which IDFind never had to fulfill, but which the underlying Flask [49] web framework can help with (such as users having *useful* profiles with “per-user” data). Work (including exploratory work) on such features will be completed in the weeks allocated to the respective stories.

2.4 System design

As mentioned in §2.2.2, the initial FundFind codebase is based on the IDFind project. The relevant classes which have been kept in the transition are displayed on Figure 1. The design is typical of simple RESTful web applications. The main way to extend it is to add more modules and classes such as “Importer” which get used by particular “Web” functions which in turn respond to specific HTTP routes (e.g. GET <fundfind>/funding/3 or <fundfind>/funders/10).

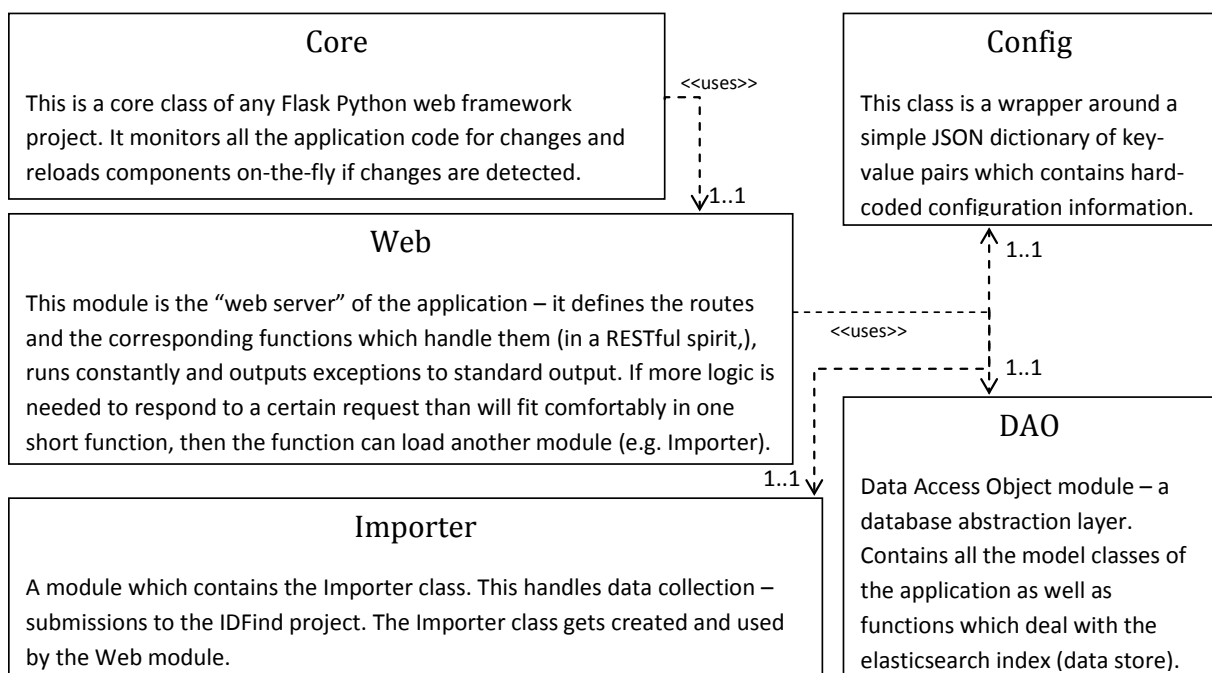


Figure 1: UML diagram of relevant IDFind classes which will become FundFind’s classes as more logic is added so that it can actually be called a different project.

2.5 Customer interaction - ensuring value

One of the most important aspects to consider when planning a project, selecting a software development methodology and so on is the audience of potential users (“customers”, “stakeholders”).

2.5.1 Potential audience

The potential audience of this project was discovered to be quite varied:

1. People looking for funding:
 - accomplished scholars (i.e. academics - lecturers, professors)
 - more “junior” academics such as Postgraduate candidates / students and Research Associates

- research development officers (professionals who find more funding opportunities for scholars)
2. People looking to advertise funding opportunities. Funding and Programme managers at various institutions such as:
 - the Research Councils in the UK
 - JISC (digital infrastructure in education/research) [19]
 - various charitable Foundations, Institutes, Endowments and so on which fund scholarly activities (e.g. the Wellcome Trust [56], Shuttleworth Foundation [37], Knight Foundation [20]). Anybody who wants to spend money on scholarship.
 3. There is a group which has markedly different needs from the first two groups - people interested in analysing how scholarly money is allocated and/or spent:
 - software developers, journalists and others interested in Open Knowledge and visualising data for the purposes of transparency or advancing the digital economy [26]
 - politicians, analysts, bloggers
 - the general public

2.5.2 Focusing on certain audience groups

Satisfying the major requirements of all these users is not a suitable scope for this project (it may be for a follow-up project).

Therefore, this project will focus on the first group of users. The main reasons are time constraints and ease of access to such users. Searching for opportunities can be significantly harder than submitting them. In a way, providing data to academic users will pave the way for providing data to a more general public (third group) for re-use and analysis.

The project will nonetheless provide basic information submission (“digestion”) capabilities and basic general raw data access via an open machine-ready interface (“API”) to satisfy some of the requirements of the second and third groups. The author belongs to the third group. The Open Knowledge Foundation [26] and Cottage Labs LLP [7] will be briefly consulted on the completed project. Cottage Labs has stated that the project will be valuable to Open Knowledge if executed well and is happy to provide feedback.

2.5.3 Project constraints due to audience characteristics

All these user groups are professionals. This project is not being developed for a specific client - the time users can dedicate is very limited. As a general rule, user engagement (meetings, e-mail) will stay below 30 minutes per month per user.

2.5.4 Audience involvement due to audience characteristics

Since the potential audience is so large, this project will work with:

- a Humanities scholar - awaiting response
- Science, but not Social or Computer sciences - awaiting response
- Computer Science (in addition to supervisor) - accepted
- Supervisor - accepted
- non-scholars (less regular meetings) - Cottage Labs LLP - accepted

All scholars work at Aberystwyth University.

2.5.5 Measuring success

This project includes regular releases and evaluation by users as part of its iterative development methodology so that it can demonstrate gradually increasing value. However, towards the end of the development phase there are two weeks dedicated to final, slightly longer evaluation meetings.

Success will be measured by comparing the provided feedback to the final version of the aims of the project. The chosen development methodology harnesses changing requirements to provide more value.

2.6 Potential issues and risks

2.6.1 Potential legal issues

A possible legal issue the feature which will parse and understand e-mail digests with funding opportunities meant for human scholarly users. The terms and conditions of each funding organisation will have to be carefully checked, or the organisation will simply have to be contacted directly before such processing is done.

2.6.2 Technical issues

The greatest risk to building software is a bad estimate of how much time it will take, which in turn is notoriously difficult to estimate well.

In this project, the integration of the back-end and front-end technologies could take longer than the allocated week. Spike work on the Javascript visualisation libraries or parsing e-mail digests with funding opportunities could take longer than anticipated.

This project has two general risk mitigation strategies:

1. Reordering: during each iteration, do spike work before starting to write tests followed by documentation and code. If the spike work does not seem to make good progress, drop the feature to the end of the queue.
2. Early value: The main requirements of the software (collecting and presenting data to humans) rely on readily available components to function. These simply have to be brought together with slight modifications - this is the first feature. The software should still be quite valuable to users even if all but that first feature are dropped.

2.7 Planning progress

A software development methodology has been chosen and the initial planning phase (as it fits in the chosen methodology) has been completed. Furthermore, suitable project management tools have been chosen and the current plan has been filled in.

2.7.1 Project management tools

A project management piece of software suited for Agile approaches was chosen to manage this project: PivotalTracker. The project's progress and an ordered list of features to be developed is publicly available at [41]. It is recommended that the reader finishes the Planning section of this report before accessing this list since PivotalTracker uses a lot of development methodology-specific terms.

A private source code version control repository was set up with GitHub [15]. This will be made public when it is decided what licence will be used for the software output of the project and the code is of sufficient quality to be publicly released, towards the end of the project.

3 Planning

Planning methods, techniques and the shape of a project's timeline are usually influenced by the chosen development methodology (or mix of methodologies). The methodology also influences activities around the project - for example, more regular meetings with users (or, by convention, "customers") fit the spirit of Agile methodologies.

3.1 Development methodology

3.1.1 Considered methodologies

Development methodologies which were considered include Waterfall [40], V-shaped model [38], double-V model (a.k.a. W-model) [14], spiral model [3], Behaviour-Driven Development [46], Feature-Driven Development [30].

3.1.2 Chosen methodology

This project has adopted an Agile approach to both its software and documentation development (including the final report). There is a great variety of Agile practices and some are better suited for groups of developers, so certain well-known examples like Pair Programming [36] or full Scrum [35] were found to be unsuitable.

The reasons were varied, but generally included Agile's advantage when it comes to capturing fluctuating user requirements over time. Another advantage is the apparent ease of communication - the user does not have to provide a full functional specification of what they want, they can use more vague descriptions in the form of "stories" which describe the feature(s) they want [34]. It is worth noting that Feature-Driven Development is also an agile approach but is more suitable for larger projects. Behaviour-Driven Development is an interesting idea, but requires more effort / time on the users' side (the need to learn a domain-specific language).

Requiring less customer time (see §2.5.3) and exploiting a very lean project management and development methodology is particularly important in this project. With this and the Agile Manifesto [5] in mind, a methodology of an Agile character was laid out, using the Extreme Programming lifecycle as a base [10].

The length of one iteration will be one week. Considering the tight deliverable deadlines (see §3.2), this will keep work on the project focused and will force the definition of user stories of a manageable size.

The basic steps are:

1. Meet with a variety of potential users from the chosen user group (§2.5.2), trying to pick them so that each one has a different professional perspective on scholarship.
2. Define *and prioritise* the functionality of the project with each user in the form of the "stories" mentioned above. Record the results using the chosen project management tools (§2.7.1).
3. Release planning - prepare an initial timeline of which project features are to be released when and denote "release points". This work can be repeated as required - the order and the features themselves might change on the basis of user feedback over time (this is how Agile projects try to deliver more value by responding to changing requirements). The total size of the work and the size of each feature will most probably remain the same. This piece of work produces the "Project Timeline" and the current version is included in §3.2.
4. Iterate - every week is a self-contained unit of work consisting of coding, documentation and write-up work. Before starting to code each week, break up that week's story into

tasks so goals can be tracked more effectively and there is something to be accomplished each day.

5. Repeat items 1 and 2 - meaning regular meetings with previously chosen users - one per user per month, demonstrate the latest features and record the feedback.

Within the coding part of each iteration, Test-Driven Development is employed using both unit and integration tests (the latter via browser automation with Selenium [53]). This will be the evidence (on a basic technical level) for the “correctness” of the software. A basic form of Continuous Integration [21] will also be used - the Jenkins [50] software is currently being evaluated for the job. It basically runs all automated tests continuously, helping to detect regressions in code quality.

This methodology makes for a considerably simpler lifecycle than the Extreme Programming one (a visualisation of which can be accessed at [10]). This is intentional as there are only about 11 (potentially 13) weeks which allow for programming work as can be seen in §3.2.

3.2 Current project timeline

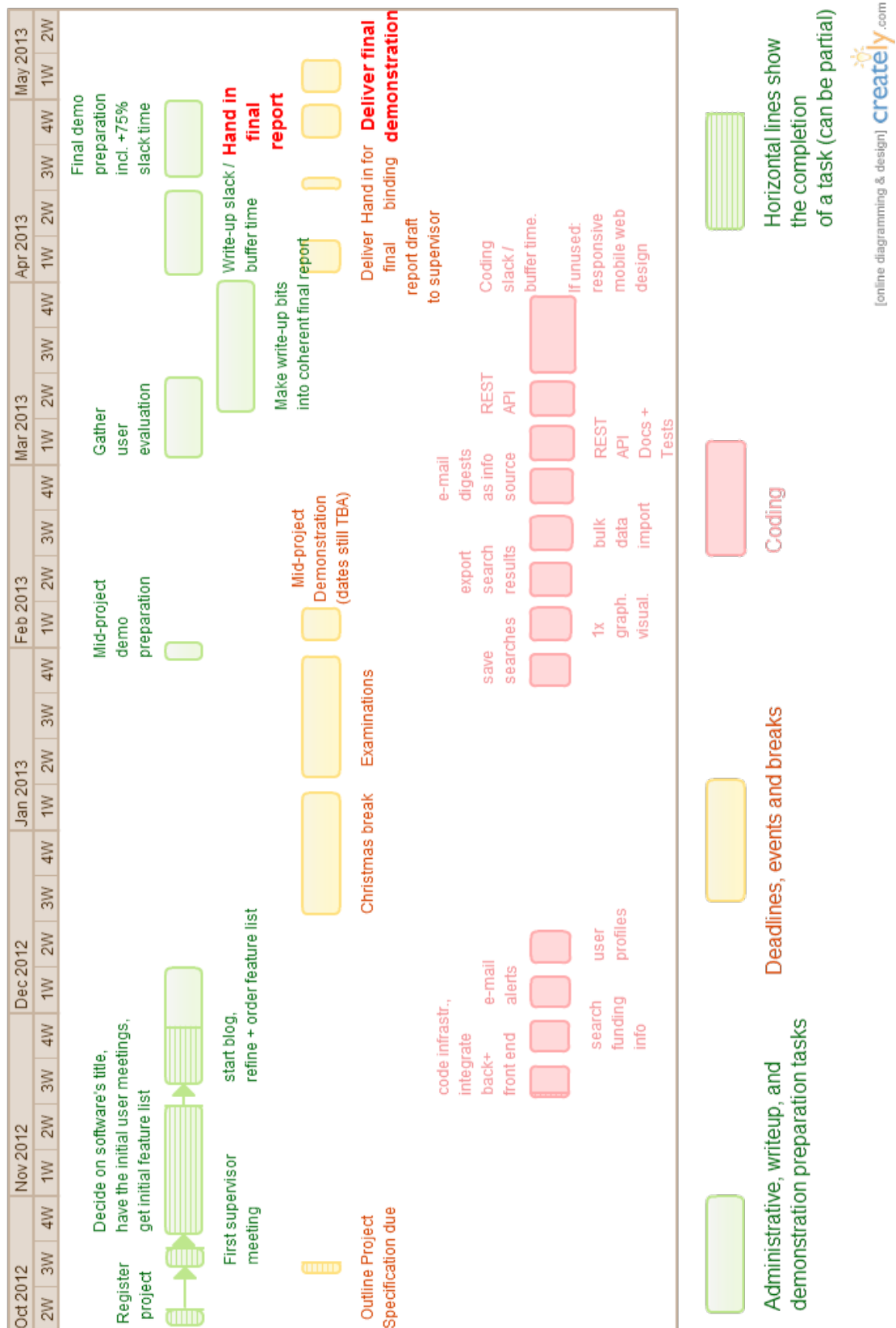


Figure 2: FundFind project timeline

Figure 2 visualises the current project timeline. It is being referred to as the “current” one since the order of or the actual features being implemented could change due to the Agile project development methodology, as described in §3.1.2. There could also be other factors which might influence the order of implementation of features, such as implementing “flashier” features slightly earlier than the users’ feedback dictates - for example, before the Mid-project demonstration. “Flashy” means producing easily recognisable valuable output such as summaries and graphical visualisations.

The latest list of features with longer descriptions and more information than the timeline Gantt chart can be accessed at [41].

It should be noted there are necessary chores which do not add value to the software. Examples: publicly hosting the project; project management tasks; writing reports such as this one. This is reflected in the following breakdown of the 400-hour time budget:

3.2.1 Time budget breakdown

Available in Table 3.

Type of activity	Activity	Repetition	Time (1 occurrence)	Total time
discussion	5x (Nov-March) regular meetings with 4 users	20	0.5h	10h
discussion	Supervisor meetings	23	0.5h	11.5h
discussion	E-mail / talk to Cottage Labs / OKFN	4	0.25h	1h
production	Project Outline Specification	1	3h	3h
production	Progress report	2	6h	12h
production	Mid-project demonstration	4	1h	4h
production	Write-up (ongoing through lifetime of project)	92	1h	92h
production	Final demonstration	4	2h	8h
admin	Project management	30	1h	30h
production	Software development	200	1h	200h
buffer	Slack available if any of these tasks take longer than estimated	N/A	N/A	28.5h
meta	Total project time	N/A	N/A	400h

Table 3: Project time budget

3.3 Demonstrating value

No special equipment or software will be necessary on the demonstration computer for both of the demonstrations, just a relatively modern graphical web browser.

Preparatory work for both demonstrations includes making sure that:

1. application is running in a publicly visible location
2. the Continuous Integration tool is working and all tests are passing
3. there is enough data loaded into the application to demonstrate more complex features such as faceted search [55]
4. the application works as intended in the web browser(s) on the demonstration computer

The project will be demonstrated as a fully working application in both demonstrations (although user feedback could have altered the feature list). The timeline §3.2 shows which features are currently planned for completion before (therefore inclusion into) each demonstration.

3.3.1 Mid-project demonstration

The first demonstration will show a “less valuable” version of the software with basic capabilities for collecting funding data and searching through it. Currently, the timeline also includes one basic graphic visualisation as a feature for this demonstration.

3.3.2 Final demonstration

The final demonstration will include all of the features in the timeline (subject to the usual Agile user feedback influence).

The RESTful API will be demonstrated using a small Python web application which will try to get the data exposed via the API and output details to the screen. The API documentation will then be shown on screen and brief comments on the API’s conformity to RESTful principles will be made.

Annotated Bibliography

- [1] AHRC, "AHRC Funding Opportunities," <http://www.ahrc.ac.uk/Funding-Opportunities/Pages/Funding-Opportunities.aspx>, accessed November 2012.

The root funding page of AHRC, the Arts and Humanities Research Council.

- [2] Alok Jha, "Open access is the future of academic publishing, says Finch report," <http://www.guardian.co.uk/science/2012/jun/19/open-access-academic-publishing-finch-report>, June 2012, accessed November 2012.

A Guardian article describing the Finch report which looked into Open Access and its benefits for academic publishing and society at large.

- [3] O. Balci, W. S. Gilley, R. J. Adams, E. Tunar, and N. D. Barnette, "The Spiral Model," <http://courses.cs.vt.edu/csonline/SE/Lessons/Spiral/index.html>, accessed November 2012.

A textbook section-length explanation of the Spiral software development model with a graphic visualisation and a table enumerating the various artefacts of the model. Although this an online educational resource from an academic institution, there is no publication date available for the project itself or the section on the Spiral model.

- [4] BBSRC, "BBSRC Research Funding," <http://www.bbsrc.ac.uk/funding/funding-index.aspx>, accessed November 2012.

The root funding page of BBSRC, the Biotechnology and Biological Sciences Research Council.

- [5] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, and D. Thomas, "Manifesto for Agile Software Development," <http://agilemanifesto.org/>, accessed November 2012.

A succinct set of statements enumerating the core values behind the Agile software development collection of approaches to software development.

- [6] Browshot, "Browshot - Powerful web screenshots," <https://browshot.com/>, accessed November 2012.

Homepage of the Browshot service. This allows a developer to make screenshots of web pages in any screen size, as any device including Personal Computers and mobile devices, in a variety of resolutions.

- [7] Cottage Labs LLP, "cottage labs - we don't sell products, we trade our skills," <http://cottage labs.com/>, accessed November 2012.

The homepage of a partnership of freelance software developers working together to produce open source software for the benefit of Higher Education and commercial companies around the world. Dedicated to Open Knowledge and Open Scholarship.

- [8] Cottage Labs LLP and Open Knowledge Foundation, "FacetView - Pure Javascript Frontend for SOLR and Elasticsearch," <http://okfnlabs.org/facetview/>, accessed November 2012.

Homepage of the Facetview project, a faceted Javascript browser (front-end) to indexing servers such as elasticsearch.

- [9] Craig Brierley, "Wellcome Trust strengthens its open access policy," <http://www.wellcome.ac.uk/News/Media-office/Press-releases/2012/WTVM055745.htm>, June 2012, accessed November 2012.

A press release by the Wellcome Trust stating it is strengthening its Open Access policy and related punishment for breaking the terms of the policy, as well as linking to a full version of the policy.

- [10] Don Wells, "Extreme Programming Project," <http://www.extremeprogramming.org/map/project.html>, accessed November 2012.

A diagram showcasing the Extreme Programming (Agile software development practice) project lifecycle. Different sections of the diagram provide links to more detailed explanations of the various parts of the project lifecycle.

- [11] Emanuil Tolev and Cottage Labs LLP, "IDFind - Got an ID? Not sure what it is? What you can do with it? Well, use this!" <https://github.com/CottageLabs/idfind/>, accessed November 2012.

The current homepage and version control repository page of the IDFind project. It tries to guess what kind of identifier an unknown string is.

- [12] EPSRC, "EPSRC Research Funding," <http://www.epsrc.ac.uk/funding/Pages/default.aspx>, accessed November 2012.

The root funding page of EPSRC, the Engineering and Physical Sciences Research Council.

- [13] Eric S. Raymond, "The Cathedral and the Bazaar," <http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/>, Sept. 2000, accessed November 2012.

A discussion on why / how JSON maps better to intuitive human understanding of the world than XML does and why this is important for a data representation format.

- [14] Gerrard Consulting, "The W-Model," <http://www.gerrardconsulting.com/?q=node/531>, accessed November 2012.

Explains the W software development lifecycle model and how it follows on from (or has evolved from) the V-model. This model could also be referred to as the double-V model.

- [15] GitHub Inc, "GitHub - Homepage," <https://github.com/>, accessed November 2012.

Currently popular decentralised version control system, usually used for software source code (but suitable for any artefacts although less suitable for binary formats or large files).

- [16] Google Inc., "Google Maps," <https://maps.google.co.uk/>, accessed November 2012.

A geographical mapping service by a commercial company.

- [17] Google Inc. and various authors - an open-source project, "Google Refine, a power tool for working with messy data (formerly Freebase Gridworks)," <http://code.google.com/p/google-refine/>, accessed November 2012.

A piece of open-source software for handling large semistructured datasets, making en-masse changes to them and cleaning them up. Google has now relinquished control and ownership of the project to the community, it is now the OpenRefine project.

- [18] Ian Sample, "Free access to British scientific research within two years," <http://www.guardian.co.uk/science/2012/jul/15/free-access-british-scientific-research>, July 2012, accessed November 2012.

An article in the electronic edition in the Guardian newspaper detailing the UK Government's policy on Open Access publications in 2014.

- [19] JISC, "JISC - We are the UK's expert on information and digital technologies for education and research," <http://www.jisc.ac.uk/>, accessed November 2012.

A UK governmental institution focused on enhancing the usage of digital technology in education and research in the UK.

- [20] Knight Foundation, "Knight Foundation - Informed & Engaged Communities," <http://www.knightfoundation.org/>, accessed November 2012.

The homepage of a global charitable foundation which supports (financially and otherwise) ideas that aim to change the world. They are particularly interested in quality journalism, media innovation, engaging the various communities around the world / in society and in fostering the arts.

- [21] Martin Fowler, "Continuous Integration," <http://martinfowler.com/articles/continuousIntegration.html>, accessed November 2012.

An article describing the principle of Continuous Integration in software development and illustrating why a developer or a development team might consider adopting this practice.

- [22] NHMRC / Australian Government, "NHMRC research funding datasets 1990 - 2012," <http://www.nhmrc.gov.au/grants/research-funding-statistics-and-data/funding-datasets>, accessed November 2012.

The NHMRC research funding data for the period 1990 to 2012 covering all new and continuing grants funded and active in that 23 year period.

- [23] Open Knowledge Foundation, "Open Definition," <http://opendefinition.org/>, accessed November 2012.

The Open Definition - defines what "open" is in relation to content and data.

- [24] —, "Open Knowledge Foundation Github Organisation Page," <https://github.com/okfn>, accessed November 2012.

The homepage of the Open Knowledge Foundation organisation on Github, the currently popular version control system.

- [25] —, "Open Knowledge Labs creates services and tools that unlock the digital commons for everyone." <http://okfnlabs.org/>, accessed November 2012.

The homepage of a collection of software projects which have sprung up to support or resulted from the Open Knowledge Foundation's pursuit of spreading the Openness cause in society.

- [26] —, “Our Vision / We Believe in the Power of Openness,” <http://okfn.org/about/vision/>, accessed November 2012.

Notes on the Open Knowledge Foundation’s vision of how openness can be beneficial for society (meaning all of current human civilisation).

- [27] —, “Radial Bubble Tree Visualization,” <https://github.com/okfn/bubbletree>, accessed November 2012.

Homepage of BubbleTree, a higher-level Javascript graphical visualisation library.

- [28] —, “Recline.js - relax with your data,” <http://okfnlabs.org/recline/>, accessed November 2012.

A library for building applications which handle potentially large amounts of the data and intend to allow and focus on more or less advanced data manipulation.

- [29] —, “Where Does My Money Go? - Showing where your taxes get spent - Country & Regional analysis,” <http://wheredoesmymoneygo.org/bubbletree-map.html>, accessed November 2012.

A BubbleTree visualisation of UK public spending.

- [30] S. Palmer, “An Introduction to Feature-Driven Development,” <http://agile.dzone.com/articles/introduction-feature-driven>, Nov. 2009, accessed November 2012.

A good introductory overview of what the Feature-Driven software development methodology is about, why it is considered an Agile methodology and when it is applicable (it evolved as an Agile approach for larger projects).

- [31] RCUK, “RCUK India - Opportunities provided by UK Research Councils,” <http://www.rcuk.ac.uk/international/Offices/OfficeinIndia/Funding/Pages/Home.aspx>, accessed November 2012.

The root funding page of RCUK / International Office / India.

- [32] Research Councils UK, “Welcome to Research Councils UK - Excellence with Impact,” <http://www.rcuk.ac.uk/Pages/Home.aspx>, accessed November 2012.

The home page of Research Councils UK.

- [33] Ron Knott, “Fibonacci Numbers and the Golden Section,” <http://www.maths.surrey.ac.uk/hosted-sites/R.Knott/Fibonacci/fib.html>, June 2012, accessed November 2012.

An explanation of the Fibonacci number sequence and related mathematical concepts. The Fibonacci sequence is being used as the complexity estimate for each user story in this Agile-managed project.

- [34] Scott Ambler + Associates, “Introduction to User Stories,” <http://www.agilemodeling.com/artifacts/userStory.htm>, accessed November 2012.

A explanation of what “stories” or “user stories” are when the term is used in the context of Agile approaches to software development.

- [35] Scrum Alliance Inc., “Scrum Is an Innovative Approach to Getting Work Done,” http://scrumalliance.org/pages/what_is_scrum, accessed November 2012.

A succinct description of the Scrum framework of Agile software development practices.

- [36] J. Shore, "The Art of Agile Development," http://www.jamesshore.com/Agile-Book/pair_programming.html, Feb. 2010, accessed November 2012.

The online edition of a famous work (book) on Agile Development. This section focuses on and provides a good description of Pair Programming, an Agile software development practice.

- [37] Shuttleworth Foundation, "Shuttleworth Foundation - Supporting exceptional people to change the world," <http://www.shuttleworthfoundation.org/>, accessed November 2012.

The homepage of a global charitable foundation which funds individuals with its scholarships and then multiplies the money they invest in projects which aim to bring about social change for the better.

- [38] SoftDevTeam, "V-shaped lifecycle model," <http://www.softdevteam.com/V-shaped-lifecycle.asp>, accessed November 2012.

Explains the V-shaped software development lifecycle model. Also has a nice succinct summary of advantages and disadvantages compared to other models (mostly Waterfall) which seems to fit previous knowledge I have of this model from less reliable sources such as Wikipedia and more reliable ones such as the teaching accompanying my Computer Science degree at Aberystwyth University.

- [39] The Document Foundation, "LibreOffice Calc - a Spreadsheet That Meets Any Need," <http://www.libreoffice.org/features/calc/>, accessed November 2012.

A piece of open-source spreadsheet viewing and editing software, part of the LibreOffice office productivity suite.

- [40] The Smart Method Ltd, "The Waterfall Development Methodology," http://www.learnaccessvba.com/application_development/waterfall_method.htm, accessed November 2012.

A brief explanation of the Waterfall software development methodology. Also (briefly) compares Waterfall to Agile approaches.

- [41] E. Tolev, "PivotalTracker - FundFind," <https://www.pivotaltracker.com/projects/688967>, Nov. 2012, accessed November 2012.

Publicly visible project management for the software component of this project. By default, it will show what the author is currently working on and what is in the backlog. Click on "Icebox" (in the top left-hand corner of the web page, next to the "CURRENT" and "BACKLOG" links) to see all recorded user stories and chores for getting the project done. Stories in the Icebox are ordered in descending order in terms of priority, i.e. the more important ones according to aggregated user opinion are at the top.

- [42] Twitter Inc., "Bootstrap - Sleek, intuitive, and powerful front-end framework for faster and easier web development." <http://twitter.github.com/bootstrap/>, accessed November 2012.

The homepage of the Twitter Bootstrap User Interface library.

- [43] UK Government - Unknown individual name, "DATA.GOV.UK - OPENING UP GOVERNMENT," <http://data.gov.uk/>, July 2012, accessed November 2012.

The UK Government Open Data portal containing various datasets released under Open licences.

- [44] Various, "What are the pros and cons of XML and JSON?" <http://stackoverflow.com/questions/3536893/what-are-the-pros-and-cons-of-xml-and-json>, accessed November 2012.

A discussion on why / how JSON maps better to intuitive human understanding of the world than XML does and why this is important for a data representation format.

- [45] Various (a UK Government Working Group), chaired by Dame Janet Finch, "Accessibility, sustainability, excellence: how to expand access to research publications," <http://www.researchinfonet.org/wp-content/uploads/2012/06/Finch-Group-report-FINAL-VERSION.pdf>, June 2012, accessed November 2012.

The Finch report - looked into Open Access and its benefits for academic publishing and society at large, commissioned by the UK Government.

- [46] Various authors, "Behaviour-Driven Development," <http://behaviour-driven.org/>, Jan. 2009, accessed November 2012.

A description of Behaviour-Driven Development. This is a Wiki-style web resource, so the year and month of publication are taken from the last edited date and may change by the time the reader accesses this resource.

- [47] Various authors - an open-source project, "D3 - Data-Driven Documents," <http://d3js.org/>, accessed November 2012.

Homepage of D3.js, a lower-level Javascript graphical visualisation library.

- [48] —, "elasticsearch - You know, for Search," <http://www.elasticsearch.org/>, accessed November 2012.

The homepage of the elasticsearch indexing server.

- [49] —, "Flask - Web development, one drop at a time," <http://flask.pocoo.org/>, accessed November 2012.

Homepage of the Flask Python web framework project, helps build RESTful web applications in Python.

- [50] —, "Jenkins - An extendable open source continuous integration server," <http://jenkins-ci.org/>, accessed November 2012.

Homepage of the Jenkins Continuous Integration server, a tool which helps a developer or a team of developers practice the Continuous Integration software development principle.

- [51] —, "jQuery is a new kind of JavaScript Library." <http://jquery.com/>, accessed November 2012.

The homepage of the jQuery Javascript and User Interface library.

- [52] —, “OpenRefine,” <https://github.com/OpenRefine/OpenRefine>, accessed November 2012.

A piece of open-source software for handling large semistructured datasets, making en-masse changes to them and cleaning them up. Used to be the Google Refine project.

- [53] —, “SeleniumHQ - Web application testing system,” <http://seleniumhq.org/>, accessed November 2012.

The homepage of an open-source software project which enables the automation of web browsers and other testing techniques for the purposes of testing web applications.

- [54] Various authors - an open-source project hosting open data, “OpenStreetMap - The Free Wiki World Map,” <http://www.openstreetmap.org/>, accessed November 2012.

An open-source, open data geographical mapping service.

- [55] Various authors / Wikipedia, “Faceted search,” http://en.wikipedia.org/wiki/Faceted_search, accessed November 2012.

The Wikipedia page on faceted search. Would not make much sense to a reader unfamiliar with the topic, but should do with the example in the report body.

- [56] Wellcome Trust, “Wellcome Trust - We are a global charitable foundation dedicated to achieving extraordinary improvements in health by supporting the brightest minds.” <http://www.wellcome.ac.uk/>, accessed November 2012.

The homepage of a global charitable foundation which funds lots of scholarly activities in a variety of areas.