

Interim Report - UniNot.es: A Collaborative Note-Taking Website

Author: George Nixon
Supervisor: Dr. Alia Abdelmoty
Moderator: Prof. Nick Avis
23/12/2011

Abstract

This report looks at the problem of collaborative note-taking within the context of a learning or knowledge-sharing environment, examines existing solutions, and sets out the decisions taken thus far in the project and the reasons behind them.

Table of Contents

[Interim Report - UniNot.es: A Collaborative Note-Taking Website](#)

[Abstract](#)

[Table of Contents](#)

[1. Introduction](#)

[2. Background](#)

[2.1 Competitive Analysis](#)

[2.1.1 Learning Central](#)

[2.1.2 General Purpose Collaboration Websites](#)

[2.1.2.1 Private Collaborative sites](#)

[2.1.2.2 Siloed Collaborative Sites](#)

[2.1.2.3 Public Collaborative Sites](#)

[2.1.3 Similar to UniNot.es](#)

[2.2 MVC rapid development framework](#)

[3. Approach](#)

[3.1 Population Analysis](#)

[3.2 Choice of Methodology](#)

[3.3 Proposed features](#)

[3.3.1 Basics](#)

[3.3.2 Collaboration](#)

[3.3.3 User Dashboard](#)

[3.3.4 Policing](#)

[3.3.5 Discoverability](#)

[3.4 Choice of Language and Framework](#)

[4. Conclusions](#)

[Appendices](#)

[Appendix A - User Personas](#)

[Learners](#)

[Teachers](#)

1. Introduction

Approaches in learning and knowledge sharing have traditionally been dominated by a teacher-student model - an apprentice studies under a master tradesman; a schoolchild studies a curriculum taught by a schoolteacher; an undergraduate is lectured to by many professors. These methods, in general, rely on perceived experts, or information authorities, to impart

wisdom based on experience, understanding and knowledge gained over the course of many years of study or practice.

In the past decade, the web has revolutionized this field, and new models based on collaboration, in which anyone with some knowledge or the capacity to perform research can participate. Wikipedia, an encyclopedia exemplifying this concept, has become the world's largest and most widely used encyclopedia¹, displacing past tomes such as the Encyclopedia Britannica and even software forms such as Microsoft's Encarta. While price may have been a factor in the early days (Wikipedia being free to browse and the latter examples requiring a subscription where available online), this is no longer the case (EB is also free, and Encarta has been discontinued in the face of Wikipedia's supremacy).

It could be argued that the breadth of Wikipedia's articles has been key to its success, and that the cause of this is its greater number of contributors, owing to its open status as a wiki. Other examples of this approach include WikiHow², a how-to guide to many different subjects written by its users, abundant subject-specific wikis such as LyricWiki³ and WikiCars⁴ (some of which are built on the same OSS framework as Wikipedia, known as MediaWiki), and the blogosphere and social web, which enable the sharing of thoughts and ideas without the requirement for an authoritative author (as opposed to the traditional print media and its heritage of editorial selectivity in choosing what to publish).

A prominent extant example of the teacher-student model is the education sector. This project aims to bring collaboration to the field of academic study. It assumes that while a lecturer on a topic may contribute large amounts of important information, the potential contributions of her students, fellow academics and other parties may also be of value. By harnessing the power of the web to allow many people to access and edit information pertaining to subject of study or research, it is possible that students and academics may be presented with greater resources with which to learn and thrive. Thus, this project was conceived with the intention of creating a website that facilitates this collaboration, providing access to a range of tools for knowledge sharing from a single location. It would be enhanced by granting access to, and being discoverable by, a wide range of users and contributors, such that the corpus of information is enhanced, and a greater number of people could benefit from the wisdom imparted. Furthermore, by basing notes around events such as lectures, scholars disadvantaged by occasional absence or disabilities such as hearing impairment may benefit from their peers' interpretations of such presentations.

I have registered the domain name <http://uninot.es> for the purpose of building this site and hosting test and final versions of my efforts. Hence I will, at times, hereafter in this document refer to the project as "UniNot.es". To me, the name both connotes university life, as "uni" in the

¹ Look Who's Using Wikipedia - Bill Tancer - March 1 2007 - <http://www.time.com/time/business/article/0,8599,1595184,00.html>

² <http://www.wikihow.com>

³ <http://lyrics.wikia.com>

⁴ <http://wikicars.org>

UK is a popular contraction of the word “university”, and also co-operation and collaboration, as in “unite”.

2. Background

2.1 Competitive Analysis

Before beginning work on this project, I undertook an in-depth study into existing solutions and their strengths and weaknesses. Here I present some of my findings:

2.1.1 Learning Central

Taking my own university as an example, I tried to find existing ways to perform the tasks described in the introduction using web applications provided by academic institutions. At Cardiff University, the core of this effort is known as Learning Central⁵.

In terms of collaboration, two features are present in a category of web app called “Collaboration Sessions”. The first is “Lecture Hall”, whose description refers to it as a “virtual classroom”. The second is called “Office Hours”, which is described as a “chat” application.

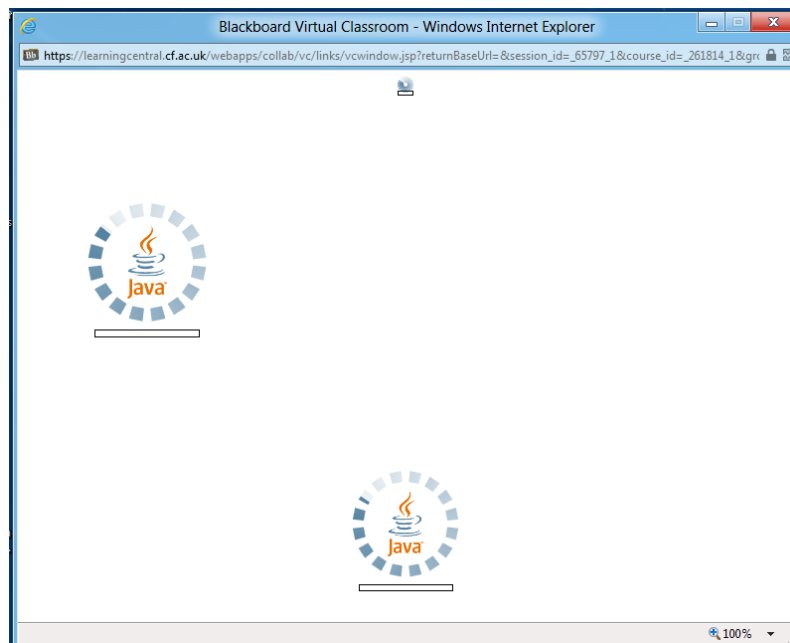


Figure 1: A screen capture of the malfunctioning Learning Central “Virtual Classroom” application

Unfortunately, I was unable to get these to work, as demonstrated in Figure 1. This was the case on more than one platform (Ubuntu, Windows 8 Developer Preview), and more than one browser (Chromium, Internet Explorer 9). Whatever their intended functionality, it would appear that they are not widely used, or else the problem might be expected to have been addressed by IT support by the point at which I accessed it, five weeks into the academic year. Anecdotally, I have not heard it mentioned by any of the students with whom I have worked or socialized. If collaboration requires a good number of users in order to populate together the

⁵ <https://learningcentral.cf.ac.uk>

shared knowledge base or to provide demand for an expert to do so, this factor appears to be absent in Learning Central's offerings.

2.1.2 General Purpose Collaboration Websites

The following are some significant examples of various types of collaboration websites. I have grouped them by publicness, which I think is key to collaboration on a wider scale. Here we have private collaborative websites, which limit access to a select group, siloed⁶ websites which have public access but only serve one particular purpose (as opposed to facilitating collaboration in various guises) and fully public websites.

2.1.2.1 Private Collaborative sites

Basecamp⁷ describes itself as “the leading online project collaboration tool”. Many sites like this exist to serve the needs of a team of people working together on a particular project, which has some parallels with the aims of this project. It has some standout features, such as the ability to share messages in threads around a particular topic, and the ability to store documents like presentation slides and pdfs in a central location, with automatic versioning. These would be useful to students as they could ask questions or add comments around a subject, and store new versions of lecture handouts with annotations added. A main focus of the site is around project management however, with other features such as task assignment which would not be relevant to learning. It is also a paid service and use of the site is restricted to members, with the added repercussion that some organizing would be needed to set up and maintain the space and discoverability would be limited to word-of-mouth.

Other examples of private, project-focused websites include pbworks⁸ and ProjectPier⁹.

2.1.2.2 Siloed Collaborative Sites

Some sites have useful collaborative features that could aid study, but lack the cohesive nature of the above project management websites that could help build differing forms of knowledge around a subject of study or research. As such content on these sites tends to be siloed rather than interwoven. A possible outcome of this project would be to define a single place on the web around a subject and offer direct links to collaborative sessions on such sites, rather than try to reproduce their features (a venture that could be a major undertaking beyond the scope of this project).

⁶ Siloing is the practice of keeping information in a contained system that interacts poorly (if at all) with other systems. See http://en.wikipedia.org/wiki/Information_silo for more on this.

⁷ <http://basecamp.com>

⁸ <http://pbworks.com>

⁹ <http://www.projectpier.org>



Figure 3: screenshot of MindMeister distributed on the company's press page¹⁰

Mindmeister¹¹ excels at mind-mapping in the browser, and even provides live collaboration with multiple users. Users can create public, rich mindmaps like the one shown in Figure 2 well-suited to sharing of ideas. However, it is a single-function website and does not allow other types of content, and is not ideal for collaboration as many people are still unfamiliar with the concept of mindmaps. It also has limited free functionality and setting up a large sharing system would involve paid subscription by members or an organization. It is also endemic to mindmaps that their contents are not crawlable by search engines, impairing discoverability by uninitiated users and the public.

Another siloed site in this vein is Twiddla¹², which allows white-board style collaboration and chat. Once again, regardless of the alacrity with which Twiddla performs this function, this type of collaboration is of limited use when attempting to completely explore a subject.

2.1.2.3 Public Collaborative Sites

Zoho¹³ offers a wide variety of free collaborative tools. These include Zoho Docs, Discussions

¹⁰ <http://www.mindmeister.com/press>

¹¹ <http://www.mindmeister.com>

¹² <http://twiddla.com>

¹³ <http://www.zoho.com>

(forums), Meetings (web conferencing) and Wiki. Many services such as the Wiki are only free for a limited number of users, after which various grades of pricing comes into effect. There is an attempt to bring together these disparate functions into a single space, entitled Zoho Share. This has some noteworthy features that might prove useful to adapt for my project, such as rating documents, bookmarking them, and displaying thumbnails of related documents¹⁴. Unfortunately these are not framed around an organization but around particular people and their documents; an organization-based version is pending, but these spaces will be closed and only accessible to Zoho Business subscribers. A university or conference may set up such an account and pay for its members to gain access in a private capacity, but an independent community set up by its users is unlikely to develop on this basis.

WikiSpaces¹⁵ is a completely free, ad-supported host of public Wikis, which anyone can set up. Because Wikis can contain many different forms of information and can assist in merging simultaneous edits, it would be possible to create a Wiki for an organization expanding to many subjects, each with detailed sets of notes and links. My project would do well to take cues from such a system, perhaps with more conventions and focus on enabling users to achieve particular goals relating to study, as the open-ended nature of the wiki could be seen as both a blessing and a curse in creating relevant content.

2.1.3 Similar to UniNot.es

In the course of my research, I came across several examples of sites with similar aims to mine, and indeed had features that I had not considered, but all seemed either abandoned or otherwise failing to live up to their potential. It would appear there may be some merit to this project as a niche not yet fully exploited.

MyNotelt¹⁶ has several ideas in common with my project, in that it presents a common structure based around schools and colleges. Each of these institutions can have a number of courses, and these courses have a course notes, course assignments and course details, such as name, professor name and contact details. As the name of the site would suggest, the focus is on keeping notes inside of this structure, and the sharing of notes fellow students. The downside is that the site is difficult to navigate in order to find a course relevant to the user, and the search functionality appears not to work at all; in other words, discoverability is poor. I also was able to delete a course from the system while browsing MyNotelt, which raises the potential issue of vandalism and unintended consequences that I may need to consider for my project. Finally, the site did not seem to have many users, and at time of writing appears to be offline.

NoteMesh¹⁷ is similar to MyNotelt, and adds other features such as class details with meeting times and rooms, a wiki for each class with history of revisions and a style guide, and a tour to introduce new users. I found it was too rigidly structured around a US college system (e.g.

¹⁴ See a video demonstration of Zoho share here: <http://www.viddler.com/explore/Raju/videos/24>

¹⁵ <http://www.wikispaces.com>

¹⁶ MyNotelt.com could not be accessed at time of report, but is described on another site here: <http://www.makeuseof.com/dir/mynoteit-online-notes-organizer-for-students/>

¹⁷ <http://notemesh.com>

forcing the naming convention CS 101) which would make it difficult to use as a British student. It was evidently abandoned, and the “create class” function only accepted dates up to 2009.

Other examples exist in the category such as Campusbug¹⁸, which describes itself as “the first social learning network” but is difficult to register for and was not clear in what it delivered, and Loomagoo¹⁹, which despite the odd name offers some noteworthy properties. Chief among these is the best discoverability that I encountered among such sites, with an ability to connect to a user’s Facebook account to discover friends also using Loomagoo, and a drop-down list of existing colleges in the systems. Again, Loomagoo did not appear to be thriving, and seems to have been taken down at time of writing.

2.2 MVC rapid development framework

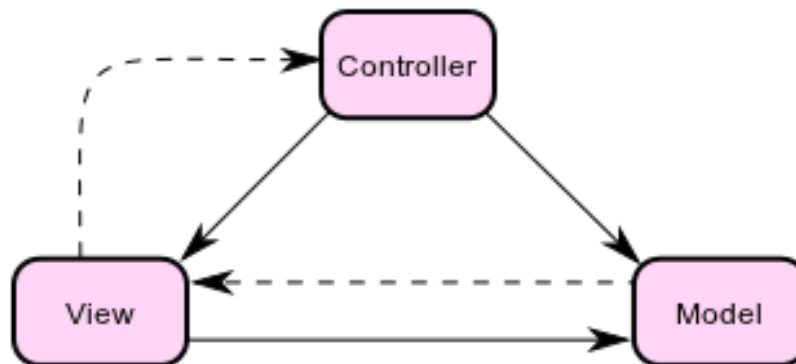


Figure 3: Diagram of the MVC pattern.²⁰

The Model-View-Controller (MVC) pattern was popularized by Ruby on Rails (RoR), creating a lasting impact on the web development community and inspiring similar frameworks to be developed for languages other than Ruby, such as Microsoft’s ASP.NET MVC and various PHP frameworks like Zend Framework and CakePHP. The reason for this remarkable success lies in the creators’ observation that dynamic, database-driven websites all tend to deal with the same familiar problems, i.e. storing, retrieving and displaying records, maintaining user accounts, and so on. Having a framework that takes care of some of these fundamental tasks leaves the developer free to work on higher-level tasks based around object relationships and user interactions, while low-level tasks such as forming and performing SQL queries and outputting browser-compliant HTML code can be handled automatically.

The core concepts of MVC are represented in Figure 3. A Model is an object-oriented

¹⁸ <http://campusbug.com/>

¹⁹ <http://loomagoo.com/> appears to have been taken down and replaced with an empty blog as of 22/12/2011, however some traces of the service can be found at <https://www.facebook.com/pages/Loomagoo/102902916886>

²⁰ This is a public domain image downloaded from <http://en.wikipedia.org/wiki/File:ModelViewControllerDiagram2.svg>.

representation of what would traditionally in relational databases be termed a record - it is tightly-coupled to the underlying database so that the record's data is available properties of the object. A Model also usually contains business logic, for example providing a method returning a Person's age by comparing his stored date-of-birth and today's date, and other less trivial operations such as validation. A View is a script for generating web pages, which displays various aspects of a Model to the user. The Controller is, as the name suggests, in charge of controlling these other two objects, in the following (slightly simplified) manner:

1. A user queries the website, perhaps by clicking a link or submitting a form
2. The request is routed to the appropriate controller, e.g. the Song controller
3. The controller uses the parameters from the user to retrieve (or sometimes store) matching Model objects in the database
4. The controller passes the Model to the View
5. The View displays the Model as a web page, along with other contextual HTML elements in the layout such as navigation links, site headers and footer

Simply put, using such a framework negates the necessity for a lot of low-level manual coding. The two main benefits of this are a) less time spent writing and testing basic functionality ("re-inventing the wheel") and b) easier "chopping and changing" of various parts of the system without requiring a major rewrite of the underlying structure. As such it is well-suited to agile, iterative development in a short time frame.

For more, see Wikipedia article "Model-view-controller"²¹.

²¹ <http://en.wikipedia.org/wiki/Model-view-controller>

3. Approach

3.1 Population Analysis

In order to serve the needs of various potential users of UniNot.es, I considered four stereotypes, as detailed in Appendix A. These personae were chosen to portray both the teacher and the student roles in two forms of learning experience: undergraduate study and business and academic conferences. I shall refer to these at various times in this interim report as well as the final report.

3.2 Choice of Methodology

When deciding how to approach the problem, I was faced with a choice between two approaches - the traditional waterfall model of specification, design, implementation etc, or agile. I have gained experience in both of these models working for two different companies, and my instinct was that agile would suit this project best.

While the waterfall model has its advantages, such as carefully planned and (in theory) well executed design work leading to what may be a better product, in fact I was aware that there were so many uncertainties at the start of the project that it made more sense to pick the methodology built around changing requirements. Examples of these unknowns were what language I would use, what framework to use, what external services would need integrating, what problems would occur when implementing them, who the users would be, and so on.

I feel justified in choosing agile development as I've already encountered unexpected turns in the project path, having chosen to use CakePHP over Zend Framework, then discovering the Google Docs library is built specifically for Zend Framework, and the latter would now make more sense for the project (more on this in the final report). It is too late to change my mind as too much work has been done in CakePHP, and in fact the two can probably co-exist happily, but it highlights the problems with trying to plan a six-month project in advance, when not enough is yet known about the problem area. I would apply this same logic to the design of functionality as well - I've tried to iteratively build a basic system thus far and written user stories that serve an immediate need for one or more personas ("Users can log in", "Registering user is not a bot", "User has a personalized dashboard", etc).

3.3 Proposed features

As described above, I am using an agile, lean approach which forgoes a full specification in favour of rapid iteration and user stories. However, based on my background research and population analysis, I have identified some beneficial features which I will be adding to the backlog²² and attempting to implement before the end of the year:

3.3.1 Basics

²² an agile term for the list of user stories to finish

To provide focus on particular learning activities, I will impose a convention for structuring collaborations:

- Organizations (i.e. conferences, universities) have many subjects (presentation topics, modules)
- Subject has a series of one or more events (presentations, lectures)
- Events have one or more presenters (speakers, lecturers)
- Users can “follow” any of the above, and each other

All of the above can have details (names, places, contact information, etc) and associated notes/documents/links.

3.3.2 Collaboration

The website will provide a way for users to create, edit and share notes and other content around particular knowledge-sharing events, such as lectures, classes, workshops and so on. Since these events are often attended by many people, and since there is likely to be a clustering of activity on UniNot.es at times close to the events (shortly preceding, during and afterwards), it would be advantageous to support real-time collaboration, viz. simultaneous editing of a set of notes by more than one party.

3.3.3 User Dashboard

When logged in, users like Rory need a home screen displaying links and updates for the subjects, organizations and other users that they follow. This in turn requires these relationships to exist in the system, and the ability to choose to follow any of these entities.

3.3.4 Policing

As there will inevitably be times when users take actions on the site that, intentionally or by accident, are perceived by others to negatively affect shared content, cause offense, or for users like Elana, violate their intellectual property rights. As such a revision history and roll-back operation would be very useful, and it should be easy to report a problem user or document. To prevent wider abuse of the system, precautions should ensure that registering users are humans and not bots.

3.3.5 Discoverability

It is a key priority that content is discoverable, i.e. new and existing users can discover information relevant to them, thus broadening the number of potential contributors as well as delivering benefit to them directly in terms of serving them content.

One side of this is search engine optimization, as search engines are large drivers of traffic to websites - if notes are held externally to the site, it would be beneficial to have an abstract of the document to help Google and other crawlers understand the nature of the page and what subjects it relates to.

Other ways of finding relevant content would be to make use of the geo-location functionality of HTML5-compatible web browsers to find events near the user, and to connect a user's account

to a social network such as Facebook or Twitter to find her friends using UniNot.es and what subjects or people they follow.

3.4 Choice of Language and Framework

Already aware that I wanted to use an MVC framework, having previously worked on a couple of Ruby on Rails projects, I surveyed the options available. Microsoft ASP.NET MVC was an option, although having worked as a support developer on SQL Server instances, I suspected it was probably quite heavy-duty, a lot to manage and basically surplus to my needs. It would also only run on Windows and I didn't have a Windows machine. I also considered Ruby on Rails, but the language is slightly obscure and I am not as well acquainted with it as the other options. I have also found in the past that RoR hosting is quite complicated as it requires maintaining an instance of a server and some installation etc just to deploy an application which works on one's local machine. There are good two good solution providers, Heroku and EngineYard, for this problem, but they can be expensive and I did not have the budget.

This left Java or PHP. I have some experience in both, and both are free and open source. However, Java is not supported by my web host, and most hosting websites only provide Java support on the more expensive plans, such as Virtual Private Server sliced hosting. The MVC availability in Java isn't too impressive either; there are a few different MVC frameworks, like Spring, Struts, and even a strange variant on Java and Ruby on Rails called Groovy on Grails, but my research did not suggest any of these were what I was looking for.

I looked into PHP MVC frameworks therefore, as PHP is a language I have already written a complex database-driven site in and it has run well for me on my web host. Deployment is often simply a case of dragging and dropping some files in FTP and the site will run with very little configuration required. From searching the web, it became apparent that there was a number of well-received MVC frameworks for in this language: Symfony, Zend Framework, CakePHP, CodeIgniter and so on. I picked CakePHP because I had already read a book on it, although I had never actually written anything on it, it struck me as an elegantly simple and friendly language that I would like to try. I read the book again, going through the tutorials step-by-step, and was happy with the results. It is the framework I will use for the rest of this project.

4. Conclusions

I have defined a problem, explored existing solutions to it, worked out who the users are likely to be and begun work on an solution. I have picked a language and a framework and worked out some of the features I will need to implement. I will spend the rest of the year working on user stories based around my personas and hope to get a working implementation by the time of the final report, or possibly to have already launched by then and responded to user feedback.

Appendices

Appendix A - User Personas

Learners

Name: Rory

Role: Undergraduate at a university

Sex: Male

Age: 21

Profile:

Rory would like to be able to follow certain modules and people at his university, and be able to access them and see updates on them from his home screen. He may need to keep notes for himself, such as reminders for actions, which he doesn't want to share publicly. He needs to be able to find relevant entities, such as his university and his friends easily. He should be able to take it upon himself to make full use of the site without relying upon his university to set up the relevant accounts, modules etc.

Name: Marissa

Role: Conference Attendee, also works as agile software team leader

Sex: Female

Age: 35

Profile:

Marissa occasionally attends conferences on agile methods relating to her job. She would like to be able to participate in creating notes and post links to useful sites and videos based around the events she attends, and also the ones she is unable to travel to in foreign countries. She would like to discover events near her on the site, rather than having to guess what title it may already be available under, thus avoiding duplication of events and redundant note-taking. Marissa also occasionally leads her own conferences and so does not wish to maintain separate accounts for her roles as both learner and teacher.

Teachers

Name: Elana

Role: University lecturer

Sex: Female

Age: 33

Profile:

Elana teaches several modules at her university, as well as conducting research and occasionally attending conferences on her subject of expertise. She wants to be able to register her modules on the site in order to share additional resources with her students to the presentation slides at her lectures, such as links to further reading and books, and to encourage her students to actively learn outside of the classroom environment. Although she posts her handouts on her own website, she does not necessarily want her content redistributed and needs to be able to report violations of her copyrights.

Name: Steve

Role: Conference speaker and Chief Technical Officer at a large IT company

Sex: Male

Age: 44

Profile:

Steve is a busy, high-level executive at a major software firm. He occasionally gives popular speeches at well-known conferences on the future of software and the web. He wants to get his message across to a large number of people, thus giving him a better voice and presence in the IT community. Although he does not have time to create or manage a profile on UniNot.es, he welcomes his followers setting up a profile for the events he attends so they can disseminate and add to his material.