FACULTY OF SCIENCE, ENGINEERING AND COMPUTING

School of *Computing and Information Systems*

BSc DEGREE
IN
Computer Science Bsc

PROJECT DISSERTATION

Name: Shakir Ahmad

ID Number: K1018112

Project Title: RYA information system

Project Type: Design

Date: 02/04/2015

Supervisor: Dr Graeme Jones

Kingston University London

Did you submit a draft of your dissertation to your supervisor?	Yes
Did you receive feedback from your supervisor on any submitted draft?	Yes

Plagiarism Declaration

The following declaration should be signed and dated and inserted directly after the title page of your report:

Declaration

I have read and understood the University regulations on plagiarism and I understand the meaning of the word *plagiarism*. I declare that this report is entirely my own work. Any other sources are duly acknowledged and referenced according to the requirements of the School of Computing and Information Science. All verbatim citations are indicated by double quotation marks ("…"). Neither in part nor in its entirety I have made use of another student's work and pretended that it is my own. I have not asked anybody to contribute to this project in the form of code, text or drawings. I did not allow and will not allow anyone to copy my work with the intention of presenting it as his or her own work.

Date _09/04/2015_____

Signature __SHAKIR AHAMD_____

Contents Page:

Contents

1. Introduction

To introduce this report a client will be interacted with in order to create a website for the RYA Sailing Club. The RYA Sailing Club is a training facility which enables scouts to enrol on courses that are taught be instructors. Qualifications can be earned on these courses which are handed to scouts by instructors once a scout completes all components of a stage; in order to proceed to the next stage. The RYA Sailing Club has a system which is the current system and it needs to be replaced with a new system. The old system uses a paper based model which is tedious, cumbersome and is hard to apply to an everyday training facility that constantly updates records. The new system however is to replace the old system with a digital format that is able to rectify the issues of the old system. The problems that are present in the old system include loss of paper records such as scout records, difficulty in modifying records, visual aspect when viewing records being in an environment where there is water can lead to the paper being wet and searching for a particular record can be time-consuming. The new system is a digitalised website in which it has the capability to amend all the issues stated previously, the new system is used on computers where it is easy, flexible and a faster way of handling records. The feature it has is being able to update records on the fly where before it was done on paper and there was no convenient way of altering scout records. Other features include being able to search for a scout by name when input in the search bar, preventative measures for data loss as data is always backed up on another server and widgets in order for viewing records seamlessly between each element that includes scouts, grades and course enrolled.

Access Website:

URL: http://studentnet.kingston.ac.uk/k1018112/fyp/FYP/

Login:

Instructor:

Username : i

Password : i

Scout:

Username : s

Password : s

Course Planner:

Username : p

Password : p

2. Literature Review

Introduction:

This chapter consists of a literature review that contains the topic of the Data Protection Act with subheadings discussing the different elements to the Data Protection Act. The topic will be introduced with what is the Data Protection Act and when was it made available, followed by the core principles of the Data Protection Act, security in terms of the Data Protection Act, ownership of data and lastly case studies.

2.1 Data Protection Act

2.1.1 What is DPA and when and why was it introduced?

The current Data Protection Act was introduced in the United Kingdom in 1998 the reason being is to comply with the right to privacy of personal information being treated as confidential between the user and the party that is willing to obtain such information. This law is known as a human right, the main reason the Data Protection Act 1998 was implemented as a law is to correct the contradiction of previous data protection laws which were the two principle aims of data protection laws which are:

- The protection of privacy during the processing of personal data.
- The maintenance of free flows of personal data between countries.

[A]

The maintenance of free flows of personal data between countries contradicts the first aim so to compensate for this there is high level protection for personal data that is undergoing processing in order to ensure privacy for the user.

2.1.2 What is the core principle of DPA?

The core principle of the Data Protection Act 1998 is privacy of data, governing bodies across the world have set four common practices called the First Principles which are ": (1) data quality; (2) transparency or openness of processing; (3) treatment of particularly sensitive data, often defined as data about health, race, religious beliefs, and sexual life among other attributes; and (4) enforcement mechanisms." [C] The First Principles was implemented in three motions, the first being a comprehensive data protection law which gave rights and responsibilities of personal information similar to how human rights have been implemented as a law which is applied towards citizens. The second approach is used to protect privacy through the norm and terms and conditions rather than relying on the government as the terms and conditions is used to protect both the user and the company when it comes to data. The last motion is used to implement policy rules for data where it shows the user what the company does with the data, which data gets leased to who such as third parties or whether the company do not disclose any information to any party.

In the United Kingdom a professor by the name of Colin Bennett making standards regarding privacy of personal information into ten elements shown in the following:

- Must be accountable for all personal information in its possession;
- Should identify the purposes for which the information is processed at or before the time of collection;
- Should only collect personal information with the knowledge and consent of the individual (except under specified circumstances);
- Should limit the collection of personal information to that which is necessary for pursuing the identified purposes;
- Should not use or disclose personal information for purposes other than those identified, except with the consent of the individual (the finality principle);
- Should retain information only as long as necessary;
- Should ensure that personal information is kept accurate, complete, and up to date;
- Should protect personal information with appropriate security safeguards;
- Should be open about its policies and practices and maintain no secret in- formation systems;
- Should allow data subjects access to their personal information, with an ability to amend it if necessary.

[B]

The first element shows that for example if a company has access to a user's personal information, that company must be accountable for that data. What this means is that the company must be liable for all the information on user's the company possess where if the company were to lose or modify data the company would have to comply with the policies that the company set and may be liable to court within the rights of the user.

The second standard shows that a company must show the user the purpose of the information the company requires in order to allow the user to grasp the concept of why the company needs the information in order for the user to be comfortable in allowing the company to obtain personal information.

The third element states that the consent of the user must be obtained in order for a company to uphold information about the user where if the user does not agree to the company's terms and conditions, the company will not be able to store data on that individual.

The fourth standard states that only necessary information should be obtained in relation to the second standard as unnecessary information would not be needed and would not be a priority when it comes to processing information.

The fifth element is an important statement where personal information should not be disclosed to anyone unless that party such as a third party has been identified or if the individual has given consent. The reason being is to ensure privacy of the user so that it can protect the user from an array of vulnerabilities such as exposure of personal information can lead to identity fraud and other swindles.

The sixth standard states that information may be retained as long as it is necessary in conjunction with the fourth standard otherwise the data should be deleted as it is redundant.

The seventh element states that all information should be accurate, whole and current where information that is inaccurate should not be kept within a company as it can lead to confusion for the company as well as the user.

The eighth standard states that all personal information should be protected with security safeguards for example encryption comes into mind where data should be secured where if the data was obtained unlawfully the data that would be shown to the individual that obtained the data unlawfully will be shown data that would be scrambled due to encryption. Secure Cipher Current Standard used by the government is AES which is impossible to hack with brute force and can only be hacked by mathematical algorithm which is currently not possible to do, if an unauthorised person obtained this data it would be near impossible for the unauthorised person to decrypt and obtain sensitive information.

The ninth element shows that the company should be open about the company's policies and practices where no secret information systems should be implemented as this contradicts the third element as if the company were to implement this secret information practice it would not comply with the company's own policies where the consent of the user must be given in order to uphold information about said user.

The last standard states that a user should be able to access and modify personal information as it is within the user's rights to be able to access the data and edit the information at any given time.

2.1.3 What is the definition of data and under what context is DPA valid?

There are two types of data which are electronic data and manual data, the definition of data is information to be specific personal data means "any information relating to an identified or identifiable individual." [D]

2.1.4

The Data Protection Act 1998 is valid if a party is processing personal data, it does not apply if the party is not processing personal data. If the party is processing personal data and it is established in the UK and processing data within a UK establishment then the Data Protection Act 1998 applies.

2.1.5 Who is responsible for defining the purpose and regulating the processing of data?

The data controller is responsible for collecting the data of the recipient where the recipient is the person who discloses the data and is processed by the data controller. It also concerns third parties which is anyone other than the data controller that may receive data about a recipient. The only time where the Data Protection Act 1998 does not apply is when legal powers are involved such as an Information Commissioner or the police may be able to obtain data of a recipient without the consent of the recipient.

2.1.6 Consent of information for parties who are unable to consent themselves

The Data Protection Act complies with the laws that are upheld in order to protect people who are unable to consent for themselves. People such as young children or people with disabilities who are unable to consent for themselves need parental guidance or guidance from a guardian in order to be able to access services an organisation provides for users. If a child is too young to consent for themselves a parent can step in and allow them to have access to services based on the parent's permission if the parent is willing to give consent for the child.

2.2 Security

2.2.1 Encryption Algorithms

A symmetric encryption method is used to pass messages from one person to another using a secret key where the secret key is used to decrypt the messages passed on from each person in order to identify what message has been sent by converting the cipher text in to plain text which can be read.

An asymmetric key however corrects the faults of the symmetric encryption method where by using a symmetric key it is prone to many flaws such as if the secret key were to fall in to the wrong hands then that person who obtained the key would be able to decipher messages and be able to create false messages whilst the original persons would not be able to identify if it is someone else. In order to rectify these flaws two keys would have been created where there is a public and a private key where the public key would be made for anyone to use and the private key would be kept secret and can only decrypt messages if the keys both public and private are matching.

2.2.2 Restricted Access

The users have access to a specific part to a database in which the user can view the database without the option to alter the way the database works. The use of application programming interfaces is what allows for differentiation in users to show what users can access what elements of the database. The first application programming interface for example could be

MySQL followed by PHP followed by HTTP; these three interfaces intercommunicate with each other in order to form the application. The basic user has the ability to view the HTTP source in which the user can view a website containing the infrastructure of the database itself. The three interfaces communicate with each other in an orderly manner being that the last interface cannot communicate vice versa for example HTTP cannot communicate with MySQL in order to obtain information and has to user the PHP link between MySQL and PHP in order to establish the connection. This technique is used by developers to restrict the access of elements containing the database where the basic user is limited to viewing the contents and the developer has access to every application programming interface that has been used.

2.2.3 The practice of implementing different form views for users with different access privileges.

Based on the aforementioned of basic users, there a different basic users that have a subset of user privileges for example on BlackBoard a teacher will be able to view and modify elements within BlackBoard whereas a student may only view elements such as grades and assignments. This is a feature implemented by developers to give certain users priviliges within the user's limitations of what the user can do for example a teacher would be able to set grades based on what has been submitted, upload coursework and have access to lectures beforehand whereas the student can only submit coursework, view grades and view lectures upon a lecturer uploading the lecture. In order for this system to work and distinguish the users, a login system would need to be created where students would have an ordinary login whereas the teacher or lecturer would have a unique login so only that particular user has advanced privileges when compared to the student.

2.3 Ownership of data

2.3.1 Who owns the data?

If a user has personal data and has given the consent to allow an organisation to obtain this data then who would have more priority as to who owns the data. The user is the original owner of the data as it is the user who is using the data to identify the user, however if the user has given consent to distribute the data to an organisation this causes an argument as to who owns the data. Since the data has been given to the organisation, the organisation now has the privilege to use the data as it pleases that complies with what the user has agreed to for example the privacy policy and whether the user allowed third parties to obtain the data.

Some may argue that the data is solely the user's and no other persons however if the user has agreed to allow an organisation to uphold its information then evidently the organisation has rights to the data as much as the user does so in theory both these persons have the right to the data depending on how the privacy policy has been agreed to on both ends.

2.3.2 What are the implications of Policy Change in terms of DPA for the Organization and for the user of the system

The argument imposed here is that if a user were to agree to the old policies of an organisation, however if the organisation were to change the policies to the point where the organisation has full rights to the data and the user did not agree to this how this would be solved.

The solution is simple, in order for the organisation to keep and have full access to a user's personal information due to policy changes the organisation must make a mandatory update to the policy changes and inform the user that there is a new policy. The user must then view these policy changes and must choose an option to agree or decline these changes where if the user were to decline then the organisation would no longer have access to that particular user's information. If the user were to agree then the organisation would have full access to the user's personal information as agreed upon.

2.3.3

If the user cancels a membership associated with an organisation is personal information stored for archival purposes or deleted and mandatory information required from the user to use services and optional information.

If a user were to be a member that was associated with an organisation by creating an account with the company which contains the user's personal information, if the user were to cancel the membership with the company would the company keep or remove personal information on that user.

Depending on the company's policies the company would have the right to either do both in the aforementioned where in some cases the company would keep that information in archives to document that the user was a member in order to keep consistency and avoid confusion but tend to remove all aspects of that member's data as the company must comply with Data Protection Act as the company no longer has the right to uphold this information.

2.4 Case Studies:

2.4.1 Who owns data stored on servers and who is responsible for ensuring privacy of data: iCloud Data Leak: this attack was made possible because the user did not create a strong enough password not with the system implemented by apple.

The iCloud data leak was caused by security vulnerabilities where if a person has access to a user's password then that person would have access to all data stored on that iCloud server even if the data was sensitive. In order to obtain the password the BruteForce method could be used in order to obtain a password of the user, one of the security vulnerabilities of Apple is that there was no lock-out feature where if a person were to attempt to enter a password many times the system would lock that person out so that the person could not attempt to enter another password similar to how the iPhone passcode works. Therefore the person

attempting to breach an iCloud account would have unlimited freedom as to how many times that person could enter a password.

Conclusion

In conclusion the literature review that was documented in this chapter discussed the Data Protection Act and why it was introduced. It also discussed the core principle to the Data Protection Act and the ten standards regarding privacy policies. Other topics included were the validity of the Data Protection Act, who was responsible for controlling data such as data controllers, consent for minors and people with disabilities in terms of the Data Protection Act, security and ownership of data with its respective sub sections. Case studies were also mentioned in this chapter which discusses real life scenarios and how it complies with the Data Protection Act.

Bibliography

2.1

Reidenberg, J. R. (2000). Resolving conflicting international data privacy rules in cyberspace. *Stanford Law Review*, 1315-1371.

2.2

Room, S. (2007). Data protection and compliance in context. Swindon, U.K.: British Computer Society.

2.3.1

Room, S. (2007). Data protection and compliance in context. Swindon, U.K.: British Computer Society.

2.3.2

Room, S. (2007). Data protection and compliance in context. Swindon, U.K.: British Computer Society.

2.4

Room, S. (2007). Data protection and compliance in context. Swindon, U.K.: British Computer Society.

3 Analysis

Introduction

In this chapter analysis will be used to examine the entire scope of the project, to see whether the project is feasible and whether it should be carried out. User stories and a class diagram will be implemented in order to show what type of functionality the user requires from the website, as well showing the main basis for the database respectively. Functional and non-functional requirements will be used to showcase how the each user would interact with the website as well as the attributes and time-constraints the project presents.

3.1 SWOT Analysis

Introduction:

In order to find out whether this project will be worthwhile, a SWOT analysis will be made in order to confirm its worth shown below. This will give the client a good idea of whether the project should be carried out in order to show the feasibility of the project.

3.1.1 Strengths of the project:

- Scout records would not be lost as it is on a digital format not paper based
- Records will uploaded on a back-up server for extra preventative measures
- A system that can keep track of Scout progression digitally
- Scout records can be added, altered and deleted with ease
- Widgets can be used in order to alternate between different content
- System access can be restricted to certain users with certain privileges

3.1.2 Weaknesses of the project:

- Back-up server could fail or be destroyed that contains the data
- Can only be accessed online no paper records

3.1.3 Opportunities of the project:

- Functionality can be extended to mobile devices
- Data can be used to generate graphical meta data

3.1.4 Threats of the project:

- Unauthorized persons may obtain data that was not intended for them to see

Conclusion:

Due to what has been listed in the SWOT analysis, this project will be worthwhile as the strengths outweigh the weaknesses and it has many opportunities to expand the project therefore being a viable project.

3.2 Requirements Elicitation

3.2.1 Client requirements for Information System

The old system that is currently in use is a paper-based model that stores scout records; the scout records are constantly added and updated. The paper-based model has a lot of disadvantages that are time-consuming and difficult when it comes to applying modifications to scout records. Applying alterations to scout records can be inconvenient and problematic as using a paper-based model is unable to format in to human-readable content and mistakes can be made when altering. Adding a new scout is also inefficient in the old system as creating an entire form for each scout is an ineffective way of creating scout records. Scout records can also be lost or incomprehensible when going through the paper-based model in the old system, as the RYA Sailing Club is in an environment that is surrounded by water which can be problematic when dealing with records as records can easily be plummeted in to the water.

3.2.2 Client requirements for user interface

Weekly client meetings were held to obtain Client feedback with regards to plans with implementing functional requirements into the website. The client required that the website was to be made available on mobile platforms in order for marking scout progress to be made in an efficient way. The client also wanted the website to be mobile friendly where the website would adapt to any mobile platform and would be able to fit the content on a smaller screen. For example images being able to resize itself based on whether the user is zooming in or out on the interface and being able to adapt to any mobile platform regardless of the size of the screen being able to display the same content as the website without limitations.

3.3 User Stories

Introduction:

User stories in correlation with the RYA project are used in order to find out what the user needs for the website so that these features the users ask for could be implemented based on priority of the user story.

User Story-1:

“As a member scout of RYA I am keen to improve my sailing skills and wish to view my progress on the course. I would like easy access to be able to see my progress from any Internet enabled deice through a web browser. ”

Christopher Reed is a scout member at RYA sailing club who is keen to improve his skills after many after signing up for a stage 2 course and completing his theory he has now completed the first three physical components of the course and would like to view his progress. Christopher carries a smart phone that can connect to the Internet and would like to view his progress online.

User Story-2:
"As Instructor of the RYA sailing club I often have to decide which qualification I should teach and which students I should invite on the course. I would like to be able to view all scouts at the RYA sailing club and be able to invite students on the course at the click of a button."

Mark Hummingbird aged 42, is a trainer at RYA sailing club, he would like a more efficient method for inviting sea scouts to his courses.

User Story-3:
"As a keen Parent of sea scout of the RYA sailing club, I'd like to view the progress of my child so that I can see if my child is enjoying and progressing in the sailing club.

User Story-4:
"As a Course Designer I would like to see the current progress of all scouts select a course stage with the least amount of scouts recently qualified but with the most amount of scouts who have not which will enable the most scouts to progress to the next stage."

User Story-5:
"As a Course Designer at the RYA sailing club, I would like to see the current progress of all scouts select a course stage with the least amount of scouts recently qualified but with the most amount of scouts who have not which will enable the most scouts to progress to the next stage."

User-Story 6:
"As a Curriculum designer for RYA sailing clubs, I would like to receive feedback from scouts and trainers regarding both physical and theoretical components of qualifications. This will enable me to modify existing courses to enable scouts to learn and develop sailing skills."

User Story 7:
"As an instructor I would like to able to view scouts and make modifications or delete information of scouts in order to keep information up-to-date and accurate."

User Story 8:
"As a scout I would like to see all my badges and awards I have achieved throughout the course in order to keep a track of what I have done during the course."

User Story 9:
"As an instructor I would like a search function in order to search for scouts to see what courses they have enrolled on, their progression and their badges they have received in order to make things easier instead of using the cumbersome paper-based format."

User Story 10:
"As an instructor I would like to have an invitation system where I can invite scouts to the next stage of their course such as stage 2 scouts, inviting them to stage 3 and so forth."

***User Story 11*:**

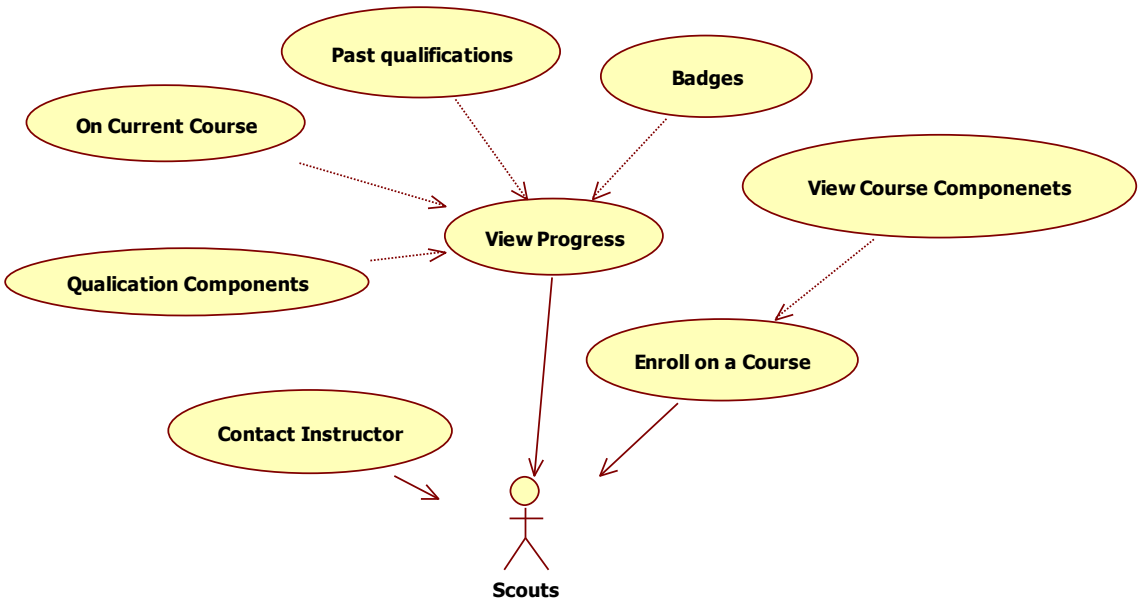
"As a scout I would like to be able to enrol on a course online by filling out a form and being able to choose a desired course."

User Story 12:
"As an instructor I would like to have my own unique login so that I can view information that is limited to scouts such as current grade for a recent exam, where the results have not been published."

Conclusion

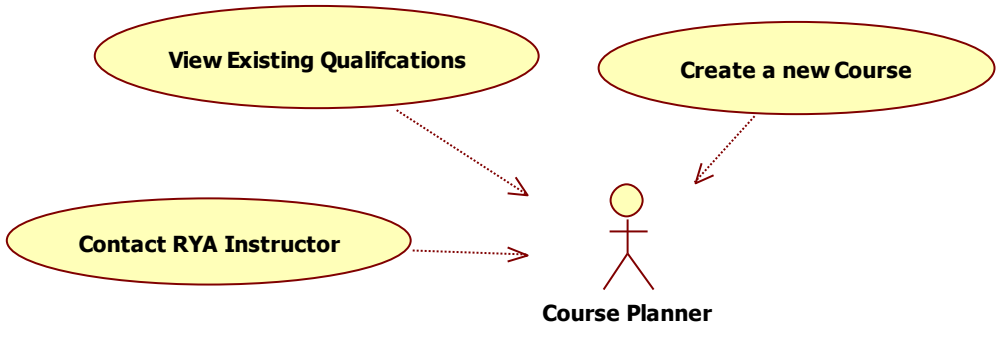
To conclude user stories show what type of functionality should be included based on the specific user story showing what functions should be implemented based on what the user said in the user story. This allows for the client to get a good grasp of what the user may want similar to feedback but before the website has been fully implemented and working.

3.3 Use Case Diagram



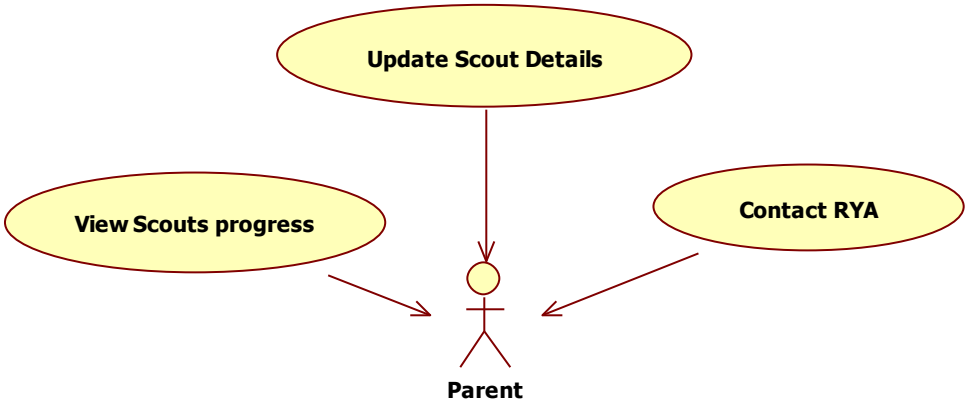
Scouts

Scouts will interact with the system by going through a series of different options to go through on the website. The scout can contact an instructor to ask for assistance, enrol on a course and view course components in which the scout can apply online to enrol on a course and the scout can view progress such as which qualification components completed.



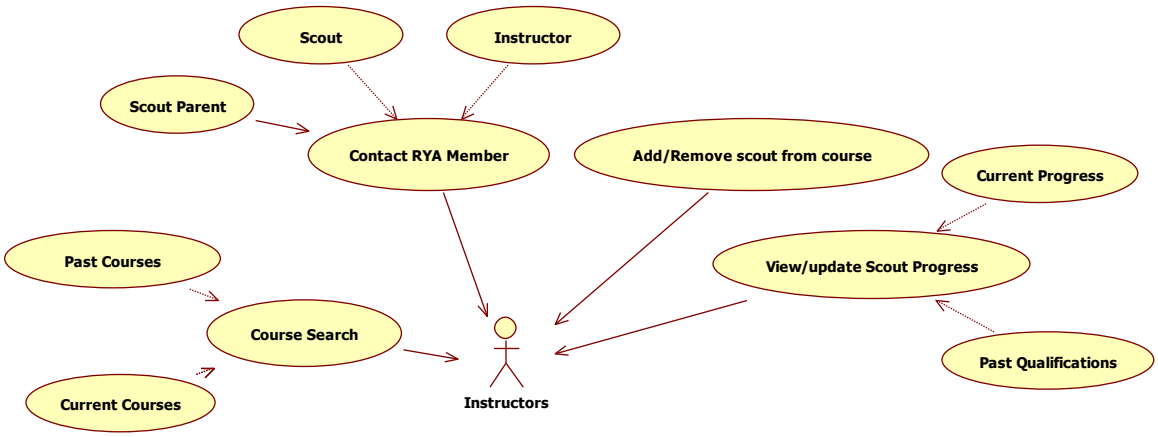
Course Planner

Course planners can view existing qualifications and create new courses that could be based off the old qualifications. When creating a new course these new courses will be applied and added to the website so that any scout may be able to apply for that course.



Parent

The parent of the scout checks and updates scout records whilst giving consent to the scout in order for the scout to be able to enrol on a course, provided if personal information has been given only if the parent agrees to the terms of the RYA Sailing Club. If a scout is too young then the parent will have to give consent to give scout personal details as well as updating scout details if scout is unable to do so on the scout's own accord. The parent can also contact the RYA Sailing Club in which the parent is able to check on a scout's progress and the scout's well-being which is important for parent's to know about the scout. This coincides with the Data Protection Act in which parental guidance must be required in order to give consent to the parent's child, in order for that child to be able to have certain privileges and whether that child will be able to enrol on a course. (Refer to Literature Review 2.1.6)



Instructor

The instructor interacts with the system by being able to add and remove scouts from a course as well as being able to update scout progress. Instructors can also search for courses and add particular scouts to a particular course; instructors can also award qualifications once a scout reaches a certain completion of components for each stage.

3.4 Functional Requirements

Introduction

Functional requirements are used to show functions within the RYA project where it is used to show the step-by-step process of how the different users would interact with the system.

Non-Functional requirements are shown to display how the system would work for example having a logins being unique whilst using the website for scouts and instructors, time constraints where the deadline must be met for the project and the costs that may be needed in order for the project to be Create.

3.4.1 Functional Requirements

Primary Actor	Use Case
Scout	- Earn Awards and badges at the end of the course. - Scout is now enrolled. - Register as a Scout after fully completing the form. - Data collected from the form is now stored on the database. - Fill in the information required to enrol on desired course. -Check progression of the course enrolled on. - Enrol on a desired course.
Instructor (Already has an account)	- Logout of the system. -Search for specified Scout and list information about Scout. -Alter or delete information about Scout. -Allocate spaces for and inform scouts for new course. - Login in using unique login given to Instructors.
Course Designer	-Create Course for Scout to enrol on. -Design scheme for each course.
Parent	-Give consent to Instructor to uphold detail of Scout. -View and track Scout progression.

3.4.2 Non-Functional Requirements

Performance:
Processing – Will be able to process data every time a Scout enrols and stores it on to a database.

Security:
Login- Different levels of priority should be enforced for Scouts and Instructors where Instructors have a unique login that has access to more content.

Passwords – Passwords are case sensitive and have varying level of degree of Strong to weak passwords.

Multiple Attempts- Locks out the user after attempting to enter the wrong password multiple times

Costs:
Budget- There is no budget as all the utilities can be used at the university free of charge.

Timescale:
Timescale- Project will be completed by March 2015

User Interfaces:
Scouts enrolled can use the website to check their progression, awards and badges and can view and accept invitation to a new course.

Instructors can use the website to view, delete and modify information about Scouts and their courses that they have enrolled on.

Software interfaces:
The software used such as PHP is compatible with different operating systems.

Capacity:
Can hold a large amount of data due to new Scouts enrolling each year and Instructors adding information to Scout data.

Sustainability:

The sustainability of the website is dependent on the server where if all else fails then backups will be used as a counter measure.

Monitoring:

Surveillance measures will be made in order to maintain the server and protect invaluable data stored on the server. All user login and logout activities will be logged.

Recovery:

All data will be backed up on another server in case the main server loses invaluable data of the whole system.

Reliability

The minimum to maximum down-time for the server should be 1 hour to 24 hours.

Availability:
Availability of data will be available as long as the server is up whilst hosting the website

Documentation:
Records will be filed on a computer; none of these records will be filed via paper-based format.

Conclusion

In conclusion the reason why functional and non-functional requirements are used is in order to show the scalability of the project and its limitations. Referring to the latter it shows what the client expect from the project in order to garner a good idea of how the system works and the time-constraints applied to the project. It also shows how the user would use the website and what options the user has based on the type of user and the user's privileges.

3.5 MOSCOW

Primary Actor	Use Cases
MUST	-Enrol on a desired course. -Fill in the information required to enrol on desired course. -Register as a Scout after fully completing the form. -Data collected from the form is now stored on the database. -Scout is now enrolled. -Check progression of the course enrolled on. -Earn Awards and badges at the end of the course.
	-Login in using unique login given to Instructors. -Search for specified Scout and list information about Scout. -Alter or delete information about Scout. -Allocate spaces for and inform scouts for new course. -Logout of the system.
	-Create Course for Scout to enrol on. -Design scheme for each course.
	-Give consent to Instructor to uphold detail of Scout. -View and track Scout progression.

3.6 Conclusion

To summarise the content explained in the analysis chapter was used in order to capture the scope the project as well as the feasibility of the project, in order to narrow down the advantages and disadvantages of carrying out the project. In conclusion the project will be carried out and it is worthwhile as it completely removes the paper-based format the old system had with the new digital system that allows for reports to be generated, modified and removed with ease therefore outweighing the cons with pros.

4 Design Chapter

Introduction

The design chapter is heavily focused on planning out the infrastructure of the project as a whole where client meetings take place, in order to discuss what kind of notions could be implemented in the project. The structure of the project is key to maintaining consistency in the project and to follow through with a plan, being a crucial element in implementing ideas and designs in order for the project to be carried out with the least amount of flaws/

4.2 Database Design

4.2.1 Class Diagram

The diagram shown in Figure 1 overleaf is used to express the logic side of the project as well as the infrastructure of the database without hindering the project. It is used as a plan to progress the project in an efficient way in order to prevent the vulnerabilities that can be present within a database. For example, making errors from the beginning of the creation of the database can lead to serious revisions which can be time-consuming which is why it is important to get a well-organised plan in order for the project to reach its goals consistently without being held back by cumbersome errors leading to the aforementioned elapsing of time.

4.2.2 Sub-systems of RYA Database

The final class diagram is split into three subsystems, the first being RYA Membership which shows all the classes related to the user side of the project including instructors and scouts with respective logins for each user. The second is Active Workflows at RYA showing dynamic tables that are constantly changing data wise where the Scout table is interconnected with composite tables which is linked to a course. The third subsystem is Qualifications, Qualifications have finite tables that do not change or are changed slightly but always between certain thresholds of data being the opposite of the dynamic tables mentioned in the second subsystem.

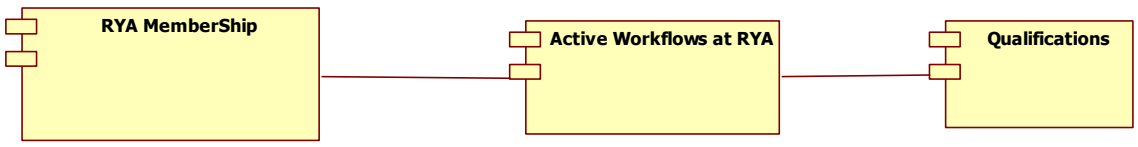


Figure 1

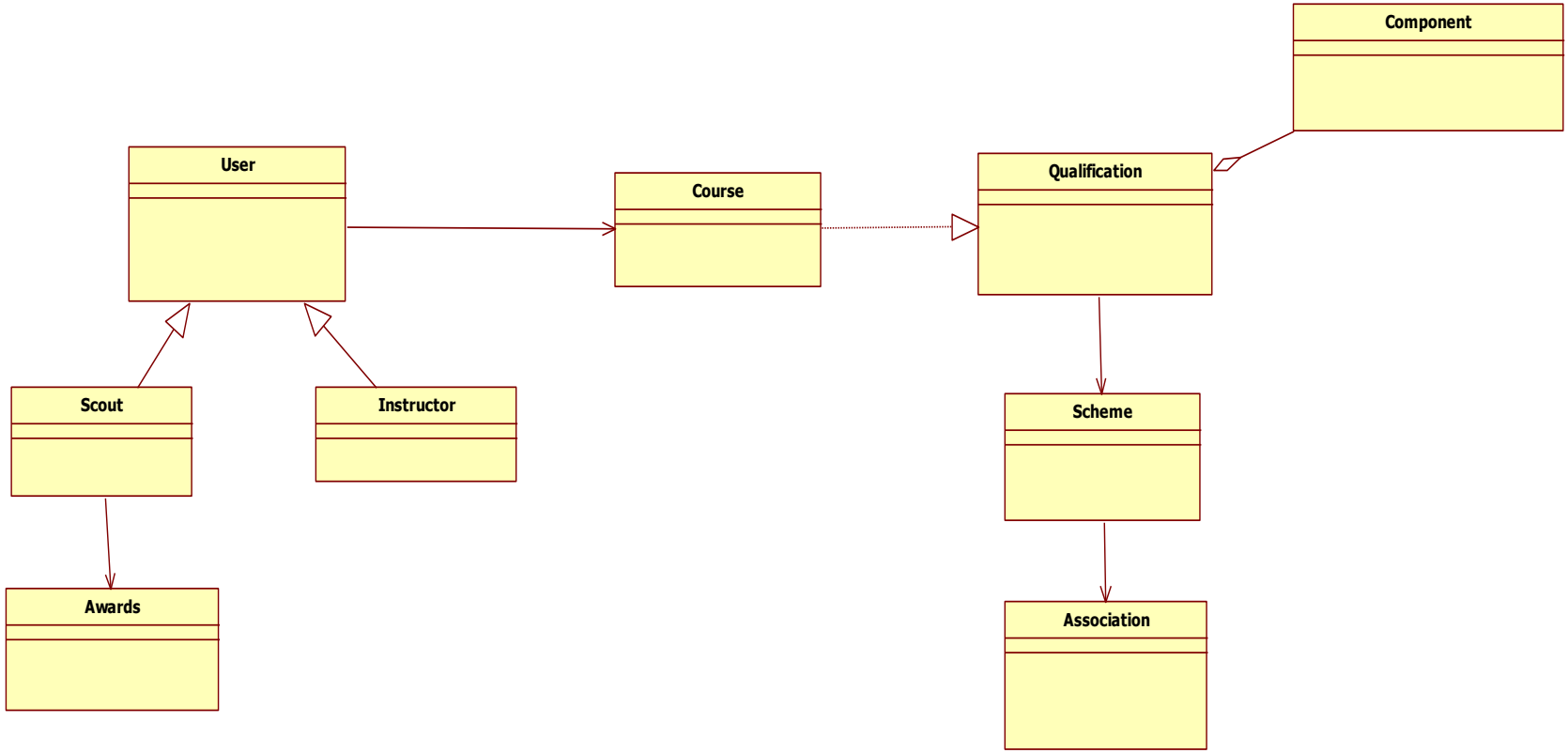


Figure 1: Class Diagram RYA Information System

4.2.3 RYA Database

This is the final class diagram that has every component a conceptual model needs, which is then converted to a relational model. This is the logic used for the database and every class shown will be in the database along with the attributes being the column headers for the database. Many revisions were made to this diagram where class were dropped and added as well as attributes altered in the same sense which lead to the final diagram.

4.2.4 Assumptions of RYA Database

- All members of the system will have a login and a user type
- All courses will have at least one instructor and scout before its status can become in progress
- A scout cannot be awarded a qualification unless the scout has all requisite entries in the scout component table.
- A member cannot both enrol on a course as a scout and instructor

4.2.5 Subsystems:

Membership

The Membership subsystem consists of classes alongside attributes showing the user side of the database. It consists of the users, membership for each user and the respective logins for instructors and scouts. The User Type table was created in order to differentiate between each user so that when a user logs into the website, the website will have the functionality to comprehend which user has been logged whether it would be a scout or an instructor which would then limit each user based on the user's status.

The user table consists of scouts and instructors which are two different users that have different privileges in terms of the website. The user table defines a scout and instructor where it displays details of both users which are universal and shared by both users.

Scouts have different privileges to the instructor in which the scout can view progression in a course as well as altering scout personal details. Instructors have more privileges than scouts where instructors can do everything a scout can do as well as being able to add entirely new scouts and remove scouts.

The login and user type table are shown in order to convey the different logins of each user by having a different login for scouts and instructors, by having these tables it will allow the website in order to coincide with the database in which each user will be defined. Defining each user allows restrictions to scouts but allow privileges to instructors in order to differentiate the users and how each user interacts with the website.

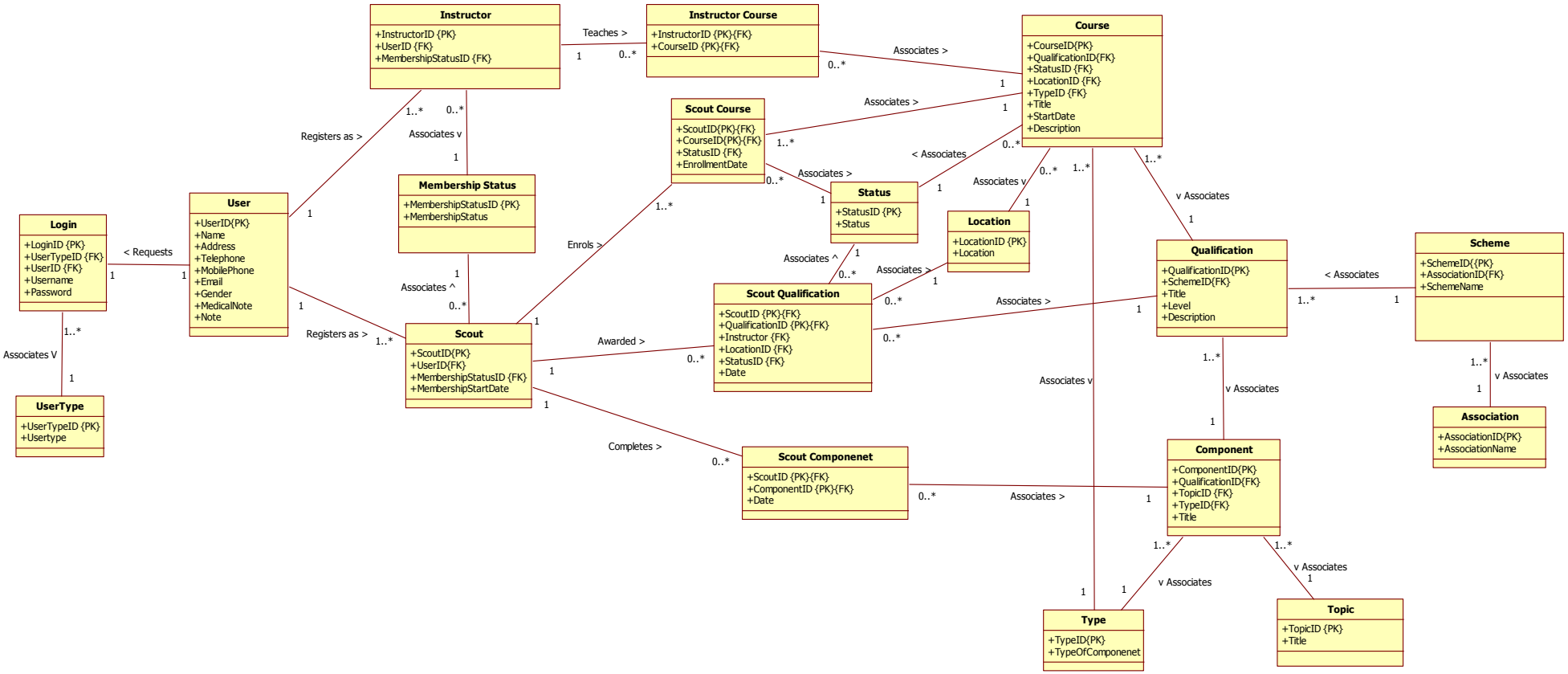


Figure 2: RYA Database

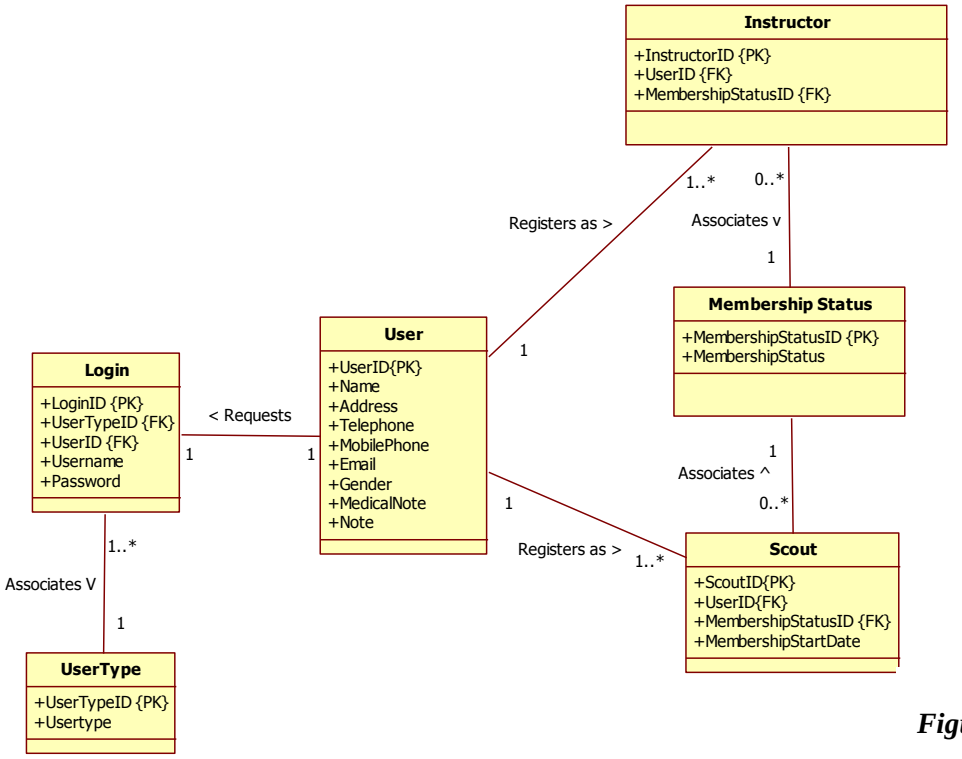

Instructor
+InstructorID {PK}
+UserID {FK}
+MembershipStatusID {FK}
1..*
0..*
Registers as >
Associates v
1
Membership Status
+MembershipStatusID {PK}
+MembershipStatus
User
+UserID{PK}
+Name
+Address
+Telephone
+MobilePhone
+Email
+Gender
+MedicalNote
+Note
1
Login
+LoginID {PK}
+UserTypeID {FK}
+UserID {FK}
+Username
+Password
< Requests
1
1
1
Associates ^
0..*
1
Registers as >
1..*
Scout
+ScoutID{PK}
+UserID{FK}
+MembershipStatusID {FK}
+MembershipStartDate
1..*
Associates V
1
UserType
+UserTypeID {PK}
+Usertype

Figure 3

Qualifications:

The Qualification subsystem consists of classes with finite amount of data which means that the data is never or hardly changing at all so entries would unlikely be added to these tables as the data is fixed.

The qualification table expresses what type of data will be displayed in the database, the qualification table is used in order to show what type of qualifications a scout can achieve through the components the scout has completed in the component table.

The scheme table refers to a set of qualifications created and regulated by an association such as RYA in which the standards are similar to how AQA and OCR work.

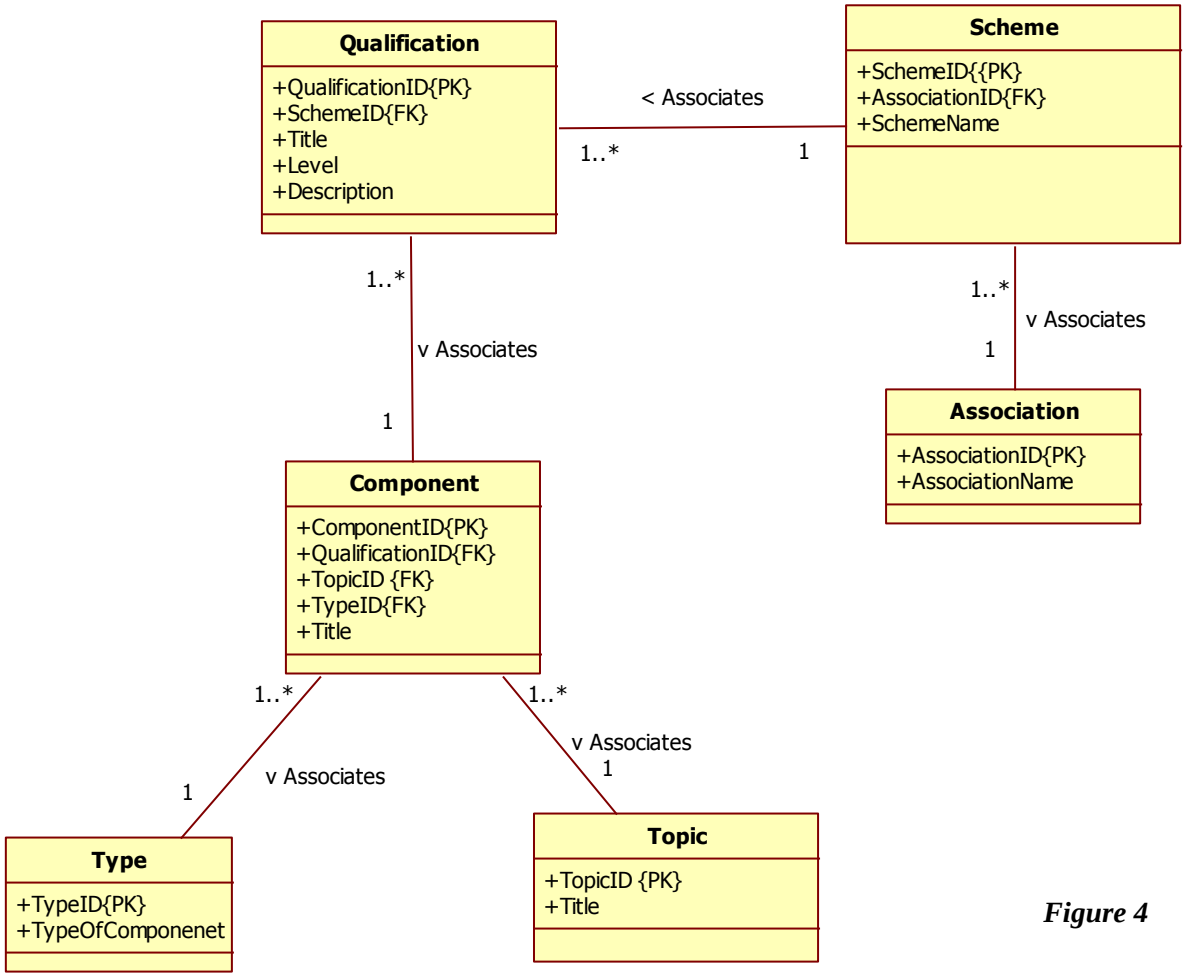


Figure 4

Active Workflows at RYA

The second subsystem includes dynamic tables that are ever changing and continuously adding new entries every time a new scout or an instructor is created therefore being the most dynamic out of all the subsystems.

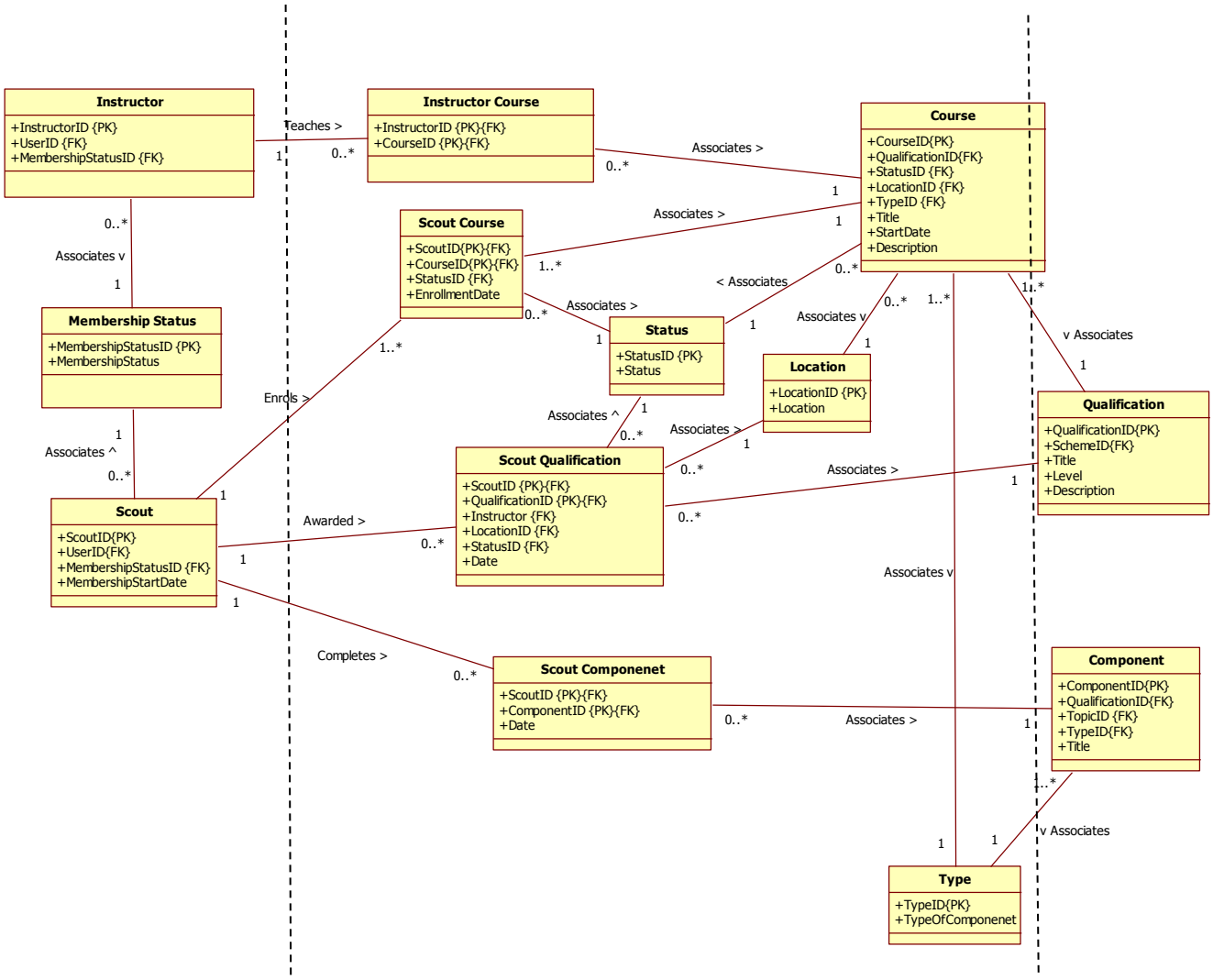


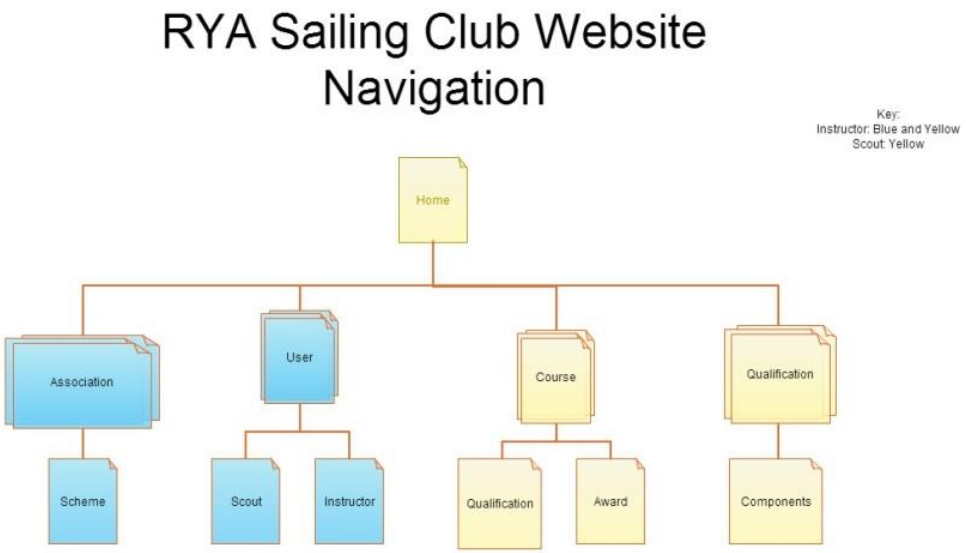
Figure 3 ***Figure 5*** ***Figure 4***

In this diagram it shows how the subsystem interacts, the scout table in figure 3 interacts with the qualification table in figure 4 through the scout qualification table in figure 5. These tables interlink with each other in order for the scout to gain qualifications, every time a scout achieves a qualification the composite table scout qualification will be updated with the relevant data to present that the scout has achieved a qualification. The status table shown in figure 5 is also plays a part which coincides with the three tables stated in the aforementioned in which the status will be stated as completed based on the status ID in scout qualification will enable the instructor to update the records accordingly.

4.3 User interface Design:

4.3.1 Navigation

Navigation is used in order to show the step-by-step process of each page and how the pages are used to proceed seamlessly from one page to another showing a tree like structure



4.4.2 Wiretrames

Introduction

Wireframes are sketched out designs of what the final product may look like showing the structure and layout of the website, it is not the final design but it used to get a rough idea of what it would look like and present the wireframes to the client to make sure that the client agrees to the format.

Low fidelity mock-up

Low fidelity mock ups show a basic design structure of how the final website may look like in the end. The reason for creating low fidelity mock ups is to get a good gist of what the website could look like as well as being able to design the website from scratch whilst having the foundations of the website to fall on. The graphical user interface of the website maintains a consistent design and low fidelity mock ups played a role in being able to having consistency in the design. Widgets were also implemented in the low fidelity mock up stage in which these widgets show the transition between each tab while showing the expansion of the widget, this was made in order to implement widgets in the final website in which having a plan for the website made it easier to implement since the whole structure was there from the beginning of the project.

Global Navigation [Figure 1.0]

(Wireframes can be found in the appendix)

High fidelity mock up

The high fidelity mock ups were created and influenced by the low fidelity mock ups and was used as a basis in order to structure the website. The low fidelity mock ups developed and transitioned into the high fidelity mock ups and most of the design made it into the final website design structure.

K1018112 – Shakir Ahmad

Figure 1.0

Mobile View

RYA	<UserName	Sign-

Navigation Drop Down: User Specific

Navigation: Bread Crumbs (displays

Page Header: States the Function

Page Content:

Will be

Specific to the function of the page

Desktop/Tablet

RYA

<UserName Sign-

Navigation Bar: User Specific

Navigation: Bread Crumbs (displays the current directory tree)

Page Header: States the Function of the Page

Page Content:

Will be

Specific to the function of the page

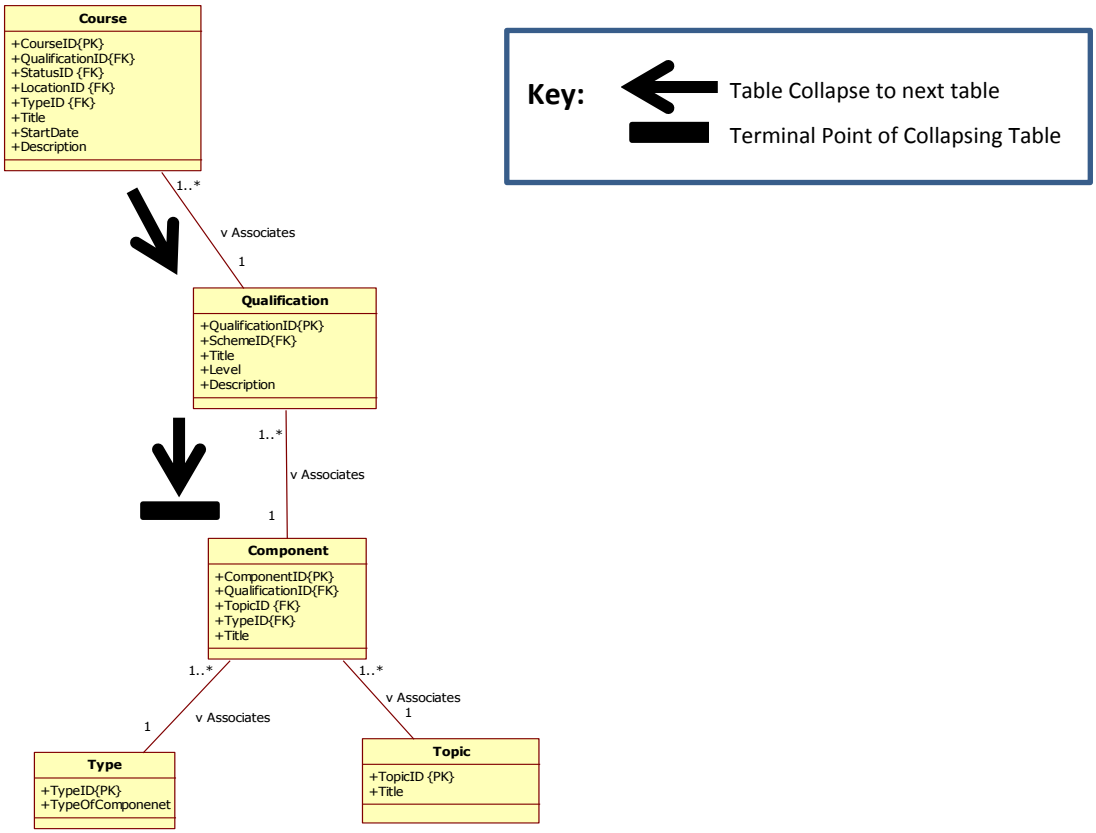
4.6 Design Patterns: Widgets

Introduction

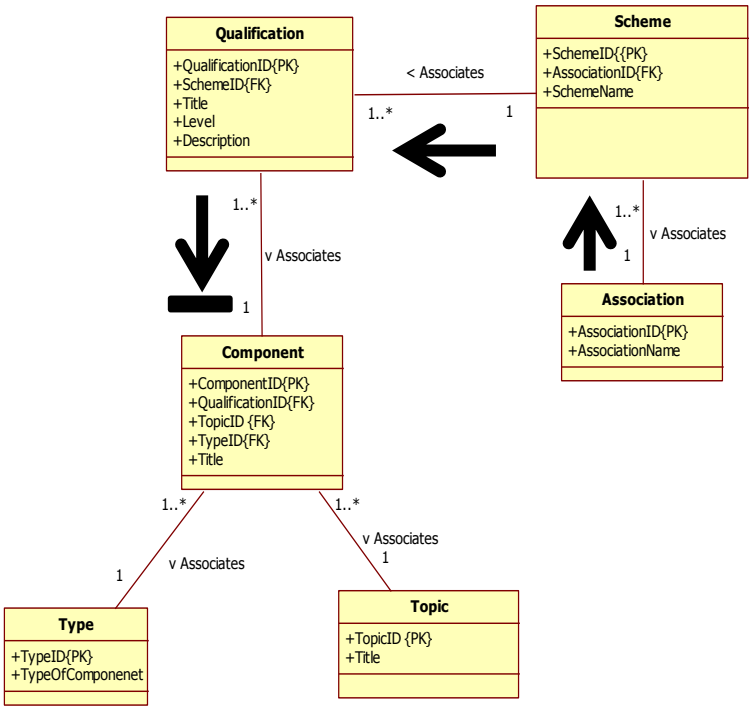
Widgets are a design concept used based on code base JQuery; widgets in terms of the RYA Sailing Club website are used in order generate reports. The reports are generated through cascading tables where table would appear then another table would appear based on the accordion of the table that was clicked. Tabs are also used in-conjunction with cascading tables; tabs are interchangeable buttons that display reports that have the ability to change what is displayed in the table without having to refresh the page otherwise known as AJAX. The reason why widgets were implemented in the website is to benefit the user when viewing records making it a simple and an efficient way of viewing data.

4.6.1 Collapsible Tables

Collapsible tables are shown in this diagram which shows class-to-class of how each table would collapse into different tables when viewing the website, for example the Course table in website view would display all the data within the Course table then it would breakdown into another table called Qualification which would then display all the data for Qualification and the process would go on for each collapsible table.



UML Modelling of Collapsible table for Course



UML Modelling of Collapsible table for Course

Example Wireframe for collapsible table

Adaptable to mobile view

The widgets system is adaptable to both mobile and desktop platforms where collapsible tables are used in essence of the website, when clicking on an accordion it leads to another set of data based on what heading of the accordion was clicked. It adapts to mobile view as it would display the accordion on the whole screen of whatever mobile is in use allowing for human-readable content to be displayed without any problems

Consistent Design Theme

The widgets system allows for a consistent design showing that all the different web pages having the same theme as well as its own accordions specific to each page. Having a consistent theme showcases the efficiency of widgets therefore having a professional look to the website.

Easily Extendable to other UML modelling tables

Widgets can be used and applied to other subsystems where it allows as stated in the aforementioned for a consistent design, this system is easily transferable to other elements of the website allowing the user to have a feel of how consistent the web pages are. This allows the user to interact with the interface with ease and allows the user to remember how the website works since all the web pages are consistent and function in the same way.

K1018112 – Shakir Ahmad

Example Wireframe of collapsible Tables for Associations

Mobile View

[Association Name] (Drop down clicked)	
[Scheme Name] (Drop down clicked)	
[Qualification Name] (Drop down clicked)	
[Components]	
[TYPE: Practical]	
[TOPIC]	[COMPONENT] [COMPONENT]
[TOPIC]	[COMPONENT] [COMPONENT] [COMPONENT]
[TYPE: Theory]	
[TOPIC]	[COMPONENT] [COMPONENT]
[Association Name]	

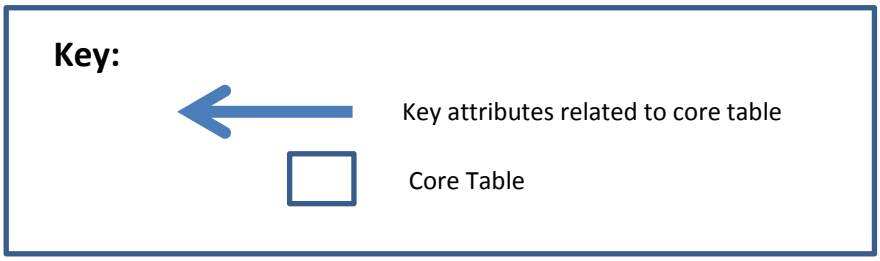
Desktop/Tablet View

Immediate Sense of Context

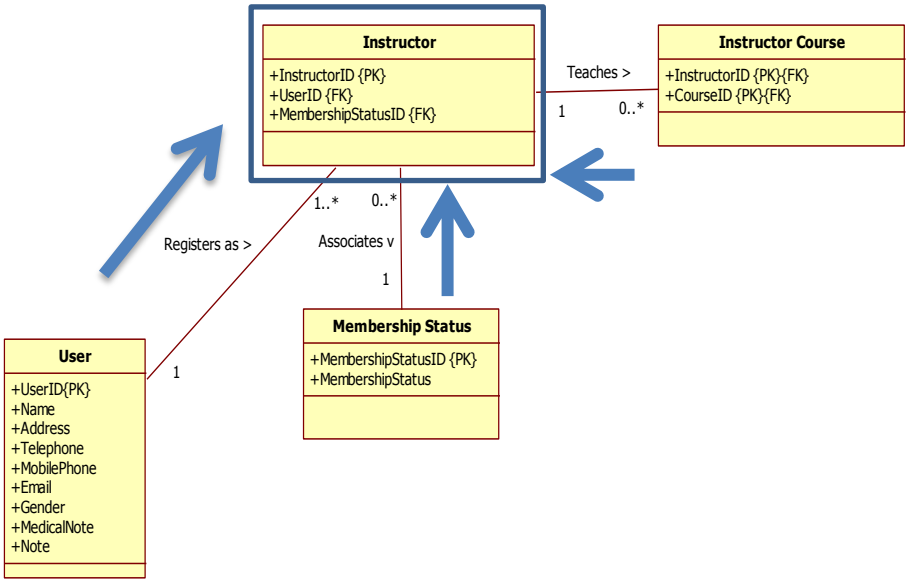
[Association Name] (Drop down clicked)	
[Scheme Name] (Drop down clicked)	
[Qualification Name] (Drop down clicked)	
[Components]	
[TYPE: Practical]	
[TOPIC]	[COMPONENT] [COMPONENT]
[TOPIC]	[COMPONENT] [COMPONENT] [COMPONENT]
[TYPE: Theory]	
[TOPIC]	[COMPONENT] [COMPONENT]
[Association Name]	

4.6.2 Quick Glance View

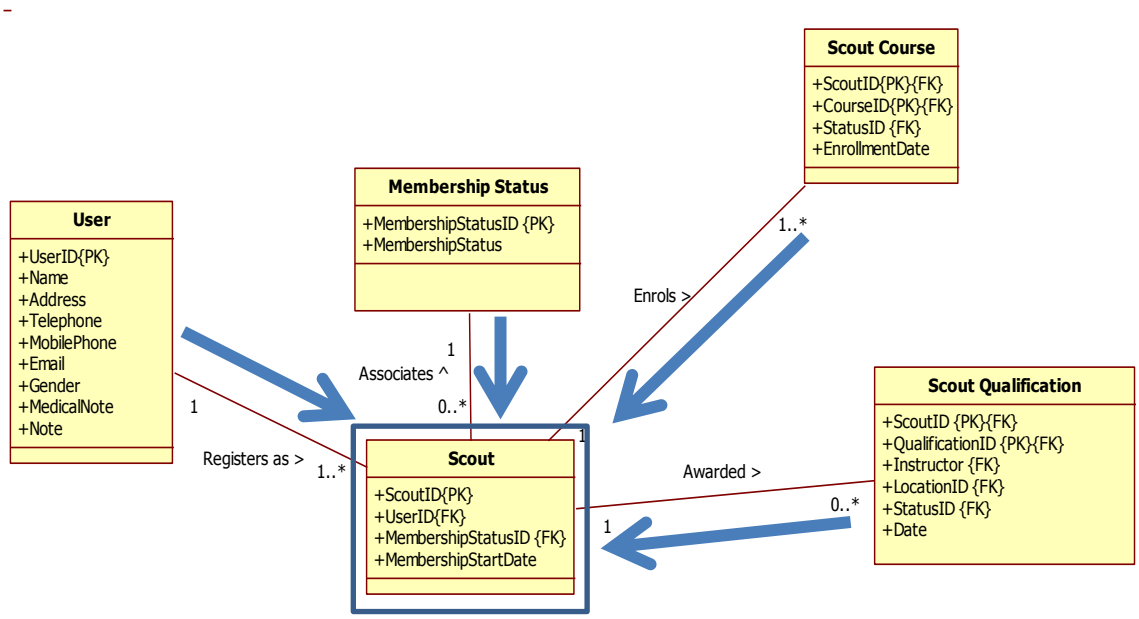
Quick glance in terms of the website is used display records where if an instructor were to click on a scouts record, that scout's record would appear in a collapsible table and every time an instructor would click on a new scout record another collapsible table would appear being on one page without having to refresh the page which is the essence of why widgets were used in this project.



UML Modelling of Quick Glance View table for Instructor



UML Modelling of Quick Glance View table for Scout



Example Wireframe for Quick Glance view

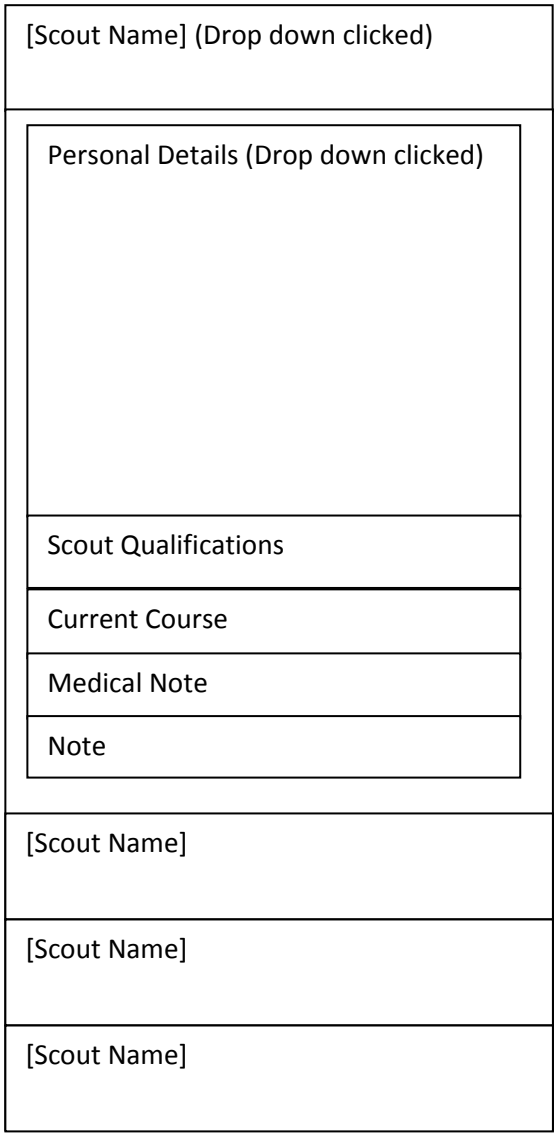
Adaptable to mobile and desktop/tablet view

Accordions are used for quick glance as it displays an eye-catching interface that allows for an immediate view for specific data based on the accordion that was used. This type of interface makes use of the space, more specifically to mobile users where data is manipulated through each accordion and can be displayed in different ways based on the specific accordion. It also adapts to the screen size where the data would fit the whole screen based on the each corresponding accordion that was used, where once the accordion was clicked on by the user the screen would shapeshift to display different subsets of data about the entity which in this example are the attributes related to Scouts. This widget like the collapsible table widget is also extendable and can be easily applied to other entities such as Instructors whilst maintaining a consistent design.

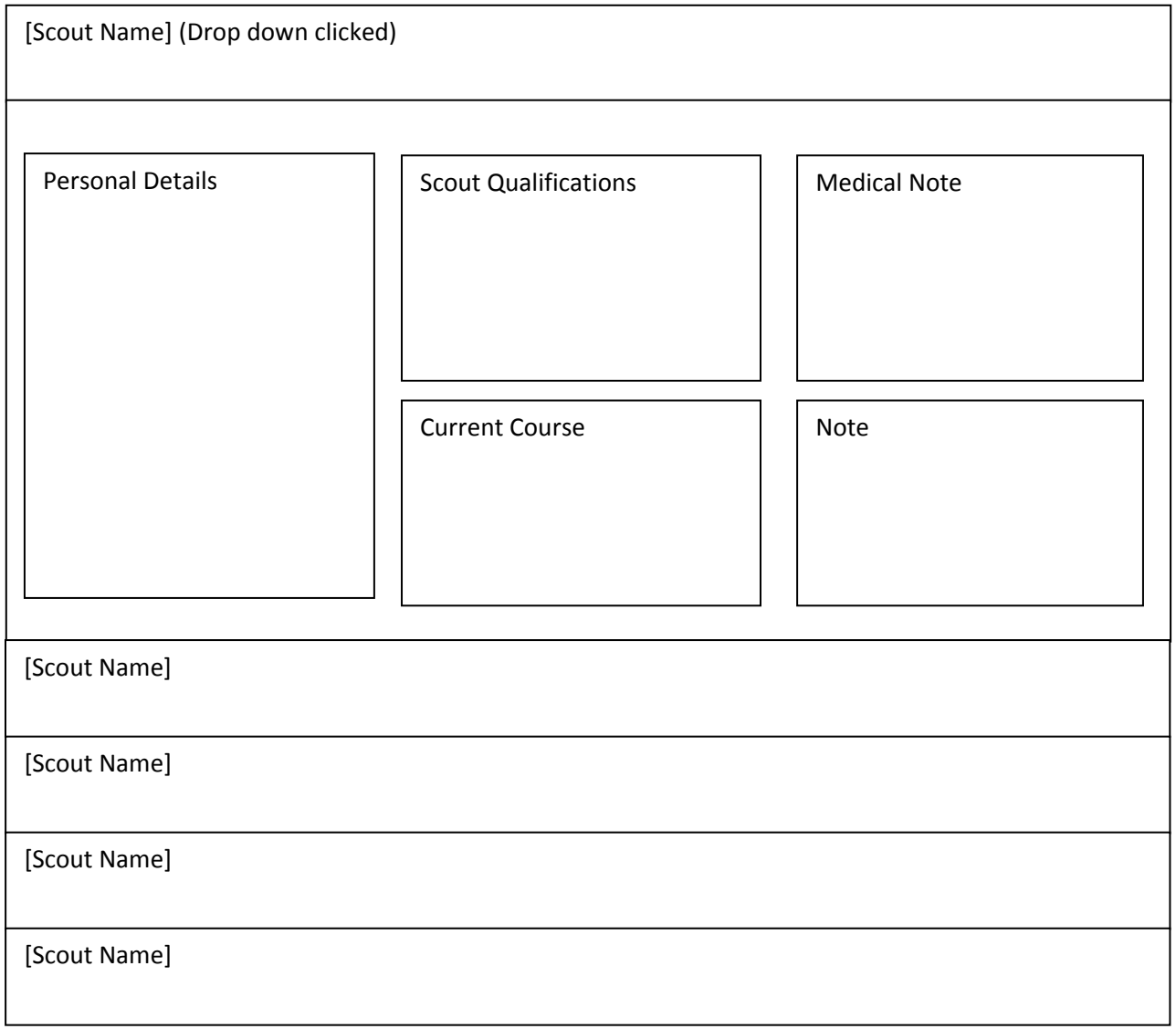
K1018112 – Shakir Ahmad

Example Wireframe of quick glance view for Scout

Mobile View



Desktop/Tablet View



Conclusion

In conclusion the design chapter is a key asset and is used in unison with implementation in order to garner a good idea of what kind of elements must be implemented within the project. The design of the project is main bulk of this project, getting the plan right is crucial when it comes to implementing the website and the database as having no basis to stand upon can be a risk to the project. The design chapter displayed diagrams with explanations of each diagram such as sub systems for the database design and widgets for the website design, this is in order to grasp how the implementation side of the database and the website will be applied by having a basis to fall back on by having consistency in the design stage.

5 Implementation Chapter

Introduction

The implementation chapter explains what kind of platforms and technologies were used and what kind of downfalls were to arise whilst implementing in this project. The chapter will discuss how all the technologies were used in unison and how the technologies mesh together to form a website that has data pulled from a database in order to display records for scouts and instructors.

5.1 Technologies implemented

5.1.1 Potential Platforms (Technologies):

5.1.2 Interface: HTML, CSS Bootstrap

HTML is a language that is used as an interface for websites which allows for the application of the visual aspect of websites. It is used in-conjunction with other languages such as PHP, CSS, CSS Bootstrap and JavaScript in order to have all the elements of a website based on the languages that are used in unification.

HTML was used as part of the RYA Sailing Club website where HTML was applied in the sense of the navigation bar of the website in order to display the navigation bar at the top of the website, allowing users of the website scouts and instructors a like to navigate through different pages whilst retaining the navigation bar on each page by calling a function in PHP called ‘include’ followed by the navigation bar file.

CSS Bootstrap is used in order to display the aesthetics of the website which is the design of how the website will look. Bootstrap is applied to all pages similarly to how HTML calls an ‘include’ function but implemented differently through the use of an attribute. Different types of Bootstrap can be implemented based on what type of Bootstrap is used which applies a design to the website which gives the website a consistent and professional look to the website.

CSS Bootstrap was applied to the RYA website where it was used in unison with HTML in order to output the visuals of the website which was applied to each page giving consistency. The reason why Bootstrap was implemented is because it is more efficient to use than a CSS style sheet. CSS style sheets have to be changed constantly and then reapplied to each page, where it may appear in a way where the style sheets are not aligned in terms of aesthetics where Bootstrap rectifies this issue and it is a simplified version of CSS style sheet where if one component is changed in Bootstrap then all the components for each page will be changed which is where the consistent and professional look comes in to play.

5.1.3 Interchange language: PHP

PHP is the code behind the website that provides functionality of the website which is used in unison with HTML where it is applied in the body of HTML. PHP is used to provide

functionality such as clicking buttons otherwise known as ‘onclick’ that can lead to another page, retrieve data from a database via a connect function to the database to output reports based on data from databases and being able to call if functions.

PHP in terms of the RYA Sailing Club website was used to create an interactive navigation bar which allows users to seamlessly access web pages based on the page that user would want to access via ‘onclick’. PHP was also used to create the dynamics of each page for example if an instructor wanted to check a scout’s record and modify its components then PHP would be applied to this situation. In order for this to be made possible an SQL query was used in synch with a form tag in order for the modifications of the scout’s records to be altered by the instructor making PHP the main bulk of the website being the most important asset in this project.

5.1.4 Database platform: MySQL

A database is an application that allows tables in association with attributes to be created that stores data within each table based on the attribute’s name or in layman’s terms columns. For example if there was a ‘Customer’ table that had the attribute ‘First Name’ then all the data under this attribute would have the list of customer first names. The reason why databases are used is to store large amounts of data as well as using a digital form of keeping records rather than a paper-based form, which is what the project is mainly based on.

MySQL was the database platform that was used for the project and it was used in synchronisation with the website in order for the website to pull data from the database, allowing the website to display all the data which is based on the specific SQL queries that were made in order to output results to the web page.

5.2 Development Tools

Notepad++

Notepad++ was used to construct the website by applying all the languages on one program, the reason why Notepad++ was used was in order to colour code the elements such as HTML in blue and PHP in red in order to differentiate between the different languages to keep on track.

NetBeans

NetBeans is a program for coding where it allows programmers to run source code to a console and being able to find errors in code making coding a whole lot easier, in this instance it was used for other languages such as HTML, PHP and JavaScript allowing for the ability to automatically find erroneous code within the whole code.

NetBeans was also used to construct the website but for a different reason, the reason being so that error codes would be displayed when there was a mistake in the code showing a specific error code on a specific line. Having these attributes of being able to see the errors would make for easy amendments to the code and saving what would have been a time-consuming objective in to a fast pace debugging process backed by NetBeans.

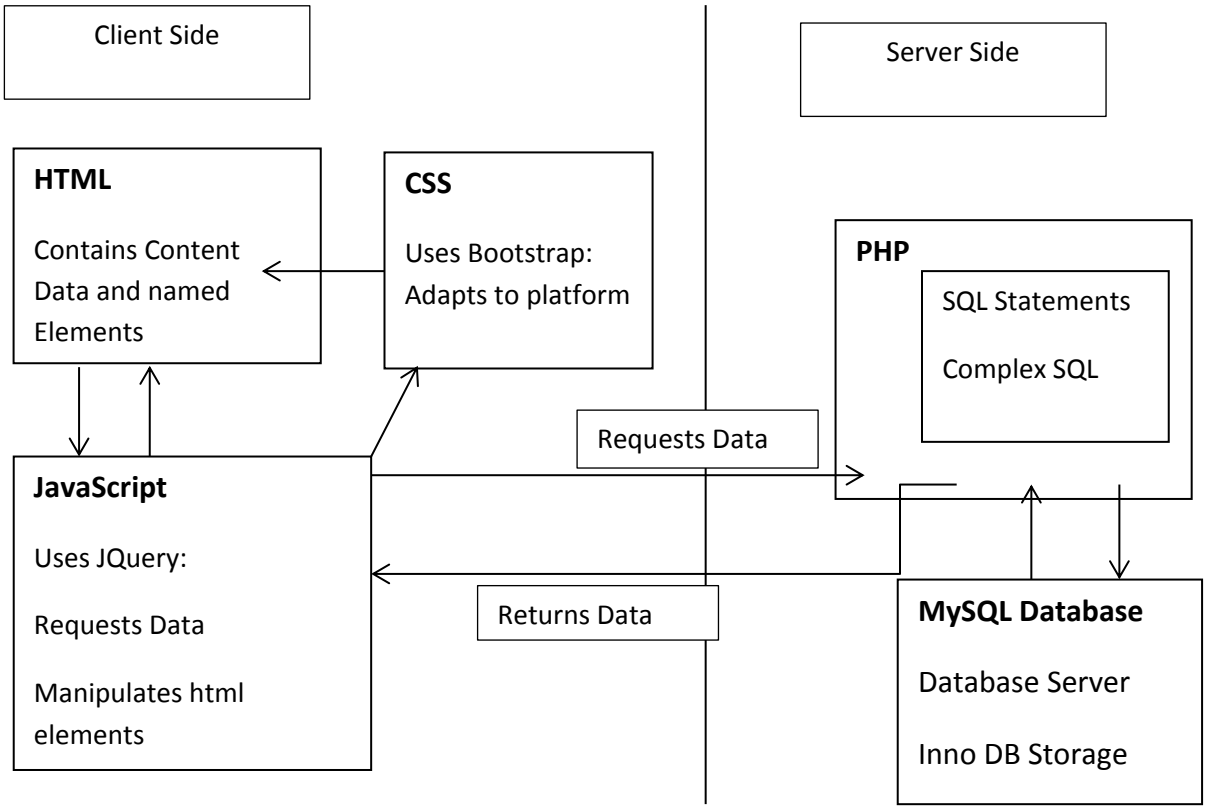
FileZilla

FileZilla is a server protocol program that allows transferring of files onto a server which is used in essence of creating the website, it is used to upload PHP files to the server in order for the website to read and interpret the code and output a result based in HTML.

5.3 Architecture Infrastructure

5.3.1 Client side and Server side technology and Architecture

The website architecture for how the website works using different technologies where client side is shown on the left and server side is shown on the right. The diagram shows how these technologies interlink with each other where all the technologies on the client side is done through the browser and server side is technologies and files stored on the server.



5.2 Database

5.2.1 Complex SQL Command

Complex SQL statements were used to show the consistency of the database in order to display that the database is coherent and is able to be unified with each table showing a consistent structure of data. For example if there were scouts that were enrolled on a course,

the database would show all the scouts enrolled for that course corresponding to the amount of scouts there were in the Scout table. Depending on the course it would show a specific list of scouts that passed a course such as stage two scouts would be shown if the scouts completed stage one but if any scouts failed those scouts would still be in stage one and this would be displayed on the database showing that the database has ingenuity.

The complex SQL statements were also used in unison with PHP were PHP served as the function and the complex SQL statements were applied to the function of PHP for example PHP was used to create the entry on a button click where it retrieved the necessary data from the complex SQL statements in order for the report to generate a new entry based on the tables and attributes of the database.

5.2.2 Security

5.2.3 Scope and Limitations of implementations

When implementing the database the main focus at the beginning of the creation process of the database is the logistics. Having sound logic for the database is up-most priority as it can influence how the database works and reacts to sensitive data as well as preventing mistakes within the database such as having a specific value for data for example having a scouts name as ten characters long in the database but then having a name that consists of eleven characters which exceeds the ten character count for that data type, so the database was well thought out at the beginning prior to making the database in order for these errors not to occur within the database.

Other issues include having the correct foreign key that corresponds with the correct table in order for the relationship of those tables to be established which key in retaining the database's consistency as well as being able to retrieve data relative to the relationship that was made with those tables so that inner joins would be implemented through complex SQL statements which allows for data to be pulled out of any table in order to output the data on the website.

5.4 Website

5.4.1 User interface

Forms and Reports

5.4.2 Security

A login system was implemented based on the client's requirements for the RYA sailing website as it adds a layer of security so that scout could not login as an instructor and vice versa as well as keeping data confidential for the instructor. The login system is also used for differentiating between a scout and an instructor by giving privileges to certain users using the website. For example a scout cannot access any part of the database nor can the scout

alter the database in any way but is limited in only viewing the specific scout's account which consists of what course the scout is enrolled on and what awards the scout has. The instructor however has more administrative rights over the scout as the instructor is able to alter the database in a way that is able to add, alter and delete scout records within a certain parameter but is not able to directly access the database as that is held only for the client.

5.5 Development Phase

The development phase shows proof of concept that the database and the RYA Sailing Club website were created with screen dumps that were taken whilst implementing both the database and the website.

Database Development Phase:

Figure 1

In figure 1 it shows that data that was imported from a master data population method that was created in excel. The query proved to be successful for all the tables and this is one of the examples shown displaying that all the data that was generated in the master data population table was applied to MySQL with minimal problems.

Figure 2

Figure 2 shows the master data population that was created in excel which were used in conjunction with the application shown in figure 1.

Figure 1

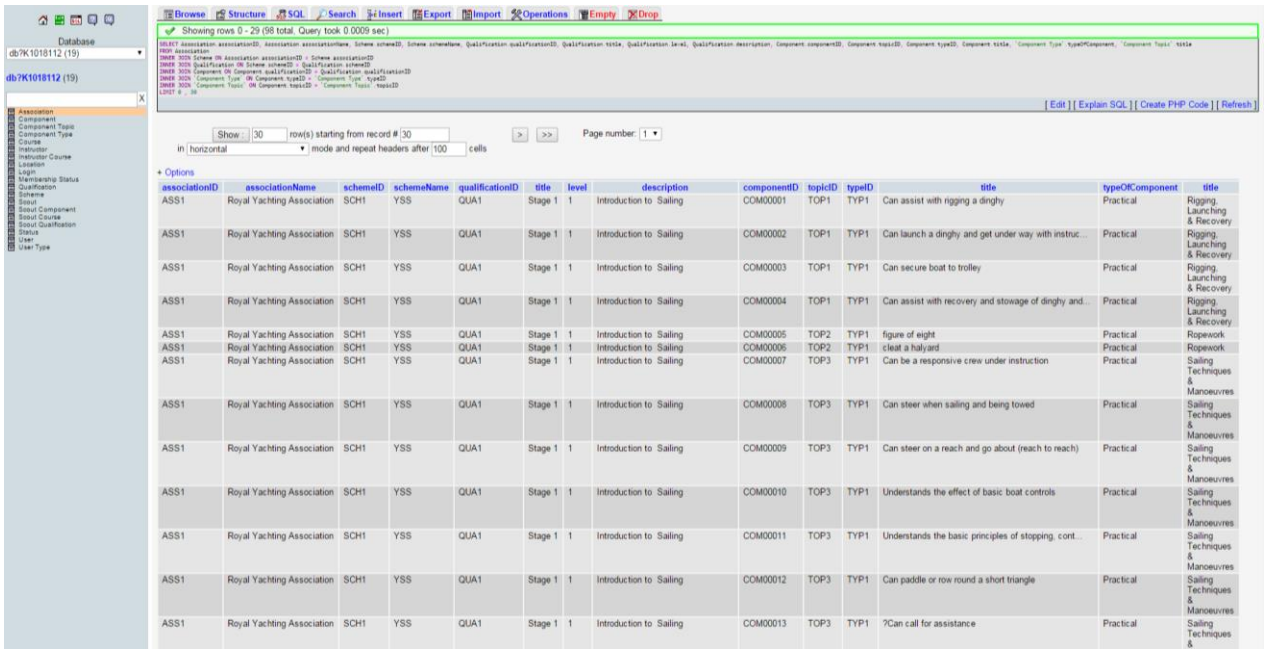
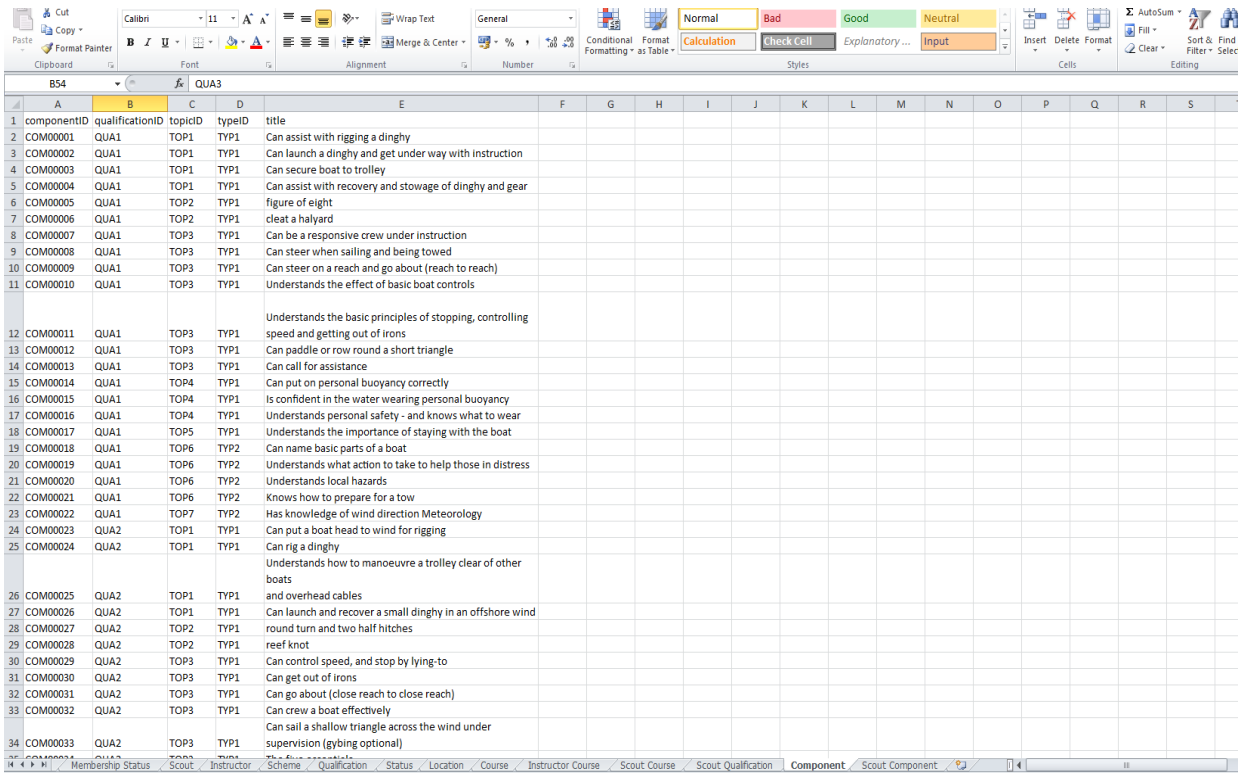


Figure 2



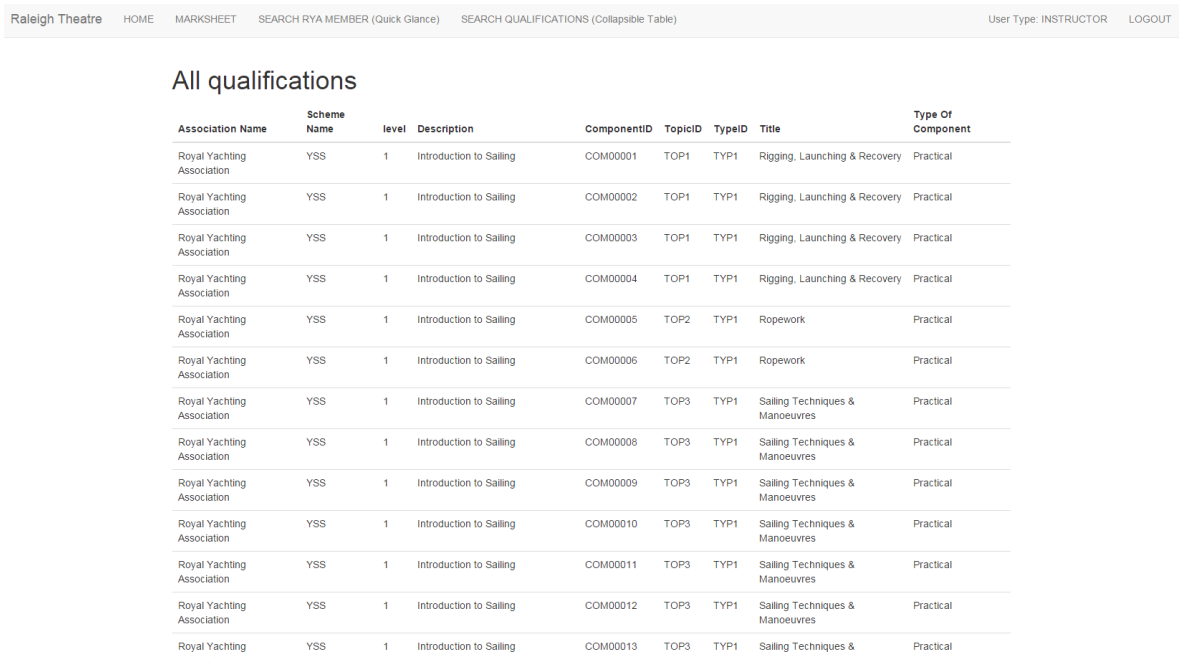
componentID	qualificationID	topicID	typeID	title
COM00001	QUA1	TOP1	TYP1	Can assist with rigging a dinghy
COM00002	QUA1	TOP1	TYP1	Can launch a dinghy and get under way with instruction
COM00003	QUA1	TOP1	TYP1	Can secure boat to trolley
COM00004	QUA1	TOP1	TYP1	Can assist with recovery and stowage of dinghy and gear
COM00005	QUA1	TOP2	TYP1	figure of eight
COM00006	QUA1	TOP2	TYP1	cleat a halyard
COM00007	QUA1	TOP3	TYP1	Can be a responsive crew under instruction
COM00008	QUA1	TOP3	TYP1	Can steer when sailing and being towed
COM00009	QUA1	TOP3	TYP1	Can steer on a reach and go about (reach to reach)
COM00010	QUA1	TOP3	TYP1	Understands the effect of basic boat controls
COM00011	QUA1	TOP3	TYP1	Understands the basic principles of stopping, controlling speed and getting out of irons
COM00012	QUA1	TOP3	TYP1	Can paddle or row round a short triangle
COM00013	QUA1	TOP3	TYP1	Can call for assistance
COM00014	QUA1	TOP4	TYP1	Can put on personal buoyancy correctly
COM00015	QUA1	TOP4	TYP1	Is confident in the water wearing personal buoyancy
COM00016	QUA1	TOP4	TYP1	Understands personal safety - and knows what to wear
COM00017	QUA1	TOP5	TYP1	Understands the importance of staying with the boat
COM00018	QUA1	TOP6	TYP2	Can name basic parts of a boat
COM00019	QUA1	TOP6	TYP2	Understands what action to take to help those in distress
COM00020	QUA1	TOP6	TYP2	Understands local hazards
COM00021	QUA1	TOP6	TYP2	Knows how to prepare for a tow
COM00022	QUA1	TOP7	TYP2	Has knowledge of wind direction Meteorology
COM00023	QUA2	TOP1	TYP1	Can put a boat head to wind for rigging
COM00024	QUA2	TOP1	TYP1	Can rig a dinghy
COM00025	QUA2	TOP1	TYP1	Understands how to manoeuvre a trolley clear of other boats and overhead cables
COM00026	QUA2	TOP1	TYP1	Can launch and recover a small dinghy in an offshore wind
COM00027	QUA2	TOP2	TYP1	round turn and two half hitches
COM00028	QUA2	TOP2	TYP1	reef knot
COM00029	QUA2	TOP3	TYP1	Can control speed, and stop by lying-to
COM00030	QUA2	TOP3	TYP1	Can get out of irons
COM00031	QUA2	TOP3	TYP1	Can go about (close reach to close reach)
COM00032	QUA2	TOP3	TYP1	Can crew a boat effectively
COM00033	QUA2	TOP3	TYP1	Can sail a shallow triangle across the wind under supervision (gybing optional)

Website Development Phase:

Figure 1

In the screen dump taken below in figure 1, shows the collapsible table concept that was applied to the website showing courses with its associations, qualifications and the components that must be completed in order for a scout to gain a qualification. The database was used in unison with the website as displayed below by allowing a MySQL query from the database to be shown on the website. It also shows which user is logged in by user type which was pulled from the user type table.

Figure 1



Association Name	Scheme Name	level	Description	ComponentID	TopicID	TypeID	Title	Type Of Component
Royal Yachting Association	YSS	1	Introduction to Sailing	COM00001	TOP1	TYP1	Rigging, Launching & Recovery	Practical
Royal Yachting Association	YSS	1	Introduction to Sailing	COM00002	TOP1	TYP1	Rigging, Launching & Recovery	Practical
Royal Yachting Association	YSS	1	Introduction to Sailing	COM00003	TOP1	TYP1	Rigging, Launching & Recovery	Practical
Royal Yachting Association	YSS	1	Introduction to Sailing	COM00004	TOP1	TYP1	Rigging, Launching & Recovery	Practical
Royal Yachting Association	YSS	1	Introduction to Sailing	COM00005	TOP2	TYP1	Ropework	Practical
Royal Yachting Association	YSS	1	Introduction to Sailing	COM00006	TOP2	TYP1	Ropework	Practical
Royal Yachting Association	YSS	1	Introduction to Sailing	COM00007	TOP3	TYP1	Sailing Techniques & Manoeuvres	Practical
Royal Yachting Association	YSS	1	Introduction to Sailing	COM00008	TOP3	TYP1	Sailing Techniques & Manoeuvres	Practical
Royal Yachting Association	YSS	1	Introduction to Sailing	COM00009	TOP3	TYP1	Sailing Techniques & Manoeuvres	Practical
Royal Yachting Association	YSS	1	Introduction to Sailing	COM00010	TOP3	TYP1	Sailing Techniques & Manoeuvres	Practical
Royal Yachting Association	YSS	1	Introduction to Sailing	COM00011	TOP3	TYP1	Sailing Techniques & Manoeuvres	Practical
Royal Yachting Association	YSS	1	Introduction to Sailing	COM00012	TOP3	TYP1	Sailing Techniques & Manoeuvres	Practical
Royal Yachting	YSS	1	Introduction to Sailing	COM00013	TOP3	TYP1	Sailing Techniques &	Practical

Figure 2

In figure 2 it shows the login page that is universal for each user but has different access rights depending on the user that is logged in. The login is interlinked with the database table login and user type. The login page will only let scouts, instructors and course planners log in to the system with certain restrictions on scouts but different features of instructors.

Figure 2

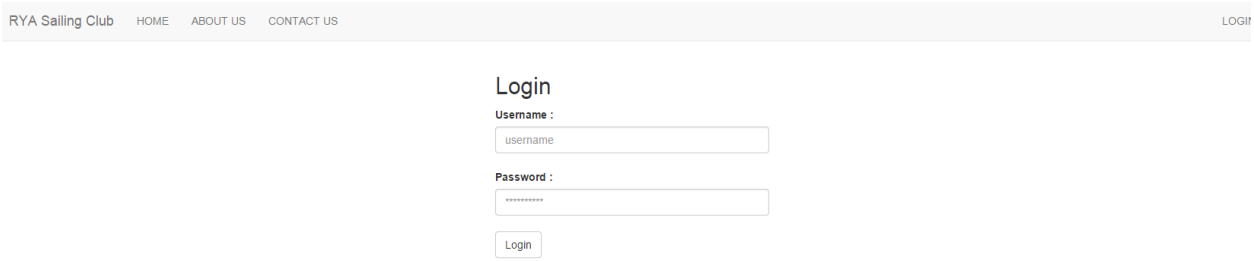
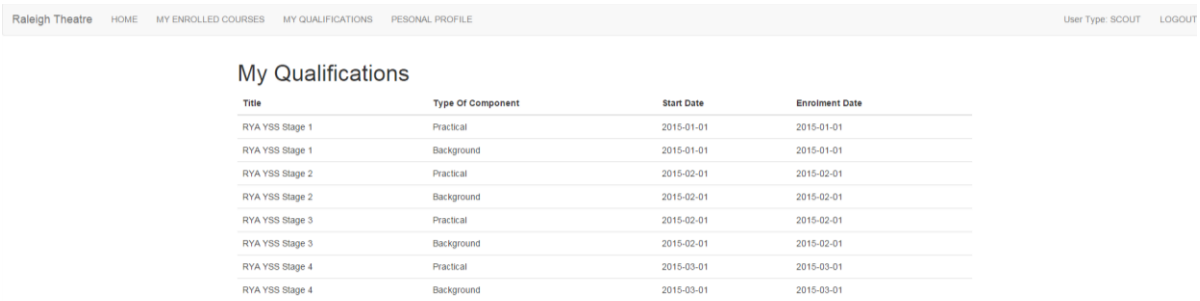


Figure 3



Conclusion

To conclude with this chapter, implementation based on the languages that were used showed the issues that arose while implementing the website and the database. The chapter explained how each language was used in order to implement the website as well as how each language was used in unison together in order to achieve an end result which was the final website and database.

6 Testing and Evaluation

Introduction

Extensive testing was made throughout the project in order to ensure that the website is fully functioning and that it can handle any dynamic interaction the user makes with the website to mesh together without any flaws as possible.

6.1 Database Testing:

Testing the database plays an important role as without the database the website would not have any data to pull from the database so testing the database is key. In order to test the database, SQL queries were made in order to be able to show the width of how far the database extends and to test whether the SQL query outputs the correct results. An example of using a SQL is to use a series of inner joins to show whether the database can handle the queries, by checking if all foreign keys are in place that would allow the SQL query to output a result to show the correct relationships between each table. Testing was also applied to the data by setting commands such as a ‘where’ and a ‘between’ clause to a two specific dates to see whether the query would output all the dates within the threshold of two dates. This allows showing whether the data was correct within the database to minimalize the amount flaws as possible within the database, using SQL queries is a good method in testing the database as it can test the limitations and scope of the database for the desired results.

6.2 Website Testing:

In order to test the website, debugging and navigating through the website was employed which allowed the website to have as few errors as possible. Debugging was used in-conjunction with navigating the website as checking the website to see whether it outputs any results on each web page, if it did not output a result then extensive debugging was used in order to pinpoint the errors where it could be a simple syntax error that could have been the problem. Navigating the website and using logic in order to identify the problem based on error codes or simply viewing a blank page or result would help isolate the errors, as it gives a good indication of what the error may be. Commenting out certain code and running the website also was used when testing the website, it shows that what code blocks are working and which are not functioning in order to discover the error it could be where errors could be varying, from a simple missing comma or a semi-colon in the wrong place thereby ending the line of code where it was not intended are some of the examples that was come across whilst testing.

Conclusion

To summarize this chapter, different testing methods were explained in order to gather erroneous data when it came to building the website alongside the database. These methods showed the most efficient way of testing both components of the project in order to gather the desired result and to make this project come to fruition.

7 Critical Review and Conclusion

Introduction

To introduce the critical review and conclusion chapter, this is the final chapter that will summarise the entirety of the report which will discuss what has been accomplished and what could have been done to improve the website functionality.

7.1 Critical Review

The overall scope of the project came to fruition and all the targets that were delivered by the client were reached. However the website could have been improved upon for example the use of procedures could have been implemented in to the website in which the procedure would be able to be activated to ensure that certain constraints were met within the system, for example a course cannot change its status from planned to running without an instructor assigned to teach the course and without a minimum of five scouts enrolled. Another example of procedures in terms of the project could be used in order to set off an automated event that checks whether a scout has completed all the components to be awarded a qualification and if the scout has not completed all the components, then an instructor would not be able to award a scout unless the scout has met all the prerequisites to be awarded with a qualification.

Conclusion

In conclusion to the report, the requirements for the project were met and improvements that could have been made were discussed in this chapter. Different elements could have been implemented in the website such as procedures which were mentioned in the critical review and the varying procedures that could have been implemented for additional functionality.

APPENDIX A: Create table Statements:

```
CREATE TABLE `User Type`
    (
        userTypeID          VARCHAR(8)          ,
        typeName            VARCHAR(50)     NOT NULL,

        PRIMARY KEY (userTypeID)
    )ENGINE = InnoDB ;

CREATE TABLE `Component Type`
    (
        typeID              VARCHAR(8)          ,
        typeOfComponent     VARCHAR(50)         NOT NULL,

        PRIMARY KEY (typeID)
    )ENGINE = InnoDB ;

CREATE TABLE `Component Topic`
    (
        topicID                 VARCHAR(8)          ,
        title               VARCHAR(100)    NOT NULL,

        PRIMARY KEY (topicID)
    )ENGINE = InnoDB ;

CREATE TABLE `Association`
    (
```

```
        associationID           VARCHAR(8)          ,
        associationName         VARCHAR(50)         NOT
NULL,

        PRIMARY KEY (associationID)
    )ENGINE = InnoDB ;

CREATE TABLE User
    (
        userID          VARCHAR(8)          ,
        name            VARCHAR(100)        NOT NULL,
        address         VARCHAR(100)        NOT NULL,
        telephone       VARCHAR(11)         NOT NULL,
        mobilePhone     VARCHAR(11)             NOT NULL,
        email           VARCHAR(50)             NOT NULL,
        gender          VARCHAR(1)              NOT NULL,
        medicalNote     VARCHAR(500)        ,
        note            VARCHAR(500)        ,

        PRIMARY KEY (userID)

    )ENGINE = InnoDB ;

CREATE TABLE `Login`
    (
        loginID             VARCHAR(8)          ,
        userTypeID          VARCHAR(8)          NOT NULL,
```

```
        userID              VARCHAR(8)          NOT NULL,
        username            VARCHAR(50)     NOT NULL,
        password            VARCHAR(50)     NOT NULL,

        PRIMARY KEY (loginID, userTypeID),
        FOREIGN KEY (userTypeID)        REFERENCES `User Type`(userTypeID),
        FOREIGN KEY (userID)        REFERENCES `User`(userID)
    )ENGINE = InnoDB ;

CREATE TABLE `Membership Status`
    (
        membershipID                VARCHAR(8)
    ,
        membershipStatus            VARCHAR(50)
    NOT NULL,

        PRIMARY KEY (membershipID)
    )ENGINE = InnoDB ;

CREATE TABLE Scout
    (
        scoutID                     VARCHAR(8)          ,
        userID                  VARCHAR(8)          NOT NULL,
        membershipID        VARCHAR(8)          NOT NULL,
        membershipStartDate DATE                NOT NULL,
```

```
        PRIMARY KEY (scoutID),
        FOREIGN KEY (userID)            REFERENCES User(userID),
        FOREIGN KEY (membershipID)      REFERENCES `Membership
Status`(membershipID)
    )ENGINE = InnoDB ;

CREATE TABLE Instructor
    (
        instructorID        VARCHAR(8)      NOT NULL,
        userID              VARCHAR(8)      NOT
NULL,
        membershipID        VARCHAR(8)      NOT NULL,

        PRIMARY KEY (instructorID),
        FOREIGN KEY (userID)            REFERENCES User(userID),
        FOREIGN KEY (membershipID)      REFERENCES `Membership
Status`(membershipID)
    )ENGINE = InnoDB ;

CREATE TABLE Scheme
    (
        schemeID            VARCHAR(8)      NOT
NULL,
        associationID       VARCHAR(8)      NOT NULL,
        schemeName          VARCHAR(20)     NOT NULL,

        PRIMARY KEY (schemeID),
        FOREIGN KEY (associationID)     REFERENCES
Association(associationID)
```

```
    )ENGINE = InnoDB ;


CREATE TABLE Qualification
    (
        qualificationID         VARCHAR(8)          NOT
NULL,
        schemeID                VARCHAR(8)          NOT
NULL,
        title                   VARCHAR(20)         NOT
NULL,
        level                   VARCHAR(20)     NOT NULL,
        description             VARCHAR(500)    NOT NULL,

        PRIMARY KEY (qualificationID),
        FOREIGN KEY (schemeID)      REFERENCES Scheme(schemeID)
    )ENGINE = InnoDB ;


CREATE TABLE `Status`
    (
        statusID                VARCHAR(8)                  ,
        status                  VARCHAR(50)         NOT
NULL,

        PRIMARY KEY (statusID)
    )ENGINE = InnoDB ;


CREATE TABLE `Location`
```

```
(
        locationID          VARCHAR(8)          ,
        location            VARCHAR(100)        NOT
NULL,

        PRIMARY KEY (locationID)
)ENGINE = InnoDB ;


CREATE TABLE `Course`
(
        courseID            VARCHAR(8)          ,
        qualificationID     VARCHAR(8)          NOT
NULL,
        statusID            VARCHAR(8)          NOT
NULL,
        locationID          VARCHAR(8)          NOT
NULL,
        typeID              VARCHAR(8)          NOT
NULL,
        title               VARCHAR(200)    NOT NULL,
        startDate           Date            NOT NULL,
        description         VARCHAR(500)    NOT NULL,

        PRIMARY KEY (courseID),
        FOREIGN KEY (qualificationID)       REFERENCES
`Qualification`(qualificationID),
        FOREIGN KEY (statusID)              REFERENCES
`Status`(statusID),
```

```
        FOREIGN KEY (locationID)        REFERENCES
`Location`(locationID),
        FOREIGN KEY (typeID)                REFERENCES
`Component Type`(typeID)

    )ENGINE = InnoDB ;

CREATE TABLE `Instructor Course`
    (
        instructorID        VARCHAR(8)        NOT NULL,
        courseID            VARCHAR(8)        NOT
NULL,

        PRIMARY KEY (instructorID, courseID),
        FOREIGN KEY (instructorID)  REFERENCES Instructor(instructorID),
        FOREIGN KEY (courseID)      REFERENCES `Course`(courseID)
    )ENGINE = InnoDB ;

CREATE TABLE `Scout Course`
    (
        scoutID             VARCHAR(8)      NOT NULL,
        courseID            VARCHAR(8)      NOT NULL,
        statusID            VARCHAR(8)      NOT NULL,
        enrolmentDate       Date            NOT NULL,

        PRIMARY KEY (scoutID, CourseID),
        FOREIGN KEY (scoutID)       REFERENCES Scout(scoutID),
```

```
        FOREIGN KEY (courseID)          REFERENCES `Course`(courseID),
        FOREIGN KEY (statusID)          REFERENCES `Status`(statusID)
    )ENGINE = InnoDB ;

CREATE TABLE `Scout Qualification`
    (
        scoutID                         VARCHAR(8)      NOT NULL,
        qualificationID             VARCHAR(8)      NOT NULL,
        instructorID            VARCHAR(8)      NOT NULL,
        locationID                  VARCHAR(8)      NOT NULL,
        statusID                    VARCHAR(8)      NOT NULL,
        date                        Date            NOT NULL,

        PRIMARY KEY (scoutID, qualificationID),
        FOREIGN KEY (scoutID)               REFERENCES Scout(scoutID),
        FOREIGN KEY (qualificationID)   REFERENCES
Qualification(qualificationID),
        FOREIGN KEY (instructorID)          REFERENCES
Instructor(instructorID),
        FOREIGN KEY (locationID)        REFERENCES `Location`(locationID),
        FOREIGN KEY (statusID)              REFERENCES
`Status`(statusID)

    )ENGINE = InnoDB ;

CREATE TABLE Component
    (
```

```
        componentID                 VARCHAR(8)          NOT
NULL,
        qualificationID             VARCHAR(8)          NOT
NULL,
        topicID                         VARCHAR(8)
    NOT NULL,
        typeID                      VARCHAR(8)          NOT
NULL,
        title                       VARCHAR(500)    NOT NULL,

        PRIMARY KEY (componentID),
        FOREIGN KEY (qualificationID)       REFERENCES
Qualification(qualificationID),
        FOREIGN KEY (topicID)   REFERENCES `Component Topic`(topicID),
        FOREIGN KEY (typeID)    REFERENCES `Component Type`(typeID)
    )ENGINE = InnoDB ;

CREATE TABLE `Scout Component`
    (
        scoutID                     VARCHAR(8)      NOT NULL,
        componentID             VARCHAR(8)      NOT NULL,
        date                        Date            NOT NULL,

        PRIMARY KEY (scoutID, componentID),
        FOREIGN KEY (scoutID)           REFERENCES Scout(scoutID),
        FOREIGN KEY (componentID)   REFERENCES
Component(componentID)
    )ENGINE = InnoDB ;
```

APPENDIX B: Data Dictionary

Association

Field	Type	Null	Default	Comments
associationID	varchar(8)	No		
associationName	varchar(50)	No		

Component Topic

Field	Type	Null	Default	Comments
topicID	varchar(8)	No		
title	varchar(100)	No		

Component Type

Field	Type	Null	Default	Comments
typeID	varchar(8)	No		
typeOfComponent	varchar(50)	No		

Component

Field	Type	Null	Default	Links to	Comments
componentID	varchar(8)	No			
qualificationID	varchar(8)	No		Qualification -> qualificationID	
topicID	varchar(8)	No		Component Topic -> topicID	
typeID	varchar(8)	No		Component Type -> typeID	
title	varchar(500)	No			

Course

Field	Type	Null	Default	Links to	Comments
courseID	varchar(8)	No			
qualificationID	varchar(8)	No		Qualification -> qualificationID	
statusID	varchar(8)	No		Status -> statusID	
locationID	varchar(8)	No		Location -> locationID	
typeID	varchar(8)	No		Component Type -> typeID	
title	varchar(200)	No			
startDate	date	No			
description	varchar(500)	No			

Instructor Course

Field	Type	Null	Default	Links to	Comments
instructorID	varchar(8)	No		Instructor -> instructorID	
courseID	varchar(8)	No		Course -> courseID	

Instructor

Field	Type	Null	Default	Links to	Comments
instructorID	varchar(8)	No			
userID	varchar(8)	No		User -> userID	
membershipID	varchar(8)	No		Membership Status -> membershipID	

Location

Field	Type	Null	Default	Comments
locationID	varchar(8)	No		
location	varchar(100)	No		

Login

Field	Type	Null	Default	Links to	Comments
loginID	varchar(8)	No			
userTypeID	varchar(8)	No		User Type -> userTypeID	
userID	varchar(8)	No		User -> userID	
username	varchar(50)	No			
password	varchar(50)	No			

Membership Status

Field	Type	Null	Default	Comments
membershipID	varchar(8)	No		
membershipStatus	varchar(50)	No		

Qualification

Field	Type	Null	Default	Links to	Comments
qualificationID	varchar(8)	No			
schemeID	varchar(8)	No		Scheme -> schemeID	
title	varchar(20)	No			
level	varchar(20)	No			
description	varchar(500)	No			

Scheme

Field	Type	Null	Default	Links to	Comments
schemeID	varchar(8)	No			
associationID	varchar(8)	No		Association -> associationID	
schemeName	varchar(20)	No			

Scout Component

Field	Type	Null	Default	Links to	Comments
scoutID	varchar(8)	No		Scout -> scoutID	
componentID	varchar(8)	No		Component -> componentID	
date	date	No			

Scout Course

Field	Type	Null	Default	Links to	Comments
scoutID	varchar(8)	No		Scout -> scoutID	
courseID	varchar(8)	No		Course -> courseID	
statusID	varchar(8)	No		Status -> statusID	
enrolmentDate	date	No			

Scout Qualification

Field	Type	Null	Default	Links to	Comments
scoutID	varchar(8)	No		Scout -> scoutID	
qualificationID	varchar(8)	No		Qualification -> qualificationID	
instructorID	varchar(8)	No		Instructor -> instructorID	
locationID	varchar(8)	No		Location -> locationID	
statusID	varchar(8)	No		Status -> statusID	
date	date	No			

Scout

Field	Type	Null	Default	Links to	Comments
scoutID	varchar(8)	No			
userID	varchar(8)	No		User -> userID	
membershipID	varchar(8)	No		Membership Status -> membershipID	

membershipStartDate	date	No			

Status

Field	Type	Null	Default	Comments
statusID	varchar(8)	No		
status	varchar(50)	No		

User Type

Field	Type	Null	Default	Comments
userTypeID	varchar(8)	No		
typeName	varchar(50)	No		

User

Field	Type	Null	Default	Comments
userID	varchar(8)	No		
name	varchar(100)	No		
address	varchar(100)	No		
telephone	varchar(11)	No		
mobilePhone	varchar(11)	No		
email	varchar(50)	No		
gender	varchar(1)	No		
medicalNote	varchar(500)	Yes	*NULL*	
note	varchar(500)	Yes	*NULL*	

APPENDIX C: Wireframes

RYA Sailing Club

information Management System

Trainer John Doe

Sign Out

Home

Home	Association	User	Course	Qualification

Image

(Quick view of courses trainer currently teaches)

Course Name	Location	Start Date
Stage 1 Course Weekday	RYA Sailing Club	12/02/12
Stage 1 Course Weekend	RYA Sailing Club	12/02/12
Stage 1 Course Half-Term	RYA Sailing Club	12/03/12
Stage 2 Course Weekday	RYA Sailing Club	12/04/12
Stage 2 Course Weekend	RYA Sailing Club	12/04/12

Quick Access to User Search
<Button>

Quick Access to Course Search
<Button>

About Us	Contact Us	Careers	

RYA Sailing Club

information Management System

Trainer John Doe | Sign Out

Association

Home	Association	User	Course	Qualification

RYA [drop down clicked]

List of Qualifications:

Stage 1

Quick View Stage 1: More details

Practical	**Theory**
Tying Rope	Using a Map
Using Compass	Checking Weather

Stage 2

NSS

K1018112 – Shakir Ahmad

RYA Sailing Club

information Management System

Trainer John Doe

Sign Out

Scheme

Home	Association	User	Course	Qualification

Table of schemes for associations

About Us	Contact Us	Careers	

RYA Sailing Club

information Management System

Trainer John Doe | Sign Out

User

Home	Association	User	Course	Qualification

Instructor | Scout

Search for RYA Member | Search

Alice Yonder

Bob Thamos [Drop down clicked]

Quick View Bob Details: More details

Contact Details:

Emergency Contact Number: 020867249389

Email: BobThomas@Mail.com

Qualification:

Stage 1 [ ] Stage 2 [ ] Stage 3 [ ] Stage 4 [ ]

Current Enrolled Course: Stage 4 Course

Medical Note:

Severally asthmatic: Make sure Brown asthma pump is with scout at all times before sailing.

Chris Benzi

Dennis Lambert

RYA Sailing Club

information Management System

Trainer John Doe

Sign Out

Scout

Home	Association	User	Course	Qualification

Scout Personal details

Name: Bob Thamos

Address: holobrook street

Telephone: 020 8 656 7536

Emergency Telephone:

020 8 656 7536

Email: bobThamos@mail.com

Courses | Qualifications

Scout Qualifications

Stage 1 [drop down clicked]

Qualification Status:	Complete
Location of Award:	Sailing Club
Date of Award:	12/12/12

Stage 2

Stage 3

Medical Note

Severally asthmatic: Make sure Brown asthma pump is with scout at all times before sailing.

Note

RYA Sailing Club

information Management System

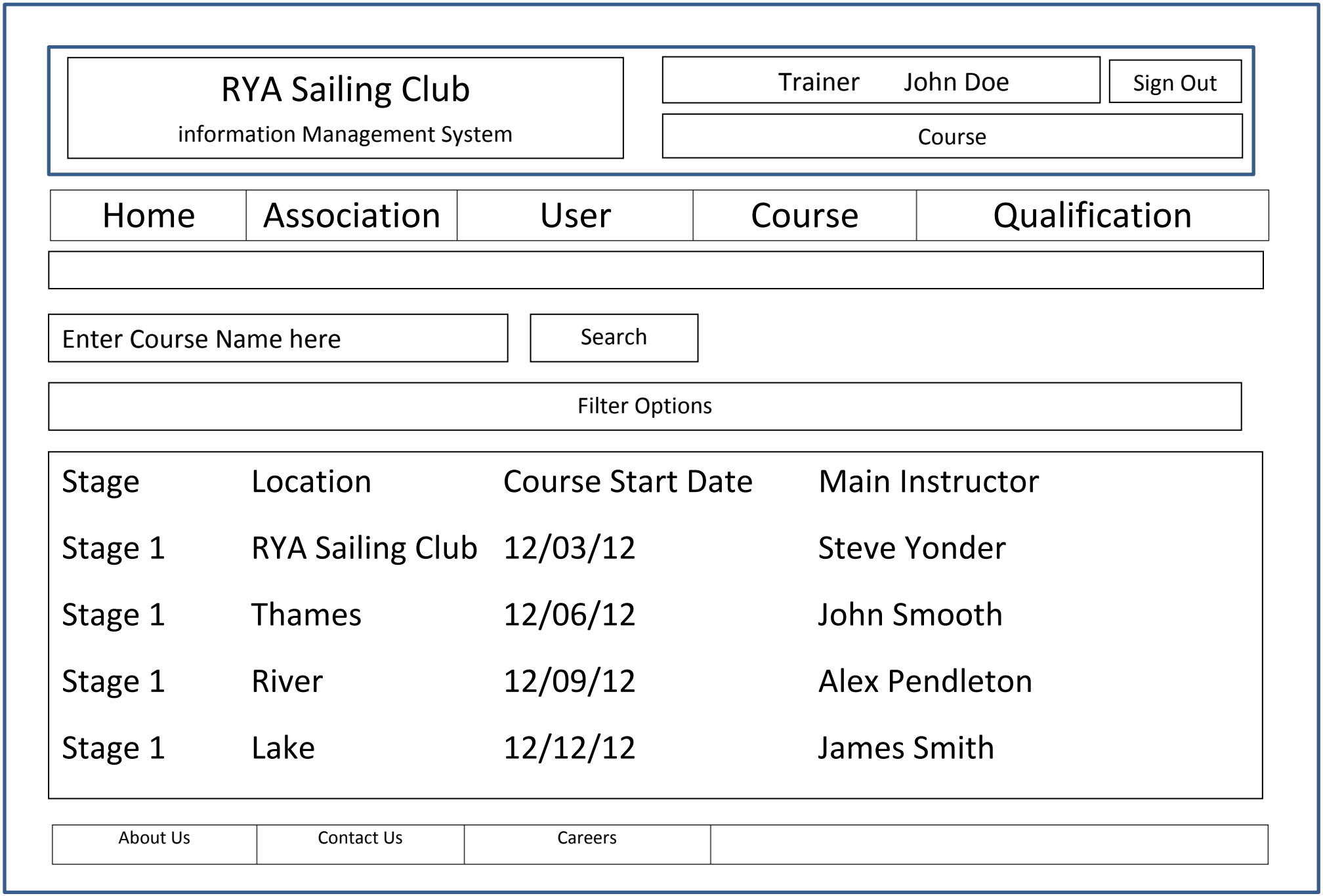
Trainer John Doe

Sign Out

Instructor

Home	Association	User	Course	Qualification

Instructor details


RYA Sailing Club
information Management System
Trainer John Doe
Sign Out
Course
Home
Association
User
Course
Qualification
Enter Course Name here
Search
Filter Options
Stage Location Course Start Date Main Instructor
Stage 1 RYA Sailing Club 12/03/12 Steve Yonder
Stage 1 Thames 12/06/12 John Smooth
Stage 1 River 12/09/12 Alex Pendleton
Stage 1 Lake 12/12/12 James Smith
About Us
Contact Us
Careers

RYA Sailing Club

information Management System

Trainer John Doe

Sign Out

Qualification

Home	Association	User	Course	Qualification

Drop down of Qualifications

RYA Sailing Club

information Management System

Trainer John Doe

Sign Out

Scout Awards and Badges

Home	Association	User	Course	Qualification

List of courses and components successfully completed by scout

RYA Sailing Club

information Management System

Trainer John Doe

Sign Out

Qualification

Home	Association	User	Course	Qualification

List of past and current qualifications being taught

RYA Sailing Club

information Management System

Trainer John Doe

Sign Out

Course Mark Sheet

Home	Association	User	Course	Qualification

List of Current Courses <Dropdown>

List of scouts enrolled on course

Check List of course components

About Us	Contact Us	Careers	

RYA Sailing Club

information Management System

Trainer John Doe

Sign Out

Qualification

Home	Association	User	Course	Qualification

List of components of qualification