Final Year Project Report

Full Unit - Final Report

Local Exchange Trading System

James Black

A report submitted in part fulfilment of the degree of

BSc (Hons) in Computer Science

Supervisor: Nuno Barreiro

Department of Computer Science
Royal Holloway, University of London

April 1, 2015

Declaration

This report has been prepared on the basis of my own work. Where other published and unpublished source materials have been used, these have been acknowledged.

Word Count: 17,158 (w/ appendix : 23,253)

Student Name: James Black

Date of Submission: 01/04/2015

Signature:

Table of Contents

Abstract

In situations of lower income, one of the main problems is being able to afford professional services. There are few examples of this being more prevalent, than in the student community. In such a case it is possible to provide a solution that 'enables' students to offer and exchange services between each other, providing mutual benefits with no financial cost to endure. The Local Exchange Trading System(LETS) designed for students, will provide a centralised hub for enabling the exchange of 'favours' between each other. This will encourage those of different cultural backgrounds to share ideas and skills, with a secondary benefit of building a strong community between students. The LETS system will be designed as a social networking website, whose main feature will be to enable the users to offer and exchange 'favours' between one another. The system will have modern social networking principles in its design and approach, in order to appeal to the student body. Through modern software-engineering principles, the website will be constructed in a professional manner in order to build a robust system. It will then be deployed with its code available as open-source. Since the website will deal with sensitive data, security aspects of social networks will be a large consideration of the website. The focus of this report is to provide a detailed analysis of the techniques and considerations applied throughout this project, including but not limited to :

1. *software-engineering principles used*
2. *security-aspects for websites*
3. *specifics of social networking principles*

The careful development of this application has lead to a more clear understanding of the processes involved with developing a website. This includes

- *the process of planning*
- *deciding on technologies*
- *implementing the features*
- *applying security and privacy aspects*

The application of these processes have contributed to the final output of the project and vastly improved my understanding of all of the concepts. I now have a knowledgeable base of how to apply these ideas, but also an understanding as to how to find the 'right' information, and apply that information into developing a substantial piece of work. This will be highly useful when undertaking any future works.

Chapter 1: Introduction

1.1 Project Specification

A local-exchange trade system(LETS) is an alternative to the current economic monetary system, serving as a way to make an exchange for goods or services. It enables members to offer services in exchange for credits, which can then be exchanged for a service provided by another user. These can be viewed of as 'favours', where credits and favours are exchanged acting as the forms of currency. It provides a system that can mutually benefit all of its members. As a secondary benefit, it breaks down barriers between social groups by encouraging a wide variety of mixing between many different types of people. In a region of low-income, it enables the transfer of skills and services to improve a person or group's future prospects, by providing them with a new skill such as learning a new language, or simply providing them with an essential service such as plumbing their kitchen.

It is this approach of transferring skills that will have a wide impact in a student environment. The traditional financial model that can leave students with very little disposable income, the use of LETS will enable them to try new things. There will also be a more general impact of greater social interaction, as it will encourage groups or individuals to encounter and experience new ideas and concepts in a friendly and safe context.

1.2 Motivation

The project specification requires the website to embrace the concept of the LETS, by developing a social network for the student body that can perform all of the features that a LETS provides for its users. This includes

- creating a profile, logging in with that profile
- create favours for other users to respond to
- allow messaging system between users, including the users to manage an inbox
- providing a search facility for viewing users/posts

As a modern social network, there is a responsibility to the users to ensure that it has the most modern security features that can be implemented, including an appropriate method for a validation process for allowing access to the website.

1.3 Goals

This project uses a wide variety of web-technologies, some of which need careful research and study to be able use. Over the course of the project, careful planning is necessary in order to learn how these technologies will work together and create a website that is seamless in its execution while providing a useful service for the community.

When considering the wider social context, plans can be made to appropriately take this into account, and construct a social network, that has a positive community input with a social conscience that is mindful of the professional issues that will arise from such a project.

The challenge is in being able to produce such a website, that utilises every technology required and employs software engineering principles. This will improve my skills as a software developer while still helping the wider community. This project will demonstrate the output from constructing a large scale project, and the management required to construct each aspect of it. Careful planning, design and organisation will guarantee that it is the best program it can be. There is also the implication that this website will be a showcase of my work, and so careful use of the framework, with a particular emphasis on the implications that such a website has on the social context, and in the more sensitive aspect of privacy and security, will demonstrate my skill in managing such a project.

Chapter 2: Theoretical Aspects

This section is dedicated to talking about the theoretical aspects associated with this project. There will be a detailed discussion of the topics that are necessary to supply a bit of background information in order to better understand the practical aspects of the project.

2.1 Programming Disciplines

AGILE

During the development, I made use of a product backlog. This is essentially a breakdown of all of the things that need to be achieved. It is a detailed list of the use cases, which is derived from the user stories, which is itself an analysis of the system specification. During development, as it became clear which features need extending or changing, they could be added to the backlog as a reference to add at a future date. Product backlogs are used in AGILE, to divide the workload amongst a team. It is a useful tool to keeping track of the development of a large-scale project.

An AGILE work mentality uses the following values :

- 'Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan' [3]

The implication of this, is to start the development of the software as soon as possible, without relying on having detailed plans from which to reference. It encourages releasing as soon as possible, with a near continuous release cycle. The philosophy encourages the 'quit fast' mentality, which aims to speed up the development process by allowing developers to utilise many different solutions until one that works is found. As a solo project, there are still implications for applying the AGILE philosophy to my work ethic, including valuing the interactions with the product owner (project supervisors) over processes and tools, to provide a constant stream of working software over documentation, and being highly responsive to any changes that will be required to implement during the development of the project.

The application and code structure is currently in its fourth prototype. Through agile development and its constant development, the website now has a vastly improved structure and organisation. This provides a fuller featured and more professionally developed application. The main advantage of developing in this manner is the reduction of the negative effects brought about by low cohesion and high coupling.

Cohesion and Coupling

Two concepts which were born through the idea of OOP; coupling is a measure of how reliant classes are on each other and cohesion is a measure of how specific the methods of a class are

to that class. High coupling are methods and classes that 'depend' very highly on each other. Low cohesion are classes that have elements that are not really related to each other. It is more desirable to have code with low coupling, and high cohesion, because this means, that classes are very specific to their tasks, and the classes have no added dependencies on each other. An attempt to implement an application with high cohesion and low coupling, reduces the complexity of a website and increases the re-usability of a piece of code. This gives a quicker development time and allows for more readable code thus more extensible/modular program without little to no significant impact on other pieces of the functionality.

Object-oriented programming

The root component of modular and extensible programming lies in the concept of object-oriented programming. Object-oriented server programming boasts a significant improvement over those applications that do not use this approach without. Sophisticated web-pages can now be constructed that perform a wide array of tasks, and are incredibly well structured. These new types of web-applications place an emphasis on continuous development, in a way that mirrors the design structure of their non web-oriented application brothers. OOP development goes hand in hand with software design patterns, which are essential to employ in such a large scale project such as this.

Design-Patterns

Design patterns are patterns for designing code that feature solutions to commonly repeated tasks in the area of software engineering. The most famous work on design patterns is Design Patterns: Elements of Reusable Object-Oriented Software [4]. They provide very general solutions to the problems that are faced by many software engineers. The main design pattern used in this project is the MVC or Model-View-Controller, of which I used an augmented version. The use of mvc provides a coherent manner for developing, encouraging a consistent and efficient development style.

The idea of a View is to be used for displaying content, and a Model for handling all of the logic behind the application, and a controller that manages the interactions between them all. Modern applications, use either straight mvc, or a modified version of this. It is important for reusing components, reducing the complexity of code, in order to provide low coupling between classe, and ensuring that classes have a high cohesion. For web-applications there exist many frameworks that have predefined structures, that can handle a lot of the structure 'out-of-the-box', and they can be very useful for an experienced developer who wishes to jump straight into the development of websites, expecting quick turn-around to achieve high-level results. However I took a more inquisitive approach to the development of the application, and decided to develop a light-weight version of a framework. This was definitely a lot more involved than using a predefined framework. This approach helped me to understand the routing mechanism involved with the application, and gain a clear understanding of the mvc structure that was being applied to the website. This approach gave a clear understanding of the underlying structure behind the website, and ultimately granted me more control in the structure and flow of the application. I will go into a further detail as to how the file structure, and the mvc segment of the code works in the next chapter.

Coding Style

The coding style of a development process is considered an assessment on the human element of coding. It includes developing a style of coding and sticking to it. Using a consistent style makes it much easier for debugging. It also makes it easier to understand for any person who may adopt the code in the future. Writing in an object-oriented language, using design patterns, and writing in an environment that uses TDD encourages a discipline in the order of writing code. However other things can also be managed to provide a strict organisation for things like,

- white-space allocation
- class/field naming conventions
- checkstyle

2.2 Dealing with program faults

2.2.1 Tools for code analysis

There are ways to check the quality of the code you write, which are embedded in your development environment of choice. Using these tools provides a technique whereby the 'style' of the code is assessed. In the case of a linter, the code is dynamically (as typed)checked to determine if it is syntactically and grammatically correct. This improves the development time by spotting syntax errors in the code before running. Fortunately there exists a plugin for Sublime, which was my text-editor throughout the development. A check-style goes one step further than linter, and checks the styling of a piece of code. It is an evolution of assessing the consistency of a project, and organisations often employ a policy for their software engineers. They ensure that everyone adheres to coding principles through policing the check-style meaning that everybody adheres to the same set of coding standards. However not all problems can be derived from syntax errors.

During the development of a piece of software, it is somewhat inevitable that something can go wrong. There are certain ways to deal with this. One way is through the analysis of code smells. They are common things that can go wrong with code. There is extensive documentation on what to look for, and how to avoid them. In a nutshell: "a code smell is a surface indication that usually corresponds to a deeper problem in the system".[5] Fortunately there are ways to look out for these code smells, in order to avoid them from appearing and causing inherent problems within the code base. Check-style is one method of looking for recurring problems within a code base. Finding significant code smells can lead to code re-factoring.

An important aspect of development is finding and removing software bugs. Bugs are software breaking elements of a program, where segments of a code are not doing exactly what they are designed to do. Or they can be where the use of a feature brings about unforeseen side effects. Finding and removing bugs is unique skill that involves practice and patience. Thankfully, there are tools that can be used to help locate a program breaking bug, and use it to diagnose the problem and fix the bug. The method I used to debug for this project was XDebug for PHP, which provides more meaningful messages when PHP throws an error, this helps to diagnose the problems that can occur during development. For the java-script, I used console messages, in order to verify certain aspects about the state of the execution, in order to achieve the desired output without introducing any bugs. My focus was based on aversion of

bugs through careful development, but with appropriate knowledge of the skills required to diagnose if necessary.

There are sophisticated tools, such as the PHPStorm IDE out there that integrate breakpoints and a console emulator, to present the 'state' of a execution and the value of the variables, in order to help diagnose a bug. These types of tools would be a suitable environment, if I were to develop more with PHP in the future.

2.2.2 Making changes to existing code

Re-factoring is a process of fixing the problems that occur through a software project. It involves updating the structure and design of the code, in order to improve the development or running of a program, without affecting the front-end components of a program. The reasons for re-factoring could be for removing code smells, for reorganisation to improve the coupling/cohesion dynamic, in order to employ a specific design pattern, for optimisation of a code base or for fixing any bugs.

In order to sustain the legacy and re-usability of the code, it is useful to provide documentation. This provides a way to present information on what you have written, which can be used by other developers or yourself in order to understand how code works. The documentation provided for a piece of code can ensure that the developers involved in extending the code, know what a method or module is meant to achieve, and details about the variables it needs in order to work. It is said that with a strict naming convention most code can be self-documenting, but it is also important to add extra bits of documentation in order to improve the communication between one programmer to another. Fortunately there exists a utility for PHP called PHPdocumentation that can automatically generate documentation for PHP code.

2.3 Information preservation

2.3.1 How data is stored

Due to the nature of the website, it is important to have a system of data storage that has secure storage mechanisms that also allows for the efficient retrieval and update of that information. The features that will need storage are :

- user-information for logging in and password verification.
- favour data, for storing information about each individual favour, and allowing for users to search and view favours

Using an object-relation model database, the above features can be implemented. Using a search query, it is possible to tailor the query results to be specific to exactly what is needed for that particular view-component.

2.3.2 Preserving the information

Using mySQLadmin, it is possible to export the entire contents of the database, including meta information about the database, to a different location. This portability makes it highly convenient for backing up the data or for transferring of servers.

The structure of the database is a crucial component for describing the relationships between stored objects. It can provide a dynamic dataset, that can present many different components of information specific to the whatever application feature that may require it. Since the website is a networked application, that uses a live connection in order to access information, it is essential to protect against database attacks that could occur, such as SQL injections. These are queries that have extraneous components, designed to perform operations beyond the scope of the intended purpose. More information relating to specifically how I have protected against SQL injections are in a later chapter.

Another important aspect of preserving information is related to the encryption of passwords. It is important to ensure that the user's password, is secure, because without secure passwords, it is possible that if an attacker got access to a user's account through unencrypted passwords, it is possible they could use those passwords to violate their accounts on this website, or that password could be used n other accounts if they use the same account details. Thus using encrypting along with a salted encryption mechanism, ensures that there is a protection against these forms of attacks on the confidentiality of the information.

2.4 Ensuring Security and privacy of Data

Security on the internet is a very important aspect to developing a web application, due to the nature of the open-networked distribution of the internet there are many opportunities for loop-holes to be manipulated in order to take advantage of the insecure websites. I will talk about some of the important aspects to protect against in the website I have developed, and in a later chapter I will discuss the specific methods that were employed in order to combat these aspects.

2.4.1 Types of attacks on a website

Denial Of Service

Denial of Service (DoS) is an attack on a website, which can originate from one or multiple sources, where requests for information flood the server buffers, causing the site to experience slow to zero download speed. Its main side effect is to prevent legitimate users from using the website.

Spoofing

Spoofing is the use of sending requests to a service, which is not the original source location. There are many uses of this, which can be performed in many ways, but it's most common use is in initiating a DoS attack. Generally it involves manipulating an IP vulnerability, for example when there is no authentication of source in IP headers. This can be used by Users who can send multiple requests at a time, changing the source location each time, to make it appear like the requests are form different locations. This can have massive implications on

the running operation of a website.

Other variations on IP-spoofing attacks are:

- man in the middle attack: used in communications between multiple users. The attacker gets in transit packages, and alters the packages to try and get destination to reveal certain information, such as server config files.
- blind spoofing: the attacker analyses the package sequences, and then inserts his own in place of a certain package.
- non-blind spoofing: the attacker sits on the same subnet as the network, sniffing the packets. And so doesn't need to analyse the package sequence in order to determine the correct packages to remove and replace.

These attacks have general techniques that can be implemented in order to prevent against:

- use an key-based authentication method as means of communication between two processes
- use an access-control list to deny private IP address from accessing a downstream interface.
- apply a filter on inbound and outbound traffic
- apply a configuration to reject any packets that originate from outside the local network that claims to have originated from inside.
- enable encryption sessions so only trusted hosts outside the network are able to securely communicate with the your hosts.

The above techniques are high-level techniques that are used to configure a server in order to tighten control on the traffic to a website, or a network. However the most commonly used way of preventing the Spoofing attacks, are by using a https server. This provides a layer of encryption between host and server, which means that an entity who is listening on the transmission path cannot decipher the content between the two machines. A further method of security for clients to determine the safety of a server application is through secure-socket layer certification. This allows a client to determine the safety of a website, using a responsible third-party to issue certificates that state that the certificate is valid, and that the user is who they say they are.

Full path disclosure

Using the address bar, it is possible for the user to manipulate the URL of a web-server and view source-files of the website. This could be undesirable if the code contains private information(such as database access credentials) or private code (if you wish to hide the structure of the program). Alternatively the knowledgeable attacker can gain full-path disclosure, which gives them knowledge of the file structure this can then lead to identifying other vulnerabilities within the file structure. Alternatively they, can use the knowledge of the file system to access certain files and/or obtain sensitive data, such as getting the phpconfig file and manipulating or using the information to exploit other vulnerabilities. This is protected against by manipulating the http access protocols that manage the transferring of files over the internet. More specifically, the .htaccess files.

Remote-file inclusion

Another vulnerability with PHP is using remote file inclusion.[6] This is the user injecting a script by using include/require statements to perform operations beyond the desired scope of the script. This can be used to perform get or post requests which can be used to manipulate or obtain sensitive data. It occurs in instances of user input or URL manipulation. It is prevented by adding adequate validation to the URL or form input to ensure no extra information is sent to the scripts on the server.

Local file inclusion

Related to remote file inclusion, is local file inclusion[7] where the vulnerabilities exploited are those that exist in files that currently exist on the server.

The methods available for providing security against these attacks are not exhaustive, they do not guarantee that the server code is safe. However they provide a means to ensuring the file system is as secure as possible. The methods I employed are:

- configuring the .htacess file
- sanitize URL before parsing
- configuring 'php.ini' file, open_basedir option to set the highest level directory accessible from the server
- configuring 'php.ini' file, allow_url_include option to prohibit representing a page inside the host page

If this website was to be deployed and hosted on a live server, it would also be a priority to provide an encryption mechanism using a https protocol, to provide a guarantee that there is no network-sniffing occuring.

Code Injection

Related to file inclusion, code injection is remotely including lines of code, in order to execute certain functions in order to access information or perform certain operations.[8]

2.5 Conclusion

I have discussed the considerations that have been the driving force behind this project, including

- Software principles used in the project
- dealing with errors in the code
- maintaining preservation of information
- upholding strong security in the website.

In the next section I will go into detail as to how these theoretical aspects have been formulated into creating a working website application.

Chapter 3: Practical Aspects

This chapter will look at the practical considerations taken for the development of the website. It will also include an extension of the topics discussed in the theoretical section, including how specific elements were applied to provide a functional application that provides a secure environment while facilitating all of the use cases.

The following is a list of technologies or achievements that I implemented throughout the project:

- PHP object-oriented approach including mvc framework
- PHPMailer(plugin for using SMTP server)
- implementing Email Verification process
- Mysql, and php to Mysql transactions
- applying encryption to traffic with ssl and self-certification
- Ajax Communications
- jQuery
- javascript
- bootstrap
- porting site to live server

During this section I will describe the implementation for the above listed concepts.

3.1 Developing applications for the internet

There are many languages that can be used for designing web applications. Some are specifically designed for a particular reason, others have been re-purposed to perform some task. In designing the website, it was important to decide on the right tools to use. The choice of tools would define *how* the project could be designed and exactly *what* could be achieved with the given combination of languages.

3.1.1 The Right Tool for the Job

The website requires to have a certain amount of data that is stored and retrieved, for example when a user logs in to the website or when making certain searches for a piece of information. The technology used to support this should be well documented, and well supported, with the features attributed to it to enable the guarantee of security for any sensitive data that may be handled. The technologies should also provide adequate libraries to enable the conducive operation in the day to day working of the system.

3.1.2 PHP

PHP is a widely used server-scripting language that has been in use since 1994. It is used in 82% [26] of the websites that are currently on the internet. It provides libraries for every feature that will be required by the website. Which include:

- well documented
- plugin support
- encryption algorithms
- database connectivity
- 'user-session' support

No external libraries are required to run these packages since they come pre-installed, so they are easy to use. It has an online documentation which provides a complete API.[22] It is provided on a open-source licence, and subsequently free to use.

The possible alternatives are:

- node.js - a server language written in javascript
- asp.net - a popular language using the Microsoft .NET framework
- ruby-on-rails - a web framework based on the language Ruby[23]
- django - an entire framework written in Python.[24]

Plug-ins, or external libraries have an incredible ease of use. There are a large collection of them, due to the length of time that PHP has existed, and how easy it is to develop for. As such, for any additional features that I want to implement, it is highly probable there will be a set of code that exists that can perform that function. An example of this is the mysqli framework, that extends php capability to provide mysql database connectivity, or the PHPmailer project.

3.1.3 Server Language Conclusion

PHP is a very popular language, with a lot of online support and published library extensions. The benefit of using PHP is that it is quite simple to pick-up, and it also provides all of the features that are necessary in order to develop this application.

3.2 Developing the back-end architecture

This section will include details of the directory structure, and how each of the components will interact in order to create an efficient application. It will define how a framework has been developed in order to provide a suitable structure that enables modular development.

3.2.1 Choosing a Framework

Using frameworks designed for the web has many benefits:

- provides an efficient implementation that reduces programming time
- encourages a faster turnaround for developing applications
- allows smaller developments to reap benefits of sophisticated frameworks, thus saving costs

Some available php frameworks are:

- Cake PHP
- Zend
- Laravel

Each of these frameworks provide different advantages, with each suited to serve different tasks.

3.2.2 Choosing To Hard-code a framework

Alternatively, it is possible to hard-code a framework myself. The advantages of hard-coding a framework are:

- improved understanding of underlying code
- can configure framework to suit needs
- application can be developed in a manner that is suitable to the developer

It was the benefits of hard-coding a framework, that influenced the decision to write my own php framework. The application framework, uses an augmented mvc for defining the code structure, and a routing mechanism for implementing the mvc components. Writing the framework in this way has provided a clear understanding of how a php mvc structure can be applied, and the mechanism behind it. Writing it in this way, has given me the knowledge to comfortably apply any framework I should chose in the future, and be able to adapt it to the needs of the specification.

3.2.3 Augmenting The MVC

The architectural design pattern MVC is a fundamental aspect for web development, as it allows to separate the user view from the application logic, to improve the performance of the website, and to prevent the user from ever having to deal with the operations that make the application work. It creates a basis of a framework that allows the application to have a controller that interchanges easily between the views and models.

This web application has a single view, so it is important to have a robust mechanism for loading new view components. Moving from the home screen to the login or the profile-edit screen are examples of the website using routing to different models, and thus loading different view components. The use of the model and controller allows for simple reuse of modular code, and it also provides an abstraction that provides an efficient structure to base the development of the application. An example of the modular code in the website is the database connection and result retrieval.

3.2.4 Developing the Structure

The current iteration of the framework features a directory structure with:

- 'includes' for libs or images
- 'public' for any non-private source files
- 'app' folder which houses the main code for running the application inside an mvc and the routing algorithm

The application makes use of the object-oriented features of php to enable the construction of a rudimentary mvc routing application. The framework was created in order to define a logical structure for the files, while the modular nature provides an extensibility in support for the future development of the project.

The application setup works by principally using 4 technologies, html5, jQuery/js, ajax and php. The general behaviour of the application works as follows:

- The user will navigate the application using a combination of hyper-links and buttons
- The use of a hyperlink posts an URL into the address bar, which is then parsed by a core-app controller
- This core controller then loads a model-specific controller, a model and any variables that are passed in
- The application uses a single view, and depending on the routed URL which would load a specific model,
- The model, when initialised collects html data from an included PHP file, and this is echoed into the controller, which then loads it into the view for the user
- While the model is collecting the data it will query any information passing that along with the view data, or perform any other operations that it needs to do.

The components for the view are split into modules, which promotes re-usability for the components. The separating of the html, provides a cleaner interface, and makes editing or extending different parts a much simpler task due to the methods being distinctly separate from each other.

3.2.5 Implementing The Structure

The image below describes the structure of the classes defined in the structure. It represents and augmented version of mvc, along with a core app initialisation and routing algorithm.

What follows is a description of the classes and elements in the code that support these actions and provide the back-end support for the functionality.

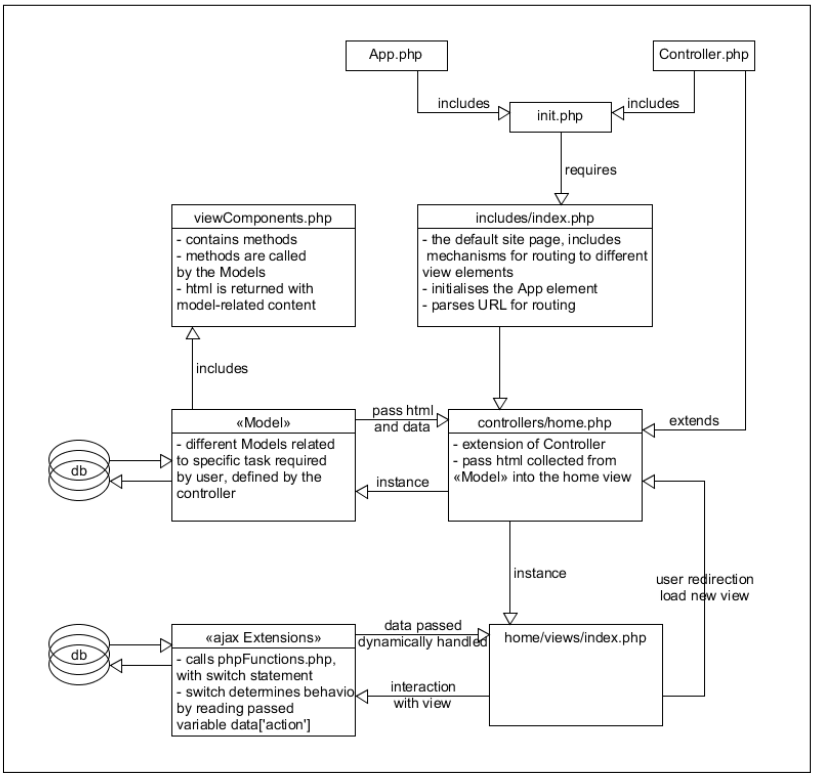


Figure 3.1: Directory Structure.

Details of the directory structure

The table 3.1 describes the behaviour associated with each file, and how it relates to other files:

The .htaccess contents to allow URL rewriting

```
Options -MultiViews //allows multiple views to be created
RewriteEngine On //allows urls to be defined in the address bar

RewriteBase /LETS/mvc/public/
//defines the base from which urls are constructed,
//which allows us too define the index.php file we
//want to specify later with the rewrite rule

RewriteCond %{REQUEST_FILENAME} !-d //checks the directories
RewriteCond %{REQUEST_FILENAME} !-f //checks the files

ReWriteRule ^(.+)$ index.php?url=$1 [QSA,L]
//matches everything, passes it to the index.php,
//with a query string url= and then the param $1,
// this defines anything after the rewrite base to
// be defined as the variable url, which can be
//obtained by accessing $_GET['url']
```

Directory/File	Description
public/index.php	Has an includes statement for app/init.php and initializes the new app object that is defined in app/init.php.
public/.htaccess	rewrites the url, and allows it to pass it through a GET variabl
includes/sqlconnect.php	Contains the information and methods required to make a connection to the db, and perform any database interactions
app/	contains all of the application specific content, including the controller, the view and the model
app/init.php	Includes the core/App.php and core/Controller.php, which is used by the public/init.php to initialize the application
app/controllers/	Includes the controller used for initialising the models for the home view. The controller extends app/Controller.php to use the construct methods and pass in the required class to be constructed depending on the url that is to be routed. The controller will initialize the model, and call the related model-methods in order to intialize a view, that is related to that model
includes/viewComponents.php	Contains all of the html elements split into separate methods. The model calls the individual methods when constructing a specific view, and this file echoes them back to the controller, that passes it into the view
controllers/home.php	The primary controller at the application level, it takes the parsed URL, and loads the model/method required, passing any variables and initializing any components required for the view. It contains a default method defined as index
app/core/App.php	Allows the application to route to other views. The class parses the string and initializes the parsed controller, then reroutes to a different controller and method depending on the URL specified.
app/Controller.php	A parent class, that provides methods for initialising the model and the view
app/models	Contains all the specific models that are created that perform the business logic for the view associated with it
app/views/	Contains all of the directories that contain the views and content that will be displayed on the application
includes/script.js	defines the behaviour for the view (the current page). Includes handling the ajax call, including the variables and html element ids associated with the call. On submitting the call, it sets the data['action'] value to the required phpFunction to be called on the server. It is also responsible for handling the response from the phpFunctions.php, and deals with the returned data to dynamically alter the page, or performing any other view-specific behaviour
includes/phpFunctions.php	includes all of the Functions that are called from an ajax requests from the client side. It includes a switch statement, that reads the argument data['action'] value. Performs its necessary behaviour, then sends response to the ajax segment that called it

Table 3.1: Directory Table[12]

Specific methods are called from the app/controller by using:

```
call_user_func_array([$controller,$method],$params);
```

3.2.6 Frameworks Conclusion

The underlying knowledge I now have with regards to developing frameworks means that I can customize such a framework to suit my needs. It also means that if a project requires the selection of a framework, I am now in a better position to understand the differences, and thus choose an option that best suits my requirements. Using a predefined framework would allow me to deploy solutions faster, using a clearly defined structure, and an effective development mechanism.

The website benefits from the modular framework structure it now has applied. It enables a more structured and faster way to develop the application.

Frameworks are similar to design patterns, in that they have been designed in order to improve the process of development, but also the quality of the design of a website.

Using my augmented-mvc, I have gained a thorough understanding of the process involved with developing a structured mvc with a routing mechanism, and effective file organisation. It is this benefit that drove me not to use a pre-defined framework. The advantage of this is that, in future projects, I will have a stronger understanding of the underlying principles that drive the frameworks, and be in a better position to be able to compare and understand the differences between, and find the most appropriate framework for my chosen project.

3.3 Security In Social Networks

Social networks demand a level of input from the user that often requires them to submit sensitive data. This data is used to distinguish each user, but it can also be used to protect the information that the user has submitted. It is important to use the most modern encryption algorithms, to ensure that the user's information is protected. I will discuss the encryption method employed, including the use of a salt.

When asking outside users to use a website, there is a risk of malicious users attempting to harm or damage the website. As a result there are certain measures that can be taken that prevent or stop these actions occuring. There are measures that can be applied to prevent against SQL injections, which include the use of prepared statements, that strip the data of any potentially harmful characters and by using validation, which has these benefits:

- ensures the user has entered data that is correct
- can give a guarantee that the user hasn't entered anything that can harm or steal the data structures

3.3.1 MySQL

A large part of protecting a user's data is to carefully encrypt their data. The encryption is useless without a robust database storage mechanism.

For the database language MySQL will be used. MySQL was chosen because there exists a complete MySQL reference on their website[20], which is useful when developing the database and developing more sophisticated search queries. MySQL is fast for performing read operations, and also features built-in encryption methods[21]. These two features are important especially for this website, due to required encryption needed and the speed of service a user demands from a public website and also due to the feature of displaying relevant data from the database and for encrypting passwords in the db.

It provides the following describe the features considered when deciding which SQL-technology to use. All of them need to be supported by the chosen technology, with adequate methods providing functionality:

- well documented
- connectable through chosen server language
- light-weight

Altogether, it is the most suitable option for developing the application due to the requirements of the website. However whenever storing data in a remote location, using SQL language and user-inputed queries, there are a set of security principles that must be implemented in order to ensure protecting a user's privacy.

3.3.2 Protecting Privacy

For this application these are the main aspects involved with protecting the user's and their data:

- encryption
- salt
- prepared statements
- .htaccess
- email verification

3.3.3 Encryption

- providing an effective encryption algorithm for the user password, to ensure no person can guess the password, or use the stored version of the passwords to compromise the security of individual users accounts

The use of encryption is to prevent an attacker with access to the database from being able to use the encrypted passwords to compromise the users' accounts. There are two ways that encryption is used in the website. The first is for encrypting the passwords in the initial instance, and the other is to provide a secure mechanism for determining if the user has entered the correct password.[15]

The application uses a one way encryption algorithm. Which takes a user input as a password and outputs a unique hashed output.

When password checking, the application receives user input of the password, and checks to see if the entered password applied to the encryption algorithm is the same as the encrypted password stored in the db. This is done every time that the user wants to login, with access granted only if the correct password is entered. In this way, the encrypted password is unintelligible to anybody who would want to access a user's account. Only the user with the knowledge of the password may log in.

The application uses an algorithm that is provided by php, which is a default algorithm defined by the php developers. When a more sophisticated algorithm is made public, php will develop a release that uses this new algorithm. The current default algorithm is the blowfish algorithm, that is the the currently the must secure algorithm for one-way encryption.

The name of the method that is used to provide the encryption is:

```
password_hash($password, PASSWORD_DEFAULT, $options);
```

[16]

This method takes the password to be encrypted, the option of the password to be used, and an array of options. For this application, the array contains a salt, and a 'cost' of the algorithm, or the amount of iterations it runs as part of the encryption process.

The combination of the password hashing, and the verification procedure, provides a one-way system, that is unbreakable if the user should gain access to the db passwords, and can only provide access to the verified users. passwords

3.3.4 Salt

- provide a salt for the periodic application to the algorithm, to keep it safe and protected against hackers.

The use of a salt, is to provide an additional layer of protection to the encryption mechanism used for passwords in an application. It is a layer of the encryption algorithm that requires a defined value to be set, and as such the output of the algorithm will be different for the same password when using different salts.[18]

Using salt is a way to change the encryption algorithm. In order to prevent any users that have determined a table of hash-mappings from input to hashed output from using that data, the salt effectively makes those mappings(known as a rainbow table) useless.

Changing a salt value for new passwords and the periodic reapplication of a salt to all existing passwords, can nullify the work that a mapping achieves. Thus helping to ensure the continuing security of the passwords and the user's data.

3.3.5 Prepared Statements

An SQL injection is where SQL code is typed into a form, processed by the server and submitted to the database. Any website that uses a form has to protect against SQL injections, where undesirable code can reach the database, and cause things to happen that are not good for the integrity of the database or the privacy of users.

The danger of SQL injections means that users can use forms to enter unvalidated data. These forms when processed make requests of a database that does not result in the desired outcome. With no validation of the entry, it is possible for the user to write something which contains an additional command for the db to perform some action

For example, if a user is logging in to a website, a typical form entry is username and password. The recovered value for the from entry could be:

```
$name = $_POST['name'];
```

The running of the command:

```
echo $name = James; Drop all tables;
```

would print "James; Drop all tables;".

The use of drop all tables, is a very dangerous function that basically deletes the entire contents of a db. However there exists more manipulative queries which can be used that causes the user to gain access to sensitive data stored in the db.

If an attacker can guess the table structure, they can mount an injection such as:

```
$name = James; Select bank_details from user;
```

Here the result returned is every bank_details stored for each user. This is harmful if the user can manipulate the application code to then display this data. Thus a solution is to employ the strategy of prepared statements, that provide a statement stripping function. There are 3 processes. A template is defined for each form which is pre-prepared statement for the query that will be applied.

```
$stmt=$connection->prepareStatement($statement);
$stmt->bind_params($params,$params...)
$stmt->execute();
```

```
prepareStatement($statement);
```

Works by using a statement that is predefined by the developer. The method will return a form that is stripped of all escape characters and is safely prepared for use in making queries to the db. The form that statements should be is:

SQL Command and table(?,?...)

an example is:

```
$statement="INSERT INTO user (firstname,familyname) VALUES (?,?)"
```

Here the statement consists of two parts:

- The statement and the operation to be completed.
- The unknown values to be entered.

The statement, adequately stripped of escape chars and prepared for binding with the params, is then returned to the $stmt value. The number of ?'s depends on the number of params used in the query.

```
bind_params("ss",$params,$params)
```

Here the method has two parts:

- The first argument is a string that indicates to the method the type of paramater that is being passed in. Here we have two string types, so we use "ss" as the value passed in.
- The consequent arguments are the values that correspond to the ? values in the prepared statement.

```
$stmt->execute();
```

When the statement is prepared, and the params are bound without error, the statement can be executed.preparedStatements

3.3.6 .htaccess

This is used for ensuring that the module for accessing the db is protected, with unreachable file in the directory and using a secure password.

Users navigate a website by changing the structure of the URL typed into the browser. A user can navigate the file structure by directly manipulating the sequence of the directories.[12]

An example (from my appplication) is:

- http://localhost/LETS/mvc/ - this is the root of the directory structure for the application
- http://localhost/LETS/mvc/public - takes the user to the public interface of the application, which is defaulted as the home
- http://localhost/LETS/mvc/app - attempting to access the app/ directory results in an error "access forbidden".

The structure of the application requires a file to contain crucial information for accessing the db. This information in the wrong hands can have devastating effects on the privacy of the users of the website. Someone with access to this information can log in to the db, and perform any actions that they require form the db. These actions include dropping tables, selecting or changing data entries.

Thus the use of the .htaccess file with the line "Options -Indexes" is applied to every file in the same directory as itself. This option prevents users from remotely accessing the files stored in that database. [13][14]

There were some changes that were needed to be made to apache in order to apply .htaccess to the server:

run command:

```
$ a2enmod rewrite;
$ service apache2 restart;
```

change apache2.conf

```
<Directory /var/www/>
...
AllowOverride All
...
</Directory>
```

and finally,

```
$ service apache2 restart;
```

3.3.7 Email Verification

An important aspect to preventing identity forging, of users creating user accounts for emails they do not possess, is to include a Email Verification process. This is implemented by using the library of the open-source php code called PHPMailer. PHPMailer provides a message function, for sending messages over the SMTP protocol.(Appendix D.1 for snippet)

Being able to use this code requires, that ssl encryption is enabled. In order to enable ssl, a user must get a signature from a certification authority. However I implemented a self-certification signature, which allows for PHPMailer to work while providing an encryption on the data-stream.

Once a user has 'created' an account, they cannot login to their account without first verifying their account. On creation, a hash is stored under their email_address. With a verified signature, and a verified connection to the smtp server, the mailer program sends a message with the following information:

- Username
- temporary password for logging in
- link to verifypassword page, with unique hash and email address as an argument in the URL

Upon clicking the link, the php will check the hash against the email_address in the db. if true, it will allow them to login, which they must use their temporary password, sent via email.

3.3.8 SSL and Self-Certification

The concept of applying ssl to a website allows traffic between a server and client to be encrypted. This protects traffic from any entity that may be monitoring the traffic. The mechanism works by using two-way encryption. Both entities release a public key, and the data to be sent is encrypted with the recipients public key. On receiving the data, the recipient can decrypt the data using the key that was used to encrypt the data. Using certification authorities, entities can have their identity authorised by an independent third-party, which adds an additional layer of trust that the content received is from who the sender says they are.

The website uses a version of self-certification, which provides an encryption layer, however does not have it's identity verified by a certification authority. If the website was to be deployed in a professional context, gaining a verified certificate would be a top priority.

The ssl is implemented by using a series of commands to install openssl layer to the server, and making some amendments to the apache conf files. I have detailed the sequence I followed to implement the ssl layer in appendix F .

3.3.9 Dealing With Security

As a developer of a website that uses personal information, it is crucially important to protect privacy of the individual user by ensuring that any sensitive information is kept private.

The use of salt combined with encryption, provides a secure mechanism by which the information can be protected in the database. This must use the most up-to date encryption methods, otherwise it is possible for the knowledgeable attacker to be able to mount sophisticated attacks that exploit the weaknesses of outdated encryption algorithms, such as MD5.

Also ensuring that the .htaccess files are set to private, ensures that only people with access to the server can view the information held within the files, this is especially important for files that hold database access information or any other sensitive data.

Finally, the use of prepared statements is currently the most efficient method of preventing SQL injections.

3.3.10 Privacy and Security Conclusion

The process of developing a web-application has many important considerations for ensuring the safety and security of its users. There does not exist one robust system that can guarantee complete security. This is due to the open-networked nature of the web, since there are just too many loopholes that can be exploited and too many users with dubious intentions, who try to break, or steal information. As a developer for the web and while the application maintains active development, I must ensure that all applications are encoded with the most modern safety principles in mind.

With all of the methods I have mentioned combined to form a strategy that protects the user of the website and to ensure that it can maintain its feasibility as a project. However, it is imperative for the designer of any website to employ the most modern security and encryption methods as they are developed. Only by using the most modern methods can the application be deemed secure.

3.4 The Front-end details

The front-end of an application has two aspects for consideration, the User-interface and the event handling and providing a dynamic user-experience. For the event-handling, I have used a combination of jQuery and ajax calls, and for the User-Interface I have used Bootstrap, which is a framework which provides css style-implementation.

3.4.1 jQuery and Ajax

An important aspect to the program is the use of javascript, jQuery and ajax, to provide dynamic behaviour to the website. Javascript is the language used here, which utilizes the collection of libraries from jQuery, to perform actions for the user. jQuery provides a convenient method for making an ajax request.

The following code includes a function that is automatically called as soon as it is loaded. It makes an ajax post that uses the arguments:

request to the location :

```
http://46.101.34.183/mvc/mvc/includes/phpFunctions.php
```

Using the array:

```
{ 'action' : 'checkNewMessages' }
```

And an anonymous function with an argument that is called on successful return from the php file:

```
function(data){
..
}
```

Here is the entire snippet in action:

```
$(function(){
$.post('http://46.101.34.183/mvc/mvc/includes/phpFunctions.php',
    { 'action' : 'checkNewMessages' }, function(data){
var unread = data['unread'];
if(unread==true){
$('#viewMessagesNav').text("Messages (New)");
}
});
```

The purpose of the above ajax snippet is to call phpFunctions.php, using action variable set to checkNewMessages. The function in php will look for new messages, and return an array containing the unread variable set to true, it is set to false otherwise. If data['unread'] is set to true, then the callback function will dynamically update the html element with id #viewMessagesNav to have the text "Messages (New)".

Ajax is used to make a get or post request to the server(Appendix D.2 for snippet).

By using jQuery, ajax calls can be easily implemented, and the combination of the two allows for dynamic updating of the user-interface.

3.4.2 Bootstrap

Bootstrap provides an implementation of CSS libraries that can be added to the html tags, in order to provide a fluid-interface that is supported across devices. It provides a set of attributes for easily adding style to document elements. The majority of the styling is implemented with Bootstrap styles.

This image is a representation of the website login screen using Bootstrap:

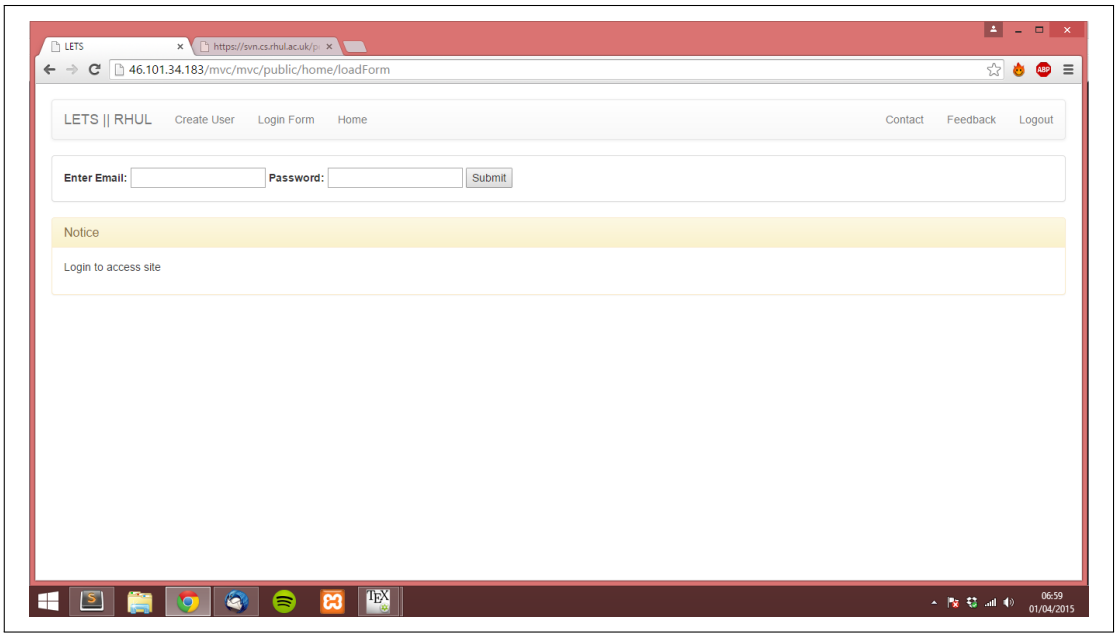


Figure 3.2: Website with Bootstrap.

Using the Bootstrap implementation it is possible to create clean looking applications that still perform their intended function.

3.4.3 Front-end Conclusion

By using bootstrap the website can have style easily applied to it that is highly adaptable to a user's specifications. Combining an intuitive style interface with dynamic events and user interactions enables the website to perform all of the required tasks, while appealing to the target audience in it's presentation and ease of use.

Chapter 4: Critical Analysis

4.1 Project Achievements

Through the course of this project, I have made some major milestones in the understanding and development of the project itself. The main achievement was developing the framework in having a working application that provides an implementation of the specification. This can be developed by applying more styling to make it more appealing to the target audience (Students). The application of styling used Bootstrap and additional style-sheets to provide the graphical elements in the web-page.

The website uses modern security features:

- encryption, and salt
- prepared statements
- htaccess
- other configurations applied to php.ini file
- ssl layer

However it is impossible to guarantee that the website is truly secure. There are always new evolving threats, and it is important to remain up-to-date with the most modern threats in order to provide the most secure website possible.

PHP uses object orientation, this is most prominent in the SqlConnect.php class. It makes use of objects, to provide a modular and reusable approach to programming.

The front-end section of the code makes extensive use of jQuery to provide a dynamic user interaction. This implements the employing ajax and improved selection of html elements. jQuery provides a lot of helper classes to speed up development time and to make it easier to read, however there are other frameworks out there that are likely to be much faster in their execution. jQuery is useful, because it is so well supported, and it can do so much. The website also makes use of a jQuery plug-in to provide additional modern features. An example of this is the 'datePicker' widget. These plug-ins are good for providing a modern application, with nice visual features that apply to the desired user base.

4.2 Personal Experience

I learnt a variety of technologies that were new to me; php, jQuery, Ajax and Bootstrap. Through using these technologies, I developed a light-weight framework, that utilised a routing application and implemented an mvc structure. This allowed me to progress from writing a website to developing a web-application. This type of application has many applicable features for modern web-sites, so in understanding the underlying principles has given me a good knowledge to apply in industry. Also in learning these languages, I also have developed a more detailed understanding of their nuances, their advantages/disadvantages, so this puts me in a better position when deciding which language to use for any future projects.

The process of finding out how to implement all of these features allowed me to find out how to learn new things. This means that when it comes to my next project, I will have the skills in order to be able to research the best means of completing such a project, and delivering a product which conforms to the required standards and specification.

The value of writing a report is in deciding how best to present the information that has been learnt over the course of a project. I have put a lot of thought into this presentation, and it has shown me some of the best ways to present that information across, in order to inform others of what you have learnt.

I have also learnt and applied important safety features that are mandatory among modern networked applications. It has given me a technical understanding of these concepts, and while they are by no means exhaustive, it has given me a good basis from which to start a life-long learning in providing the most upto date security features to all possible software.

I now understand the importance of creating a plan, and seeing the 'bigger' picture, when creating that plan to consider different approaches to developing project, and to weigh up the differences between all of the different approaches.

4.3 Enhancements

4.3.1 Developing The Project

Form Feedback

Through analysis of the form feedback provided for users (included as Appendix F), there are some trends that become clear about the website. Most of the criticism is aimed at the design and layout of the website. Comments like, "It's not clear how to use the website from the start", tell me that the website is not very intuitive. To improve this impression, I would add a small tutorial page, that includes annotated images to act as a tour guide and describe the features, and how they work. Another comment of, "Upon completion the home page should have some kind of design or picture, at the moment it looks slightly bare and incomplete." confirms that I have a lot to learn with respect to design principles for websites. In order to improve this I would further research into popular and modern websites, and look at how they are structured and try and emulate the parts that give them good presentation and reflect that in the website.

Style

To improve the style of the application, I would use the Bootstrap classes to optimise the views to be enable a fluid interface that can be resized to fit any screen-size. This would be especially beneficial for any mobile users of the website.

In order to make the site more appealing to users, and students in particular, I would use research into popular websites but also work with the feedback given by users to develop an intuitive and engaging website. I would put research into the use of Bootstrap in making websites portable to different view-port sizes and appealing to wider target audiences. In particular, I would research additional frameworks such as Angular JS, which provide additional CSS components with javascript support, to create easily adaptable applications with support for common tasks that are part of the html structure. A framework such as angular can provide a very effective way to develop modern looking applications that have a vastly

reduced development time.

Testing

Testing should be applied to the student body. An implementation of this would be to apply white-box testing, or a bug-report form for users to fill in and return. This would provide useful feedback in order to help in finding bugs and being able to apply continuous development in the form of patches to improve the code-base or develop features discovered necessary through testing. An additional method could be to write a segment of code that logs all put and get requests that are made to the server. This would be an effective way of monitoring traffic and analysing for any exceptions that get thrown.

Unit-testing

With a sophisticated application using OOP, it is a very common engineering technique to use Unit-testing. This is especially true with a modular program that faces continuous extensions. On developing a new feature, existing test suites can be run to ensure that new features adheres to existing standards. Unit testing is a principle that I would integrate if I was to develop any future project. Not only is it a sophisticated way of defining standards, but it is a way to prevent the most common of bugs while providing a formal process for finding and removing bugs.

TDD

Test-driven-development (TDD) is an extension of unit-testing. It involves creating the test and building the class, objects and methods around the test. It is a mechanism for developing robust applications with sophisticated and precise unit tests. It enables an framework that is easily adaptable for developing extensible features/patches.

Documentation

Providing documentation is an integral part for the body of any code. PHPdocumentor is a documenting plug-in that automatically documents PHP code. My code is somewhat self documenting in terms of the variables and class names, and comments inside the methods. But with the use of a documentation system, it can improve the legacy of code, and allow it to be easily understood by other developers, thus it can be adapted or extended as necessary.

Cohesion

The state of some of the classes are too big, they are have low cohesion. In order to improve their structure and cohesion, the methods and classes can be broken into smaller more specified classes, in order to make it more modular. This is especially true considering that some of the methods are basically performing the same task, but with different variables. The methods can be reduced into fewer more modular methods as a way to improve the efficiency of the code. This can be achieved with careful refactoring, to ensure that none of the classes stop performing their required tasks.

Language Considerations

Another consideration would be to reassess what is the best language for the application. Such an application has many options available to be used for the language. Each language available has a particular aspect that makes it different from other languages. PHP has been a reliable choice for this project, however with the knowledge I now have, I may have chosen differently. I could use the knowledge of frameworks, and choose a based on the requirements of the project. Additionally, with the knowledge I now have of php, and more specifically how frameworks are implemented, I would use a pre-defined framework in my future projects. Fully featured frameworks such a Ruby-on-Rails provide a lot of support for the tasks that take up a lot of a developer's time. Again it would depend on the specification, but it is something that would I would put a lot of thought into in order to choose the framework that is best suited for the task at hand.

LDAP

LDAP is light-weight directory access protocol, and it is responsible for managing and accessing a directory services database. With further research, the benefits of this system would be able to be applied to such an application. The feature allows for remote directory structures to be applied to networked applications, in order to take benefit of the features of the remote directory structure. In this case the file structure, would have been that of the RHUL database, and the user-name/password fields stored in their database would have been the same fields used to access/login to my own website. This would benefit this application by providing a means for using using the user-name/password defined by the college (RHUL), having the added benefit for the user of not having to 'sign-up' to the website, using their existing details. It would also mean that as an administrator, would not be responsible for managing user-accounts, in terms of password or account recovery.

Security Optimisations

Currently the website has some security considerations, however there are certain aspects that can be developed in order to create a more secure application. These are:

- Improve the validation applications for the forms at the server and client side of the application
- Obtain a certificate from a certification authority, which would allow users to trust in the security of the website
- improve the .htaccess files across the application to tighten the controls to the application server directories
- research and employ configurations for apache.conf, and php.ini in order to optimise the security, and the user-experience of the application

4.4 Conclusion

I have learnt a lot from this project. I had never developed a web-site before undertaking this project, so it was a big challenge to understand the concepts needed in order to develop a sophisticated web-application. The project has undergone many prototypes, and each stage has improved its concept and technical sophistication. With each development stage, I have

gained vastly more knowledge, and now have a stronger understanding in some of the best practices for developing modern web-applications. I found it very useful to use form-feedback, because this gave an almost immediate feedback on user-satisfaction of the website, this allowed me to make fixes to any bugs or broken features of the website almost immediately. This rolling improvement of the website allows for dynamic and engaging applications. It was also satisfying to engage with some of the users and learn their opinions about the final product. It is the constant progression with ever improving technologies that has inspired me to pursue a career in this area of web-development.

The driving concept behind the LETS is one of an altruistic nature, and it is this idea that encourages me to find and pursue active projects in the future, both to improve my craft using the most modern techniques and tools that are currently available, but also to try and improve the lives of others.

Chapter 5: Professional Aspects

5.1 Issues in IT

Professional Issues are areas of concern for any person working in a professional capacity. Issues are any concerns that may come up for a person working in this industry, either on a daily basis or as a long-term consideration when undertaking work in that professional capacity. The ideas behind them are meant to guide a person away from making choices that will negatively impact themselves, the general public, relevant authorities or the profession as a whole. On an individual level they are designed to maintain the competence and integrity of the professional worker. On an public level, they are designed to maintain the discriminatory rights, privacy and security of those who will be impacted, either directly or indirectly as a result of the professional's work. For the relevant authorities involved in the work, they are designed to guarantee that the correct flow of responsibility is upheld and parties are held accountable for any wrong-doing that may occur. And most importantly, to uphold the integrity of the profession as a discipline, to ensure its upheld opinion in the public and professional perspective, and to ensure the continued progression of the field.

A code of conduct is a set of principles which are issued by a professional body assigned the responsibility for ensuring that professional issues are maintained for its represented industry. In this case the field is IT, and the body is the BCS, the British Computing Society. The BCS provides: " access to a range of great services and tools to support you now and throughout your career " [1]. One of the aspects it uses to help maintain professional issues within the industry, is by providing a code of conduct for it's members (included in its entirety as appendix 7.4). The code is a tool designed to guide a person throughout their professional career, and which upon agreeing to follow, allows a person to become a member of the society. Society members who operate within this code of conduct are permitted to apply BCS accreditation to their name. This can be an important consideration for specific sub-industries, companies or employment agencies, who demand this accreditation as a prerequisite for hiring. Adhering to this code is crucial for maintaining respect in the field, and for ensuring a person is behaving in the most professional manner possible.

I will refer to the BCS code of conduct when analysing aspects of my project, but it is important to note that there are other respected professional bodies out there with similar codes, which expect their members to adhere to (eg. ACM, IEEE, AITP among many others). They all have different requirements but all adhere to very similar principles: maintaining a duty to yourself, to the general-public, to an authority and to the profession.

I will refer to the code of conduct by BCS as the basis for how I should conduct, using it as a reference for particular aspects of my project, including what I have done to adhere to it, and particularly WHY it is important in the wider context of my progression as IT professional.

5.2 Issues related to my project

5.2.1 Privacy and Security

Ensuring that my project adheres to Privacy and Security regulations has overlap with these ideas:

- carry out due care and diligence
- accept responsibility (morally, legally, professionally)
- protect confidential information
- uphold integrity of information.

Building a website such as the LETS, which requires users to post identifying information which could be personal in nature, has a requirement to protect the privacy of the individual user.

From the Code:

- 1.a) have due regard for public health, privacy, security and well-being of others and the environment.
- 2.f) avoid injuring others, their property, reputation, or employment by false or malicious or negligent action or inaction.

The implication here is that a a developer I have a social responsibility to protect the individual user. In this way, it is important to efficiently manage the data in a secure fashion. In this instance, when a user creates an account, they also have to create a password. Passwords are devices used to protect an account, so that only one user has access to that account, the account 'owner'. It is important to store this password using appropriate encryption tools to ensure it is stored in a safe manner. This has been followed by using modern techniques to encrypt the password and verify for account entry. This is to ensure that only the account owner's password remains private, and their account details are kept secure.

Other means that these codes can be violated in this project is with:

- data snooping - hackers tunnelling though an insecure http connection to spy on the users of the website
- sql injection - entering extraneous code into form fields to extract/harm the data stored in db
- malware injection - legit users with malware become conduits for malicious users to access sensitive data
- ddos - flooding the server with requests for data, to crash the system, this prevents legit users from accessing data.

These attacks are protected from by using the techniques which I will describe in the Practical Aspects section.

5.2.2 Respecting the legitimate rights of third parties

I have demonstrated ways that the user's integrity can be compromised, but as a software professional, I have a commitment to the legal process and to the authority that I represent(the department of Computer Science: Royal Holloway) and to any third parties who may be affected by my code (for example the technology providers and the licences they hold).

From the code:

- 1.b) have due regard for the legitimate rights of Third Parties.

This includes the rights of any other party that may be reflected, by the process of the work, or the outcome of the work itself. This means that the following have the potential to be affected by such a statement:

- technology providers, eg PHP, jQuery, Bootstrap, etc...
- the users of the website
- the service provider (based on the assumption the website eventually becomes hosted on a live server)

I have ensured to uphold the rights of all third party technology providers, by ensuring that all licence conditions for using code has been met. For the users and the server, I have made research into the security and privacy issues related to modern websites, and provided solutions to ensure their rights are not violated. I have gone into detail of the issues regarding security and privacy in the 'Practical Aspects' chapter.

5.2.3 Presenting my best professional competence

As an IT professional, it is my duty to ensure that all work that I undertake meets at least the minimum acceptable standards for the quality and integrity of software production. This idea involves working to my best professional competence, including accepting a duty to uphold reputation of profession. As an individual who represents the IT industry, it is my responsibility that the items I am involved with developing are developed with the utmost professional capacity. This is in order to maintain the status of the profession, and not bring it into disrepute, either by action or mis-action. Not only will this help maintain industry reputation as a whole, but it will also ensure, that the work that can be achieved on an individual basis, is to the best of my ability. This is maintained by following these parts of the code:

- 2.a) only undertake to do work or provide a service that is within your professional competence.
- 2.b) NOT claim any level of competence that you do not possess.
- 2.c) develop your professional knowledge, skills and competence on a continuing basis, maintaining awareness of technological developments, procedures, and standards that are relevant to your field.
- 2.e) respect and value alternative viewpoints and, seek, accept and offer honest criticisms of work.

By following these rules in the decisions I make regarding my work, and the development of a project, means I can ensure that the work is of a required quality and high standard.

5.2.4 Understanding legislation

It is important to have an understanding of the legal aspects regarding the implications of designing and constructing a work of software may have. Specific considerations would be

preservation of data, uphold user's private information, or not building anything that could potentially aid in committing any other crimes.

This is related to legal matters, that are there to uphold the safety and security of third parties. But in following these codes, it provides a protection for yourself and the industry reputation.

- 2.d) ensure that you have the knowledge and understanding of Legislation and that you comply with such Legislation, in carrying out your professional responsibilities.
- 2.f) avoid injuring others, their property, reputation, or employment by false or malicious or negligent action or inaction.
- 2.g) reject and will not make any offer of bribery or unethical inducement.

5.3 Professional Issues Evaluation

The BCS as an institute "collaborates with government, industry and relevant bodies" [2] It provides an excellent means for maintaining Professional Issues within the IT industry, and the workplace in general. While parts of the code of conduct may seem to be 'common sense', it is a means to provide a direction in situations of uncertainty, and since it is a globally recognised institution, the following of the code will help to:

- protect the public interest
- maintain professional competence and integrity
- ensure upheld duty to relevant authority
- ensure upheld duty to the profession as a whole.

It is Professional Issues are an area of industry that has ambiguous decisions by default, however they are an important aspect any professional's considerations.

5.4 Open-source and choice of licence

Open-source code is source-code developed by a person or team, that is 'available' in the sense that the code is freely open to access, in order to study or use to develop in order to advance the original source, or to create an entirely new piece of software using the original as a basis. The Open-Source Initiative (OSI) state a definition[9] for what Open-source software is, this includes the following details:

- to allow no restrictions to the redistribution of the software related to the code
- there is an allocation of source-code which goes along with the software
- derived works are allowed to be distributed, but under conditions stated in the originator's licence

- maintain integrity of author's source code by a licence condition stipulating that derived works may have to carry different version number, or software name, based on the individual licence.
- there is no discrimination of persons or groups who can use or redistribute works
- there is no discrimination against fields of endeavour within the field
- distribution of licence must apply to any who use the program, without the provision of an additional licence by any new parties
- licence must not tie to a specific software distribution, it must be usable beyond such set
- licence must not restrict other software from being used appropriately alongside distributed software
- there can be no provision placed on the specific technologies that a software should/could be used with

Open-source software is managed using 'licences' which are managed by the Open-Source Initiative[10]. The licence applied to a piece of software comes with a set of regulations, that dictate the state of ownership that must be stated upon redistributing a new piece of software that has been taken from original open-source piece of code. The licences reflect the definitions as stated by the OSI, including any variations that are specific to a specific licence. There are many licences, some which state that future licences must also hold the same licence, or some that dictate how the new piece of code can be monetised, if it can be at all. Different licences are used for different types of work.

My code should be open-source, because it is a work in education, and so other parties should be able to benefit from it being distributed as open-source. It gives others the opportunity to read and analyse the code, in order to either improve it, or to learn from it. The licence I will choose will allow the code to be freely distributed with no limitation on what or how it is then to be used. The use of the licence also gives a specific indication to who the author is of a piece of code, and so, there can be no ambiguity if there is a discussion as to the original author of the code.

Taking my requirements into consideration, I'm going to choose the MIT licence. This allows the work to be freely distributed with the least restrictions. It ensures that the source originator, cannot be held responsible, and that they are given acknowledgement as the originator of the work, while guaranteeing that any derivative work is re-distributed into the community, to share the knowledge.[11]

Bibliography

[1] British Computing Society *http://www.bcs.org/content/conCertification/115*

[2] British Computing Society Description *http://www.bcs.org/category/5651*

[3] The agile manifesto *http://www.agilemanifesto.org/*

[4] Using Design Patterns *"Design Patterns: Elements of Reusable Object-Oriented Software", Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides (1995)*

[5] Coding Smells *"Refactoring: Improving The Design of Existing Code", Martin Fowler (1999)*

[6] Remote File Inclusion *"https://www.owasp.org/index.php/Testing_for_Remote_File_Inclusion", OWASP.org*

[7] Local File Inclusion *"https://www.owasp.org/index.php/Testing_for_Local_File_Inclusion", OWASP.org*

[8] *"https://www.owasp.org/index.php/Code_Injection", Code-Injection - OWASP.org*

[9] *"http://opensource.org/docs/osd#fields-of-endeavor" Open-source Definition, OSI website*

[10] *"http://opensource.org/licenses" Licence Information, OSI website*

[11] *"http://choosealicense.com/licenses/mit/" MIT licence description*

[12] "https://www.youtube.com/playlist?list=PLfdtiltiRHWGXVHXX09fxXDi-DqInchFD" Alex Garrett, 2014

[13] "http://unixhelp.ed.ac.uk/manual/mod/core.html#options" Official Apache Documentation,

[14] "http://docs.oracle.com/cd/E19146-01/821-1828/gdeun/index.html" Official Oracle Documentation,

[15] "http://php.net/manual/en/faq.passwords.php#faq.passwords.hashing" Official PHP documentation, password hashing,

[16] "http://php.net/manual/en/function.password-hash.php" Official PHP documentation, password_hash()

[17] "PHP 6 and MySQL 5 for Dynamic Web Sites", Larry Ullman, 2009,

[18] "http://php.net/manual/en/faq.passwords.php#faq.passwords.salt" Official PHP documentation, salt,

[19] "Design Patterns: Elements of Reusable Object-Oriented Software" Gamma,Helm,Johnson,Vlissides, 1994

[20] "http://dev.mysql.com/doc/"

[21] "https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems" Server Provider analysis of different RDBMS services.

[22] "http://php.net/" PHP Official Documentation,

[23] "http://guides.rubyonrails.org/getting_started.html" Ruby-On-Rails Documentation,

[24] "https://docs.djangoproject.com/en/1.7/intro/install/" Django Installation guide,

[25] "https://github.com/joyent/node/tags?after=v0.0.4" Official Node documentation,

[26] "http://w3techs.com/technologies/overview/programming_language/all" W3Techs, December 2014,

Appendices

Appendix A: **Running the software**

Requirements

In order to run the software, you need:

- access to an email address
- access to a web browser

Instructions

Type this url 'https://rhul-lets.uk/mvc/mvc/public/home' into the web browser.

This takes you to the home page. From here you can navigate the site using the links in the navbar, and follow the onscreen prompts and messages.

If you have any problems with the site, please contact the site administrator at hostmaster@rhul-lets.uk.

Appendix B: **User Guide**

Accompanying this Guide is a video, which will describe the process for using this website. When accessing this site, there is ssl encryption provided, however there is no verification for the certificate by a certificate authority. As such the browser will falsely complain that it is untrustworthy site due to unverified signature. You can continue onto the site, ignoring the complaints from the browser.

Create an Account

The first thing you should do is to create a user account on the website. Create an account by clicking CreateUser form. After entering your email, you must check your email account, and click the link it provided for you in the email. Upon clicking the link, it will take you to the login page, where you should login with the password given to you in the email. It is highly recomended that on logging in, that the first thing you do is to change your password to something that is more secure.

Searching

The search feature can search your choice of Users, Posts or Skills. As the site becomes more populated with more content added, the search feature will give better results. However for now, try typing 'James' under Users to see all of the users that match that search. Now try typing 'French' under Skills to see any skill that includes that word. Finally try searching 'post' under Post to see a skill. Any search can be typed in here, and results are returned in date order for Posts and alphabetically for Users and Skills.

Make Favour

Has a form for creating a new favour for others to see and respond to. Try creating one, then moving over to 'View Details' on the nav-bar.

View Details

The option defaults to show all of the information about you that has been entered previously. If you select Active Favours on the secondary nav-bar to see all of your 'active' favours. These are favours created, and not yet expired, nor have an exchange of credits made for. They're listed in order of expiry date, you should be able to see the favour you just created in the correct place among the favours. Active Favours also feature the choice to retire, if you want to remove them from circulation before they're expiry.

If you link to Unredeemed favours, you'll see the favours that you have received credits for but the other party has not 'redeemed' the favour yet.

By selecting retired favours you can see all of the favours that you have ever created that have either expired, been redeemed or have been manually retired by you.

Open Favours are all of the favours that you have given credits for, but you have not yet redeemed.

The final option allows you view the entire chronological list (most-recent first) of transaction history between you and other users. All of these options from the secondary-nav-bar will be pretty empty because you've just created a new user, but if you logout, then login with the details:

- username : guest@email.com
- password : password

I have prepared some sample information for you to see.

All of the lists of data have selectable links that can tell you more information about the favour or the user that is involved. Try them out.

Edit Profile

If you decide that you want to change some of your account details, then make your way to the Edit Profile section. Here there are a lot of different forms, depending on what specifically you want to change. Go ahead and try out some of the edit forms then check View Details to see the changes reflected in your account.

The profile information you can change is email, username, password or currently added skill levels. You can also add a previously created skill or you can create a skill to add.

Messages

When selecting Messages, there are 2 main features. The first, Send Message, allows you to send to a user by using their email address associated with their RHUL_LETS account. Try this by typing in the email currently associated with the account you're logged in with, if it's still the guest account type 'guest@email.com' into the recipient box. Fill in the other fields and hit send. From here go to Conversations, you will now see a notification of (New) and a list of previously started conversations. All conversations are sorted in order of last-recieved message, so you should be able to see the conversation (with yourself no less) sat right at the top of the list. Hit that to see details of the new message, and a transcript of the conversation.

Contact, Feedback and Logout

If you experience problems using the site, please use the contact button on the nav-bar to load a mail to the administrator, this will allow your to voice your concern through the medium of email. Alternatively, It is encouraged for all new users to the site to record your experiences, this can be done by hitting Feedback. All responses are anonymous, and help to gather important information about the usability and design of the website, including any application breaking bugs, all of which help to maintain and improve the website.

To return to the home screen and logout, press the 'Logout' button on the nav-bar.

Appendix C: **BCS code of conduct**

Public Interest

You shall:

a) have due regard for public health, privacy, security and wellbeing of others and the environment.
b) have due regard for the legitimate rights of Third Parties*.
c) conduct your professional activities without discrimination on the grounds of sex, sexual orientation, marital status, nationality, colour, race, ethnic origin, religion, age or disability, or of any other condition or requirement
d) promote equal access to the benefits of IT and seek to promote the inclusion of all sectors in society wherever opportunities arise.

Competence and Integrity

You shall:

a) only undertake to do work or provide a service that is within your professional competence.
b) NOT claim any level of competence that you do not possess.
c) develop your professional knowledge, skills and competence on a continuing basis, maintaining awareness of technological developments, procedures, and standards that are relevant to your field.
d) ensure that you have the knowledge and understanding of Legislation* and that you comply with such Legislation, in carrying out your professional responsibilities.
e) respect and value alternative viewpoints and, seek, accept and offer honest criticisms of work.
f) avoid injuring others, their property, reputation, or employment by false or malicious or negligent action or inaction.
g) reject and will not make any offer of bribery or unethical inducement.

Duty to Relevant Authority

You shall

a) carry out your professional responsibilities with due care and diligence in accordance with the Relevant AuthorityâĂŹs requirements whilst exercising your professional judgement at all times.
b) seek to avoid any situation that may give rise to a conflict of interest between you and your Relevant Authority.
c) accept professional responsibility for your work and for the work of colleagues who are defined in a given context as working under your supervision.
d) NOT disclose or authorise to be disclosed, or use for personal gain or to benefit a third party, confidential information except with the permission of your Relevant Authority, or as required by Legislation
e) NOT misrepresent or withhold information on the performance of products, systems or services (unless lawfully bound by a duty of confidentiality not to disclose such information), or take advantage of the lack of relevant knowledge or inexperience of others.

Duty to the Profession

You shall:

a) accept your personal duty to uphold the reputation of the profession and not take any action which could bring the profession into disrepute.
b) seek to improve professional standards through participation in their development, use and enforcement.
c) uphold the reputation and good standing of BCS, the Chartered Institute for IT.
d) act with integrity and respect in your professional relationships with all members of BCS and with members of other professions with whom you work in a professional capacity.
e) notify BCS if convicted of a criminal offence or upon becoming bankrupt or disqualified as a Company Director and in each case give details of the relevant jurisdiction.
f) encourage and support fellow members in their professional development

* Definitions:

Legislation - The term âĂIJLegislationâĂİ means any applicable laws, statutes and regulations.
Third Parties - The term âĂŸThird PartiesâĂŹ includes any person or organisation that might be affected by your activities in your professional capacity, irrespective of whether they are directly aware or involved in those activities.
Relevant Authority - The term âĂIJRelevant AuthorityâĂİ in this document is used to identify the person(s) or organisation(s) which has / have authority over the activity of individuals in their professional capacity. For practising BCS members this is normally an employer or client. For student members, this is normally an academic institution.

Appendix D: **Code**

PHPMailer snippet

```
//make the password
$password = rand(1000,5000);

//make the hash
$hash = md5(rand(0,1000));

//setup send variables
$to      = $email; // Send email to our user
$subject = 'Signup | Verification'; // Give the email a subject
$message =
"

\r\n Thanks for signing up!\r\n
\r\n Your account has been created,
\r\n you can login with the following credentials
\r\n after you have activated your account
\r\n by pressing the url below.

------------------------\r\n
Login:    ".$to."\r\n
Password: ".$password."\r\n
Hash:     ".$hash."\r\n
------------------------\r\n

Please click this link to activate your account:
\r\n http://https://rhul-lets.uk/mvc/mvc/includes/accountCreation.php?
email=".$email."&hash=".$hash."\r\n
";

//create SMTP connection
$mail = new PHPMailer;
$mail->isSMTP();
$mail->SMTPAuth = true;
$mail->Mailer = 'smtp';
$mail->Host = 'tls://smtp.gmail.com:587';
$mail->Username = "letsatrhul@gmail.com";
$mail->Password = "*******";
$mail->SMTPSecure = 'tls';
$mail->Port = 587;

//create message
$mail->From = "letsatrhul@gmail.com";
$mail->FromName = "Lets - Admin";
$mail->addReplyTo('replyto@example.com', 'Reply To');
$mail->addAddress($to);
$mail->Subject = $subject;
$mail->Body     = $message;//allows html markup
```

```
$mail->AltBody = 'This is a plain-text message body';
$mail->setFrom('letsatrhul@gmail.com', 'Lets Admin');

//send the message
$mail->send();
```

PHP Side of Ajax Call

```
//if 'action' is set, and 'action'==='checkNewMessages', perform the
 behaviour in this if block
else if(isset($_POST['action'])&&$_POST['action']==='checkNewMessages')
{
session_start();

$clientID=$_SESSION['client_id'];

//perform SQL query of the db for new messages for that user
$newMessageDataSet = SqlConnect::getInstance()->
validateConnection()->prepareStatement("SELECT message_id FROM
 messages WHERE reciever_id=? AND state =?")->
 bindParamsNewMessages("ss",$clientID,'unread')->stmtExecute()
 ->get_Result()->getRes();

//check size of the dataset
$size = SqlConnect::numRows($newMessageDataSet);
if($size > 0){
//edit session vars
$_SESSION['unreadMessages']= " (New)" ;
$unread = true;
}
else{
$_SESSION['unreadMessages']="";
$unread= false;
}

$verified = true;

//create return variable
$response = array('verified'=>$verified,
'unread'=>$unread,'size'=>$size);
}
//return the created json data to the ajax function that called it.
echo $response;
```

Appendix E: README

The Directory Structure for the application

> mvc - Top level directory
> mvc/README.txt - readme file
> mvc/wwaz008.final.pdf - Report file
> mvc/app/core - core app files
> mvc/controllers - controller files
> mvc/models - model files
> mvc/views - view files
> mvc/includes - supporting php and js files
> mvc/public - public content for application

Appendix F: **Apache Config**

Run this sequence of commands when logged in to server. (For Apache2, and Ubunutu 14.04)

```
$ sudo a2enmod ssl
$ sudo service apache2 restart

$ sudo mkdir /etc/apache2/ssl
$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/apache2/ssl/apache.
```

Enter extra information if you want, but you must edit: - Common Name (e.g. server FQDN or YOUR name) []:your_domain.com - Email Address []:your_emaildomain.com

```
$ sudo nano /etc/apache2/sites-available/default-ssl.conf
```

edit the above file to look like this (ignoring the comments in the file):

```
<IfModule mod_ssl.c>
    <VirtualHost _default_:443>
        ServerAdmin admin@example.com
        ServerName your_domain.com
        ServerAlias www.your_domain.com
        DocumentRoot /var/www/html
        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
        SSLEngine on
        SSLCertificateFile /etc/apache2/ssl/apache.crt
        SSLCertificateKeyFile /etc/apache2/ssl/apache.key
        <FilesMatch "\.(cgi|shtml|phtml|php)$">
                        SSLOptions +StdEnvVars
        </FilesMatch>
        <Directory /usr/lib/cgi-bin>
                        SSLOptions +StdEnvVars
        </Directory>
        BrowserMatch "MSIE [2-6]" \
                        nokeepalive ssl-unclean-shutdown \
                        downgrade-1.0 force-response-1.0
        BrowserMatch "MSIE [17-9]" ssl-unclean-shutdown
    </VirtualHost>
</IfModule>
```

Run the following commands

```
$ sudo a2ensite default-ssl.conf

$ sudo service apache2 restart
```

Appendix G: **Feedback Form**

Please see the next page for the feedback form.

9 responses

View all responses Publish analytics

Summary

Please describe your overall impression of the website?

Overall the impression is good as the website is operational

I really like the logo. The website works well and I like the idea behind it.

Good concept a little confusing to use though.

Great.

The website seemed plain, I could only see the navigation bar with all the different options at the top. The rest of the webpage was empty white space. I don't know if its because I'm using Google Chrome or the fact that the goal of this website was to demonstrate functionality and not so much design.

Like the clean look and feel, minimalist has always been a favourite of mine! As a start up user, it is perhaps not very intuitive - i.e., a brief description of what it is about and an easy start up guide would be helpful - just to get kicked off. Registering was very quick and painless, a pleasure to use. When showing the favours page, again here an example or two might be helpful?

Functionality is there and works well, I liked the way it told you whether the email address was available without having to refresh the page

Set out nicely and well presented. Easy to navigate from the simple buttons at the top of the screen. New and interesting idea, user friendly

Did you find any bugs or glitches with the program?

none.

The dialog reminding you to change your password would continue to pop during my entire tour of the website.

No

No.

None that I found, it seems to work smoothly

None found when I was looking around.

No, everything worked well. I successfully created a new account and could edit my profile.

Do you have any suggestions for improvements in the website?

Have the verify email link actually be a link rather than inputting it into a a browser

Add some instructional text to the process of signing up, so that the user knows why they are doing a step and what the next steps are. Add a welcome message, explaining what the website system does.

Add some graphics and colour, more options for specialties. Make is a bit easier to navigate such as adding a site map.

I would just place the "home" as the first thing, before the "search".

Upon completion the home page should have some kind of design or picture, and the moment it looks slightly bare and incomplete

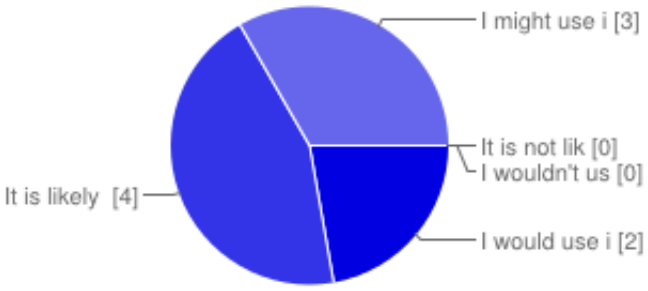
Keyword searching. So if i search for 'math' instead of receiving no results i could be given results something that may contain math in it. Have an option to disable the dialog to change password during a web session if i have already received it once.

I've mentioned my thoughts in the overall impressions box. A small 'about' section would be useful, as well as a quick set up guide, to take you through registration and how to get going with the registration of either a favour to offer or required.

It's not clear how to use the website from the start, for example I don't know how to find other people's favours. It would be nice if more of the page had content or something to easily distinguish each page from the other. If the email checker turned red when the email was unavailable would also be an improvement as it's difficult to see the difference between "email available" and "email unavailable".

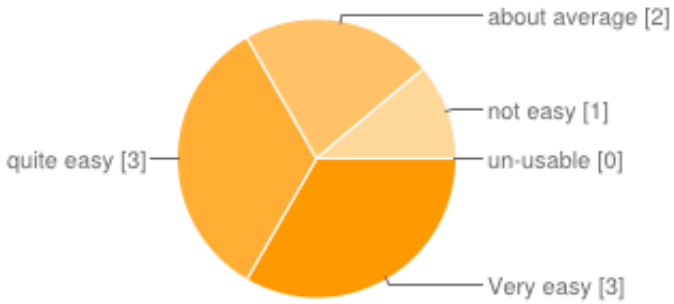
The layout.

How interested would you be in using the website and the services it provides??



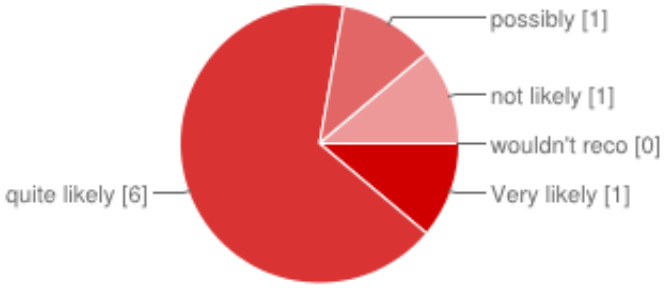
I would use it	**2**	22.2%
It is likely I would use it	**4**	44.4%
I might use it	**3**	33.3%
It is not likely I would use it	**0**	0%
I wouldn't use it	**0**	0%

How was the site to navigate?



Very easy	**3**	33.3%
quite easy	**3**	33.3%
about average	**2**	22.2%
not easy	**1**	11.1%
un-usable	**0**	0%

How likely would you recommend this site to your friends?



Very likely	**1**	11.1%
quite likely	**6**	66.7%
possibly	**1**	11.1%
not likely	**1**	11.1%
wouldn't recommend	**0**	0%

Number of daily responses