



# Knowledge Bomb

Interactive learning support with smart mobile technologies

Final Group Report

Group 5

Henry Riches, Matthew Price, Luke Bandy, Ethan Preece, Alex Stewart, Kenneth Cynric Medina  
Dasalla, Stanislav Borissov, Nearchos Nearchou, Ahmad Syazani Mohammad

Client: Professor David Marshall

# Contents

Introduction .....	3
Refinements and Modifications .....	4
Justifications .....	5
UML (Unified Modelling Language) .....	6
STN (State Transition Networks).....	7
Key Aspects of the System .....	8
System Structure .....	11
Legal, Social, Ethical and Professional Issues .....	12
Legal issues.....	12
Social issues .....	13
Ethical issues: .....	13
Professional issues .....	15
Business Case .....	16
Testing.....	17
Use cases .....	26
Future Deliverables .....	27
Evaluation .....	28

# Introduction

This report aims to describe the final solution to our software development problem, set to us by our client Professor Dave Marshall. We were tasked with building an Interactive learning support solution with smart mobile technologies. The contents of this report will reflect plans and goals from prior reports and evaluate our final outcome for the overall project. We will show evidence that we have taken these requirements from the client, discussed and evaluated them in previous reports and have either included them in the final application or excluded them with justifications.

Included in this report will be a walkthrough of the system that we have developed along with supporting material such as; refinements and modifications of the requirements, justifications for changes made throughout the software development life-cycle and overall progress towards addressing the client's problem. All work since the previous report will be documented via this report as well as any executive changes that we made to the project as a group.

Visually we will display the projects solution via diagrams such as screenshots of key aspects of the design work. Influences on legal, social, ethical and professional issues will also be included along with a detailed explanation of the business case for the project which will include how we aim to later white label and maybe sell the product to educational establishments.

This report will also contain testing that we have conducted on the application via test cases and use cases that we have provided in prior reports. Rigorous testing has been applied to the application to ensure a high level of production along with ensuring all of our necessary requirements have been delivered.

An evaluation of the overall application and design stages will also be presented. This will include justifications for design and implementation decisions and an overall evaluation based on the data gathered from testing.

# Refinements and Modifications

Throughout the process of creating our app, all the way through initial planning, design and up to building a finished product there have been many ideas thrown into the mix. We prioritised elements of our app that our client asked for and agreed on the best ideas we had ourselves. These were developed into the features of our app. Inevitably as the process goes on some of these ideas would evolve, not everything we wanted would be feasible in the time limit and non-essential features were dropped in favour of perfecting essential features.

Our biggest change was the way we handled the login system. Initially we were going to have students create a profile, including a username and password, store them in a database and use form validation. However through our research into legal, social, ethical and professional issues and regulations we began to appreciate the security requirements of sensitive data and this would have require a lot of extra work. To solve this problem we found out that we could use the existing university server and students existing ID's and passwords. This means that the security aspect of storing data is handled by the much better equipped university server. This has also streamlined the process of using the app as students no longer need to create profiles and can begin using the app straight away.

Another feature of our app that has changed is the way content is loaded. Originally we had built our app as many separate pages that linked to each other, but this method is slow and inefficient and would suffer when multiple users try and update it at once. Instead we have used ajax to dynamically load different views and update content, this means that multiple users can use the app and ask and up vote questions and keep track of content much more effectively.

# Justifications

From our specifications we created our list of requirements. We have then evaluated our requirements and implemented key features as well as some optional ones within our application. There have been various changes from our original design to our final implementation. We have made changes to suit our development team, the client and even the target audiences. In this section of the report we shall justify what we have implemented in the final specification.

As mentioned above we have used the university's LDAP server for login authentication in our application. This benefits us as we do not need to store and secure large volumes of personal data. Not only does this benefit us it will benefit our target audience. There will be no need to register an account as students already have an account within the Cardiff University network which is then used for authentication to access the application. The major downside of this being different companies and/or universities may use different styles of server authentication and therefore the LDAP authentication is limited to Cardiff University students.

Our next feature we have not implemented is the use of QR codes. This was not needed as our client believes that this will restrict those users with laptops. All throughout the development of the application, we have frequently consulted with our client and adjusted our specification accordingly.

We have also decided to not include the ability for users to register and enter their personal details on the application as we have championed anonymity. The colour blind option was also removed as the user would need to enter this in the personal details. Instead we have chosen a colour scheme that uses colours not associated with colour blindness. Users can also change the colour scheme on their own on the 'Change Styles' page.

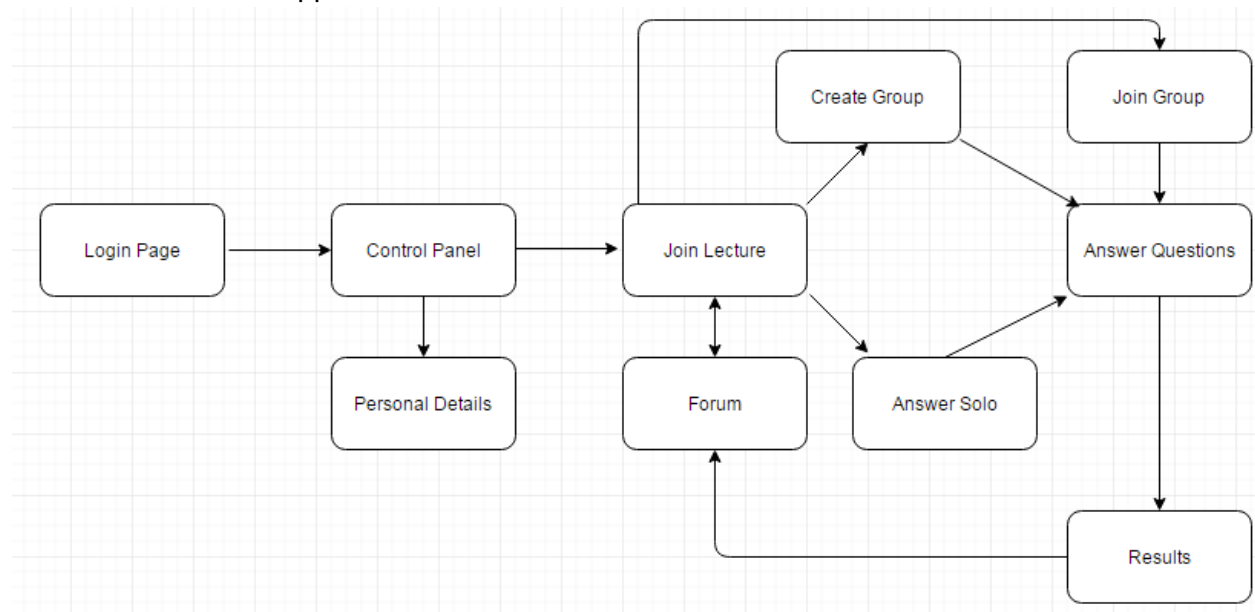
For our final justification, we have not used any classes to create the application as they were unnecessary. Some of our code has been repurposed from existing projects that our group members have worked on outside of this. The use of classes would have meant that we needed to track the objects within sessions which can be time consuming and also susceptible to errors. Instead of classes, we have made use of sessions to temporarily store variables and only the necessary variables are saved into a database for later use. When using the class diagrams, there can be many functions and may also be many objects created and therefore temporarily stored variables allows us to handle the application within getting confused.

# UML (Unified Modelling Language)

Unified Modelling Language or 'UML' is a general purpose, development language used widely in the field of software engineering. It is used to provide a standardised way of visualising the design of a system. UML allows the group to communicate the desired structure and behaviour of a system between ourselves and the client/supervisor, promote a deeper understanding of the system, highlight opportunities for simplification and allows us to evaluate and manage risk.

# STN (State Transition Networks)

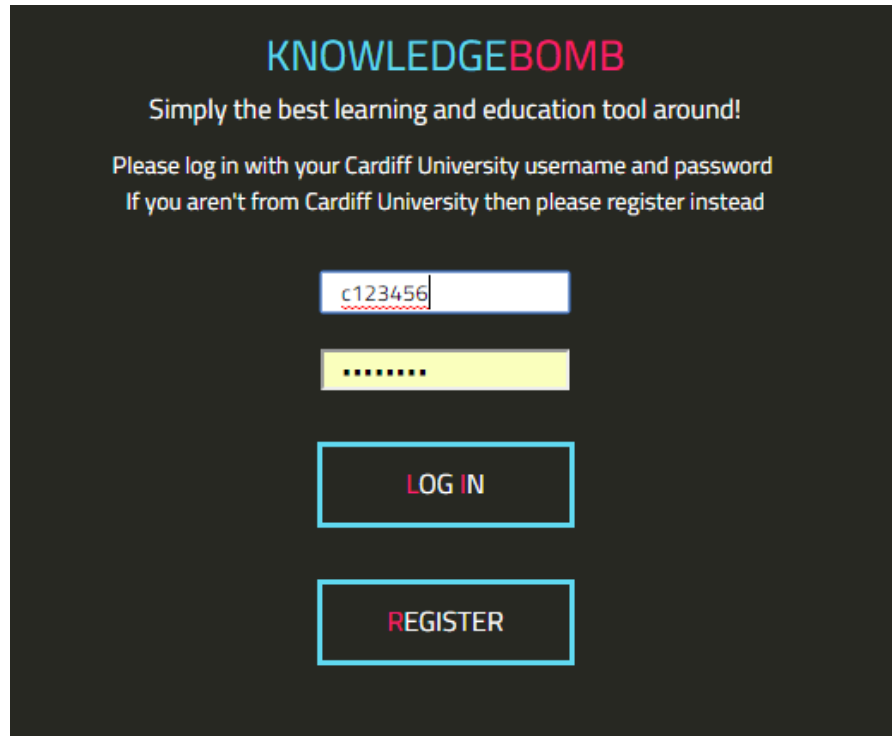
State transition networks are a visual representation of the application state by state. They give a visual representation of which pages each user can access from a particular state. Below is a STN for our finished application.



The above state transition network diagram shows the state transitions of our final application based off of our previous design work. This state diagram tightly resembles the design work defined in prior reports. This reflects that the design work defined was actually used when the final application was implemented.

# Key Aspects of the System

This section of the report displays and explains key aspect areas of the system that we feel were important, it shows a screenshot of the final version of that aspect and gives a short explanation.



The screenshot shows a dark-themed login and registration page for 'KNOWLEDGEBOMB'. The title is in large, bold, cyan and red letters. Below it, a tagline reads 'Simply the best learning and education tool around!'. Two lines of instructions follow: 'Please log in with your Cardiff University username and password' and 'If you aren't from Cardiff University then please register instead'. There are two input fields: a white one containing 'c123456' and a yellow one with masked characters. Below these are two buttons: 'LOG IN' and 'REGISTER', both with cyan borders and red text.

**KNOWLEDGEBOMB**

Simply the best learning and education tool around!

Please log in with your Cardiff University username and password  
If you aren't from Cardiff University then please register instead

c123456

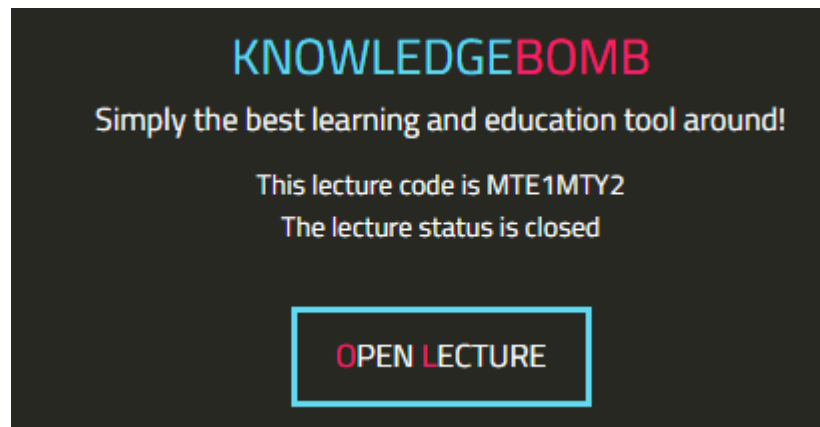
.....

LOG IN

REGISTER

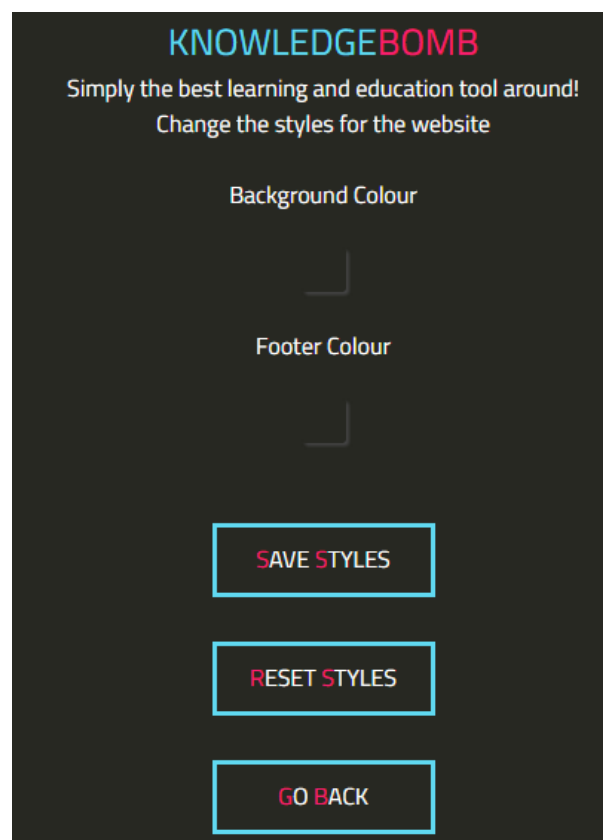
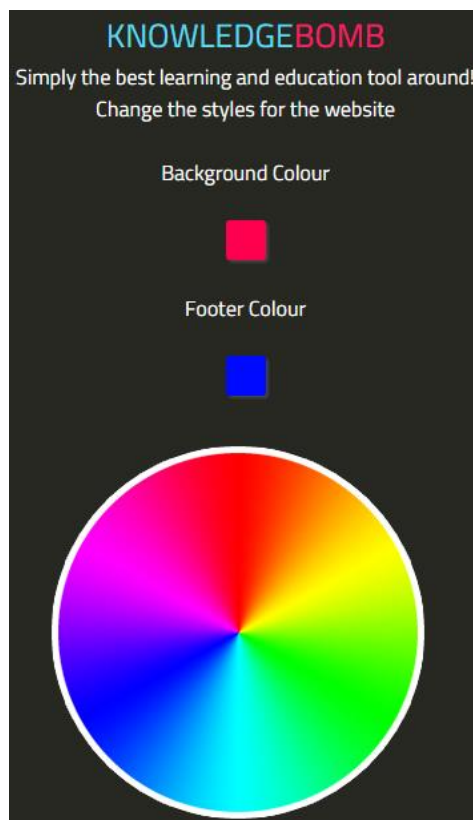


Above is a screenshot of our login screen. As we have discussed in prior reports and meetings we shall be using the university's own LDAP system as a login interface to aid usability of the application within the university. We believe that allowing users to use their existing university login to access the application allows for faster setup and increased usability. This also allows lecturers to use their university login, such that they don't have to 'sign up' before using the



application.

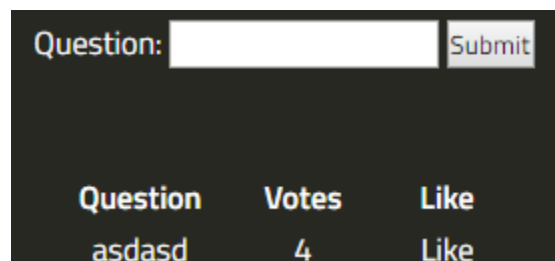
When lectures are created by an admin/lecturer, a code is randomly generated so students can join the lecture room. The way in which the code is generated allows the lecturer privacy against users trying to access lecture content in a malicious way. When the lecture is created, only students in the lecture are supplied with the code. Only students in this lecture are able to



ask questions via the forum and take quizzes set by the lecturer.

This gives student's further incentive to attend lectures as content available via the application can only be accessed by students who attend that specific lecture.

Above is a snapshot of the interface available to change the aesthetics of the application. We have implemented a system such that a user can change the colours of the application through the 'Change Styles' page. The colours of the application can be changed using our colour wheel which is used to change the background colour and footer colour. This style can be saved and be used throughout the application or it can be reset to default (dark grey).



Question	Votes	Like
asdasd	4	Like

Above is a snippet from the forum/question board that we have implemented into the application. This can be accessed during the lecture, this is where students can ask questions to the lecturer. This page can be accessed by lecturers at a later time so they can go back and see what questions were asked in a specific lecture. This allows the lecturer to see what questions were frequently asked and adjust their teaching accordingly.

# System Structure

This part of the report would normally include descriptive UML diagrams and Class Diagrams, visually displaying system architecture and structure. However, the way in which we coded the system uses very little structure and classes as almost everything is done in one main PHP file, 'home.php'. So, instead of giving UML and Class Diagrams we will look to provide a brief overview of system structure and talk about a few features and coding practices that we used whilst developing our system.

One of the main practices we used to store user data and cookies was the implementation of 'sessions' across the application. A session is an abstract concept to represent a series of HTTP requests and responses between a specific Web browser and server, sessions hold user data on the server (1 session per client) whereas cookies store user data on the client end. sessions are often built on top of cookies, the only data the client stores is a cookie holding a unique session ID which on each page request, the client sends its session ID cookie, and the server uses this to find and retrieve the client's session data. A session is defined by calling 'session\_start( );' at the top of your working file before any HTML, which signifies that a session will be started with the user. If the server hasn't seen this user before, a new session is created, otherwise, existing session data is loaded into \$\_SESSION associative array which can store data in \$\_SESSION and retrieve it on future pages

For storing of the quizzes, questions and answers are stored via a database. This database keeps a log of which questions were asked in which lecture along with the corresponding right answers to go with those questions. When a quiz is created information on each created question is taken from the generated form and sent to a PHP file via Ajax. The way information is stored from the generated questions is each question is stored in an html <div>, all divs that store questions being of the same class. These questions divs can then be looped through to obtain question information. This information is all saved to in a database in the format (QuizID, QuestionID, QuestionType, QuestionText, Options, Answers). QuizID is the primary key of the table and identifies the lecture the quiz belongs to. The variable representing the QuizID is simply the code of the lecture/room. QuestionID's are auto-incremented and are not passed to the database directly. The other four values passed are taken from the information passed in earlier when the "Create quiz" button is pressed.

# Legal, Social, Ethical and Professional Issues

## Legal issues

First of all, when creating our application we had to make sure we were adhering to the data protection act 1998. This is to ensure protection and security of everything within the application. This will look after any information or data which is being processed throughout the application. To do this we decided to add a login system in the application. This will enable the users to have their own username (Their student number) as well as their own personal password. This is very effective as it will help prevent any unauthorised access to the application. This will ensure any of the information being collected is staying in the right hands and is not at risk at being used in an unlawful way.

Another legal issue is to make sure all of the algorithms as well as all the functional aspects within the application is well secure and protected from all cases. One way in which we have been able to meet this is by connecting to the LDAP server. This is very helpful because it means we are connecting to the university server. This will enable many security methods to be applied within the application already as the university network is a very secure network and will prevent any unauthorised activities from occurring. This is also what is used to allow our login system to work efficiently. As from using the LDAP server both the login as well as the password will be the same login the user's use as their university logins. This makes it a lot more secure and effective as there is no need of changing passwords or even usernames. It also enables each of the users to have more trust in using the application as they know it is running of a secure and well protected network. This will give them confidence and encourage them to use the application more.

As we are using the LDAP server, all of the login details of the users are stored within the university server database. This means there is no personal or important data of users stored anywhere within the application. This means there is no risk of unauthorised access of accounts as the account details are located in a secure and well protected area.

## Social issues

One of the main social issues which our application could face is to ensure the security of all the information is kept to a sufficient standard and there is no risk of any unauthorised use of data/information. As mentioned above one of the main and most effective ways in which has secured our application is by using the university server (LDAP server).

This is very effective as it will prevent any unauthorised access as all of the login details are going to be stored within the university server. This means there is no risk of anyone accessing the application without having a verified login.

However, there are a few negatives from using the LDAP server. For example, if the university LDAP server goes down, this will cause the app to be unavailable for use, as it runs entirely of that server. Although this should not be a problem as the LDAP server has very efficient and top level security methods. This should prevent any errors from occurring or problems within the server.

Another social issue which we needed to ensure we done was to take possible disabilities into consideration. There may be many users who are planning to use our application who have a certain type of disability.

However, due to the time frame in which we had, it became very unlikely for us to be able to add in a personal details page. To make sure we hit the deadlines efficiently we thought the best option for us to take was to make each user anonymous when using the application. This was very effective in all areas as it ensured security and improves the reliability/trust in which the users have when using the application.

However I do believe if we were asked to create another application like this, we would dedicate time more efficiently to allow us to include a personal details page if necessary. An option which we could have implemented is to enable the user to be able to pick and choose what colour scheme they want their application to be. This is very effective and could be a huge assist for users who are colour blind. This would allow them to choose a colour scheme which is appropriate for their kind of eyes, which will enable them to easily use the application without any hassle. Due to not having this option available, it could cause users to lose interest in using the application as it may be difficult for certain users to use the application as they won't be able to see it properly.

## Ethical issues:

There are also many ethical issues which must be adhered to when creating a new application. For example, one of the main ethical issues which we had to meet efficiently is ensuring all of the software which is being used to create the application has appropriate licensing. If the software does not have the correct licensing put into place this could cause many problems in the long run. This could even be possible fines up to £110,000 if there is an unauthorised use of software.

For the majority of the time we have used a text editor which is called sublime text. This is a free text editor which is open for free download for all users, depending on how they are going to use it. If it is been downloaded and is going to be used continuously, there is a license which must be purchased. However as we are using it to create a small application and will not be used once it is finished, the free version of this software is most efficient for what we needed it for. It was a very effective and useful piece of software to use for our application as it allowed us to create our software from scratch. It is also very simple and easy to use software, so if anyone within our group are unsure on how to use it, it's very simple and easy to get the hang of. This allows all members of the production team to understand clearly what is happening and how everything is being done throughout the creation of the application.

Also another reason why we decided to use this kind of software for the creation of our product is because it saved us a lot of time and money. As it is a free application, we did not need to spend anything as the free version was sufficient for our needs. This was very effective as it prevented us from spending any unnecessary money. Also saved us a lot of time overall as we was able to start creating the application straight away without any problems from occurring. Finally we also had to ensure we were testing each part of the application thoroughly and consistently throughout the creation of the product. This was essential to do as if we did not do this, there could have been many unknown bugs or errors arisen at the end of the creation. This could have caused many problems as when the product is finally completed, it is a lot harder to try and determine where the errors are coming from. This is why we had to ensure we were checking each part of the application thoroughly while we were creating it, to tackle all of the errors as quick as possible.

To do this we had to make sure we were doing it as efficiently as possible. We set out an effective process where each of us within the group done our own specific part of the application. Once that was done we checked all of the individual parts to make sure they were working correctly and we then implemented them together. While adding each section to the application we had to consistently do valid checks to ensure that it was still working efficiently. This was very effective as it enabled us to figure out where the errors were coming from if any, also how to fix them.

By doing this it saved us a lot of time overall as the application was working efficiently and there was no errors occurring within the finished product. This also worked very well as it allowed us as a team to meet the deadlines accordingly as we kept the testing consistent throughout the production of the application.

## Professional issues

There are many professional issues which must be adhered to when creating an application. First of all as mentioned before, there are four points which must be looked at when considering the professional issues. These are the Product, price, place and promotion.

First of all I will look at the promotion. At the current moment in time, we have created this product generally as a university group project. However, we needed to ensure that there are possible capabilities of the product to be used in other real life situations. This is depending on the feedback we get of the use of the application within Cardiff University. I believe there is space for this kind of application at all a university, as it could be a very effective way of gaining feedback for students as well as lecturer.

However because we are using Cardiff University's LDAP server, this means there is no way at the moment of any other users accessing the application outside of the university. To overcome this we have implemented a white labelling system within the application which will enable all users outside the university to have access to the application. This is very effective because it gives us the option to make it more easily accessible for all users. This could allow us to promote the production very efficiently and even improve the application if necessary in the long term.

Another major professional issue which must be adhered to efficiently is the knowledge of the product. As mentioned before, it is essential that the whole production team has a good knowledge of the application and how the finished product must be. Before we finalised the application, we met up with Dave (the client) again to ensure that all of the requirements of the application are being met. This was essential to do before we finished the production of the application because if we finished it and then realised there was something missing, this would of caused a lot of time been wasted as well as money.

However as we have been meeting up on a regular basis this meant that we had met all of the requirements of the application efficiently. This was very effective as it ensured us as a team that we had met the client requirements efficiently and no problems or errors had occurred. This was also very useful as it gave our team as a whole confidence as we had completed the tasks we were required to do correctly.

Apart from these two I do not believe there are any other professional issues arisen at this stage of production. This is because the place of the application at the moment is currently within Cardiff University alone. This is down to it been a group project as well as because we are using the LDAP university server as mentioned above. Also the price of the application is not necessary at this moment in time. This is because no money has been spent within the production as it was not necessary for this kind of application. However this could change considering how well and in what way the application is going to be used within the university, also if it is successful or not.

# Business Case

This project emulates meeting the client's needs as if our group was in a real world situation. As part of a software development company we will also be including a business case to articulate a clear path to an attractive return on investment by the client as well as future investors. This argument is intended to convince a decision maker to approve further development of the system along with the distribution of our final product.

From the start of the project it has been our utmost priority to develop the application in a way in which it could be widely distributed and sold as a viable product for other educational establishments to use. This would be done through white labelling the system such that colour schemes, logos, text and interfaces could be easily changed by the purchaser to reflect their own style/company. To accomplish this white labelling technologies we have implemented an easy to use system of which anyone can use that involves clicking on different aspects on the screen and changing their colour etc. on the fly. This is done through the 'Change Styles' page available in the final version of the application.

We believe that allowing the product to be changed visually but maintaining the core functionality of the question and answer system makes for an easy to use and desired product. Combined with our rigorous testing that shows the potential the product has when used in educational establishments, we believe that the product is a desirable and viable product that could be used widely in schools, universities etc.



# Testing

## Group Testing

The first level of testing that we employed to test the application was to test it on as many devices as we could in one session. This was done in one of our weekly group meetings where we got everyone present to load the application onto their own device, join a lecture, take a quiz, and ask a question. This proved to be a successful test of the application, however, it did not run completely smooth as the university server was experiencing some 'slow' time. This slowed down the process of logging in but once everyone was logged in the test ran smoothly. This test did not only test to see if multiple users could use the application at one time but checked to see if the application worked on multiple different platforms on multiple devices. Below is a table displaying the devices that it was tested on and whether the application worked or not.

Device Name	Software	Did it work?
iPhone 6	Safari	Yes
iPhone 5	Safari	Yes
Samsung Galaxy S4	Galaxy browser	Yes
iPad	Safari	Yes
MacBook Pro	Firefox	Yes
Sony VAIO	Google Chrome	Yes
HTC One	Google Chrome	Yes
LG Nexus 5X	Chrome	Yes
Toshiba Laptop	Firefox	Yes

This is great news for us as we plan for the application to be used across all types of devices by all different aspects of users. We designed the application such that if a user's application can use a browser then the application can function.

## Test Cases

We tested our application on a system level by writing test cases for each functionality we would ideally provide to the user. The idea behind this approach is to provide us with a clear match between system reactions and user actions, giving us a way to benchmark whether or not our implementation has been successful. Based on how each functionality fits into the overall context of the system, each test goes into one of three categories - Login, Student actions or Lecturer actions. Each test case has:

- an ID associated with the functionality being tested
- a set of conditions that need to be met for the specific testing procedure to take place
- a step counter which identifies a given procedure within a test case
- procedure description - the way the user interacts with interface
- result - the system reaction to the user action
- pass/fail indicating whether or not the result in the test case reflects the actual implementation

With the completion of our application we've been able to extensively test every individual functionality on its own and in the context of the entire system. The following is the full set of completed test cases:

### First context: Logging in

All users must log-in to the system with their university credentials.

TS01: Login (student)			
Preconditions: None			
Step	Procedure	Result	Pass/Fail
1	Student inputs valid student number and password		
2	Login button is pressed	User is logged into the system, redirected to the user control panel and recognised as a student	Pass
Comments: Login verification successfully goes through the university LDAP server.			

<b>TS02: Login (lecturer)</b>			
Preconditions: None			
Step	Procedure	Result	Pass/Fail
1	Lecturer inputs valid ID and password		
2	Login button is pressed	User is logged into the system, redirected to the user control panel and recognised as a lecturer	Pass
Comments: Login verification successfully goes through the university LDAP server.			

<b>TS03: Login (negative test)</b>			
Preconditions: None			
Step	Procedure	Result	Pass/Fail
1	Invalid credentials are submitted		
2	Login button is pressed	Access is denied and a message is shown indicating this	Pass
Comments: Login verification successfully goes through the university LDAP server.			

### **Second context: Student actions**

This series of tests covers all actions available to students once they've logged into the application. Once students enter the student control panel they can join a lecture, and from there either take a quiz or submit/rate questions in the forum.

<b>TS04: Student joins lecture</b>			
Preconditions: Student has logged in and is on the user control panel			
Step	Procedure	Result	Pass/Fail
1	Student presses the “Join lecture” button	Student is presented with an input box in which they can input the lecture code of the lecture they wish to join	Pass
2	Student enters valid lecture code and presses “Join lecture”	Student joins a lecture room and is redirected to a page with the following options: “Take quiz”, “View forum”, “Go back”.	Pass
Comments:			

<b>TS05: Student adds question to the forum</b>			
Preconditions: Student has joined a lecture			
Step	Procedure	Result	Pass/Fail
1	Student presses the “View forum” button	Student gets redirected to the forum board	Pass
2	Student types in a question and submits it	Question is added to the forum board	Pass
Comments: The list of questions on the forum page is successfully updated. All existing questions are successfully retrieved from the database.			

<b>TS06: Student up votes question on the forum</b>			
Preconditions: Student has joined a lecture, at least one question exists on the forum			
Step	Procedure	Result	Pass/Fail
1	Student presses the “View forum” button	Student gets redirected to the forum board	Pass
2	Student up votes a question by pressing the “Like” button next to it	Updated question score is calculated and kept, questions are displayed in descending order of score	Pass
Comments: The new score of the upvoter question is calculated and the order of questions is changed if necessary			

<b>TS07: Student completes quiz</b>			
Preconditions: Student has joined a lecture and a quiz has been created for that lecture			
Step	Procedure	Result	Pass/Fail
1	Student presses the “Take quiz” button	Student gets redirected to the first question of the quiz	Pass
2	Student answers question		
3	Student presses the “Submit answer” button	Student is taken to the next question. Steps 2 and 3 are repeated until the quiz ends	Pass
4	Student presses the “View results” button	The student’s score on the quiz is shown along with the correct answers	Pass
Comments: Questions and their corresponding answers are successfully pulled from the database, correct answers are accurately compared to student’s answers			

### **Third context: Lecturer actions**

This series of tests covers all actions available to lecturers once they’ve logged into the application. Lecturer actions mainly involve managing lectures from the lecturer control panel in various ways - creating them, selection through various ones, creating quizzes for lectures, etc.

<b>TS08: Lecturer creates a lecture</b>			
Preconditions: Lecturer has logged in and is on the user control panel			
Step	Procedure	Result	Pass/Fail
1	Lecturer presses the "Manage lectures" button	Lecturer gets redirected to the lecture management page	Pass
2	Lecturer creates a new lecture by pressing the "Create lecture" button	A new lecture with a random name is created. The lecture should initially be closed.	Pass
Comments: New lecture is successfully added to the database and can now be entered by students			

<b>TS09: Lecturer can open/close a lecture</b>			
Preconditions: Lecturer has logged in and is on the user control panel, at least one lecture exists			
Step	Procedure	Result	Pass/Fail
1	Lecturer presses the "Manage lectures" button	Lecturer gets redirected to lecture management	Pass
2	Lecturer presses the "Edit lecture" button	Lecturer is redirected to lecture editing	Pass
3	Lecturer presses the "Open/Close" button	The lecture is opened/closed according to what the current state of the button is	Pass
Comments: The lecture status is reflected in the database - only open lectures may be joined by students			

TS10: Lecturer removes a lecture			
Preconditions: Lecturer has selected a lecture to manage			
Step	Procedure	Result	Pass/Fail
1	Lecturer presses "Manage lectures" button	Lecturer gets redirected to the lecture management page	Pass
2	Lecturer presses the "Old lectures" button	A list of already existing lectures is shown	Pass
3	Student remove a lecture by pressing the "Remove" button next to it	The lecture is removed from the list	Pass
Comments: The lecture is successfully removed from the database and is no longer shown to students			

TS11: Lecturer creates a quiz			
Preconditions: Lecturer has selected a lecture to manage			
Step	Procedure	Result	Pass/Fail
1	Lecturer presses the “Manage lectures” button	Lecturer gets redirected to the lecture management page	Pass
2	Lecturer presses the “Edit lecture” button	Lecturer is redirected to the lecture editing	Pass
3	Lecturer presses the “Create quiz” button	Lecturer is presented with quiz creator interface	Pass
4	Lecturer adds a question, selection question type, question text and possible answers	Question is added to the page	Pass
5	Step 4 is repeated as many times as the lecturer wishes	Each question is appended to the page	Pass
6	Lecturer answers all questions on the generated form		
7	“Create questionnaire” button is pressed	Lecturer is redirected to lecture management	Pass
Comments: The quiz is stored in the database so that it can be used by the students. In-depth error checking occurs for each question for problems such as unanswered questions, empty question text, trying to create a question-less quiz, etc.			



<b>TS12: Lecturer views the forum of a specific lecture</b>			
Preconditions: Lecturer has selected a lecture to manage			
Step	Procedure	Result	Pass/Fail
1	Lecturer presses the “Manage lectures” button	Lecturer gets redirected to the lecture management page	Pass
2	Lecturer presses the “Edit lecture” button	Lecturer is redirected to the lecture editing	Pass
3	Lecturer presses the “View forum” button	Lecturer is redirected to the forum where they can view all questions asked and their ratings	Pass
Comments: All questions and their ratings are successfully pulled from the database			

<b>TS13: Lecturer changes the styling of the application</b>			
Preconditions: Lecturer has logged in and is on the user control panel			
Step	Procedure	Result	Pass/Fail
1	Lecturer presses the “Change styles” button	Lecturer is redirected to the styling tool	Pass
2	Lecturer chooses an element to style - the background or the footer of the page	A colour wheel is shown	Pass
3	A colour is chosen	The selected element (background or footer) changes colour appropriately	Pass
Comments:			

## Use cases

Below is a table representation of the 'Use cases' represented in our interim report, page 25-29, these use cases were defined as a basic set of actions in which the system should be able to carry out. Below are the results taken from the final version of the application.

User case ID	User case name	Expected result	Actual Result
01	Login Page	The user's account will work on the system correctly	Login successful, expected result
02	Join a lecture	The user will now be in their specified lecture room where they can submit questions, create groups, join groups and answer a quiz solo	Join lecture successful, expected result
03	Join Group	The user will be able to view who is in their group and the leader can answer questions on behalf of the group	Unsuccessful, not yet implemented
04	Quiz as a group	The user will be in a current quiz and be able to answer questions on behalf of his/her group	Unsuccessful, not yet implemented
05	Change alias	The user's alias/nickname will be displayed to what they changed it to	Unsuccessful, not yet implemented
06	Submit a question to the Lecturer	The question submitted by the user will now be shown on the forum page	Question submitted successful, expected result
07	Up vote a question	The question the user has up voted will have 1 more vote	Up vote successful, expected result

# Future Deliverables

During the course of developing the project there were a few features which were omitted due to time constraints but are features which we would like to add to the project in the future. We have also come up with some features which could be nice additions to the project if future work was to be done.

## **Ad-hoc grouping**

This is one of the features that was omitted due to time constraints on the project, the concept was that a student could create a group for other students to join and answer the quiz together rather than individually. We have managed to create the system in such a way that this is something that could be added in the future with a small amount of work. This would be our number one priority to implement during any future development.

## **Using QR codes to join rooms**

This feature was one that we would have liked to implement from the start although the client said it wasn't necessary. By implementing the ability to join rooms via scanning a QR code we would also eliminate the issue which we have with our randomly generated room names (more on this later). Again this is a feature which could be added with a small amount of work and used in conjunction with the current system of the dynamically created room names.

## **Upgrading the dynamically created room names**

Whilst testing the application we ran into a slight issue with the room name generation. Since we are using base64 encoding to generate the name we have an issue where we can't distinguish between some characters in capitals from others in lowercase. An example of this would be an uppercase "i" and a lowercase "L". There are several ways in which we could fix this, either omit the offending characters from the room name generation or we could use a font which allows us to distinguish between the characters easily. The issue with the first approach would be that we reduce the number of available rooms in the database (it's currently  $64^8$ ) and it would go down to the region of  $60^8$  which is still more than enough rooms. The issues with the second approach mainly come down to personal preference and aesthetics.

## **Adding more advanced question types**

Another feature which we could add would be to make some more advanced question types, for example we could create a system that allows the lecturer to ask the user to create a program that outputs a certain string whilst following a specific set of instructions. This is quite out of the scope for the pub quiz aspect of the project, however it could definitely be useful in universities.

# Evaluation

During this section of the report, the last section, we aim to look back on what we have achieved as a group during this project, critically evaluate our end project along with how we worked as a group and overall bring our project to a final conclusion.

Working as a group has been an educational and enlightening experience. Every group member's skills have been tested and utilized all the way throughout our project's development.

To evaluate whether the design process of the application has been a success or not we have compared the final outcome of the application against our functional requirements set in the first group report. Below we have defined each functional requirement (both should have and could have) and written an explanation about whether we met this requirement or not.

1. **Lecturer can ask the students questions:** After meetings with our client and supervisor we decided that for lecturers to ask questions via the app was not needed as they could just ask the question conventionally which would be a lot easier. However, lecturers can also post onto the forum if they would like to define pre-set questions or ask questions themselves that they think may be popular.
2. **Students can ask the lecturer questions:** Students are available to post to the forum board of the lecture they are currently present in. This posts the questions to the board in which it can be viewed by all users in that lecture as well as the lecturer. We decided on the lecturer then answering the question conventionally in the lecture which gives the students more incentive to attend lecture classes.
3. **Ad-hoc grouping:** Although Ad-hoc grouping was a high priority functional requirement due to time constraints it is not currently implemented.
4. **Web Link:** The application is currently available via the school's server at ['https://project.cs.cf.ac.uk/RichesH/group5final/home.php'](https://project.cs.cf.ac.uk/RichesH/group5final/home.php).
5. **QR Codes:** Instead of implementing the ability for QR codes to be used to access lectures/the application, we have implemented a randomly generated code which is used to access lectures, which is supplied to the lecturer when the lecture is created. Lecture accessibility was done this way so users using laptops who would struggle with accessing QR codes can easily access the app. However, for QR code usability to be implemented it would just be a case of implementing a QR code generated for the random code generated for each lecture.
6. **Lecturer login:** We have successfully implemented a lecturer login, to be logged into an 'admin user' they must first be given access or they will only be given student access. Lecturer access allows for further functionality within the application.
7. **Content Management System (CMS):** A CMS has been successfully implemented with the ability to change the colour of the background and footer of the application.
8. **Forum boards of questions:** Forum boards for each lecture are fully implemented in which students can ask the lecturer a question and it can be voted upon by other students.

- 9. Rating Lectures:** Rating lectures was something that we said we 'could' have. Upon making the decision to make the application completely anonymous we decided not to add the ability to rate lectures because of the anonymity involved. Allowing full anonymity gives the user the ability to perhaps be malicious when it came to rating lectures and we found that this could have negative consequences.
- 10. Student Login:** A student login using the LDAP university system is fully implemented, using the university login system allows anyone in the university the ability to use the app and eliminates lengthy 'Register' forms.

From above it is clear that most of the functional requirements defined in the first report were actually successfully fulfilled in the final version, 6/10 of the functional requirements were fully implemented (with those of which were not implemented supplied with proper justification). The only thing that we hoped to include into the system that we finally did not get round to doing because of time limitations was 'Ad-hoc' grouping.

**What have we achieved:** As of now the final version of the application that we have designed and implemented, all as part of a group, it is fully functional and ready to be used in lectures. We have tested the application on a number of devices and against a number of industry standard testing procedures and have proven that it works as proposed. We believe we have sufficiently met client requirements and implemented a fully working application as proposed within the time frame set with only minimal aspects for improvement. We have gone above and beyond as a group and have worked cohesively as a group to achieve a high level of software design.

As stated above, under future deliverables, there is room for improvement and further additions to the functionality of the application, but we are happy with the application as it stands now.