UNIVERSITY of LIMERICK

OLLSCOIL LUIMNIGH

Munster Fleadh Web Development

Student Name: Turlough O'Shea

Student Number: 09004486

Course Code: LM114 - Music Media and Performance Technology

Supervisor: Chris Exton

Acknowledgements

I would like to acknowledge the support I have received from my friends and class mates. The last four years were tough, but having them around made it easier!

I would like to thank my family and Kathryn Conway for their patience and support during the completion of my FYP. They helped me get through the tough times and gave me encouragement when I needed it most.

I would also like to acknowledge my fellow students who were kind enough to complete a series of tests. Their contribution and results were important in improving the usability of the website.

I would also like to acknowledge the contribution of the clients from the Munster Fléadh. They were very easy to contact and always gave their opinions on the work presented to them. They had some clear instructions that they wished followed and this really helped the design and development process.

I would like to thank my supervisor Chris Exton for his guidance over the last few months. I feel that his presence greatly helped with the completion of my FYP.

Finally, I would like to thank my second reader Joe Griffin for showing an interest in my project and making the experience of the demo day very relaxing.

Declaration

This Final Year Project is presented in partial fulfilment of the requirements for a B.Sc in Music, Media and Performance Technology.

It is entirely my own work and has not been submitted to any other University or higher education institution, or for any other academic award in this University.

Where the use has been made of the work of other people it has been fully acknowledged and referenced.

Signature: _____

Date: _____

Project Summary

The aim of this project is to make a fully functional website for the Munster Fléadh due to take place on the University of Limerick campus in July 2013.

This project is driven by client contact and feedback. As the web developer, my most important task will be to complete the work to the client's specifications and desires. To do this, there has to be constant contact with them and presentation of the current progress for their input and comments. When working with clients, it is always possible that they may not know about the technical side of things and have very vague requirements. When we were initially brainstorming, one of key suggestions that the clients had was to get people to 'stick around'. They had no suggestions on how to achieve this so it was now down to me to try and develop something to try to keep users interested and stay for as long as possible.

When designing this website, the main idea was to create a modern and aesthetically pleasing but also fully functional website. The functionality that was to be implemented to keep people around was simple in idea but more complex in action. The new functionality of the website lets users create an account and then make their own schedule. By doing this, users would have full control over what events they were going to attend. The administrator on the website would also have a rough estimate of the numbers of users attending these events. The website for the previous Munster Fleadh had nothing like this up to now so the upcoming users will find the service more useful and hopefully take full advantage of it. There will be a mix of dynamic and static web pages. The users should be able to find directions, look for accommodation, learn about UL and the Castletroy CCÉ and much more.

As this area was new to me, much research and practice had to be done before the project was started. The languages used to develop the website were HTML, CSS, the JQuery library of Javascript and Javascript. PHP and MYSQL were used in the development of the back end and server side of the website. All images used were created or edited using Adobe Photoshop CS5.

Table of Contents

Chapter 1: Introduction

Overview

For my Final Year Project (FYP) I am going to make a website for the Munster Fleadh to take place on the University of Limerick during July of 2013. For this FYP I will be working closely with clients and will have to work to their specifications and deadlines. I will have to design and develop the entire website; from the front user-end and the back-end server side.

For the development of this website, I'm going to use a number of web technologies such as HTML, CSS, jQuery a library of Javascript, PHP and MySQL. Ruby on Rails was also considered for use, but it was decided against in lieu of PHP and MySQL. The project will be heavily driven by client interaction. I will need to complete tasks on time according to the clients' deadlines and standards. The client's main idea was to have users 'stick around'. To do this I plan to let users make their own schedule from the available events and save it for a later date.

My background is in Music, Media and Performance technology and I wanted to move away from this and expand my skills so I chose an FYP somewhat outside my comfort zone. After choosing my FYP I set about examining the existing website used for last year's Munster Fléadh. I soon came up with a design I felt was a great improvement upon the original. I wanted to keep the website somewhat similar for novice users, but also improve upon its design and functionality.

To recap, the aim of this FYP is to make a fully functional website for the Munster Fleadh set to take place on the University of Limerick campus in July 2013. Technologies such as HTML, CSS, jQuery a library of Javascript, PHP and MySQL will be utilised. For any graphic design work Adobe Photoshop CS5 will be used.

Motivation

A few of my motivations during this project are to improve my skills, more than satisfy the clients with the end product and have a new portfolio item to show to potential employers.

As I have no experience in this area, I want to improve my skills with many different technologies. I hope this will happen throughout the course of this project. Learning new skills and improving on some of my previous ones will motivate and keep me going throughout. Before I begin work on the website I will have to learn some of the technologies, such as Javascript, from scratch. I have wanted to learn this for some time so I will be quite motivated. I hope to become very adept with all of the technologies that I learn and try to improve.

I am a very goal oriented person and always complete my set tasks. This will motivate me throughout the project. Once I begin, I will set myself a list of deadlines to be completed on schedule. I will not be able to miss these deadlines and so will be motivated to complete them on time. The more work I have completed before each deadline, the less pressure will be on me for the next deadline. The idea of having less pressure upon me for deadlines will motivate me to get as much work as possible done.

Having a fully functional website up and available for viewing will help me in the long run when searching for employment after college. This should motivate me to make the website as aesthetically pleasing as possible and to have it running very smoothly. The backend of the website should also run smoothly with no errors. I will be motivated to have everything running well so I can show my final year project to a potential employer.

Objectives

My objectives for this final year project are to develop a fully functional website for the Munster Fleadh, gain and improve skills with various technologies and to get real world experience by working closely with a client.

Fully Functional Website

I aim to create a fully functioning, quality website that meets and possibly surpasses the needs of my client. As a consequence of this, the construction of this website will be an iterative process. To do this, I will need to have everything running smoothly and to error check everything. The front end of the website will need to be laid out well enough for a novice user to navigate easily. There should be appropriate

validation where required. An example of this would be for the register, login and contact forms.

The back end of the website should run without issues. Connections to the server should be smooth and there should be no errors with any queries that the user may send. Once the user makes an account, it should be available for them to come back to and log in whenever they want. The user should also be able to complete any tasks that they may have to connect to the server with ease.

Improve skills

I aim to improve my skills in some of the following areas; graphic design, web development, user/web design and presentation. These are all key skills that will be needed in working life after college.

I hope to improve my graphic design skills by editing the images that I will be supplied with or creating my own images to the client's specifications. Some of the images that I am supplied with may need to be changed or touched up for various reasons. I may also have to create a new image if the client does not have one which suits their needs.

I aim to improve my web development skills far beyond what they are. To date, I've had one module on web development that consisted in learning about basic HMTL, CSS, PHP and MySQL. This was in first year and I have had no further education in these skills. These are all technologies that I will require in the completion of my project so I hope to improve greatly in their use. I will also be using Javascript through AJAX and jQuery. These will be new to me so I will have to try and learn some of the basics to them before I begin. Depending on the needs of the project, Ruby on Rails may be learned.

In designing the website, I will have to improve my user and web design so that the user will not have any problems with using the website. Bad design of a website can turn users away and for those that stay, give them a bad user experience. For the website to be successful, the users that come will need to be satisfied with the service and stay.

My presentation skills should be improved during my meetings with the client. I will need to be able to get across what has been done in simple terms because they're unlikely to understand the technical side of things. Through all my meetings with the client, I hope to keep improving my presentation skills as they will be needed for the demo day at the end of the project.

Real World Experience

During the course of this project, I will be getting some real world experience in the form of working to deadlines and working with a client. The deadlines set will not be negotiable and I will need to complete relative tasks on time. I will also experience working to a client's timetable and expectations. I will have to showcase my work to them and make changes if they deem it necessary. This will show me what it is like to work with a client outside the college setting.

Report Structure

In this section, I will describe the structure of the report. It will be divided into the following sections; Research, Design and Implementation, Testing and Evaluation and Evaluation and Conclusions.

In the research section of the report, I will describe the research undertaken at the beginning of this project. This research included a heuristic evaluation of the previous website, reading previous students FYP reports, website design, website usability, touch based interaction, simple validation, password hashing, social network integration and implementation technologies. My research will be mainly based on academic papers in these fields. The research will help me in the design and development of the website.

In the design and implementation section I will go through my design and development of the website. The design and development were very closely linked. Sometimes the design would change while in development so I saw it fit to have these two sections together. The construction of the website is thoroughly documented and explained in this section.

In the testing and evaluation section, I will explain and go through the results of my testing. Testing is an essential part of any product. The developer must do this

to ensure that the product is what the user wants and not what the developer thinks the user wants. The initial testing was very informal while the later testing was more formal.

In the conclusions and further development sections I will discuss the conclusions that I have arrived at while doing my FYP. I will discuss ways I could have gone about some of my work to try and improve the website. I will go on to discuss what I would have done to the website if the time was not as constraining.

The references section contains all of the work I referenced while completing this project while the bibliography section contains all the work I have read while working on this project.

Chapter 2: Research

Existing Products

Research began by looking at the previous Munster Fleadh website http://www.munsterfleadh.ie/. If one applies Jakob Nielsen's Usability Heuristics (Nielson, 1995) towards making a website:

1. Visibility of system status: Each page needs to be clearly marked and let the user know where they are and where they can go.
2. Match between system and the real world: Every day conversational language should be used.
3. User control and freedom: There should be a home button on every page to help the users navigate and feel like they have control.
4. Consistency and standards: Consistency is everything. Make sure that links and page titles are the same. Try and keep formatting similar enough to other websites.
5. Error prevention: Have a good 404 page set up that always works. Error trapping and validation is very important
6. Recognition rather than recall: Have good labels and descriptive links. Have a good menu so the user can recognise where it brings them to.
7. Flexibility and efficiency of use: Do not remove links. People may have certain URL's bookmarked and if you remove them the bookmark will 404.
8. Aesthetic and minimalist design: Do not have too much or too little information on a page. Start with the most important information and link to less important.
9. Help users recognize, diagnose, and recover from errors: There should be a good 404 page with suggestions or recommendations.
10. Help and documentation: Integrate documentation into the site itself.

Relevant Research

Since much of the technology that was going to be used was relatively new to me, some research and learning needed to be done before the project began. I had not used jQuery, AJAX and had limited experience with PHP and MySQL. The best place I found to give tutorials and help people to learn such technologies was W3Schools (Refsnes Data, 2012). After giving myself some time to get used to these technologies I tested my working knowledge of them. When I felt some bit proficient, I began some proper research for this project.

Previous FYP Reports

Relational Database Driven Website (0702011)

by Alan Hayes (Student ID: 0702011)

This FYP was about the creation of a dynamic website to report the loss of stolen heavy plant equipment in the agricultural, industrial and construction sectors. Alan thought it was important to have the website look professional. He also wanted to make sure the functionality was fully implemented. There was a lot of emphasis on security. This is something that would need to be addressed during this FYP. The website was made using only HTML and PHP. Alan suggests that it would have been better to use a CMS (content management system) such as Joomla in any future projects he might work on.

Photo Hosting Website Snaps.ie (0875635)

by William Bennett (Student ID: 0874835)

This FYP was about the creation of a photo hosting website which allows photographers to upload and display photographs. There is a strong social media aspect to the website. It seems important to William to streamline the social media aspects of the website and keep it user friendly. The website is made using HTML and PHP. The website is also managed by Drupal, a CMS. William would strongly recommend the use of Drupal 6 rather than Drupal 7 because when using Drupal 7, several issues arouse and took time to fix.

Website Design

When developing a website, the developer should try to ensure that they keep the layout and overall design of the website consistent. A consistent layout will help users of the website in a number of ways. Doubé and Beh state that consistent navigation and layout of the website will help 'chunking' which will lower the cognitive load of the user (Doubé and Beh, 2012). Chunking joins together elements so that there will be fewer elements. Fewer elements mean that there is less strain on working memory (Cherry, 2012).

According to Doubé and Beh, it is easier to learn how to navigate your way through a website when the visual elements of the website are close to each other (Doubé and Beh, 2012). It can get difficult for the user to learn how to navigate the website if the visual elements are far from each other.

According to Gibson, navigation through a site is important. It is crucial to optimise page hopping. The user should be easily able to get to different pages (Gibson, 2006). He also stated that menu's and headers should be consistent. If there is a change made to one of them that change should be made to all of them. He also went on to say that page sizes should not exceed 60 kilobytes. This is because we have to assume that not everyone has a high-speed internet connection. He also goes on to use the acronym KISS which stands for 'keep it simple stupid'. This encourages you to stick to how things are usually done as much as possible to avoid confusing novice users.

There is an interesting quote from Poblete and Baeza-Yates; they say that a website "is not simply a collection of pages, it is a network of related pages" (Poblete and Baeza-Yates, 2006). I take this to mean that the pages should be connected in their appearance and function and not just an assortment of different pages linked together. There must be something common between the pages.

Liu and Yang state that individual websites are more organised then the entire web. They also go on to say that the hierarchical model is a good choice because of its ease of use and understanding (Liu and Yang, 2005). Individual websites would be more organised then the entire web for a number of reasons, mainly because that it would be a single developer or a team working on one site. However, the entire web is

made of websites from different developers with different styles. The hierarchical model is a model that uses the parent/child relationship. Each parent can have many children but each child can only have one parent/two parents. This is a many to one relationship. This can be brought over to databases with tables, rows and columns.

Website Usability

If a website has a good design, it is easier to present information well. This can help the user to process that information. However, bad design will usually lead to bad usability (Reinecke and Bernstein, 2011). This could lead the user to go to a different website to meet their needs. To try and keep the user, website designers aim to make their website as aesthetically pleasing as possible while still maintaining good usability.

Web Designers can hypothesize as to what users will consider aesthetically pleasing. Designing according to the laws of Gestalt psychology (proximity, similarity, prägnanz (figure-ground), symmetry, common fate, closure) is one way to achieve this (Tuck, 2010). It must also be noted that personal preference and culture can influence this (Reinecke and Bernstein, 2011).

Touch Based Interaction

There are many reasons that touch based interaction is going to be in the back of a developers mind as they develop a website. Touch based interaction is the most direct form of interaction. The advent of the smart phones has made touch interaction very important. There is very little separating the user from the information. Touch based interaction is very easy to use and does not require practice to get it right. Touch based interaction is more robust than a system such as the mouse (Leiva, 2011).

As touch based interaction is becoming more widespread, many websites have mobile versions. These mobile versions can vary in size. They may have been made for a specific size screen not optimised for other sizes. A solution to this would be to use the Goldilocks Approach (Quick and McKreever, 2013). With this, the website developer only need make one page and then the size of the page will be scaled to suit which ever screen it will be displayed on.

When the Goldilocks Approach is being utilised, the page is split up into columns. If one of the columns cannot be displayed in full, the content of the website will automatically scale itself to the remaining columns. By doing this, the website should be able to be displayed on all screen sizes.

Simple Validation

Most user data is obtained using HTML forms. However this can lead to some security issues. The data provided by a user can be incorrect. This can simply be a mistake or something more malicious on the users' part. As a result of this, forms often ask for more information than is required for the request to the server to happen. This is in order to "sanity-check" the data (Camenisch et al, 2006).

A lot of the information from forms is transferred over an unidentified network which can lead to identity theft and fraud. Some forms include Javascript to try and validate the information that the user tries to submit. However, this can fail if users from other countries use the form. An example of this is the length between Irish and English phone numbers. Javascript can validate anything from passwords to email addresses.

Password Hashing

Message Direct (MD) is a group of hashing functions. MD5 is one of the most popular of this group. MD5 is optimised for 32 bit processors. According to Teat and Peltsverger, the digest size of MD5 should be increased (Teat and Peltsverger, 2011). Secure Hashing Algorithm (SHA) is another group of hashing functions. SHA2 functions are the most secure in the group. However, SHA3 may yet exceed the security of SHA2 (Teat and Peltsverger, 2011). With enough time and resources, any password hash can be broken. The key is to try and limit time for malicious activity. MD5 should be suitable because of the scale of the project. It will protect the users and not use up much computational power.

Social Network Integration

As social networks have hundreds of millions of users, it makes sense to integrate them with other websites. Social networks are used daily. Integrating social networks

into your website has many benefits; users can share the site, share content from the site and as a result the website may attract new users (Northwoods Web Design, 2012). As the traffic grows for the website it's ranking in search engines also improves (Carlos, 2013).

With Twitter, the users of the site and participants of the Fléadh will be able to talk about and recommend the event in real time. People use Twitter to talk about their daily activities and share information (Java et al, 2007). With hashtags (#) and trends it is a very good idea to integrate Twitter into your website as the site itself or information about it will spread over Twitter.

Implementation Technologies

HTML/HTML5 and CSS

HTML is a standardised markup language. HTML elements form the building blocks for websites (Kasten, 1995). HTML allows the developer to embed images and objects into the web page. Scripts such as Javascript can be embedded and affect the behaviour of web pages. CSS can be used with HTML to outline the appearance and layout of the webpage.

Since HTML 5 is usable albeit still under development, it can be incorporated into aspects of the website. There is a variety of new tags along with the deprecation of older tags. With HTML5 developers can embed audio and video straight into web pages (Moreno et al, 2011).

CSS is a style sheet language that's used to define the appearance and layout of a webpage (Lie and Saarela, 1999). CSS files are generally linked in the head of the HTML file and describe elements in the document. You can have CSS code in the HTML file itself, but it is better practice to have it in a separate file as it can be linked to many HTML documents.

AJAX/jQuery

AJAX stands for Asynchronous Javascript and XML. It is a group of interrelated techniques for web development on the client side. It is used to create asynchronous web applications. It can send and receive data to the server in the background without

interfering with the current page (Wang and Zahadat, 2009). AJAX is not a single technology. HTML and CSS can be used together to edit look of the information.

JQuery is a multi-browser Javascript library. It is supposed to provide a simple and easy way to script HMTL on the client side of things. It is open source (Lerner, 2009). It helps developers to create dynamic web pages and applications. JQuery plug-ins can also be made by developers. This helps to extend its functionality. jQuery can be incorporated into websites with ease. There are a variety of scripts for image viewers, audio players and video players available for use. These scripts are there for use and to be improved upon.

Ruby on Rails

This is an open source web application framework for the ruby programming language (Lerner, 2006). It gives the web developer the ability to get information from the web server, query the database and render templates out of the box. It uses the Mode-View-Controller (MVC) architecture pattern (Lerner, 2011).

Ruby on rails would be very useful for this website; however, it may replace PHP. MySQL or SQLite3 would be used with it. Ruby on rails may not be compatible with all servers. PHP is a standard so would be. They both have advantages and disadvantages but PHP would be better suited for this FYP.

PHP

This is a general purpose server side scripting language (PHP Group, 2013). It is one of the first server side scripting languages with the ability to be embedded into html file. It does not need to call an external file. It is free to use. This may be very useful for setting up forms. It is very useful for including pieces of code (Coggeshall, 2001) on multiple pages such as the header, footer and menu. This can keep pages consistent. PHP can also have many functions when used in conjunction with MySQL. However, to view PHP files, a web development environment such as WampServer is needed (Bourdon, 2013).

SQL/MySQL

SQL is a special purpose programming language designed for use in relational database management systems (RDBMS) (Lerner, 1997). With this you are able to

insert data, update and delete, schema creation and modification and data access control.

MySQL is the worlds most used open source RDBMS. MySQL is a popular choice of database for use in web applications. phpMyAdmin (phpMyAdmin devel team, 2013) would be used in conjunction with this. It is a graphical interface that is easy to use. Databases can be made and tables can be made in these databases. The admin of the website would be able to look at the variables and contents of these tables.

Chapter 3: Design and Implementation

Development of Initial Idea

The idea at the beginning of this FYP was a simple one. After all the relevant practice and research was completed, work began on a mockup of the homepage. This can be seen in fig. 1.

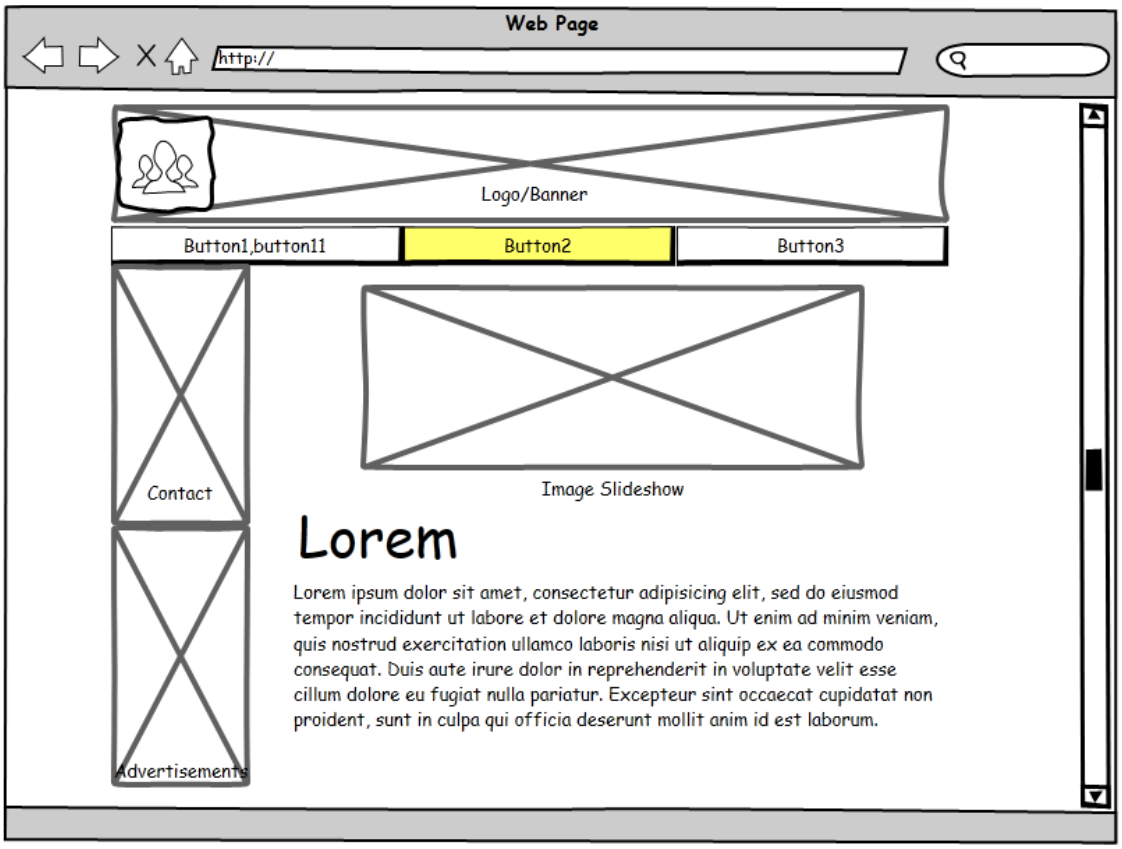


Fig. 1: Design of the initial mockup.

A menu that was always visible, but not too obvious is something I sought after greatly. I conducted some research online and found several ways to do this. I found an example of this type of menu which fit all my needs (Nikos Tsaganos, 2012) and decided to use it. Before it could be added to the website, the source code had to be edited and customised. Many things were changed; the colour, the width, the size of the tabs among others. One of the main issues with the menu was that when the page was scrolled down, the menu changed from the size it was and took up the width of the page. To fit into the aesthetic of the page, it needed to stay a certain width (980px) and not change. This was easily fixed soon thereafter. A simple banner was made as filler until the clients decided on the official banner. The banner and menu

content were then put into a separate PHP file so they could be included in each page. This way they would stay consistent throughout. This will also keep the body of my code cleaner and tidier.

The next task was to get a slideshow working on the homepage. This would showcase some images the clients wanted users of the website to see. A lot of image viewers appeared during the search, but a slideshow was needed so these were ignored. After a good search, a slideshow which was functional along with being aesthetically pleasing was selected (Nathan Searles, 2010). I then attempted to insert this slideshow into my code. With some trial and error, it began to work eventually. However, some of the new code was conflicting with the older code of the menu and needed to be fixed manually. The slideshow was appearing above the menu even though the menu was supposed to be on top. However, this was fixed easily using z.index. Soon a few other problems began to arise. The slideshow and the pagination below it were not working. Several attempts were made to fix this but to no avail. I then recalled reading something called jQuery noConflict a few weeks prior to the incident. I proceeded to read about jQuery noConflict (jQuery Foundation, 2013) and then integrated it into my code. It worked straight away! There was still one issue after this, a slight problem with the pagination; it was off centre. A quick change to some of the CSS solved this problem. The first iteration of the homepage can be seen in fig. 2.

For simple page hopping (having one click between pages) to work, I deemed it necessary to have a working menu with a sub-menu. Unfortunately, the menu I was making use of was not built with a sub-menu. This meant that the sub-menu would need to be constructed by building upon the initial code. Making this sub-menu proved to be more difficult than first anticipated. It consisted of a lot of trial and error. It took a mix of my own code and the initial code supplied to get the sub-menu working somewhat. One of the main issues with the sub-menu at this point was that it was not lining up properly underneath its parent. After trying lots of different attributes and values, the sub-menu was eventually lining up correctly.

Now that the sub-menu was working correctly, cross browser compatibility was checked. Since Chrome was primarily what the website was tested and previewed in, it ran without any issues. The website also ran in Firefox and Opera without issue.

However, when tested in IE (Internet Explorer) there were several issues. IE did not seem to recognise the '*' value which set the attributes to auto in the other browsers. As a consequence of this, the height and width needed to be set for the images. In IE, there seemed to be a green border around all the images whose border was not set to '0'. I corrected this and will continue to do so in the future. This only seemed to be an issue in IE.

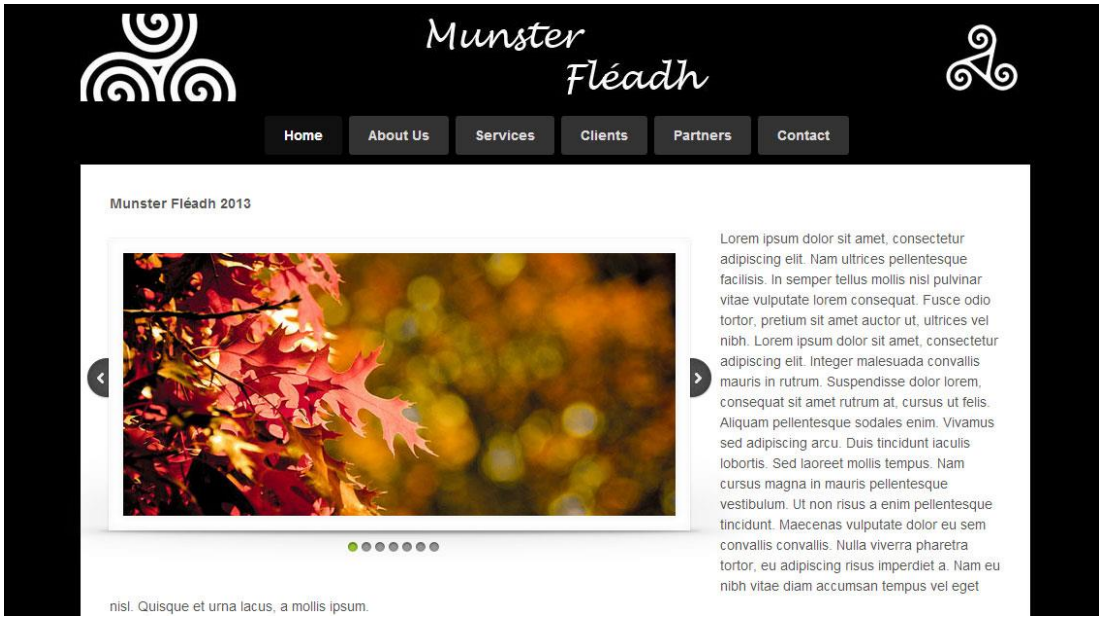


Fig. 2: The first iteration of the homepage.

I thought it was important to keep my files and code in order. As a consequence of this, the Javascript and code that was given with some of the jQuery plugins were tidied up. I thought about copying and hosting the jQuery version code, but decided against it. It would be better to receive the code from Google for two reasons; in case there were any changes made to it in the future and it would not need to be hosted by the clients.

With the current menu setup, the colour of the tab did not change upon hover. While this is not a big issue, it would be an aesthetically pleasing feature. The CSS was changed in various ways but to no avail. Eventually it started working. The initial code for the menu tab to change colour with hover was copied and applied to the sub-menu code.

At this point, it was time to start looking into an audio and video player. After some research, it seemed very advantageous to use jPlayer (Happyworm, 2009).

There were many samples of how it could be used included and it was very customisable. The playlist players seemed like the best options because this would save on having multiple players on the same page. These were separated from the rest of the samples and the files that were not needed were deleted. The colours of the players were changed from bright blue to black (using Photoshop) by changing the colour of the image that they took their buttons from. The colour changes I made to the image worked for both the audio and video since they use the same base image and CSS files. A filler spiral was put in place instead of the default 'JP' on the play button for when video player was not in use. These two players were then incorporated into two different pages and were checked to make sure that there was no conflict between them or any of the previous jQuery versions or code used.

Work on First Iteration

At this point I wanted to put the three pages that had been created together. This was a good idea because it is important during projects to be aware of the progress that has been made. The included PHP file made this quite an easy task as it only took three or four changes to the menu and the three pages were linked and working properly.

The next page to be completed was the contact form. A few suitable examples were found online. One in particular stood out (HTML Form Guide, 2011). It was customised as was done with all the code before it. Upon trying to integrate it into the previous code, a few issues arose. One of the issues concerned the placement of some of the PHP files that came with it. This proved rather difficult because of the nature of the way the code was being implemented. It was going to be included as the menu and banner had been previously. In hindsight, it may have been easier to just copy the code to the contact page, but after much work the contact form functioned properly. There was also a conflict in the new CSS file that was supplied with the contact form. In previous CSS files there was already an id named container. As this was not the first time this issue was encountered, it was quite easy to remedy. Container was simply changed to container3 as there was already a container2 from earlier. The completed contact form can be seen in fig. 3.

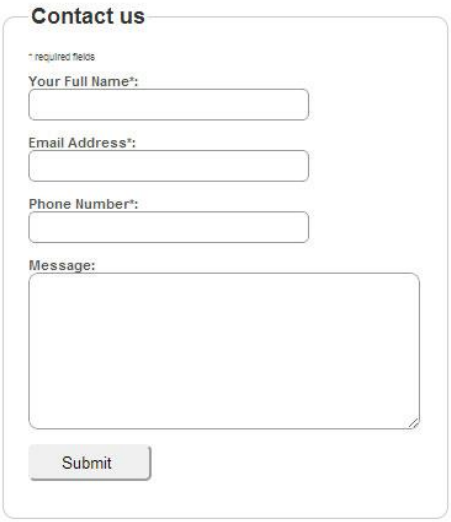


Fig. 3: Completed contact form.

The menu headings were changed to something similar to what they might actually be, although this is still subject to change. It was then time to include an image viewer in the media section. There were a few functionalities that I was looking for in the image viewer. Thumbnails, a large image viewer and pageation were very important. Through research, one was found that met all these criteria (Awkward Group 2012). There were many versions of this, but it was the second version that would be used. This still needed to be customised. A lot of code was deleted from the CSS file and some of the images that were included were deleted. Some of the HTML code was changed before it was added to the previous code. When the code was being integrated, I noticed that the jQuery framework it was using was a version newer then the one already in use. I tested this to see if the new framework would support the old one and it turned out it would. There were no other conflicts when bringing the new code into the already existing code.

At this stage, I began to think about adding more functionality to the website. I thought of a way for people to make their own schedule. There would be a form and the users would have to input their information. This way the user would be able to input their choices and then bookmark the page to come back later. With this, there was no need for a login. The form was set up so the user would have to input their name, age, phone number and email address. This is not final and may be changed. There would be check boxes for all the different events going on throughout and the user would be able to select the events that they want to attend. When the user

submits the form the input would be displayed on the next page. This is displayed in fig. 4. This was difficult to get working properly.

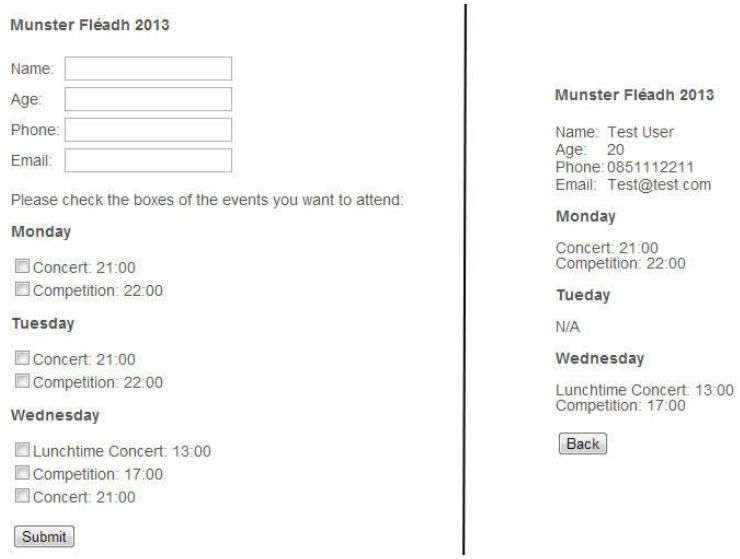


Fig. 4: First scheduling idea before and after submission.

Work then began on validation of the form. Javascript would be used because it can be done on the client side rather than the server side like PHP would have to. PHP would have to send information to the server and then wait for a response. Using Javascript for validation would be much better for the end user. I looked at a few different scripts online but could not seem to incorporate any of them into the previous code. Some very basic validation was found on W3Schools (Refsnes Data, 2012) and I then based my code from that. The code had to be made more complex to work for this situation. The name, phone number and email address would be validated and a dialog box would appear if the correct information was not filled in. The code consists of variables being stated, then an ‘if’ statement and some ‘else ifs’. The validation for the phone number contains two further ifs to check for a numeric value and number of characters. The first just checks if the field has been left empty. Some text was then added to let the user know what fields must be filed in. These fields were marked with a ‘*’.

After this I thought it would be a good idea to set up a login system. The same form would be used, but with two added fields. The extra fields were for the users’ password and another one so they could confirm their password. For this to work

properly the GET functions had to be changed to POST functions. A database and table would be needed to insert the information from the form. The data would still be displayed as normal but this time on the page with the checkbox input form. The user could log in and get their information again if they wanted. To do this I tried to make use of the where clause. However after quite a bit of trial and error work, with no success, I decided to cut my losses. A different approach was needed.

All the work that had been done with the checkboxes was thrown out. I decided to make a new database and a new table to put in the users login details. This database takes in the name, password, phone number and email of the user. By default, the userType of the regular user will be 0 and the userType of an admin will be 1. A register form was set up to take the users details when they registered. If the email address was already registered, the user would be informed and have the option to try again. A login form was set up so the user could log in.

A basic admin page was also set up. If you log in and have the correct user type you will be brought to a page that has an option to edit the tables on the database and add a new event or delete an event. You can also view the events. To add an event the admin will fill out a simple form which was somewhat validated. This was done with an insert if the event ID number was not taken. An ‘if’ statement was then set up to check whether or not the ID number was already taken. A delete was also set up where the admin just has to enter the ID number to delete a row from the table. If the ID number does not exist, the admin will get an error message. The admin can also view all the events so they can check the ID number, for this, ‘where’ and ‘while’ statements are being utilised. Where the conditions are met, the contents of the rows are displayed using a ‘while’ loop. Fig. 5 shows the tables that are being used.

A very basic user page was set up. When the user logs in, they have the option to edit and view their schedule. However, the page to edit the schedule has not been created yet, because I was having difficulty with the page to view the schedule. It is most likely something very minor.

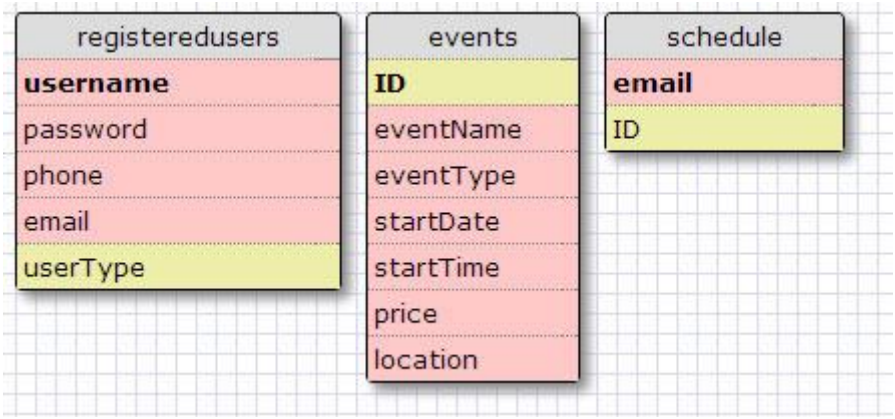


Fig. 5: Tables being used.

Work on Second Iteration

Work began on the 'View Your Schedule' section after a brief Christmas break. There were some issues with this before the break that needed to be fixed. No matter what was changed, the same error message kept appearing – 'mysql_num_rows() expects parameter 1 to be resource, boolean given'. A lot of trial and error and changing values and variables of the query led to the same result. After some research, sessions seemed like they could be a solution (Refsnes Data, 2012). After some experimentation, the sessions were working correctly. However, this lead to another issue where when the user tried to return to the previous page they got a dialog box appearing about resubmitting the form data. This can be seen in fig. 6. This issue needed to be fixed as it impacted badly upon usability.

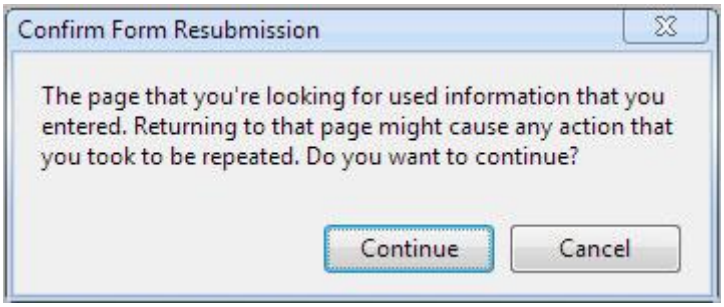


Fig. 6: Form resubmission dialog box.

Sessions were used in another way to try and solve this issue. This did not work and resulted in the same outcome. Page redirection was then tried. However, the first few attempts led to infinite loops. This meant the page would not even load so another attempt at stopping the resubmission was needed. When following a link back to the previous page, an error of undefined indexes in the username and password

fields were displayed because nothing had been submitted by the user upon the page reload.

As a result of the issues that had arisen while trying to get the login system working, I saw it fit to scrap a lot of the previous work and start again. However, some of the old code was saved. At this point, the login and register form were put onto different pages so as to be more aesthetically pleasing. The admin login would also be separate to the user login. This is on a page that only admin shall be able to access, but this may be changed at a later date. To begin, a simple working login system was needed (PHPeasystep, 2012). This then had to be customised and tested to make sure it worked with the current database. The newer and older work was then combined. After some difficulty, the two pieces of code began to work together without conflict. When registering an account, another issue became clear. The default user type is set to " " (null). For the normal users, the user type needs to be '0'. To remedy this, '0' was inserted into the table for userType when a user registered. If an admin makes an account, their user type can be changed manually.

After this, the admin side of things was created using the same setup. The same set of steps was followed, only user type 1 was selected where applicable. A lot of the older code remained and could be combined with the new code. This process was much easier than making the login for the user as the process had already been completed once. The admin section remained hidden, but as mentioned previously, it may be linked to the normal login at some point. Quite a big issue was discovered, all of the pages could be accessed even though you were supposed to be logged in to see them. There seemed to be an error with the code. After a lot of trial and error, it appeared to be working properly. However, no one could log in now. A lot of different ideas were tried but to no success. After much looking through the code, it became apparent that the session was never started. When the session was started, everything worked perfectly. This process must also be done with the user login and can be seen in fig. 7.

```php
<?php
session_start();
if(!isset($_SESSION['user'])){
    header("location:login.php"); // not logged in
}
?>
```

```php
<?php
session_start();
if(!isset($_SESSION['admin'])){
    header("location:adminLogin.php"); // not logged in
}
?>
```

Fig. 7: Session code used for the user and admin logins.

After this the process was applied to the user login. It would make sense to redirect both the user and admin if they arrived on the login page and had already logged in. This would make a lot more sense rather than have people logging in every time they went to the login page. To do this, the redirects were set so that if the users or admin were logged in, they would be redirected to the corresponding pages. This makes the website more user friendly and makes it less likely that the user will lose their way. A page was developed for the users to edit their schedule and a note was left on this page saying that it was under construction. Error trapping now needed to be done. It was discovered that some of the Javascript had broken while combining the new and old code. The Javascript for the login was an easy fix. All that needed to be done was change some of the code at the start of the form. The Javascript for the register proved more challenging. After checking nearly everything it could be, it turns out that while combining code; the input ID of name got changed from "name" to "email". Changing all of this back fixed the issue. The minimum requirement for a password was then set to six characters. The same Javascript was then applied to the Admin page to keep the website consistent.

In an attempt to personalise the user login, the users name was inserted, along with a welcome message, by using sessions when they logged in. This was easily done as there was something similar to it in the old version of the website. It uses a while loop with 'mysql_fetch_array' to work properly. The code was then brought over to the admin side of things to try and keep everything consistent. This can be seen in fig. 8.

```
echo "Welcome back " . $row['username'] . "!" . "<br /><br />" . $row['phone']
 . "<br />" . $row['email']; // To display name, phone and email
```

Fig. 8: Code to display personal details.

User passwords should be encrypted to try and protect their information. To do this, the passwords were hashed using the md5 function. This is not the most secure way to encrypt passwords, but it will do for the purposes of this website. It was fairly straight forward to do. All that was added to the code was '$password = md5 ($_POST['password']);' this can be seen in fig. 9. This took the input from the form, hashed it and made it equal to $password. This had to be added to the login screen too so when the user enters their password it would match the password stored in the

database. However, there was an issue when trying to log in after a new account was registered. The password entered did not seem to match the one stored in the database. To find out the issue, I printed out the hashed password and then compared it to what was stored in the database. The issue was that the passwords could only be 16 characters long. This was then fixed straight away. The same process was used to make sure that it also worked with the admin login. A simple password when hashed using md5 as seen in fig. 10.

```
$password = md5 ($_POST['password']);
```

```
qqqqqq = 343b1c4a3ea721b2d640fc8700db0f36
```

Fig. 9 & 10: Code to hash the password and an example of the output.

There was an issue when registering names with an apostrophe in them. It gave an error as the database was not able to handle it. To remedy this situation, an apostrophe was replaced with a space if the user inputs one into the form as seen in fig. 11. After this was working properly, I thought it would be a good idea to make all of the email entered by the user lower case as some people may enter capital letters in places. This was done simply enough by using the 'strtolower' to bring the text input by the user to lowercase as seen in fig. 12. This was then applied to the admin and user login files so everything would be consistent.

```
$name = str_replace("'", " ", $_POST['name']);
```

```
$email1 = strtolower ($_POST['email']);
```

Fig. 10 & 11: Code to fix issues in registration.

Addition of Content

At this point, it was a good idea to fill the website with content. Some of the content was taken from the current website so the clients would get a proper idea of what the website would look like when finished. After some of the text was taken and the banner changed to an image that had been sent to me, it became obvious that the whole colour scheme of the website would have to change as it did not suit the orange banner. Blue compliments orange and many different shades of blue were tested and eventually one was found that complimented the banner well. Then the colours of the menu, menu hover and menu background had to be changed to match this. Colours

that were similar, but different enough that they could be distinguished from the background, were found and used. This can be seen in fig. 13.

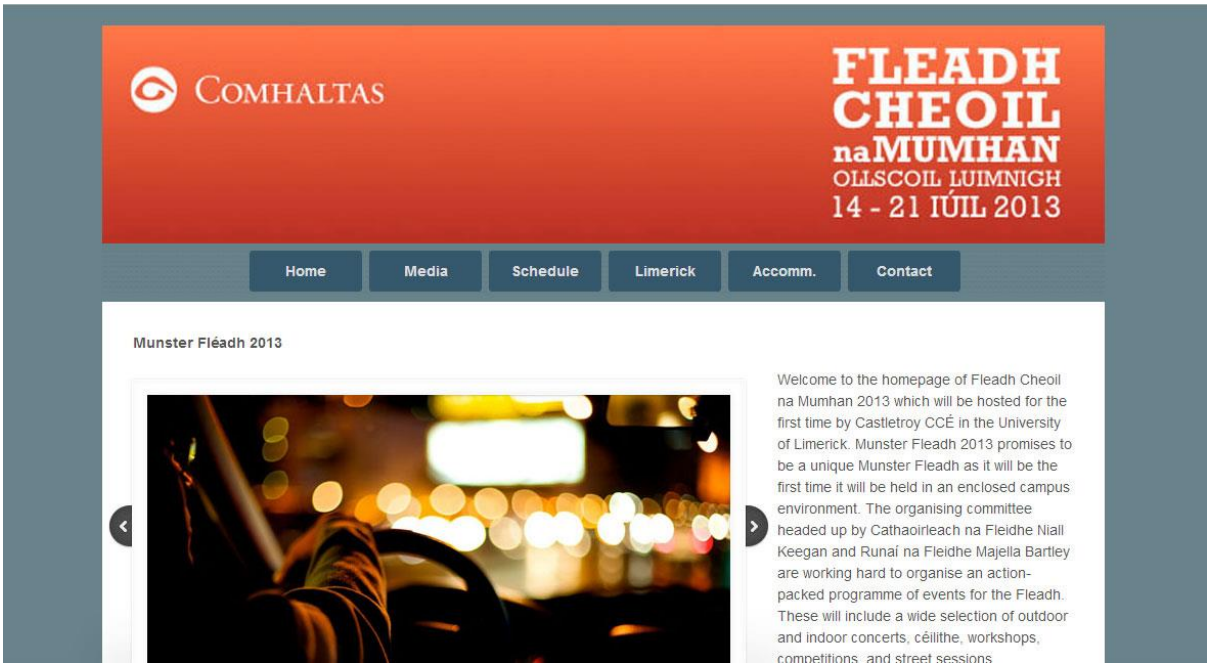


Fig. 13: Screenshot of the second iteration of the homepage.

Social media integration is a major part of the modern website. This website would have several forms of social media integration. This was achieved by the addition of a Facebook like button (Facebook Like, 2013), Twitter follow button and a Twitter 'hashtag' (#) button (Twitter Developers, 2013). The Facebook like button will allow users with Facebook to like the Munster Fleadh Facebook page. Users with Twitter will be able to Follow the Munster Fleadh on Twitter and also tweet with the #FleadhnaMumhan. After the content was inserted, it would need to be arranged in an aesthetically pleasing manner.

The social media buttons were placed in the footer so that they would appear on every page. This way the user has ease of access to them should it be required. The three social media buttons were placed on the left and the Copyright text was pushed out to the right of the footer. This still looked somewhat odd so the colour of the footer was changed. None of the colours made it look any better so instead, a horizontal line was used to separate the content and footer section. This can be seen in fig. 14. This worked quite well helped to integrate social media into the website.

Fig. 14: Screenshot of the new footer.

Continued Work on the User and Administrator Pages

It was about time to get back to the server side of the website. In an attempt to complete the page where users could edit their schedule I got a basic layout from (Anon, 2001) and adapted it to my needs. In this, the check boxes were incremented automatically. Work began on the code and all of the events were then displayed with checkboxes next to them. The checkboxes were to be checked if the user had already selected the event as can be seen is fig. 15. In an effort to get this to work, there was a new column added to the table 'schedule'. This would be to signify that the checkboxes had been selected. An 'if' statement was used to check a box if they matched the variables given to the 'if' statement, but this did not seem to work. Everything just came out checked.

```
10            $nr = 0;
11            $result = mysql_query("SELECT * FROM events ORDER BY ID");
12            $result1 = mysql_query("SELECT ID FROM schedule WHERE checkbox = 1 AND email = '$_SESSION[user]'");
13
14            echo '<form name="box" method="post" action="editCheckbox.php">';
15            echo "<table><tr class='spaceUnder'><td>Box</td><td width='100'>Event ID</td><td width='100'>Event Name</td><td width='100'>Event Type
</td><td width='100'>Date</td><td width='100'>Time</td><td width='100'>Price</td><td width='100'>Location</td></tr>";
16
17            while($row = mysql_fetch_array($result))
18            {
19                $nr++;
20                if ($result1){
21                    echo "<tr><td>". "<input type = 'checkbox' name='$nr' value=%s checked>" . "</td>";
22                }
23                else{
24                    echo "<tr><td>". "<input type = 'checkbox' name='$nr' value=%s>" . "</td>";
25                }
26                echo "<td>" . $row['ID'] . "</td>";
27                echo "<td>" . $row['eventName'] . "</td>";
28                echo "<td>" . $row['eventType'] . "</td>";
29                echo "<td>" . $row['startDate'] . "</td>";
30                echo "<td>" . $row['startTime'] . "</td>";
31                echo "<td>" . $row['price'] . "</td>";
32                echo "<td>" . $row['location'] . "</td>";
33                echo "</tr>";
34            }
```

Fig. 15: Code used to edit the users schedule.

However, after trying with the check boxes for a while to no avail, a different approach to it was taken. To add an event, all the events are displayed and the user needs to input the event ID to add it to their schedule as seen in fig. 16. To delete an event, all the events in their schedule are presented and they will have to input the event ID number to delete it from their schedule as seen in fig. 17. The user still has the option to simply view their schedule as seen in fig. 18. There may be some functionality added to when the user views their schedule.

Fig. 16: Screenshot of adding an event to the schedule.

Fig. 17: Screenshot of deleting an event from the schedule.

Fig. 18: Screenshot of viewing the schedule.

The admin page then had to be updated. Instead of everything being on one page as it had previously been, it was changed so that it was similar to the user page with the option to delete an event on a different page. The option to add a page is still on the main page but this may be changed at a later time. This can be seen in fig. 19.

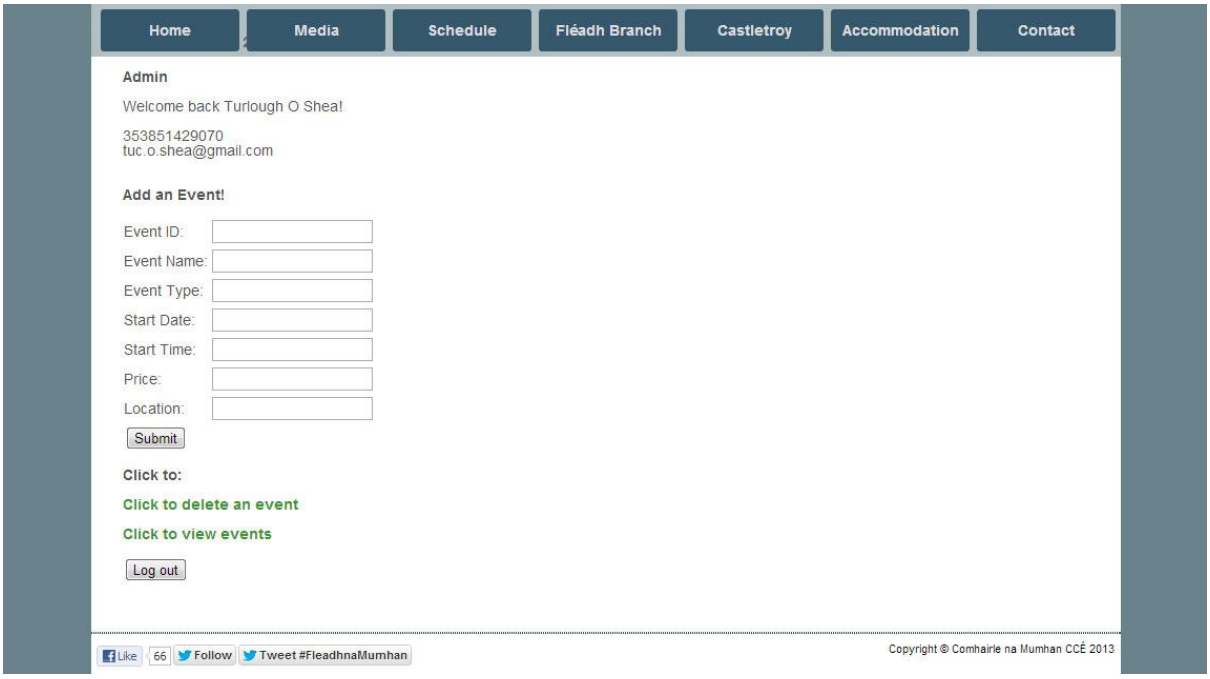


Fig. 19: Screenshot of the admin page.

Preparation for First User Test

At this time, the clients got in touch and requested a few things. They seemed to like my work so far, but wanted some changes, which was to be expected. They liked the functionality and the design of the website as a whole. They requested a few things, but as the website was not full of content, this was to be expected. The first thing that was done after the meeting was populate the contact page with contacts of some of the members of the committee. They wanted this along with a contact form which was already in place. The next thing they wanted was for the image viewer and the slideshow to be populated with images. Some of the images that they wanted were sent to me. They also said that I would have to retrieve some images for myself. A lot of the images were very large and needed to be resized and optimised for web use. This would speed up the loading of the images and web page as a whole. Images were selected for the slideshow on the home page. Some of the more appropriate photos were selected from the slideshow on the current page and then inserted into the slideshow on this homepage. The images that did not make the slideshow were resized and optimised for web use and then inserted into the image viewer. These were not captioned as the clients may have captions for the pictures themselves. The favicon.ico was changed to the one that was being used in the current site. For the changes to take effect the cache of Chrome needed to be cleared. The filler symbol on the video player was then changed to match the image of the favicon.ico.

Another request of the client was to have an extra item in the menu so this was completed straight away. After the extra tab was added to the menu it now had seven tabs. This was the perfect number according to George Miller. According to the experiments of Millers in the 1950's people could only retain between five and nine items in their short term memory (Miller, 1956). According to Jeff Atwood, if users have to remember more than 7 pieces of information while using your software, a redesign should be required (Atwood, 2006). The seven tabs were made as wide as possible to try and take up as much space as the banner and main section of the page. This was simply trial and error between the width and padding of the menu items. Using the old website as a reference, the pages of the 'Accommodation' tab were populated. However, there seemed to be text missing from the Bed and Breakfast page and the flier from the 'On-Campus' page was not accessible. The clients were

contacted to see if they could remedy this situation. Making this page had to be put on hold until the clients supplied the correct materials.

Work then began on the ‘Castletroy’ section of the website. Four pages were made in this section; ‘UL’, ‘Venue Map’, ‘Committee’ and ‘Directions to UL’. The pages were then filled with content; however, some of the images that were on the old website were still inaccessible so there would be blank spaces in these until the correct materials were supplied. They would be filled in when the materials were supplied. In the traveling section, Google maps (Google Maps, 2013) was used. This would let the user find directions to UL (University of Limerick). UL was set as the destination so all the user had to do was fill in the starting point of their journey.

The clients had supplied one audio track; I then proceeded to add it to the video player. To do this it needed to be placed in a specific folder. The code for the player then had to be changed so it called the right song. It also needed to display the correct artist and track names. The native support for the audio player seems to be .mp3 and .ogg so Audacity was used to convert the .mp3 to .ogg. They were put in the folder but to no avail; the song would not play. The audio player did not seem to be working. The reason the track did not play was because of an issue with the name of it. As it had a long title its name was changed to ‘1’. This way there would be fewer mistakes when the audio player calls the songs.

There were some drawings of characters playing instruments that the clients wanted included in the website. These were mostly .ai files (Adobe Illustrators file type). These were imported into Photoshop, scaled and optimised for web use as .jpg’s. Since the client wanted some of these spread throughout the site, they were placed in unobtrusive but obvious places. A lot of white space was removed by adding in these minimalistic images. Fig. 20 is an example of the sort of placement of the images. The images were placed in a table with two rows and first row was adjusted so the image would be in a desirable position. There was a lot of trial and error when trying to place them as the images were not the same size. Some of these tables were aligned to the left and some were aligned to the right. In some cases, a blank image had to be used to keep the content of the page in the centre.

Fig. 20: Screenshot of an example of the image placement.

It was time to change some minor parts of the site, that could not be changed before. The number to contact for accommodation was now added to the contact page. There was a page made for the Castletroy CCÉ along with a few pictures added to the bottom of the page. Pictures were also added to the bottom of the 'UL' page. With all these changes the menu needed to be checked to see if it worked for every page. Two more images were then added to the image viewer. The 'Committee' page was completed by adding a picture of the committee and some text at the bottom of the page.

After this, the clients still had a few small changes they wanted made. They supplied captions for the images in the image viewer. They wanted a certain image in the slideshow. The 'Bliain na Cruinne' logo then had to be added to the home page. Two of the hotels had to be removed for the time being. They were just commented out (using <!-- and -->) in case they needed to be re-added at a later date. The Castletroy CCÉ album was supplied, and I was instructed to include some of the songs from it on the website. To do this, the CD had to be ripped and converted from .m4a to .mp3 and .ogg. After listening to the songs, four were selected and added to the audio player. More music was then received. These were tracks from 'Goitse'. They were in aif format so they also needed to be converted .mp3 and .ogg. As there were more songs being added, the drawing on the left side had to be moved down

more. There was a heading placed at the top of the content of every page to tell the user what was on that page.

Work soon began on the ‘Fleadh Branch’ section. The clients requested some contact details put on the ‘Castletroy CCÉ’ page. It took some time and multiple attempts to get the contact details into an aesthetically pleasing position. After this, the clients wanted a page to be created for lessons that the Castletroy CCÉ supply. They also wanted a registration form so people would be able to print it and fill it out in advance. The lessons were supplied in an excel file. As a consequence of this, all of the details had to be added to a table. After this, a new section called ‘Dance Packages’ needed to be set up. This consisted of an image with prices of dance packages. The images were then added to the ‘Venue Map’ and ‘On Campus’ sections.

It was time to make sure that everything that had already been completed still worked. As a result of this, it became apparent that the contact form was not working. A few of the functions in it had become deprecated. Instead of trying to fix this, I decided to try out a new contact form entirely. After much searching, it was clear that one stood out (Rocheleau, 2012). It needed to be customised before it would be integrated into the website. However, some problems soon arose. There was a conflict with some of the jQuery. To overcome this jQuery no conflict code that was used previously (jQuery Foundation, 2013) had to be used again. There was still an issue, the text and text boxes were in the centre and would not move to the left. There was a conflict in div names. There was a crossover when the div was called container. When the name was changed to ‘container content’ it started working correctly again. Some of the finer details such as the size of the form, textboxes and where the error messages appeared had to be customised. One of the drawings was nearly included on the page, but because of the way the page changes size when the contact form is submitted, this was decided against. For some reason the footer seemed to change slightly so it just needed to put it back to the way it was before by changing the size of the table. The finished contact form can be seen in fig. 21.

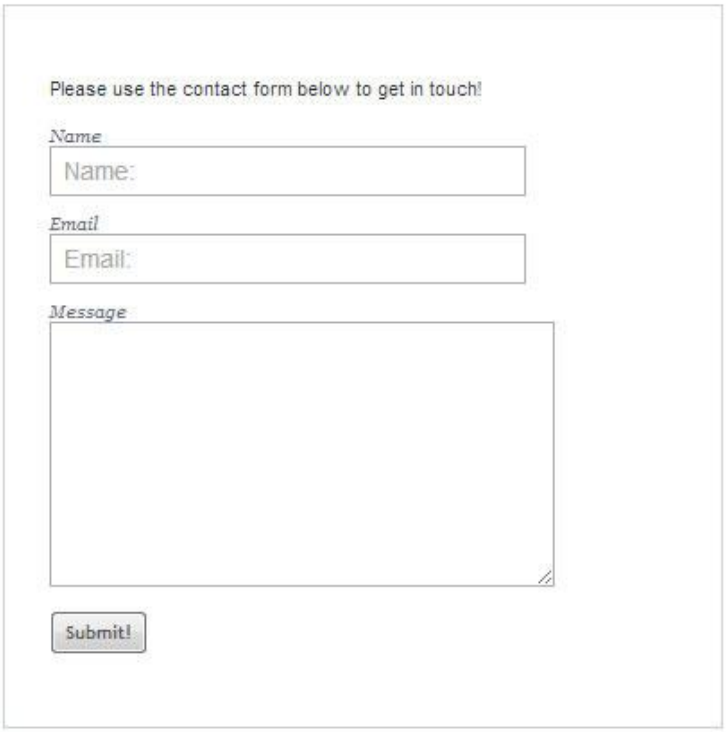


Fig. 21: Screenshot of the new contact form.

Since the admin page was not linked anywhere on the website, navigation to it was harder. As a consequence of this, it became necessary to include it. It should not be included in in the menu so instead it was linked in the login page itself. A number of places it could be inserted were considered, but in the end it was placed between the heading and the login form. The text was made smaller than usual and put it in italics. This way the admins of the website could get to their login. This was also done on the admin login. Now if someone found themselves on the wrong login screen, they would be able to return to the correct login screen easily.

Text was added to the audio, image and video section so the user would know what they were and why they were there. To populate the video section of the website videos of the artists who supplied music were acquired. Only .webm videos were supplied. Since the video player supports more types as a backup it would be a good idea to convert the videos to .ogv or .m4v. Converting to one of the above formats should be enough. Screenshots were taken during the video for the 'poster' of the videos.

More functionality needed to be included in the login section of the site. Up until now, the user had no way to change their password. This should be similar

enough to the code for registering an account. Javascript will be used to validate the form. The Javascript was basically the same as what was being used for registering aside from an extra line or two of code. If the password the users enters for the new password is the same as the old password, nothing should happen. To change the password, the password will have to be hashed using md5 and if it matches, the old password would be changed to the new password. This took many attempts because although similar, I had never attempted something like this before. There were several errors in the 'UPDATE' query (Refsnes Data, 2012) but with a lot of trial and error, these issues became resolved and the functionality for the user to change their password now works. This functionality then had to be applied to the admin pages. The code was copied across and worked successfully on the first attempt.

The next functionality to be included was for the user to delete their account. This was a similar set up to changing a password but with a few differences. On the page with the form all that was required were two text boxes; one for the email address and one for the password. The Javascript had to change to accommodate this! The setup for this page was nearly the same as the setup for the other page in the way the variables were set up and then put in if and else if statements. The difference came in the 'if' statement itself. Some examples were found of how to use the 'DELETE' query (Refsnes Data, 2012). This was straight forward and was nearly the same as my 'SELECT' query. After the delete code was working properly, the user was made log out and redirected to the login page. However, during informal testing, a participant suggested that there should be a page to confirm the deletion of the account. This was simple to do and only required changing where you were redirected after the account was deleted.

Applying Feedback from the First Test

Some preliminary testing was conducted of the login system with my brother, a student, and he was instructed to try and break the system. This was essential because the website should contain as little errors as possible. From doing this, he found that the error trapping had fallen short on numerous occasions. Every text box with user input would need to be validated. Numeric values needed to be entered for the event ID to add or delete an event. Links were added to the side of the page to help ease navigation. Users may have had issues finding some of the pages. The image had to

be split and then inserted into two different columns to accommodate the new side links. The links on the side were not consistent on every page. This would need to be changed.

Some of the error checking was lacking on the user side of things too. Only numeric values were to be accepted for adding and deleting from your schedule. The same approach was used for the admin pages. All of the side links were made consistent like they were for the admin page. There was more work in this because there were more pages to link. This would improve usability. There were still issues with adding numeric values that were not in the events table to your schedule. To fix this an ‘else if’ to the ‘if’ statement that caught the numbers that did not exist was used.

Informal tests were conducted on a few students and they provided some useful feedback. Some small details of the website were changed to try and improve usability and its aesthetics. The sub-menu was made 5px smaller so it would not overlap the next tab in the menu. The menu item ‘Castletroy’ was changed to ‘Location’ to try and cut out any confusion. The order and wording was changed in the page ‘Travelling to UL’. Its name was then changed to ‘Directions’. Its position in the sub-men was also changed. From the simple informal testing, the participants all stated that they found the wording of the menu confusing. Following this, the names of the ‘Fléadh Branch’ section and its sub-menus were changed. ‘Fléadh Branch’ became ‘Castletroy CCÉ’ and the sub-menu heading that was previously ‘Castletroy CCÉ’ became ‘About’. The text ‘Click to’ at the start of the links in the ‘Member’ page was removed. I then experimented with moving the test to the centre of the ‘Member’ page. An image was then added to the left. The new version can be seen in fig. 22. However, this soon was changed back to the way it had been previously.

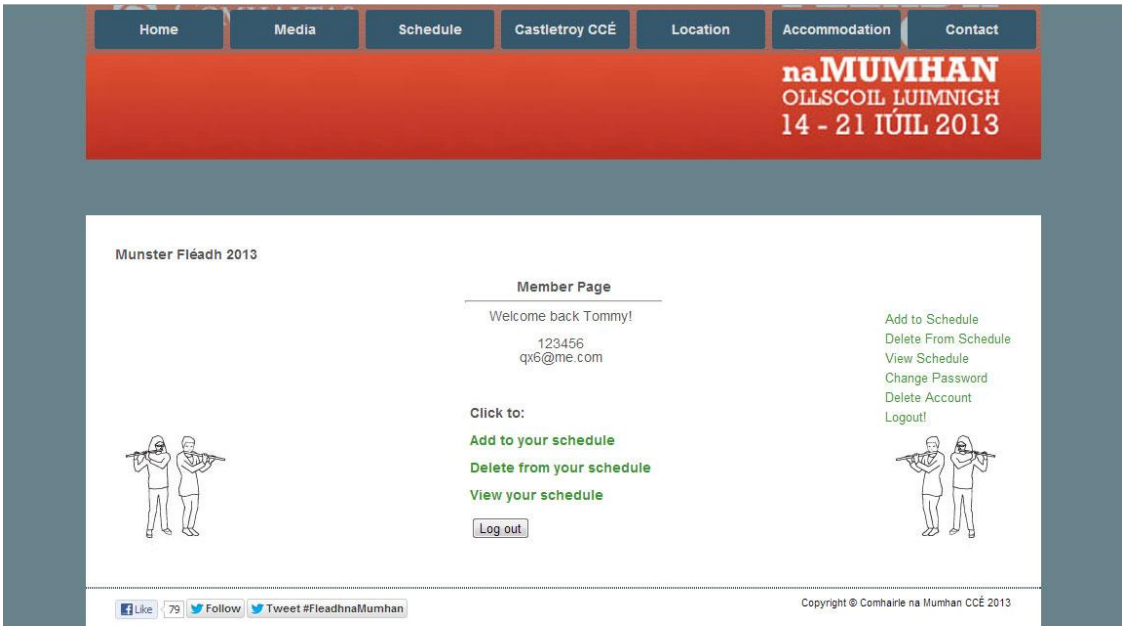


Fig. 22: Screenshot of the member page with items in the centre. This was reverted.

Links to 'Login' and 'Register' were then added to the home page. 'Schedule' was changed to 'Profile'. The sub-menu 'Programme' was then moved to underneath the 'Home' button. This may be moved in the future. 'Login' was then changed to 'Profile' in an attempt to make it more obvious. These changes were to make navigation around the site easier for the users.

Changes to Administrator Page

Yet more functionality needed to be added to the admin section of the website. A page was added so any admin would be able to see a list of registered users. To do this, a link was added on the home page to access this page. The code to view the events was taken and altered to be able to see the registered users. They were then ordered by user type. This put the admin at the bottom, but the admin accounts should be at the top. To do this, 'DESC' had to be added to the end of the MySQL query (Refsnes Data, 2012). This descended the order of the list and put the admins first. Instead of displaying the user type, *yes* should be displayed for an admin and *no* for a normal member. A nested 'if' statement within a while loop should have worked, however, it did not. The error may have had to do with concatenating the strings because it seemed to work when the '.' was removed from the code. This link was also added to all the side bars.

The next logical step after this was to let an admin change the user type of the registered users. They would be able to change normal users to admin and admin to normal users if need be. The admin enters the email address. It is validated so it has to be in the format of an email address. As an admin would not change their own account status, it did not seem fit to stop them entering their own email address when they did not want to. They may also want to step down as admins at some stage too. An 'if' statement was used to change the account type if the email was correct and they were an admin or user. It would change the account type to the one it was not currently. If the email address was not registered, they got another error message. The 'UPDATE' statement was then used to change the user type for both parts of the statement. A link to this was put on all the pages to keep the layout consistent.

Applying Feedback from the Second Test

A common comment that was received from the testing was that some of the links did not stand out. To fix this, the and <i> tags were used to make every link bold and italicized. This seemed to make them more obvious. The text 'Password must be at least ***6*** characters long.' was added to the register page so users would know before entering their password. Placeholder text was added to the forms to try and help the users. Placeholder text was then put in all of the text boxes that may need it. Some gave examples of what was to be entered. Two of the 'Back' buttons were changed to 'Add More' and 'Delete More'.

The same changes were then made in the admin section of the website. All the text boxes had examples and or a description of what was supposed to be entered into them. A lot of the 'Back' buttons had been changed to describe what they did. Some were left as 'Back' because it was the best explanation of what they did.

As suggested by some of the feedback, all pages that had large blocks of text were justified. This was done by using the code '<p align="justify">'. The <p> tag was already in use of many of these pages so the align attribute just had to be added in. It was a small change, but made a big difference to the appearance of the website. It made all of the pages with blocks of text much more aesthetically pleasing.

Further Development

At this point in the project, improving the users experience of the website became a very important thing. While looking at the 'Directions' page, one could see that there was vast room for improvement. It would be a good idea to insert a Google Gadget for giving directions (Google Directions, 2013). This would be much better than having a link to an external page. The map that was supplied was not the best of maps. It was soon replaced by an interactive map from Google (Google Maps, 2013). As a result of the testing, it was known that users had difficulty with finding directions previously. This should not be the case with the dynamic Google Maps and Directions gadgets.

The process began by customising the gadget and then requesting the code from Google. The code came in the form of a piece of Javascript code. This was then embedded on the web page and tested. It was fully functional but caused some problems. When there was a search done, it extended on size and moved all the text around. To fix this, a table was set up and aligned to the right. Now the text stayed in place, but when it extended it went past the footer. A simple solution to this was to place a white image below it so it would stretch the size of the page as needed. The text was then spaced out and changed slightly. When the user uses the gadget, the text that is in the way of it will simply move to the next line.

The old static map was then replaced with the interactive Google map. It was the same map that was used previously (the only difference is that it was zoomed in more). It's embed code was then retrieved. With this code the map could be embedded onto the page. A section of the code that Google supplied was then deleted because it brought you away from the page when it was clicked on. When the directions gadget is used, the map moves to the left like the text does. This is aesthetically pleasing. Examples of the two Google gadgets that were added can be seen in fig. 23 and 24.

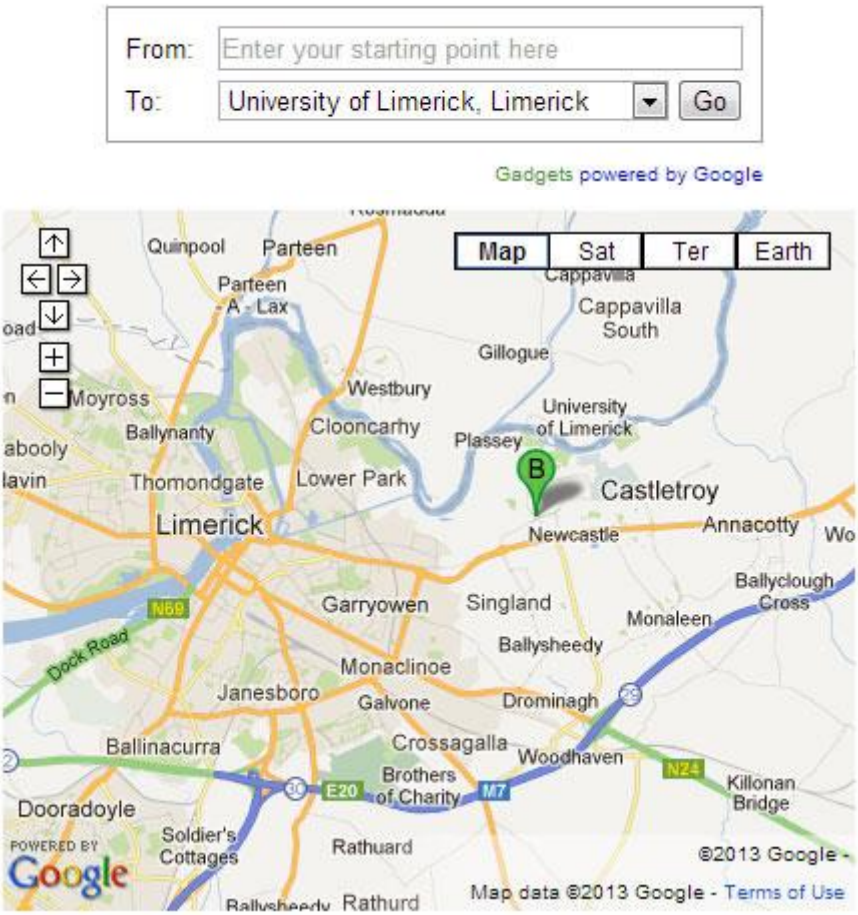


Fig. 23&24: Examples of the Google gadgets used.

Up to this point, there were just images embedded on the 'About' and 'UL' pages. They were just there as a stopgap. The search for a simple slideshow began. After a time, one that was usable and worked as required was found (Snook, 2009). It was featureless and transitioned smoothly between the images. When building the script from snippets of code on Snooks website it was decided to keep the CSS in the file because there was not much point making another CSS file for two lines of code.

This was then integrated into the code. This code was included in the 'About' page under the 'Castletroy CCÉ' menu tab. It was simple to integrate and worked straight away. The slideshow was moved because it was not aesthetically pleasing in its initial location. It was aligned to the right so it would be at the top right of the page. Now the pictures are in a constant slideshow of three and the slideshow is quite subtle. An example of the new 'About UL' page can be seen in fig. 25.

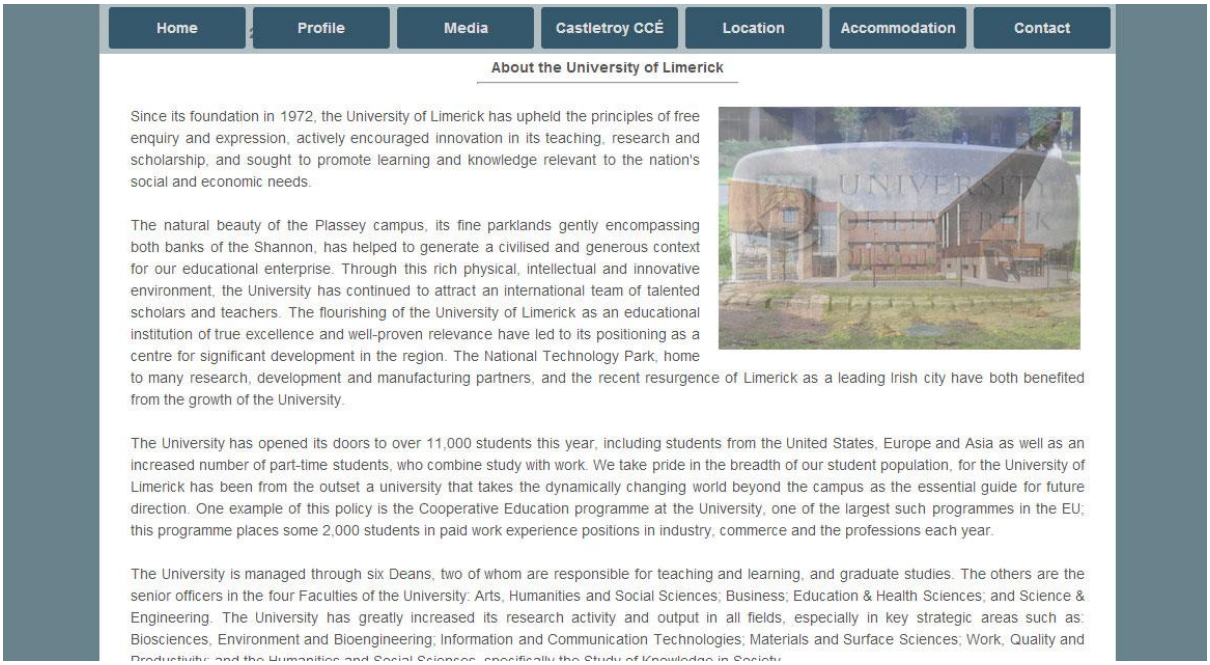


Fig. 25: Screenshot of the new 'About UL' page with slideshow in transition.

To integrate the site further with social media, a Twitter feed and #FleadhNaMumhan hashtag ticker would be embedded onto the home page. After searching, an unobtrusive Twitter jQuery script (seaofclouds, 2012) was found. This had the required functionality so customisation began. It was formatted to fit the colour scheme of the rest of the website. When this was complete it was integrated into the homepage. However, refreshing the page, the Twitter feeds did not show up. A number of different solutions were tried, including noConflict (jQuery Foundation, 2013) but to no avail.

As it was not working, I thought it best to set it up on one of the template pages. They worked on the first time of trying. This led to the conclusion that there was some sort of conflict between different versions of jQuery. After trying many different combinations of jQuery noConflict, I realised that the initial jQuery noConflict script was never deleted. When this was removed, the whole page worked perfectly. Another line of code was added to inform the user if there were no results for #FleadhNaMumhan. This can be seen in fig. 26. This had to be taken from one of the other examples given. The Twitter feeds can be seen in fig. 27.

```
}).bind("empty", function() { $(this).append("No matching tweets found"); });
```

Fig. 26: Screenshot of the code to show text if no tweets are found.

Fig. 27: Screenshot of the two Twitter feeds on the homepage.

After the addition of the Twitter feed, it was deemed to be practical to include more Facebook functionality. Something similar to the Twitter feed used previously was needed, but this proved hard to find. The Facebook developers provide some code but nothing that would meet the requirements. After a more exhaustive search, a social app was found close to what was needed (Facebook Comments, 2013).

To use this, it was a simple matter of selecting the page it sent the comments to, the width and number of posts, the theme and you could then request the code. It was just a matter if inserting the code into your site and the social plugin would appear there. It was placed inside a table and positioned it to the right of the Twitter feed to make the page aesthetically pleasing once more. A heading was then placed over it to keep it consistent with the Twitter feeds. With these latest two additions, the homepage has become much more interactive as can be seen in fig. 28.

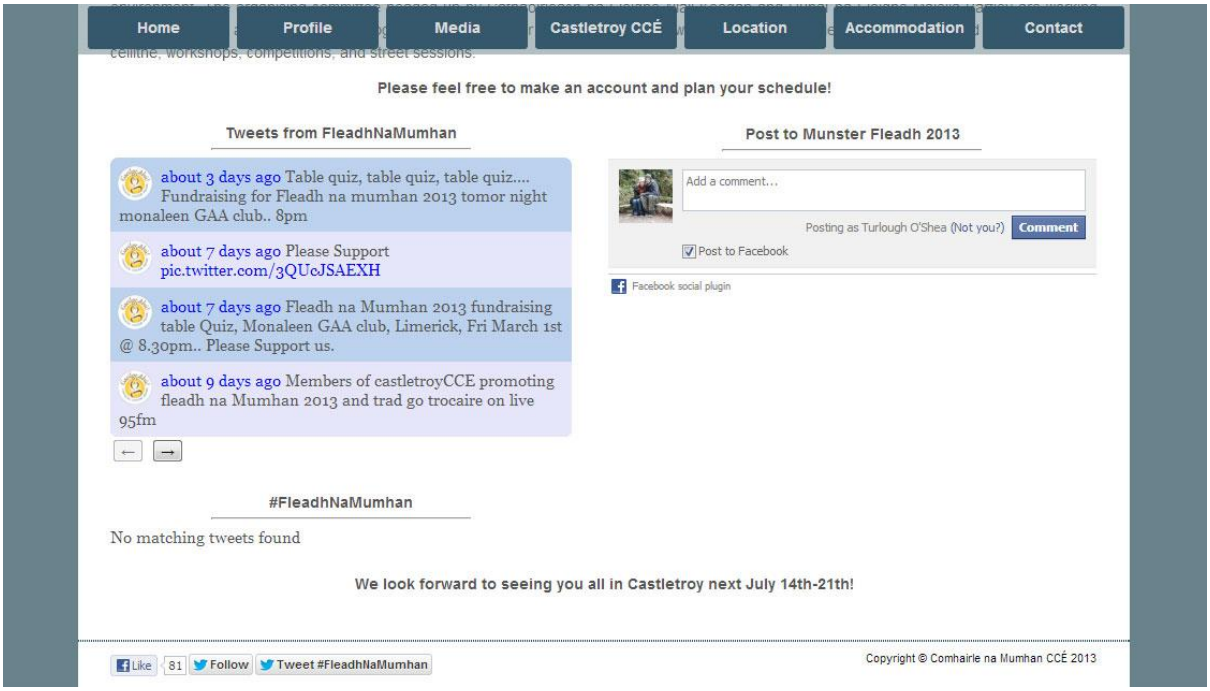


Fig. 28: Screenshot of the homepage with Twitter and Facebook functionality.

The website already had a simple 404 page, but this was made at the outset and had never been upgraded. It was time for this to change. I thought that it would be a good idea to include some sort of Google Gadget, HTML5 or Flash game. After a thorough search, nothing that met the requirements was found. The search was narrowed to something musical, perhaps with an instrument because this is a Fleadh webpage. However, this yielded no results.

Instead of having a game, a few simple instructions and an image would be put on the page instead. Some instructions were written for the users and then an image was made, using Photoshop, to fill the available space on the page. The design process was iterative and the second version of the image was selected for the 404 page. Fig. 29 shows the second iteration of the 404 image.

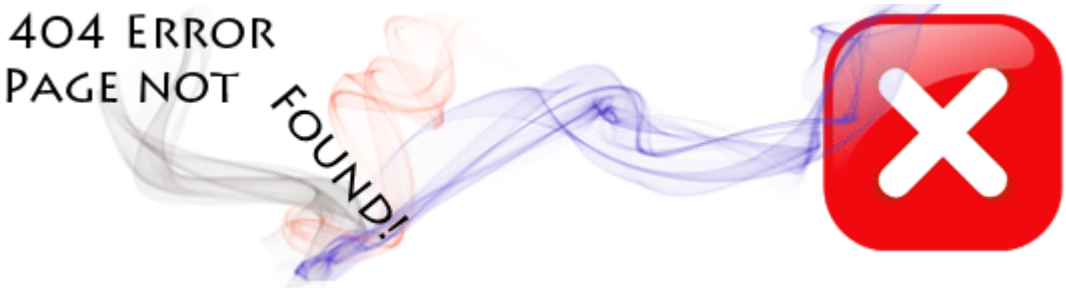


Fig. 29: The second iteration of the image used on the 404 page.

At this point some more content had to be added to the site. More details and a registration form needed to be added to the 'On Campus' page. The test was taken from the old website and then formatted so it would fit into the new website properly. The flier was aligned to the right with the on campus information to the left. There was now a lot of free space available beneath the text. As a consequence of this, some of the drawings that were used previously were added to the page.

More hotels needed to be added to the 'Hotels' page. The information was taken from the old website. However, some of the information was incorrect and needed to be fixed. Some of the images were not of a proper standard and needed to be acquired. There was also a new 'Camping' page added to this section. There was a paragraph about competing in the competitions added to the home page along with a downloadable version the rules in pdf format.

Feedback from Formal Usability Testing

After the Formal Usability testing, there were quite a few changes to be made to the website. A few of the participants commented on the banner saying a better quality one was needed. As a high quality image for the banner was not supplied, one needed to be made using Photoshop. While making this new banner, it was constantly compared to the old banner to keep them similar. A few participants thought that some of the sub-menu and menu tabs should be changed. 'Branch Classes' and 'About' were merged to form 'Castletroy CCÉ'. This was then moved so it would be under the location tab on the menu. 'Committee' was moved from under the location tab to under the 'Contact' tab. The old 'Castletroy CCÉ' menu tab was changed to 'What's On?'. Programme was moved to underneath the 'What's On?' menu tab. Dance Packages was also moved to underneath 'What's On?'. The menu headings were then made consistent for the entire site.

At least two participants thought that there should be consistency in the menu. They stated that all menu items should either be clickable, or not clickable; one or the other, not both. Before some were and some not. As the participants thought that was an issue, it would have to be remedied. Pages had to be customised for all of the menu tabs. One participant stated that they thought it was odd that you had to scroll down to see text on the home page and recommended that the text be moved above the slideshow. They also thought that the 'Register' and 'Login' links should have been

more like buttons instead of just links. After these cosmetic changes to the homepage, the text was now the first thing that the user would see. This can be seen in fig. 30.

Fig. 30: New iteration of the homepage after the formal usability testing.

Another participant recommended that some text be removed from the 'Confirmation of Deletion' screen. He also said that the schedule should be deleted when the account is deleted. To do this, an extra query had to be added in when the account was deleted. 'mysql_query("DELETE FROM schedule WHERE email='$email'");' worked and now when a user's account is deleted, their schedule is deleted too. A line telling you the benefits of registering was added to the register page.

After this, the menu tab 'What's On?' needed to be populated as it only featured 'Programme' and 'Dance'. Pages were made for 'Céilí', 'Concerts', 'Exhibitions', 'Gig Rig', 'Seas Nós', 'Singing', 'Trad Disco' and 'Workshops'. However, it was decided to combine 'Dance', 'Sean Nós' and 'Céilí'. After all this was finished, the menus were checked and made consistent.

Google Calendar Integration

There are a few ways that you can add Google Calendar widgets to a website. A button can be added so every event is added to the calendar once clicked. This would not be desirable because of the fact that someone may not want to go to all of these

events. The option to add a single event is a much better option because the users would then be able to pick and choose.

After selecting which option would be used, it was decided that the Google Calendar buttons would feature in the sub-menu pages of the 'What's On?' menu tab. It was felt that if they were put into the login section, it would draw the users away from the option of making their own schedule on the website. The Google Calendar button was added to some of the events in the sub-menu of the following pages; Concert, Dance, Trad Disco and Workshops. The page used to create the buttons was found on Google Calendar itself (Google Cal, 2013).

Further Social Network Integration

To try and integrate Facebook login with the website, I needed to join Facebook Developers (Facebook Developers, 2013). A simple form needed to be filled out to get access to their APIs. However, from reading through the Facebook developers' page about the login (Facebook Login, 2013), it appears that the site has to be live for it to work. As a consequence of this, the free hosting that they offer will be used to try and get the code working. After this, if successful, the code will be added to the website.

To test the Facebook login, an account with Heroku needed to be made (Heroku, 2013). After the account was made, the Heroku toolbelt needed to be downloaded. The Heroku toolbelt was supposed to help with building the login on a virtual machine. There were several steps to be followed in the command line software 'Git Bash'. After this, another app had to be made. This had the same name as the first but was marked as a development app. It was called 'Test – Dev'. After the local IP address of the computer was typed in (http://127.0.0.1/) a virtual host needed to be created so the files could be viewed on the local computer. Since I did not know how to make a virtual host, a solution needed to be found online. The first page seemed to have the solution (Cesaric, 2009). However, this did not work. After I had no luck with this, more reading was done on the Facebook Developers page.

After more investigation and reading of the Facebook Developers site, the Facebook register social app (Facebook Register 2013) looked like a better solution than the Facebook login. However, difficulty still arose with this. It appeared that the site needed to be live for this to work. Because of this, work on this section was called

to a halt. I then looked at another way to further integrate social networking into the website and this was to let users have the option to log in using Twitter (Twitter Login, 2013). However, it was decided to go ahead with the Facebook example.

After the website went live I began attempting to integrate the Facebook registration again. Initially there was some difficulty and an error message would appear in place of the Facebook form. The error stated 'Error: Can't parse fields attribute. Is it a JSON array or CSV list?'. A number of things were tried but to no avail. After searching for solutions, there were a few promising leads found. A comment from Juicy Scripter (Juicy Scripter, 2012) in a thread on Stack Overflow (Stack Overflow, 2013) gave quite a bit of information about making the form and introduced JSFiddle. JSFiddle (JSFiddle, 2013) is a tool for using Javascript online and was used to make the form with no errors. This can be seen in fig. 31. When the form was displaying correctly, my attention then turned to trying to insert the code into the database. Some code was taken from Ravi Tamada's tutorial (Tamada, 2011) and integrated with some of my own to insert the user input into the database. However, after several attempts nothing would go into the database. I tried hardcoding text into the database when the form was passed through. This worked and then led me to believe that there was an issue with an "'" in my name. When the name was put through a replace function ($name = str_replace("'", " ", $name1);) the user input was inserted into the database.

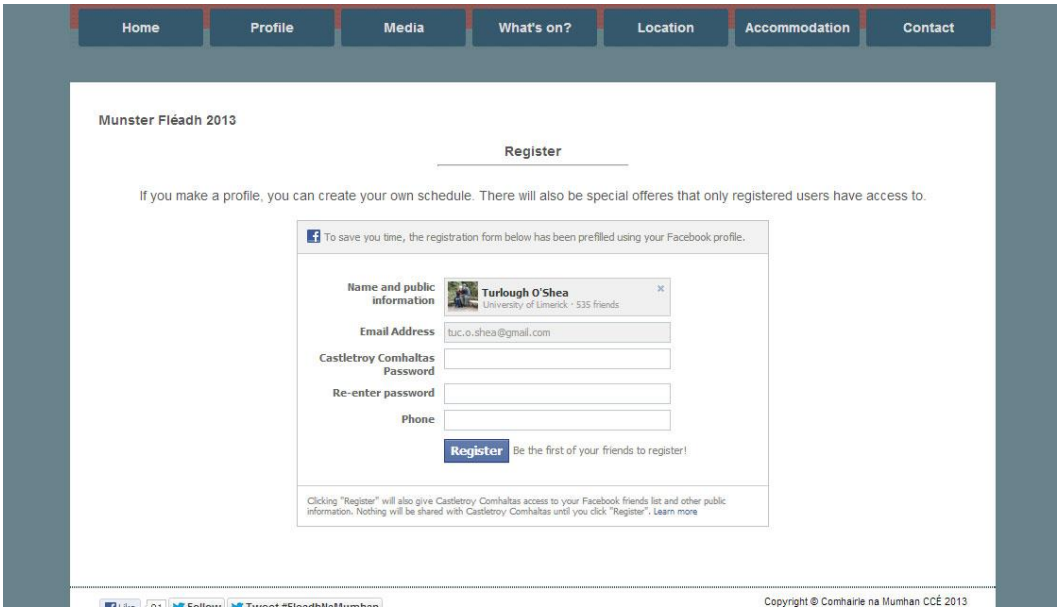


Fig. 31: The form for registering with Facebook.

Even though the user input was getting passed into the database, there was still an issue. The code did not do anything if the username was already registered. As this page was already similar to the 'Registered' page it simply took using the same structure of if statements to get an error message to show up if a user was already registered. This will be used as the main register form and the old one will be a backup that can be used in case issues arise with the new one in the future.

Htacess, Sitemap and Robots.txt Files

It was time to set up a .htacess file. To do this, a guide was used for help (Javascript Kit, 2012). The address for the 404 page was put in so that if there are any errors, the user will be redirected to the 404 page. At this point the clients informed me of a domain change. Instead of www.munsterfleadh.ie the website would now be www.castletroycomhaltas.com. This change had to be made to the .htacess file.

To stop 'bad' bots and people ripping the site for offline access, Javascript Kit recommends inserting code into the .htacess file (Javascript Kit, 2012). This should save bandwidth and decrease server resource usage. It will display a 403 Forbidden error to the 'bad' bots. It is also suggested to include a bit of code to stop 'hot-linking'. However, when this was included, it seemed to stop the jQuery plugins from working. It was then removed and everything worked. This is when someone links to an image or css directly from your site and uses your bandwidth.

After the website went live, a sitemap generator (XML Sitemap, 2013) was used to create a sitemap. However, after exploring the GoDaddy control panel (GoDaddy, 2013) I created a sitemap using their service instead. According to SEO, a sitemap is advised for a number of reasons (SEO, 2011). A sitemap helps with navigation and theme of the website. It helps users if they get lost and can help them assess the structure. Sitemaps are also used for search engine optimisation. It is a great way to get every page in the website optimised by search engines.

Initially the plan was to use robots.txt generator (McAnerin, 2010). However from further reading (WebMasterWorld, 2006) it appeared that a robots.txt file was not needed. Robots.txt is used mostly where there is a section of the website that you do not want to be crawled through by bots.

From WAMP to Server

As I have never attempted this process before, it was something that should be experimented with. Until this stage, no password had been set and the username was simply set to 'root' in phpMyAdmin. This would have to change before the website went live. Because the password and username were subject to change, it would be better to leave them as they were until the change was needed.

After the domain and hosting were purchased, the domains name servers needed to be changed to the name servers of the hosting company. The company the domain was purchased from was Register365 (Register365, 2013). The company the domain was purchased from was GoDaddy (GoDaddy, 2013). After the name servers were changed, an account was made so the files would be able to be uploaded via a FTP (File Transfer Protocol) (FileZilla, 2012). After this, the files were uploaded. Initially, the login section was omitted until the MySQL was set up.

When the login section was put onto the server, there were errors when admins and users tried to log in. Fig. 32 shows the error for the admins. And fig. 33 shows the error when users tried to log in.

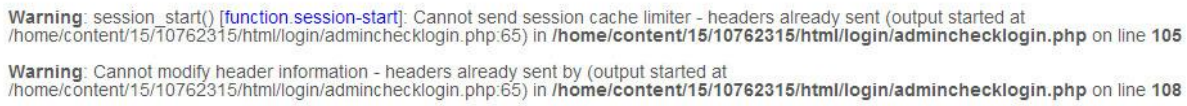


Fig. 32: Errors when admin tried to log in.

Fig. 33: Errors when user tried to log in.

According to online resources (Digital Point, 2007) and (Php Builder, 2005) seem to suggest that this error occurs if there is output to the browser before the session and or header starts. To combat this, everything was deleted before and after the php open and closing tags. However, this still did not work. On closer inspection there was a space before the opening tag. When this was removed, it worked properly. However, when a user or admin entered their password incorrectly, they were brought

to a blank screen. Everything tried would give the previous error, so this will have to do for now.

There was an error with registering an account. It was a similar error to one that showed up before. All that was needed was an extra '>' in the 'if' statement so the 'if' statement now had '>>'. There were not any other issues in the user side of the side of things aside from when the user went to delete their account. Fig 34 is the error message given.

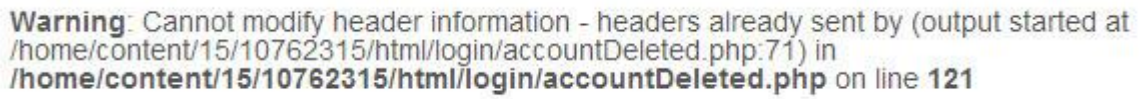


Fig. 34: Error when account is deleted.

To get around this issue, the information from the page 'Confirmation' was displayed on the page rather than redirecting to the 'Confirmation' page. As a consequence of this the side links needed to be put into the conditions where the query failed. After these initial few fixes, the site appeared to be in working order.

An email address (admin@castletroycomhaltas.com) was then set up. The contact from was then pointed to this address. The details were then given to the client and they were then told that any messages could be forwarded from this email address if they wanted to use their own email address.

Final Changes

After the website went live, the clients wanted to meet with me. They asked if they would be able to get some more email addresses made which I created during the meeting. The clients then requested to add some more hotels to the hotel section. They also wanted a name and contact number added to the homepage so it would be visible for users. The clients then wanted information regarding the competitions in the Fléadh to be available for the public. Because the music competition spread sheet was so large, I simply made that and a PDF available to everyone. Several pieces of contact information were changed. A link to the contact form was placed under the login form telling the user to use the contact form if they forgot their password. The images for the special offers were then made using Photoshop and uploaded to the website.

Work then began on making a form for the admins to be able to add results of the competition to the website during the Fléadh. To do this, a table within the database was created with the rows comp, compNo, placing, name, branch, type and day. Then the code for adding events was taken and then adapted to this table. If the comp and name matched something already in the table, an error message would appear. All of the fields were validated so they had to have some input in them. The compNo field was validated so that it could only accept numeric values. After this, the side links were made consistent throughout the admin page. The next step was then to set up the page for viewing results.

The next page to be created was the page to view the results. This used code adapted from the pages view users, events and schedule. There was a difference though. The results were displayed in days in order; Friday, Saturday and Sunday. The results for each day were then displayed under the section for each day. The results were also ordered by the competition number. After this, a page to delete results was added in case any mistakes were made during the process. This also adapted code that was used previously. It asked the admin to input a competition number and name of participant. If these matched a row in the table it would be deleted from the table. If they did not match, an error would appear. All the side links then needed to be checked for consistency.

Search functionality was then added to the results page. This would help users find what they were looking for rather than looking through all the results. A form was added and validated to the ‘Results’ page. This then directed the user to a new page called ‘Search Results’. Search Results uses six queries to search the table ‘results’ for the search term. If something is found, it will be displayed on the search page. Initially I tried to have everything in a single query, but this did not work so I split it into different queries. Many different combinations of ‘if’ statements were tried. There were supposed to show alternative text if there were no results. However, none of these different combinations actually worked. If nothing is found in the search, no results populate the table. This was not much of a problem as there is a button that the user could simply use to go back to the previous page.

Chapter 4: Testing

During the later stages of the development of the website, it would be essential to conduct some informal user testing. A list of tasks was decided upon. The participants would have to complete these tasks and then give feedback. These tasks were divided into three sections (media, login and miscellaneous) to try and help make the tests clear for the participants. For both the first and second informal test, the same participants were used.

Tasks

Below is a list of tasks, split into their sections, which the participants had to complete:

1. **Media Section**
    1.1. Find and play a video.
    1.2. Find and play a song.
    1.3. Find and look through the photographs.
2. **Login Section**
    2.1. Register an account.
    2.2. Login.
    2.3. Add something to your schedule.
    2.4. View your schedule.
    2.5. Delete something from your schedule.
    2.6. Change your password.
    2.7. Delete your account.
3. **Miscellaneous section**
    3.1. Find out the times of the classes run by the Castletroy CCÉ.
    3.2. Get directions to the Fléadh from Birr, Offaly.
    3.3. Find the hotels available.
    3.4. Use the contact form.

Results From First Informal Test

For this test, there were six participants. They were all students. As they all had different levels of skill with computers they provided good data and feedback for the development and design of the website.

All users found the tasks in the media section quite easy. Most commented that the tasks were ‘Very straight forward’. However, there were some suggestions along with a mishap. One suggestion was that the ‘Media’ tab on the menu should bring you to some page which displays the links to the different media sections, but that was not deemed necessary. Another user thought that the numbers under the image viewer looked strange and would rather something similar to the pageation used for the slideshow on the home page. One was not able to get the video playing at first, but when they realised that the big button in the centre of the video would play it, they went about with the other tasks.

Most users encountered some difficulty with the login section. This had something to do with the placement of the ‘Login’ and ‘Register’ tabs in the sub-menu. Some of the users suggested moving them and changing the name of them and the ‘Schedule’ tab in the menu itself. Two users had problems logging in after creating the account and suggested having an option to log in on the home page. Most users had no trouble adding and deleting events from their schedule, but nearly all were confused when asked to view their schedule. All but one went to the ‘Programme’ sub-menu initially. One participant tried to see if they could break the system and was successful. They recommended error checking whenever there was input from the user. The same user also thought it would be a nice touch if the buttons described what they did rather than say ‘Submit’. Another user thought that the back buttons were odd in places and recommended removing them. It was suggested to have buttons or check boxes instead of inputting a number when deleting and or adding an event to your schedule. A point was made to include the links on the side of the page on every page to keep the section consistent. One user also suggested that a page to confirm the account had been deleted should be included. This was because they were not sure if their account had been deleted.

The name of two of the tabs in the menu caused great confusion in the miscellaneous section. As a consequence of this, some users found it difficult to find

out the times of classes and to get directions. It was suggested that the layout of the 'Travelling to UL' page should be changed. It was also suggested that the link to Google Maps should be more obvious. Several users went to 'Venue Map' to try and find directions. None of the users had difficulty in finding the hotels available or using the contact form.

There were a few suggestions about the site in general. Three users remarked on the banner image. They all thought that it needed to be a higher resolution. It was obvious that it was not a good quality image. One user thought that the way the sub-menu overlapped the next menu tab should be changed so that it would not be and recommended that the sub menu tabs should be made smaller.

There was good feedback from the participants of the first informal test. There were also a number of changes recommended by the participants. Not everything that was suggested could be changed, but the main issues were fixed and some of the lesser issues were also fixed.

Results From Second Informal Test

For the second test the same six participants were used. Several aesthetic and functional changes had been made to the website. The participants were asked to try and complete the tasks as if they had never completed the first test.

Most users found the media section straightforward and very clear. There were only very slight changes made to it since the first test. The numbers from underneath the image viewer had been changed to a pageation similar to the slideshow on the home page. The 'Media' tab had also been moved to the right on the menu. All the users found these tasks quite easy and many stated this.

The users found the login section much simpler then the first time around. A register and login link had been put on the home page. There had also been a change to the menu tab changing 'Schedule' to 'Profile'. In the sub-menu, 'Login' was renamed to 'Profile' so the users would know to go there. The side links were also made more consistent which helped navigation for the different tasks. However, one user did not see them straight away and recommended that they were made stand out more. This user also noticed that a link had been left out of one page and needed to be corrected. Another user thought that it should be mentioned that the password had to be at least

six characters when registering. Another user thought that it might be a good idea to have a sort of dropdown way to login. They used the Twitter (www.twitter.com) website as an example. Another user was impressed with the changes to the login section and also recommended that changing the 'Back' button to 'Add More' or 'Delete More' would be a nice touch.

All of the participants agreed that the miscellaneous section was easier to follow when compared to the previous test. 'Fléadh Branch' had been changed to 'Castletroy CCÉ' and 'Castletroy' to 'Location'. 'Travelling to UL' had been changed to 'Directions' and some of the recommended changes had been implemented. Three users thought that the link to Google Maps should be clearer on the 'Directions' page. They recommended making it bold or italicised and possibly having a line break after it. The rest seemed to have no issues with this section.

A user came up with a few suggestions after completing the second test. They thought that the text should be justified in some sections. They also thought that it would be a good idea to have some sort of ticker implemented into the page. The Facebook ticker was given as an example. They noted the new placement of the 'Programme' sub-menu tab and wondered if it could be moved anywhere else. They also brought up the issue about replacing the banner with a better quality image.

The second informal test also gave some good feedback. The users all seemed to think that it was a big improvement upon what they had seen during the first test. Their feedback will help improve the development of the website greatly.

Formal Usability Testing

After the informal testing had been done and the feedback applied to the website, it was time for some formal usability testing to be done. Some points on usability testing were taken from (Dunn, 2013). This would be somewhat similar to the user testing, but the participants would be given a sheet with tasks and a space for feedback. In the informal testing, the participants were read the tasks and then notes were taken from their feedback. During this testing, there will be no contact with the participants, they will have to do everything themselves.

Instructions

1. You just found this webpage and want to see if they have a social network presence. Aside from 'Liking' the page on Facebook, 'following' it on Twitter or 'tweeting' with the #FleadhNaMumhan (# = hashtag) is there any other social network integration?

2. You hear that 'Goitse' will be performing at the Munster Fleadh in 2013. You have never listened to or seen any videos of them before and would like to before attending the Munster Fleadh. How could you go about this?

3. You would like to book some accommodation for the duration of the Munster Fleadh but do not know the area. How would you go about finding out what accommodation is available?

4. You live in Birr, Offaly and will be traveling to UL for the event. You decided you are going to drive but do not know the directions. How would you find directions to UL?

5. You would like to inquire about an event taking place during the Munster Fleadh – find the number of the 'runaí' or secretary.

6. You have heard that there may be special offers to people who have created an account. Create an account with the website and proceed to login.

7. When logged in, you notice that you can make your own schedule of events. You would like to add an event to your schedule.

8. After you added the previous event you realised that you would not be available that day. As a consequence of this, you need to delete this event from your schedule.

9. You think that someone may know your password, because of this, you need to change it. How do you change your password?

10. After this you decide that you do not want your account anymore. How do you delete it?

Results From Formal Usability Test

Eight participants took part in the formal usability testing. These were all students with a varying range of computer skills and familiarity. Different participants were used for this test then were used for the first two informal tests. All the participants were briefed and then asked to sign a consent form. The full consent form and instructions are available to see in the 'Appendices' section.

Before the participants began with the instructions, they were asked what they thought a website for the Munster Fléadh should have included. The general consensus was that the website for the Munster Fléadh should contain information about the Fléadh, lists of events, information of services, maps, reviews of acts, pictures from previous Fléadhs, accommodation and directions. This question was asked to see what impression the participants would have of the Munster Fléadh.

Participants were then asked to familiarise themselves with the website. They were asked what they thought of the layout and if they would change anything. Four of the participants thought it was laid out in a straightforward way and had no trouble getting around the website. They did not have any suggestions for changes to the website. The other four participants could still find their way around, but two of the participants found the menu confusing and recommended changes to it. Two participants commented on the quality of the banner image and thought it should be replaced. One user recommended that the text should come first on the home page and found it strange that you had to scroll down to read the text. Two participants were concerned with the clickability of the menu and recommended that all the menu tabs should be clickable, or none at all. One participant found the vertical spacing of the menu strange. Another participant felt that there should be some images embedded within the programme.

After the initial two questions came the ten tasks that the participants needed to carry out. For the first task in the list of instructions, four of the participants gave the desired answer. The desired answer was to point to the twitter feed and ability to post to Facebook that was on the home page. The other four answers varied quite a bit. Perhaps the question should have been reworded before the forms were handed out. One participant started to talk about a G+ account that they had. A second participant suggested that social bookmarking, such as Delicious, be included. Two

participants noted the fact that you were able to make your own profile on the website. However, half the participants saw the features that were suggested by the question.

All of the participants found the second instruction very easy and straightforward. They were asked to find out how to listen and or see videos of 'Goitse' play. Everyone noticed and navigated to the video menu and watched the video. However, none of the participants seemed to go to the audio section to try and find the 'Goitse' songs listed there. This was interesting and suggested that the participants would prefer to watch a video then they would like to listen to a song.

Again, all of the participants found the third instruction easy and straight forward. None of the participants had issues finding out about the accommodation available during the Munster Fléadh. Three participants commented on the comprehensive nature of the accommodation listed. Two participants commented on the layout of the 'Hotels' section and said that it was very easy to follow and gather the information.

The fourth instruction asked participants to find their way to UL for the Munster Fléadh if they lived in Birr in Offaly. A random location that was easy to search was selected. All of the participants seemed to be able to find their way to the directions page without any trouble. They all also completed the task without issue. One participant simply commented 'Well done!' after they had completed their task. Another participant complimented the speed and ease at which the directions were able to be found. Two participants complimented the convenience of being able to search directly from the directions page.

The fifth instruction asked the participants to find the phone number of the 'runaí' or secretary. All of the participants found the number of the secretary very easily and had no issues in finding it. Two of the participants said that they liked the way all of the contact information was laid out in front of them

The sixth instruction had the participants search for special offers once they had registered. All of the participants found registering easy. The participants also found the special offers link once they had registered. However, one participant stated

that there was no information on why a user should create an account easily available and suggested that this situation was remedied.

For the seventh instruction, participants had to add an event to their schedule. All participants found this option easily and commented on its straightforward nature. However, one participant stated that they did not know what was meant by 'Please navigate to desired page' after they had added the event. One participant stated that it was 'strange' to have the addition/deletion/viewing on different pages and suggested that they were all put onto the same page. This suggestion would not be applied to the website because of the nature of the tasks. Adding a task needs all of the available events to be displayed while deleting and viewing need only the events that have already been added to the schedule to be displayed. Technically, viewing and deleting events use the same code, aside from the queries to delete an event from the schedule.

The eighth instruction had participants delete the event that they had previously added from their schedule. All participants commented that this was easy to do and straightforward. One participant still thought it was strange to have the addition/deletion and viewing on different pages, but they still commented on the ease of the task.

The ninth instruction had participants change their password. Again, all participants found this straightforward. Five participants commented on its ease and intuitive nature. One participant commented on the secureness of the system. They had tried to reuse an old password and confirm their password with a different password but to no avail.

For the tenth instruction, participants had to delete the account that they had just made. All of the participants found the task easy and straight forward. None of the participants had any issues finding the link to delete their page. However one user expressed concerns about their schedule still being kept after their account was deleted. They recommended removing that section of text from the confirmation page and the delete the schedule of the account when the account was deleted.

After the participants completed the instructions, there was a debriefing section for them to complete. The participants were then asked about their impressions of the website. Four participants stated that the website was laid out well,

aesthetically pleasing, informative and easily accessible. One participant stated that it was easy to use and had a lot of functionality, but that some parts did not work as expected. This was referring to the clickability of the menu and some of the tabs being in the wrong place. It may also have referred to the schedule being stored even when the participant had deleted their account. Another participant stated that the website worked well and went on to say 'A test description on the homepage without scrolling would be great'. Two participants said the website was laid out well and easy to use. They also commented on the functionality and liked the way they were able to make an account.

The participants were then asked about any improvements that they could suggest for the website. Two participants were impressed with the website and could not suggest any improvements. Four participants commented on the menu and thought that some of the sub-menu tabs could be in better positions. Two participants spoke about the clickability of the menu tabs and said that they would like to see it consistent. One participant would like to see some graphics embedded into the programme. One participant would have like more information about the accounts and account management. Another participant found the vertical spacing on the menu strange and recommended more spacing. The same participant also recommended getting a better quality image for the banner. One participant suggested a sitemap to help navigation of the website. Another participant suggested that there should be button graphics for 'Register' and 'Login' on the homepage to make them seem like they could be pressed. Please note that the participants had many opinions in this section so it appears that more than eight were tested, but that was just to get all the opinions across.

The last thing asked of the participants was to recall what they could about the structure and functionality of the website. Five participants found the website easy to navigate. The five participants also thought the website was structured well. They all thought that the tasks were easy to complete and commented on the good functionality of the website. One participant questioned the positioning of one of two of the sub-menu tabs but spoke highly of the website otherwise. One participant commented on the ease of navigation around that site and also liked the way that it was not too cluttered. The last participant noted that the structure was familiar and common with most websites. They also said that the website was well structured.

Chapter 5: Evaluation and Conclusions

Evaluation

Before the website was finished, it was evaluated at least three times. These three times were the two informal tests and then formal usability test. From the initial informal tests, participants gave feedback of what they would change and what they found confusing. They were asked to complete tasks and gave feedback on these tasks. This was an evaluation in itself because the participants looked at the early stage of the website and recommended changes.

During the formal usability test, the participants had more freedom and were able to explore all aspects of the website. This led to feedback on what they would change about the site, not just the sections where they had to complete tasks. As there was some evaluation early on in the construction of this site, the site developed in a different way to how it would have if there was no evaluation.

Once the website was online, it was inspected by the clients and some of the participants of the various tests. At first glance the clients were delighted with the work that had been completed. They commented most favourably on the aesthetics of it and seemed to really like the new high quality banner.

During the demo day, I had to showcase the website to any interested parties. There was some valuable and insightful feedback given by people interested in the project. A number of people stated that all the information they would need was readily available without overloading the user. Another person commented on the ability to page hop between all pages. They stated that this was very good usability as the user should not be able to lose their way. One of the interested people was very impressed upon discovering I was in MMPT (Music, Media and Performance Technology) and did not have a background in this type of work. The same user thought that the Google gadgets in the ‘Directions’ page were a very nice touch and inquired as to where I found them. However, it was pointed out that there was no search capability in the ‘Results’ section. This was then soon changed. I was then asked how the site would be maintained after I was finished with the project. As I did not use a CMS (content management system) and the clients have never used code before this may be an issue. This is discussed in more detail in the

'Recommendations' section. I was then asked if I had tested the website on people who may be attending the Fléadh. One or two of my test participants have an interest in attending, but if I had more, some of the results I collected may have been different.

Overall, the website is aesthetically pleasing, has minimal errors and functions as expected. At the outset of this project, I wanted to make an aesthetically pleasing, modern looking and useable website and I believe that this has been accomplished. This is drawn from the comments and feedback from the participants of the formal usability testing. However, saying all this, there were problems encountered. I also have recommendations and scope for future work.

Problems

During the completion of this project, a number of issues have arisen. While many have been small and fixed quickly, there were a few larger ones that took more time to remedy. These will be discussed here.

Initial Use of MySQL

Before proper research for this project was conducted, the idea was that there would be no server side to the website. This soon changed as I realised that for any sort of login or storage of information, MySQL or a similar language and some database with tables would be required. As I had never properly used MySQL (outside of an introduction module in first year) it was quite difficult to get a handle on. Before work on the project had started, there was a lot of practice done using w3schools (Refsnes Data, 2012).

The work completed before the project had started did not do much to prepare me for the work ahead. A lot of the work done with MySQL was learned while doing. Most of the early queries used returned errors or did not work properly. It was by constant practice that the syntax and structure became clearer to me.

After some time, a lot of the old code was thrown away and work on the login system began again. This helped me to become more familiar with MySQL. The amount of errors in queries and overall errors dropped dramatically.

Initially I had a problem with MySQL, but after time, the problems seemed to diminish as I got more familiar and proficient with the language. Although I became familiar with the language, there were still a few issues with it.

Editing Schedule with Checkboxes

An issue occurred when the page to edit your schedule was being created. The idea behind this was that there would be a list of checkboxes next to the event; if it was checked it would be added to your schedule when you submitted the form. If you removed the check from the checkbox, you removed the event from your schedule with the form submission. The checkboxes were incremented automatically to be next to each of the events. An 'if' statement was used to check the box if they matched the variables. Whenever this was attempted, it either ended up with all the checkboxes being checked, or none of them.

As this did not work, another solution was needed. Editing the schedule was changed to two different pages. Adding an event and deleting an event. Instead of using checkboxes, the user must input a number into a text box to add or delete that event from the schedule. These text boxes are validated using Javascript and only allow numeric values to be entered. If an event ID that does not exist is entered, the user gets an error message.

The solution to this issue worked well, although, the initial idea would have been more practical. During the informal testing one user commented on how they would have preferred buttons when adding and deleting rather than having to input a number.

Keeping Everything Consistent

A major issue while designing and developing this website was keeping everything consistent. To keep the Banner, menu and footer consistent on every page, they were placed in different files and included using PHP. However, other elements of the website were not as easy to keep consistent.

While logged in, the user has a few side links which link to the different pages they are able to access. It took a few attempts and the informal user testing to get these links consistent. The page that was open was not displayed in the side links. This meant that the links were not the same on every page. After the testing, they all

seemed to be consistent. All the links on the website were made strong and italicised to keep them consistent.

It was difficult to keep the layout of the website consistent. As it was being designed and developed in tandem, sometimes there may be sudden changes. These changes then need to be made to all of the pages. When there are mass changes like this, sometimes a page can get skipped out and left behind with the old design. If a CMS (Content Management System) was used for this project, it would cut out a few of these issues.

Integration with Social Networks

As a lot of the work and languages used in this project are new to me, they needed to be learnt. They were learnt before the project started and then reinforced by familiarisation during the project.

However, when it came time to try and integrate aspects of social networking into the website, such as logging in or registering with ether Facebook or Twitter using their API's, some trouble arose. Both social networking sites mentioned above had their own way of doing things which were very different to the way everything was done in this project so far.

Initially the login was to be integrated into the website. However, having the register would have advantages over the login. It would speed up the register process and still leave the login the way it was. This way, users of Twitter and Facebook would be able to register using the information from these social networking sites instead of inputting the information into the form manually.

Recommendations

While completing this project, certain things would have done differently had time not been an issue.

Content Management System

For anyone embarking on a similar project, I would recommend using a CMS. A CMS helps the developer in a number of ways. For instance, with a CMS, links to a page that were created automatically will automatically be removed when the page is

deleted (SiteLeads, 2013). The style and content are stored separately, making it easy to make changes to the website.

According to Partis, when using a CMS a websites security is improved (Partis, 2003). This is because the information stored in your database system can only be accessed by using the CMS. Partis also says that when using a CMS, content will be updated more frequently as it is easier to do. This will lead to being ranked higher on search engines. It would also be easier to maintain and change the website after my involvement with it was concluded.

The three open source CMS's that are supposed to work very well are Wordpress (Wordpress, 2013), Drupal (Drupal, 2013) and Joomla (Open Source Matters, 2013). The three CMS's mentioned above all have merits; however, Wordpress seems to be better suited for blog type websites (Rackspace, 2012). As a consequence of this, I would not recommend it. Drupal is used for complex and highly customised websites. Joomla seems to fall somewhere between Wordpress and Drupal (Rackspace, 2012). For anyone doing a project similar to this, it would be beneficial to use either Joomla or Drupal.

Goldilocks Approach

If I was to start this project again, the Goldilocks Approach (Quick and McKreever, 2013) would be implemented. The Goldilocks Approach would help to facilitate the huge amount of mobile devices and tablets that are currently in use. Utilising the Goldilocks Approach would mean that only one website would need to be made for all the different screen sizes that may be viewing the Website. The size of the page is scaled to whatever the optimal size of the device being used to view it.

When the Goldilocks Approach is in use, the webpage is split up into columns. If a column cannot be displayed in full, the content of the webpage will automatically scale itself to fill the remaining columns. By utilising this, the website should be able to be displayed on all screen sizes. This is advantageous when compared to making mobile versions of sites.

If the Goldilocks Approach was used, the Munster Fléadh Website would be able to scale to any screen size. Without this, the user will have zoom and scroll to use

the site. With the above evidence, one can see that there are advantages to using the Goldilocks Approach.

Early User Testing

Something that I would suggest to someone who wants to embark on a similar project to this is to have as much user testing as possible. For this project, there were only three rounds of testing. Two informal user tests with the same participants and a formal usability test with a new group of participants. The first two tests only had six participants whereas the formal usability test had eight participants.

When working on a project it is simple to get caught up in what you're doing and miss small errors or mistakes you may have made. Getting fresh eyes to look at your work is always good. It can give you good feedback for change or maybe some suggestions of how to do something differently. It could be compared to proof reading, the author may be so familiar with their work, that they may simply not see some mistakes or issues. A developer may be so used to their work that it takes someone who has not seen it before to bring new solutions or suggestions.

The earlier that testing is brought into your work, the better. It may stop you from making mistakes in the future. I would recommend, to anyone doing this sort of project in the future, to test early and test often. It will save a lot of time and help the end product be more suited for the consumer.

Future Work

There is some scope for future work on this project. I would like to use a CMS. I am not sure of the work behind this, whether it consists of converting everything page by page or simply installing a CMS onto the server. Integrating a CMS would make it much easier in the future for the clients to make any edits that they would need to make in the future. However, since this is a project for the Munster Fléadh 2013, it was made for a once off event.

The integration of a system where the admins would be able to see what users were online at that time would be beneficial to this website. With more time perhaps this is something I could strive to complete, but as time is limited and I would have to complete much more research, it would be impossible to implement this functionality.

Something that could be integrated in the future would be the ability for users to pay for events through the site. However, for this to be viable, the clients would need to have PayPal accounts set up specifically. We discussed this early on but the clients did not show much of an interest in the idea.

The ability for users to search the site more easily could be added in the future. The only part of the site that has search capabilities is the 'Results' section. This was a late addition. The basic code for this search could be applied in other places to help users search. Another thing that could be added would be the ability for users to have their password reset. At the moment, the user uses the contact form and then their password is changed to something simple and they are emailed the password. The process would run a lot smoother if it was automated.

Getting participants who would be attending the Munster Fléadh (or any Fléadh for that matter) would be a huge benefit. This way aside from checking for errors and usability, they would be able to direct me as to what content or functionality they would like. Since they would be the people attending and using the website, their opinion would be very important.

References

Anon (2001) How Can I Auto Increment Checkbox Names? [online], *phpbuilder*, available: http://board.phpbuilder.com/showthread.php?10126995-How-can-I-auto-increment-checkbox-names [accessed 7 Feb 2013].

Atwood, J. (2006) Coding Horror: The Magical Number Seven Plus or Minus Two [online], available: http://www.codinghorror.com/blog/2006/08/the-magical-number-seven-plus-or-minus-two.html [accessed 7 Mar 2013].

Awkward Group (2012) Awkward Showcase – A jQuery Plugin [online], *Awkward Group*, available: http://www.awkwardgroup.com/sandbox/awkward-showcase-a-jquery-plugin/ [accessed 12 Dec 2012].

Bourdon, R. (2013) WampServer, the Web Development Platform on Windows - Apache, MySQL, PHP [online], available: http://www.wampserver.com/en/ [accessed 6 Mar 2013].

Camenisch, J., shelat, abhi, Sommer, D., Zimmermann, R. (2006) 'Securing user inputs for the web', in *Proceedings of the Second ACM Workshop on Digital Identity Management*, DIM '06, ACM: New York, NY, USA, 33–44, available: http://doi.acm.org/10.1145/1179529.1179536 [accessed 7 Feb 2013].

Carlos, S. (2013) Social Media Participation? Search Engine Visibility Is One More Reason to Do So! [online], *Antezeta Digital Marketing*, available: http://antezeta.com/news/search-engine-visibility-social-media [accessed 18 Feb 2013].

Cesaric, R. (2009) Create Virtual Hosts in WAMP [online], available: http://cesaric.com/?p=255 [accessed 6 Mar 2013].

Cherry, K. (2011) Chunking - Psychology Definition of the Week [online], About.com Psychology, available: http://psychology.about.com/b/2011/07/29/chunking-psychology-definition-of-the-week.htm [accessed 5 Feb 2013].

Coggeshall, J. (2001) Embedding PHP in HTML - O'Reilly Media [online], available: http://onlamp.com/pub/a/php/2001/05/03/php_foundations.html [accessed 17 Feb 2013].

Digital Point (2007) Warning: Session_start() [function.session-start]: Cannot Send Session Cache Limiter [online], available: https://forums.digitalpoint.com/threads/warning-session_start-function-session-start-cannot-send-session-cache-limiter.475425/ [accessed 23 Mar 2013].

Doubé, W., Beh, J. (2012) 'Typing over autocomplete: cognitive load in website use by older adults', in *Proceedings of the 24th Australian Computer-Human Interaction Conference*, OzCHI '12, ACM: New York, NY, USA, 97–106, available: http://doi.acm.org/10.1145/2414536.2414553 [accessed 5 Feb 2013].

Drupal (2013) Drupal - Open Source CMS | Drupal.org [online], available: http://drupal.org/ [accessed 13 Mar 2013].

Dunn, T. (2013) 8 Guidelines for Usability Testing [online], available: http://www.webcredible.co.uk/user-friendly-resources/web-usability/usability-testing.shtml [accessed 13 Mar 2013].

Facebook Comments (2013) Comments - Facebook Developers [online], available: https://developers.facebook.com/docs/reference/plugins/comments/ [accessed 4 Mar 2013].

Facebook Developers (2013) Home - Facebook Developers [online], available: http://developers.facebook.com/ [accessed 5 Mar 2013].

Facebook Like (2013) Like Button [online], *Facebook Developers*, available: https://developers.facebook.com/docs/reference/plugins/like/ [accessed 7 Feb 2013].

Facebook Login (2013) Login with Javascript SDK - Facebook Developers [online], available: https://developers.facebook.com/docs/howtos/login/getting-started/ [accessed 5 Mar 2013].

Facebook Register (2013) Registration - Facebook Developers [online], available: https://developers.facebook.com/docs/plugins/registration/ [accessed 7 Mar 2013].

FileZilla (2012) FileZilla - The Free FTP Solution [online], available: http://filezilla-project.org/ [accessed 23 Mar 2013].

Gibson, C. (2006) 'Website engineering in the real world', in *Proceedings of the 44th Annual Southeast Regional Conference*, ACM-SE 44, ACM: New York, NY, USA, 573–578, available: http://doi.acm.org/10.1145/1185448.1185573 [accessed 7 Oct 2012].

GoDaddy (2013) Domain Names | The World's Largest Domain Name Registrar - GoDaddy.com [online], available: http://www.godaddy.com/ [accessed 23 Mar 2013].

Google Cal (2013) Event Publisher Guide - Google Calendar Help [online], available: http://support.google.com/calendar/bin/answer.py?hl=en&answer=1186917 [accessed 22 Mar 2013].

Google Directions (2013) Driving Directions - Add to Your Homepage [online], available: http://www.google.ie/ig/directory?type=gadgets&url=www.google.com/ig/modules/driving_directions.xml [accessed 1 Mar 2013].

Google Maps (2013) Google Maps [online], *Google Maps*, available: https://maps.google.ie/ [accessed 7 Feb 2013].

Happyworm (2009) jPlayer : HTML5 Audio & Video for jQuery [online], *jPlayer*, available: http://www.jplayer.org/ [accessed 12 Dec 2012].

Heroku (2013) Heroku | Cloud Application Platform [online], available: http://www.heroku.com/ [accessed 6 Mar 2013].

HTML Form Guide (2011) A Simple, Easy PHP Contact Form [online], *HTML Form Guide*, available: http://www.html-form-guide.com/contact-form/simple-php-contact-form.html [accessed 12 Dec 2012].

Java, A., Song, X., Finin, T., Tseng, B. (2007) 'Why we twitter: understanding microblogging usage and communities', in *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis*, WebKDD/SNA-KDD '07, ACM: New York, NY, USA, 56–65, available: http://doi.acm.org/10.1145/1348549.1348556 [accessed 18 Feb 2013].

Javascript Kit (2012) Comprehensive Guide to .htaccess- Intro [online], available: http://www.javascriptkit.com/howto/htaccess.shtml [accessed 22 Mar 2013].

jQuery Foundation (2013) jQuery.noConflict() | jQuery API Documentation [online], *jQuery*, available: http://api.jquery.com/jQuery.noConflict/ [accessed 15 Feb 2013].

JSFiddle (2013) Create a New Fiddle - jsFiddle [online], available: http://jsfiddle.net/ [accessed 25 Mar 2013].

Juicy Scripter (2012) What Is Wrong with This Json Array Code Facebook Form - Stack Overflow [online], available: http://stackoverflow.com/questions/10411218/what-is-wrong-with-this-json-array-code-facebook-form [accessed 25 Mar 2013].

Kasten, E. (1995) 'HTML', *Linux J.*, 1995(15es), available: http://dl.acm.org/citation.cfm?id=324855.324858 [accessed 17 Feb 2013].

Leiva, L.A. (2011) 'Restyling website design via touch-based interactions', in *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '11, ACM: New York, NY, USA, 599–604, available: http://doi.acm.org/10.1145/2037373.2037467 [accessed 6 Feb 2013].Lerner, R. (1997) 'At the Forge: Speaking SQL', *Linux J.*, 1997(41es), available: http://dl.acm.org/citation.cfm?id=326952.326965 [accessed 17 Feb 2013].

Lerner, R.M. (2006) 'At the forge: assessing ruby on rails', *Linux J.*, 2006(142), 10–.

Lerner, R.M. (2009) 'At the forge: jQuery', *Linux J.*, 2009(178), available: http://dl.acm.org/citation.cfm?id=1514256.1514262 [accessed 17 Feb 2013].

Lerner, R.M. (2011) 'At the forge: Ruby on Rails 3', *Linux J.*, 2011(203), available: http://dl.acm.org/citation.cfm?id=1959466.1959472 [accessed 17 Feb 2013].

Lie, H.W., Saarela, J. (1999) 'Multipurpose Web publishing using HTML, XML, and CSS', *Commun. ACM*, 42(10), 95–101.

Liu, N., Yang, C.C. (2005) 'Extracting a website's content structure from its link structure', in *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM '05, ACM: New York, NY, USA, 345–346, available: http://doi.acm.org/10.1145/1099554.1099660 [accessed 7 Oct 2012].

McAnerin (2010) Robots.txt Generator - McAnerin International Inc. [online], available: http://www.mcanerin.com/EN/search-engine/robots-txt.asp [accessed 22 Mar 2013].

Miller, G. (1956) Classics in the History of Psychology -- Miller (1956) [online], available: http://psychclassics.yorku.ca/Miller/ [accessed 7 Mar 2013].

Moreno, L., Martínez, P., Iglesias, A., Gonzalez, M. (2011) 'HTML 5 Support for an Accessible User-Video-Interaction on the Web', in Campos, P., Graham, N., Jorge, J., Nunes, N., Palanque, P. and Winckler, M., eds., *Human-Computer Interaction – INTERACT 2011*, Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 535–539, available: http://www.springerlink.com.proxy.lib.ul.ie/content/pjn10761395257r2/abstract/ [accessed 25 Sep 2012].

Nielsen, J. (1995) 10 Heuristics for User Interface Design [online], available: http://www.nngroup.com/articles/ten-usability-heuristics/ [accessed 15 Feb 2013].

Northwoods Web Design (2012) The Importance of Integrating Social Media and Small Business Websites [online], available: http://northwoodswebdesigns.com/blog/importance-integrating-social-media-website/ [accessed 18 Feb 2013].

Open Source Matters (2013) Joomla! The CMS Trusted By Millions for Their Websites [online], available: http://www.joomla.org/ [accessed 13 Mar 2013].

Partis, A. (2003) Top 10 Benefits of a Content Management System [online], available: http://www.thundernet.com/alpartis/articles/cmstop10.shtml [accessed 13 Mar 2013].

Php Builder (2005) [RESOLVED] Warning: Session_start(): Cannot Send Session Cookie - Headers Already Sent [online], available: http://board.phpbuilder.com/showthread.php?10310794-RESOLVED-Warning-session_start()-Cannot-send-session-cookie-headers-already-sent [accessed 23 Mar 2013].

PHPeasystep (2012) PHP Login Script Tutorial [online], *PHPeasystep*, available: http://www.phpeasystep.com/phptu/6.html [accessed 7 Feb 2013].

PHP Group (2013) PHP: What Can PHP Do? - Manual [online], *php*, available: http://www.php.net/manual/en/intro-whatcando.php [accessed 17 Feb 2013].

phpMyAdmin devel team (2013) phpMyAdmin [online], available: http://www.phpmyadmin.net/home_page/index.php [accessed 17 Feb 2013].

Poblete, B., Baeza-Yates, R. (2006) ‘A content and structure website mining model’, in *Proceedings of the 15th International Conference on World Wide Web*, WWW ’06, ACM: New York, NY, USA, 957–958, available: http://doi.acm.org/10.1145/1135777.1135963 [accessed 7 Oct 2012].

Quick, V., McKreever, P. (2013) The Goldilocks Approach to Responsive Web Design [online], *The Goldilocks Approach*, available: http://goldilocksapproach.com/ [accessed 6 Feb 2013].

Rackspace Support (2012) CMS Comparison: Drupal, Joomla and Wordpress | Knowledge Center | Rackspace Hosting [online], available: http://www.rackspace.com/knowledge_center/article/cms-comparison-drupal-joomla-and-wordpress [accessed 13 Mar 2013].

Refsnes Data (2012) W3Schools Online Web Tutorials [online], *w3schools*, available: http://www.w3schools.com/ [accessed 12 Dec 2012].

Register365 (2013) Domain Name Registration - Web Hosting in Ireland - Domains from Register365 [online], available: http://www.register365.com/ [accessed 23 Mar 2013].

Reinecke, K., Bernstein, A. (2011) ‘Improving performance, perceived usability, and aesthetics with culturally adaptive user interfaces’, *ACM Trans. Comput.-Hum. Interact.*, 18(2), 8:1–8:29.

Rocheleau, J. (2012) How to Create An Ajax-Based HTML5/CSS3 Contact Form [online], *Hongkiat*, available: http://www.hongkiat.com/blog/ajax-html5-css3-contact-form-tutorial/ [accessed 14 Feb 2013].

seaofclouds (2012) Tweet! Put Twitter on Your Site with This Simple, Unobtrusive jQuery Widget [online], available: http://tweet.seaofclouds.com/ [accessed 4 Mar 2013].

Searles, N. (2010) Slides, A Slideshow Plugin for jQuery [online], *SlidesJS*, available: http://slidesjs.com/examples/images-with-captions/ [accessed 12 Dec 2012].

SEO (2011) 4 Reasons Why You Need A Site Map [online], available: http://www.sonicseo.com/why-sitemap/ [accessed 22 Mar 2013].

SiteLeads (2013) The Advantages of a (Web) Content Management System (WCMS) [online], available: http://www.siteleads.net/bm/blog/the-advantages-of-a-web-content-management-system-.shtml [accessed 13 Mar 2013].

Snook, J. (2009) Simplest jQuery Slideshow - Snook.ca [online], available: http://snook.ca/archives/javascript/simplest-jquery-slideshow [accessed 3 Mar 2013].

Stack Overflow (2013) Stack Overflow [online], available: http://stackoverflow.com/ [accessed 25 Mar 2013].

Teat, C., Peltsverger, S. (2011) 'The security of cryptographic hashes', in *Proceedings of the 49th Annual Southeast Regional Conference*, ACM-SE '11, ACM: New York, NY, USA, 103–108, available: http://doi.acm.org/10.1145/2016039.2016072 [accessed 8 Feb 2013].

Tsaganos, N. (2012) A 'Stick-to-top' Navigation Menu with jQuery - Backslash [online], available: http://www.backslash.gr/demos/jquery-sticky-navigation/ [accessed 12 Dec 2012].

Tuck, M. (2010) Gestalt Principles Applied in Design [online], *Six Revisions*, available: http://sixrevisions.com/web_design/gestalt-principles-applied-in-design/ [accessed 7 Feb 2013].

Twitter Developers (2013) Twitter Buttons [online], *Twitter*, available: https://twitter.com/about/resources/buttons [accessed 7 Feb 2013].

Twitter Login (2013) Implementing Sign in with Twitter | Twitter Developers [online], available: https://dev.twitter.com/docs/auth/implementing-sign-twitter [accessed 18 Mar 2013].

Wang, Y.D., Zahadat, N. (2009) 'Teaching web development in the web 2.0 era', in *Proceedings of the 10th ACM Conference on SIG-information Technology Education*, SIGITE '09, ACM: New York, NY, USA, 80–86, available: http://doi.acm.org/10.1145/1631728.1631753 [accessed 24 Sep 2012].

WebMasterWorld (2006) Why Should I Have a Robots.txt File? Sitemaps, Meta Data, and Robots.txt Forum at WebmasterWorld [online], available: http://www.webmasterworld.com/robots_txt/3203372.htm [accessed 22 Mar 2013].

Wordpress (2013) WordPress › Blog Tool, Publishing Platform, and CMS [online], available: http://wordpress.org/ [accessed 13 Mar 2013].

XML Sitemaps (2013) Create Your Google Sitemap Online - XML Sitemaps Generator [online], available: http://www.xml-sitemaps.com/ [accessed 22 Mar 2013].

Bibliography

Al-Shamaileh, O., Sutcliffe, A. (2012) 'The effect of website interactivity and repeated exposure on user experience', in *Proceedings of the 4th Mexican Conference on Human-Computer Interaction*, MexIHC '12, ACM: New York, NY, USA, 1–8, available: http://doi.acm.org/10.1145/2382176.2382178 [accessed 6 Feb 2013].

An, J., Chaudhuri, A., Foster, J.S. (2009) 'Static Typing for Ruby on Rails', in *Proceedings of the 2009 IEEE/ACM International Conference on Automated Software Engineering*, ASE '09, IEEE Computer Society: Washington, DC, USA, 590–594, available: http://dx.doi.org/10.1109/ASE.2009.80 [accessed 17 Feb 2013].

Anon (2001) How Can I Auto Increment Checkbox Names? [online], *phpbuilder*, available: http://board.phpbuilder.com/showthread.php?10126995-How-can-I-auto-increment-checkbox-names [accessed 7 Feb 2013].

Atwood, J. (2006) Coding Horror: The Magical Number Seven Plus or Minus Two [online], available: http://www.codinghorror.com/blog/2006/08/the-magical-number-seven-plus-or-minus-two.html [accessed 7 Mar 2013].

Awkward Group (2012) Awkward Showcase – A jQuery Plugin [online], *Awkward Group*, available: http://www.awkwardgroup.com/sandbox/awkward-showcase-a-jquery-plugin/ [accessed 12 Dec 2012].

Axmark, D., Widenius, M. (1999) 'MySQL Introduction', *Linux J.*, 1999(67es), available: http://dl.acm.org/citation.cfm?id=328036.328041 [accessed 17 Feb 2013].

Banks, L., Ye, S., Huang, Y., Wu, S.F. (2007) 'Davis social links: integrating social networks with internet routing', in *Proceedings of the 2007 Workshop on Large Scale Attack Defense*, LSAD '07, ACM: New York, NY, USA, 121–128, available: http://doi.acm.org/10.1145/1352664.1352671 [accessed 18 Feb 2013].

Barry, P. (2006) 'Web reporting with MySQL, CSS and Perl', *Linux J.*, 2006(149), 8–.

Barry, P. (2009) 'Improved scaffolding for Ruby on Rails', *Linux J.*, 2009(178), available: http://dl.acm.org/citation.cfm?id=1514256.1514259 [accessed 17 Feb 2013].

Bourdon, R. (2013) WampServer, the Web Development Platform on Windows - Apache, MySQL, PHP [online], available: http://www.wampserver.com/en/ [accessed 6 Mar 2013].

Brinck, T., Gergle, D., Wood, S. (1998) 'Website design from the trenches', in *CHI 98 Cconference Summary on Human Factors in Computing Systems*, CHI '98, ACM: New York, NY, USA, 133–134, available: http://doi.acm.org/10.1145/286498.286579 [accessed 6 Feb 2013].

Php Builder (2005) [RESOLVED] Warning: Session_start(): Cannot Send Session Cookie - Headers Already Sent [online], available: http://board.phpbuilder.com/showthread.php?10310794-RESOLVED-Warning-session_start()-Cannot-send-session-cookie-headers-already-sent [accessed 23 Mar 2013].

Butkiewicz, M., Madhyastha, H.V., Sekar, V. (2011) 'Understanding website complexity: measurements, metrics, and implications', in *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, IMC '11, ACM: New York, NY, USA, 313–328, available: http://doi.acm.org/10.1145/2068816.2068846 [accessed 7 Oct 2012].

Camenisch, J., shelat, abhi, Sommer, D., Zimmermann, R. (2006) 'Securing user inputs for the web', in *Proceedings of the Second ACM Workshop on Digital Identity Management*, DIM '06, ACM: New York, NY, USA, 33–44, available: http://doi.acm.org/10.1145/1179529.1179536 [accessed 7 Feb 2013].

Carlos, S. (2013) Social Media Participation? Search Engine Visibility Is One More Reason to Do So! [online], *Antezeta Digital Marketing*, available: http://antezeta.com/news/search-engine-visibility-social-media [accessed 18 Feb 2013].

Cesaric, R. (2009) Create Virtual Hosts in WAMP [online], available: http://cesaric.com/?p=255 [accessed 6 Mar 2013].

Chaudhuri, A., Foster, J.S. (2010) 'Symbolic security analysis of ruby-on-rails web applications', in *Proceedings of the 17th ACM Conference on Computer and Communications Security*, CCS '10, ACM: New York, NY, USA, 585–594, available: http://doi.acm.org/10.1145/1866307.1866373 [accessed 17 Feb 2013].

Cherry, K. (2011) Chunking - Psychology Definition of the Week [online], *About.com Psychology*, available: http://psychology.about.com/b/2011/07/29/chunking-psychology-definition-of-the-week.htm [accessed 5 Feb 2013].

Cholakov, N. (2008) 'On some drawbacks of the PHP platform', in *Proceedings of the 9th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing*, CompSysTech '08, ACM: New York, NY, USA, 12:II.7–12:2, available: http://doi.acm.org/10.1145/1500879.1500894 [accessed 17 Feb 2013].

Coggeshall, J. (2001) Embedding PHP in HTML - O'Reilly Media [online], available: http://onlamp.com/pub/a/php/2001/05/03/php_foundations.html [accessed 17 Feb 2013].

Comments (2010) Php - Warning: Session_start(): Cannot Send Session Cache Limiter - Headers Already Sent - Stack Overflow [online], available: http://stackoverflow.com/questions/2703547/warning-session-start-cannot-send-session-cache-limiter-headers-already-se [accessed 23 Mar 2013].

Curry, R., Kiddle, C., Simmonds, R. (2009) 'Social networking and scientific gateways', in *Proceedings of the 5th Grid Computing Environments Workshop*, GCE '09, ACM: New York, NY, USA, 4:1–4:10, available: http://doi.acm.org/10.1145/1658260.1658266 [accessed 18 Feb 2013].

Digital Point (2007) Warning: Session_start() [function.session-start]: Cannot Send Session Cache Limiter [online], available: https://forums.digitalpoint.com/threads/warning-session_start-function-session-start-cannot-send-session-cache-limiter.475425/ [accessed 23 Mar 2013].

Doubé, W., Beh, J. (2012) 'Typing over autocomplete: cognitive load in website use by older adults', in *Proceedings of the 24th Australian Computer-Human Interaction Conference*, OzCHI '12, ACM: New York, NY, USA, 97–106, available: http://doi.acm.org/10.1145/2414536.2414553 [accessed 5 Feb 2013].

Drupal (2013) Drupal - Open Source CMS | Drupal.org [online], available: http://drupal.org/ [accessed 13 Mar 2013].

Dunn, T. (2013) 8 Guidelines for Usability Testing [online], available: http://www.webcredible.co.uk/user-friendly-resources/web-usability/usability-testing.shtml [accessed 13 Mar 2013].

ECM (2011) What Are the Benefits of Web Content Management? [online], available: http://www.contentmanager.eu.com/wcmsbens.htm [accessed 13 Mar 2013].

Facebook Comments (2013) Comments - Facebook Developers [online], available: https://developers.facebook.com/docs/reference/plugins/comments/ [accessed 4 Mar 2013].

Facebook Developers (2013) Home - Facebook Developers [online], available: http://developers.facebook.com/ [accessed 5 Mar 2013].

Facebook Like (2013) Like Button [online], *Facebook Developers*, available: https://developers.facebook.com/docs/reference/plugins/like/ [accessed 7 Feb 2013].

Facebook Login (2013) Login with Javascript SDK - Facebook Developers [online], available: https://developers.facebook.com/docs/howtos/login/getting-started/ [accessed 5 Mar 2013].

Facebook Register (2013) Registration - Facebook Developers [online], available: https://developers.facebook.com/docs/plugins/registration/ [accessed 7 Mar 2013].

FileZilla (2012) FileZilla - The Free FTP Solution [online], available: http://filezilla-project.org/ [accessed 23 Mar 2013].

Garvey, G.P. (2006) '3D or not 3D - that is the question!', in *ACM SIGGRAPH 2006 Educators Program*, SIGGRAPH '06, ACM: New York, NY, USA, available: http://doi.acm.org/10.1145/1179295.1179327 [accessed 7 Feb 2013].

Geoghegan, S.J., Landers, L., Gray, T. (2009) 'The evaluation of MD5 hash tools as a digital forensics class experiment', *J. Comput. Sci. Coll.*, 24(5), 50–56.

Gergle, D., Brinck, T., Wood, S. (1999) 'Practical usability methods in website design', in *CHI '99 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '99, ACM: New York, NY, USA, 110–111, available: http://doi.acm.org/10.1145/632716.632786 [accessed 6 Feb 2013].

Gibson, C. (2006) 'Website engineering in the real world', in *Proceedings of the 44th Annual Southeast Regional Conference*, ACM-SE 44, ACM: New York, NY, USA, 573–578, available: http://doi.acm.org/10.1145/1185448.1185573 [accessed 7 Oct 2012].

GoDaddy (2013) Domain Names | The World's Largest Domain Name Registrar - GoDaddy.com [online], available: http://www.godaddy.com/ [accessed 23 Mar 2013].

Golbeck, J., Wasser, M.M. (2007) 'SocialBrowsing: integrating social networks and web browsing', in *CHI '07 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '07, ACM: New York, NY, USA, 2381–2386, available: http://doi.acm.org/10.1145/1240866.1241011 [accessed 18 Feb 2013].

Google Cal (2013) Event Publisher Guide - Google Calendar Help [online], available: http://support.google.com/calendar/bin/answer.py?hl=en&answer=1186917 [accessed 22 Mar 2013].

Google Directions (2013) Driving Directions - Add to Your Homepage [online], available: http://www.google.ie/ig/directory?type=gadgets&url=www.google.com/ig/modules/driving_directions.xml [accessed 1 Mar 2013].

Google Maps (2013) Google Maps [online], *Google Maps*, available: https://maps.google.ie/ [accessed 7 Feb 2013].

Happyworm (2009) jPlayer : HTML5 Audio & Video for jQuery [online], *jPlayer*, available: http://www.jplayer.org/ [accessed 12 Dec 2012].

Heroku (2013) Heroku | Cloud Application Platform [online], available: http://www.heroku.com/ [accessed 6 Mar 2013].

Houshmand, S., Aggarwal, S. (2012) 'Building better passwords using probabilistic techniques', in *Proceedings of the 28th Annual Computer Security Applications Conference*, ACSAC '12, ACM: New York, NY, USA, 109–118, available: http://doi.acm.org/10.1145/2420950.2420966 [accessed 8 Feb 2013].

HTML Form Guide (2011) A Simple, Easy PHP Contact Form [online], *HTML Form Guide*, available: http://www.html-form-guide.com/contact-form/simple-php-contact-form.html [accessed 12 Dec 2012].

Java, A., Song, X., Finin, T., Tseng, B. (2007) 'Why we twitter: understanding microblogging usage and communities', in *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis*, WebKDD/SNA-KDD '07, ACM: New York, NY, USA, 56–65, available: http://doi.acm.org/10.1145/1348549.1348556 [accessed 18 Feb 2013].

Javascript Kit (2012) Comprehensive Guide to .htaccess- Intro [online], available: http://www.javascriptkit.com/howto/htaccess.shtml [accessed 22 Mar 2013].

jQuery Foundation (2013) jQuery.noConflict() | jQuery API Documentation [online], *jQuery*, available: http://api.jquery.com/jQuery.noConflict/ [accessed 15 Feb 2013].

JSFiddle (2013) Create a New Fiddle - jsFiddle [online], available: http://jsfiddle.net/ [accessed 25 Mar 2013].

Juicy Scripter (2012) What Is Wrong with This Json Array Code Facebook Form - Stack Overflow [online], available: http://stackoverflow.com/questions/10411218/what-is-wrong-with-this-json-array-code-facebook-form [accessed 25 Mar 2013].

Kasten, E. (1995a) 'HTML', *Linux J.*, 1995(15es), available: http://dl.acm.org/citation.cfm?id=324855.324858 [accessed 17 Feb 2013].

Kasten, E. (1995b) 'HTML Forms', *Linux J.*, 1995(16es), available: http://dl.acm.org/citation.cfm?id=324870.324871 [accessed 17 Feb 2013].

Kwak, H., Lee, C., Park, H., Moon, S. (2010) 'What is Twitter, a social network or a news media?', in *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, ACM: New York, NY, USA, 591–600, available: http://doi.acm.org/10.1145/1772690.1772751 [accessed 18 Feb 2013].

Leiva, L.A. (2011) 'Restyling website design via touch-based interactions', in *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '11, ACM: New York, NY, USA, 599–604, available: http://doi.acm.org/10.1145/2037373.2037467 [accessed 6 Feb 2013].

Lerner, R. (1997) 'At the Forge: Speaking SQL', *Linux J.*, 1997(41es), available: http://dl.acm.org/citation.cfm?id=326952.326965 [accessed 17 Feb 2013].

Lerner, R.M. (2006a) 'At the forge: assessing ruby on rails', *Linux J.*, 2006(142), 10–.

Lerner, R.M. (2006b) 'At the forge: JavaScript, Forms and Ajax', *Linux J.*, 2006(150), 8–.

Lerner, R.M. (2007) 'MySQL deserves a double take', *Linux J.*, 2007(156), 18–.

Lerner, R.M. (2009a) 'At the forge: checking your html', *Linux J.*, 2009(182), available: http://dl.acm.org/citation.cfm?id=1554324.1554329 [accessed 17 Feb 2013].

Lerner, R.M. (2009b) 'At the forge: jQuery', *Linux J.*, 2009(178), available: http://dl.acm.org/citation.cfm?id=1514256.1514262 [accessed 17 Feb 2013].

Lerner, R.M. (2009c) ‘At the forge: writing jQuery plugins’, *Linux J.*, 2009(180), available: http://dl.acm.org/citation.cfm?id=1524272.1524279 [accessed 17 Feb 2013].

Lerner, R.M. (2011) ‘At the forge: Ruby on Rails 3’, *Linux J.*, 2011(203), available: http://dl.acm.org/citation.cfm?id=1959466.1959472 [accessed 17 Feb 2013].

Lie, H.W., Saarela, J. (1999) ‘Multipurpose Web publishing using HTML, XML, and CSS’, *Commun. ACM*, 42(10), 95–101.

Liu, N., Yang, C.C. (2005) ‘Extracting a website’s content structure from its link structure’, in *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM ’05, ACM: New York, NY, USA, 345–346, available: http://doi.acm.org/10.1145/1099554.1099660 [accessed 7 Oct 2012].

Liu, N., Yang, C.C. (2007) ‘A link classification based approach to website topic hierarchy generation’, in *Proceedings of the 16th International Conference on World Wide Web*, WWW ’07, ACM: New York, NY, USA, 1127–1128, available: http://doi.acm.org/10.1145/1242572.1242728 [accessed 6 Feb 2013].

Mahmud, J., Drews, C., Collins, M., Carreno-Fuentes, A., Bullard, A., Vickstrom, M., Cho, M. (2012) ‘Test-driven development for the web: increasing efficiency of web development’, in *CHI ’12 Extended Abstracts on Human Factors in Computing Systems*, CHI EA ’12, ACM: New York, NY, USA, 1931–1936, available: http://doi.acm.org/10.1145/2212776.2223731 [accessed 7 Feb 2013].

McAnerin (2010) Robots.txt Generator - McAnerin International Inc. [online], available: http://www.mcanerin.com/EN/search-engine/robots-txt.asp [accessed 22 Mar 2013].

McCormick, E., De Volder, K. (2004) ‘JQuery: finding your way through tangled code’, in *Companion to the 19th Annual ACM SIGPLAN Conference on Object-oriented Programming Systems, Languages, and Applications*, OOPSLA ’04, ACM: New York, NY, USA, 9–10, available: http://doi.acm.org/10.1145/1028664.1028670 [accessed 17 Feb 2013].

Miller, G. (1956) Classics in the History of Psychology -- Miller (1956) [online], available: http://psychclassics.yorku.ca/Miller/ [accessed 7 Mar 2013].

Moreno, L., Martínez, P., Iglesias, A., Gonzalez, M. (2011) ‘HTML 5 Support for an Accessible User-Video-Interaction on the Web’, in Campos, P., Graham, N., Jorge, J., Nunes, N., Palanque, P. and Winckler, M., eds., *Human-Computer Interaction – INTERACT 2011*, Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 535–539, available: http://www.springerlink.com.proxy.lib.ul.ie/content/pjn10761395257r2/abstract/ [accessed 25 Sep 2012].

Nielsen, J. (1995) 10 Heuristics for User Interface Design [online], available: http://www.nngroup.com/articles/ten-usability-heuristics/ [accessed 15 Feb 2013].

Northwoods Web Design (2012) The Importance of Integrating Social Media and Small Business Websites [online], available: http://northwoodswebdesigns.com/blog/importance-integrating-social-media-website/ [accessed 18 Feb 2013].

Open Source Matters (2013) Joomla! The CMS Trusted By Millions for Their Websites [online], available: http://www.joomla.org/ [accessed 13 Mar 2013].

Partis, A. (2003) Top 10 Benefits of a Content Management System [online], available: http://www.thundernet.com/alpartis/articles/cmstop10.shtml [accessed 13 Mar 2013].

Petrie, H., Hamilton, F., King, N. (2004) 'Tension, what tension?: Website accessibility and visual design', in *Proceedings of the 2004 International Cross-disciplinary Workshop on Web Accessibility (W4A)*, W4A '04, ACM: New York, NY, USA, 13–18, available: http://doi.acm.org/10.1145/990657.990660 [accessed 6 Feb 2013].

PHP Group (2013) PHP: What Can PHP Do? - Manual [online], *php*, available: http://www.php.net/manual/en/intro-whatcando.php [accessed 17 Feb 2013].

PHPeasystep (2012) PHP Login Script Tutorial [online], *PHPeasystep*, available: http://www.phpeasystep.com/phptu/6.html [accessed 7 Feb 2013].

phpMyAdmin devel team (2013) phpMyAdmin [online], available: http://www.phpmyadmin.net/home_page/index.php [accessed 17 Feb 2013].

Poblete, B., Baeza-Yates, R. (2006) 'A content and structure website mining model', in *Proceedings of the 15th International Conference on World Wide Web*, WWW '06, ACM: New York, NY, USA, 957–958, available: http://doi.acm.org/10.1145/1135777.1135963 [accessed 7 Oct 2012].

Puerta, A., Micheletti, M., Mak, A. (2005) 'The UI pilot: a model-based tool to guide early interface design', in *Proceedings of the 10th International Conference on Intelligent User Interfaces*, IUI '05, ACM: New York, NY, USA, 215–222, available: http://doi.acm.org/10.1145/1040830.1040877 [accessed 7 Feb 2013].

Quick, V., McKreever, P. (2013) The Goldilocks Approach to Responsive Web Design [online], *The Goldilocks Approach*, available: http://goldilocksapproach.com/ [accessed 6 Feb 2013].

Rackspace Support (2012) CMS Comparison: Drupal, Joomla and Wordpress | Knowledge Center | Rackspace Hosting [online], available: http://www.rackspace.com/knowledge_center/article/cms-comparison-drupal-joomla-and-wordpress [accessed 13 Mar 2013].

Refsnes Data (2012) W3Schools Online Web Tutorials [online], *w3schools*, available: http://www.w3schools.com/ [accessed 12 Dec 2012].

Register365 (2013) Domain Name Registration - Web Hosting in Ireland - Domains from Register365 [online], available: http://www.register365.com/ [accessed 23 Mar 2013].

Reinecke, K., Bernstein, A. (2011) 'Improving performance, perceived usability, and aesthetics with culturally adaptive user interfaces', *ACM Trans. Comput.-Hum. Interact.*, 18(2), 8:1–8:29.

Robots (2007a) The Web Robots Pages [online], available: http://www.robotstxt.org/ [accessed 22 Mar 2013].

Robots (2007b) The Web Robots Pages | About [online], available: http://www.robotstxt.org/robotstxt.html [accessed 22 Mar 2013].

Rocheleau, J. (2012) How to Create An Ajax-Based HTML5/CSS3 Contact Form [online], *Hongkiat*, available: http://www.hongkiat.com/blog/ajax-html5-css3-contact-form-tutorial/ [accessed 14 Feb 2013].

seaofclouds (2012) Tweet! Put Twitter on Your Site with This Simple, Unobtrusive jQuery Widget [online], available: http://tweet.seaofclouds.com/ [accessed 4 Mar 2013].

Searles, N. (2010) Slides, A Slideshow Plugin for jQuery [online], *SlidesJS*, available: http://slidesjs.com/examples/images-with-captions/ [accessed 12 Dec 2012].

SEO (2011) 4 Reasons Why You Need A Site Map [online], available: http://www.sonicseo.com/why-sitemap/ [accessed 22 Mar 2013].

Shoogle Design (2010) Create a Funky 404 Page with jQuery Plugin jParallax | Shoogle Designs' Blog [online], available: http://shoogledesigns.com/blog/blog/2010/10/04/create-a-funky-404-page-with-jquery-plugin-jparallax/ [accessed 5 Mar 2013].

SiteLeads (2013) The Advantages of a (Web) Content Management System (WCMS) [online], available: http://www.siteleads.net/bm/blog/the-advantages-of-a-web-content-management-system-.shtml [accessed 13 Mar 2013].

Snook, J. (2009) Simplest jQuery Slideshow - Snook.ca [online], available: http://snook.ca/archives/javascript/simplest-jquery-slideshow [accessed 3 Mar 2013].

Stack Overflow (2013) Stack Overflow [online], available: http://stackoverflow.com/ [accessed 25 Mar 2013].

Stevens, S.M., Dornburg, C.C. (2009) 'Utilizing pathfinder in the design of an intranet website', in *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, ACM: New York, NY, USA, 4087–4092, available: http://doi.acm.org/10.1145/1520340.1520622 [accessed 6 Feb 2013].

Tamada, R. (2011) User Signup Using Facebook Data [online], available: http://www.9lessons.info/2011/01/user-signup-using-facebook-data.html [accessed 25 Mar 2013].

Tatsubori, M., Tozawa, A., Suzumura, T., Trent, S., Onodera, T. (2010) 'Evaluation of a just-in-time compiler retrofitted for PHP', in *Proceedings of the 6th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments*, VEE '10, ACM: New York, NY, USA, 121–132, available: http://doi.acm.org/10.1145/1735997.1736015 [accessed 17 Feb 2013].

Teat, C., Peltsverger, S. (2011) 'The security of cryptographic hashes', in *Proceedings of the 49th Annual Southeast Regional Conference*, ACM-SE '11, ACM: New York, NY, USA, 103–108, available: http://doi.acm.org/10.1145/2016039.2016072 [accessed 8 Feb 2013].

Touch, J.D. (1995) 'Performance analysis of MD5', *SIGCOMM Comput. Commun. Rev.*, 25(4), 77–86.

Tsaganos, N. (2012) A 'Stick-to-top' Navigation Menu with jQuery - Backslash [online], available: http://www.backslash.gr/demos/jquery-sticky-navigation/ [accessed 12 Dec 2012].

Tuck, M. (2010) Gestalt Principles Applied in Design [online], *Six Revisions*, available: http://sixrevisions.com/web_design/gestalt-principles-applied-in-design/ [accessed 7 Feb 2013].

Twitter Login (2013) Implementing Sign in with Twitter | Twitter Developers [online], available: https://dev.twitter.com/docs/auth/implementing-sign-twitter [accessed 18 Mar 2013].

Twtter Developers (2013) Twitter Buttons [online], *Twitter*, available: https://twitter.com/about/resources/buttons [accessed 7 Feb 2013].

Vowels, S.A. (2005) 'DIY-IT: an empirical study of website development staffing by non-profit organizations', in *Proceedings of the 2005 ACM SIGMIS CPR Conference on Computer Personnel Research*, SIGMIS CPR '05, ACM: New York, NY, USA, 59–61, available: http://doi.acm.org/10.1145/1055973.1055985 [accessed 7 Feb 2013].

Wang, Y.D., Zahadat, N. (2009) 'Teaching web development in the web 2.0 era', in *Proceedings of the 10th ACM Conference on SIG-information Technology Education*, SIGITE '09, ACM: New York, NY, USA, 80–86, available: http://doi.acm.org/10.1145/1631728.1631753 [accessed 24 Sep 2012].

Web Design Shock (2011) 404 Error Page Designs, 200+ Best for Inspiration [online], available: http://www.webdesignshock.com/404-error-page-designs/ [accessed 5 Mar 2013].

WebMasterWorld (2006) Why Should I Have a Robots.txt File? Sitemaps, Meta Data, and Robots.txt Forum at WebmasterWorld [online], available: http://www.webmasterworld.com/robots_txt/3203372.htm [accessed 22 Mar 2013].

Weingroff, M., Bhushman, S. (2005) 'Tools for managing collaboration, communication, and website content development in a distributed digital library community', in *Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '05, ACM: New York, NY, USA, 401–401, available: http://doi.acm.org/10.1145/1065385.1065504 [accessed 7 Feb 2013].

Weischedel, B., Huizingh, E.K.R.E. (2006) 'Website optimization with web metrics: a case study', in *Proceedings of the 8th International Conference on Electronic Commerce: The New E-commerce: Innovations for Conquering Current Barriers, Obstacles and Limitations to Conducting Successful Business on the Internet*, ICEC '06, ACM: New York, NY, USA, 463–470, available: http://doi.acm.org/10.1145/1151454.1151525 [accessed 7 Oct 2012].

Wellman, B., Haase, A.Q., Witte, J., Hampton, K. (2001) 'Does the Internet Increase, Decrease, or Supplement Social Capital? Social Networks, Participation, and Community Commitment', *American Behavioral Scientist*, 45(3), 436–455.

Wikipedia (2013) Content Management System - Wikipedia, the Free Encyclopedia [online], available: http://en.wikipedia.org/wiki/Content_management_system [accessed 13 Mar 2013].

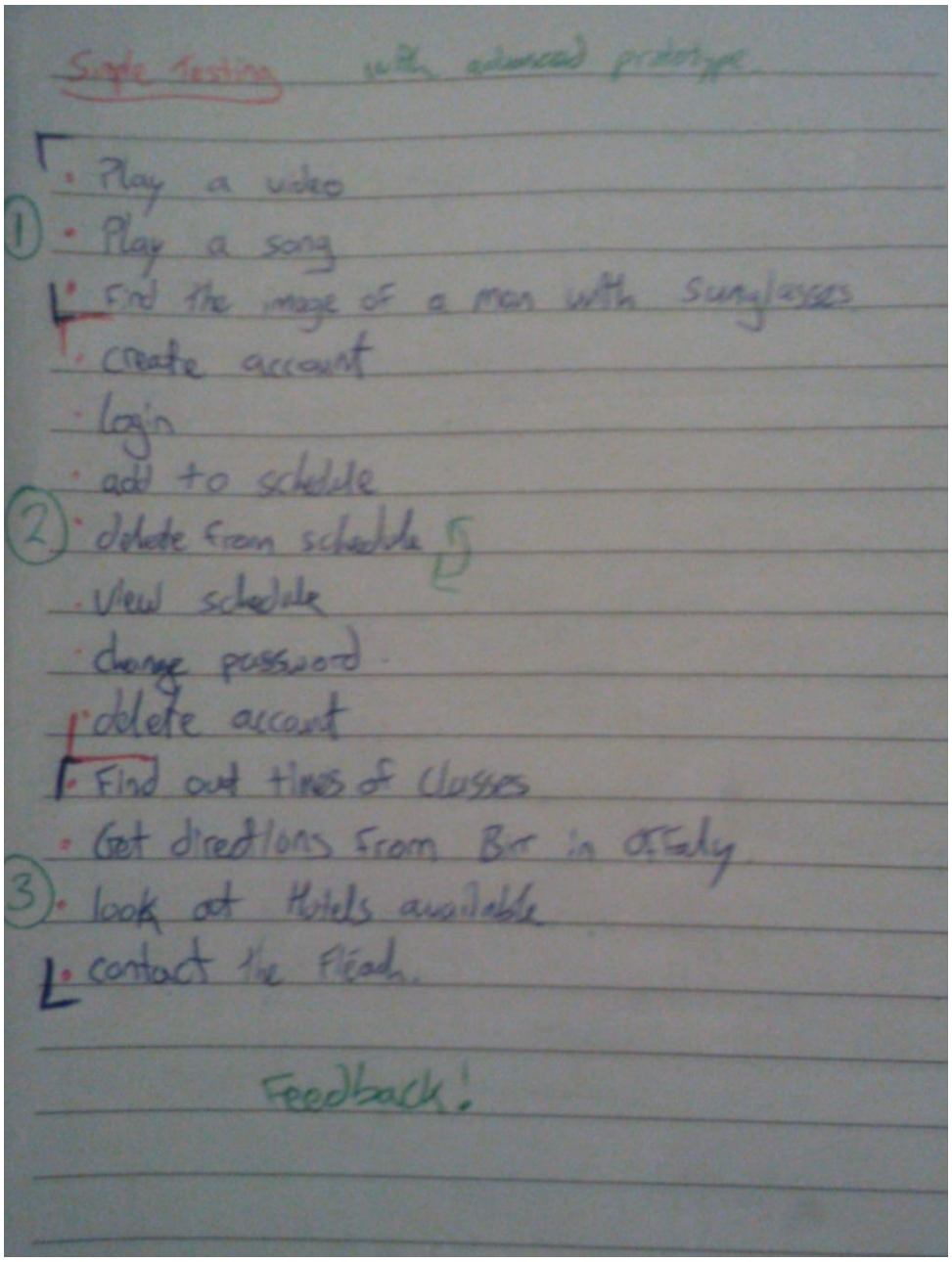
Wolff, D.A. (2001) 'MySQL, PostgreSQL, and PHP: open source technologies for a database management course', *J. Comput. Sci. Coll.*, 17(2), 91–92.

Wordpress (2013) WordPress › Blog Tool, Publishing Platform, and CMS [online], available: http://wordpress.org/ [accessed 13 Mar 2013].

XML Sitemaps (2013) Create Your Google Sitemap Online - XML Sitemaps Generator [online], available: http://www.xml-sitemaps.com/ [accessed 22 Mar 2013].

Appendices

Tasks from the Informal Tests



Tasks from the first and second informal test.

Formal Usability Testing Forms

UNIVERSITY *of* LIMERICK

Declaration of Informed Consent

I, the undersigned, declare that I am willing to take part in research for the project entitled "Munster Fléadh Web Development".

- I declare that I have been fully briefed on the nature of this study and my role in it and have been given the opportunity to ask questions before agreeing to participate.
- The nature of my participation has been explained to me and I have full knowledge of how the information collected will be used.
- I am also aware that my participation in this study may be recorded (video/audio) and I agree to this. However, should I feel uncomfortable at any time I can request that the recording equipment be switched off. I am entitled to copies of all recordings made and am fully informed as to what will happen to these recordings once the study is completed.
- I fully understand that there is no obligation on me to participate in this study I fully understand that I am free to withdraw my participation at any time without having to explain or give a reason.
- I am also entitled to full confidentiality in terms of my participation and personal details.

_____ _____

Signature of participant Date

Contact Information

Researcher: Turlough O'Shea, 09004486@studentmail.ul.ie
Supervisor: Chris Exton, chris.exton@ul.ie

Please write down whatever comes to mind as you read through the following!

The whole process should take between 5 – 10 minutes.

Before Completing the Instructions

The address of this website will be www.munsterfleadh.ie. (This was later changed to www.castletroycomhaltas.com) With an address like that, what would you imagine would be included within this website?

_____

_____

_____

Please take a few minutes to familiarise yourself with this website. From this initial viewing, what do you think of the layout? Is there anything you would change?

_____

_____

_____

Instructions

Please complete the following tasks/instructions while writing down thoughts!

All Tasks should be able to be completed without leaving the website.

1. You just found this webpage and want to see if they have a social network presence. Aside from 'Liking' the page on Facebook, 'following' it on Twitter or 'tweeting' with the #FleadhNaMumhan (# = hashtag) is there any other social network integration?

_____

_____

_____

2. You hear that 'Goitse' will be performing at the Munster Fleadh in 2013. You have never listened to or seen any videos of them before and would like to before attending the Munster Fleadh. How could you go about this?

_____

_____

_____

3. You would like to book some accommodation for the duration of the Munster Fleadh but do not know the area. How would you go about finding out what accommodation is available?

_____

_____

_____

4. You live in Birr, Offaly and will be traveling to UL for the event. You decided you are going to drive but do not know the directions. How would you find directions to UL?

_____

_____

_____

5. You would like to inquire about an event taking place during the Munster Fleadh – find the number of the ‘runaí’ or secretary.

_____

_____

_____

6. You have heard that there may be special offers to people who have created an account. Create an account with the website and proceed to login.

_____

_____

_____

7. When logged in, you notice that you can make your own schedule of events. You would like to add an event to your schedule.

_____

_____

_____

8. After you added the previous event you realised that you would not be available that day. As a consequence of this, you need to delete this event from your schedule.

_____

_____

_____

9. You think that someone may know your password, because of this, you need to change it. How do you change your password?

_____

_____

_____

10. After this you decide that you do not want your account anymore. How do you delete it?

_____

_____

_____

After Completing the Instructions

What were your impressions of the website?

_____

_____

_____

Do you have any suggestions on how the site could be improved?

_____

_____

_____

What do you recall about the structure and functionality of the website?

_____

_____

_____