

Lecture Communication
(“LectComm”)



Cameron Ward

x12378906

BSc in Computing

Network & Mobile Technologies

2015/2016



National
College *of*
Ireland

Supervisor:

Paul Hayes

Declaration Cover Sheet for Project Submission

SECTION 1

Name:
Student ID:
Supervisor:

SECTION 2 Confirmation of Authorship

I confirm that I have read the College statement on plagiarism (summarised overleaf and printed in full in the Student Handbook) and that the work I have submitted for assessment is entirely my own work.

Signature: _____

Date: _____

NB. If it is suspected that your assignment contains the work of others falsely represented as your own, it will be referred to the College's Disciplinary Committee. Should the Committee be satisfied that plagiarism has occurred this is likely to lead to your failing the module and possibly to your being suspended or expelled from college.

Table of Contents

Executive Summary	5
1 Introduction	6
1.1 Background.....	6
1.2 Aims.....	7
1.3 Technologies	8
2 System.....	10
2.1 Requirements	10
2.1.1 Functional requirements.....	10
2.1.2 Data requirements.....	11
2.1.3 User requirements.....	11
2.1.4 Environmental requirements	12
2.1.5 Usability requirements.....	13
2.2 Design and Architecture.....	13
2.3 Use Case Diagrams.....	14
2.4 Implementation	16
2.4.1 Creating Database Tables.....	16
2.4.2 LectComm.....	17
2.4.3 Forum.....	17
2.4.4 Maps	18
2.4.5 To Do List.....	19
2.5 Testing.....	19
2.5.1 Database Connection Test.....	20
2.5.2 LectComm Login Test	20
2.5.3 LectComm Compose Test.....	20
2.5.4 LectComm Inbox Test	21
2.5.5 LectComm Reply Test.....	21
2.5.6 Forum Login Test	21
2.5.7 Category View Test	22
2.5.8 Topic Creation Test.....	23
2.5.9 Views Test.....	23
2.5.10 Reply Compose Test.....	23

2.5.11	Reply View Test	24
2.5.12	Maps Test	24
2.5.13	To Do List Test.....	25
2.6	Graphical User Interface (GUI) Layout.....	26
2.7	Customer testing.....	34
3	Conclusions	35
4	Further development or research.....	36
5	References	37
6	Appendix.....	38
6.1	MySQL Queries	38
6.1.1	Users.....	38
6.1.2	Private Messages.....	38
6.1.3	Topics.....	38
6.1.4	Posts	39
6.1.5	Categories.....	39
6.2	Project Proposal	40
6.3	Project Plan	43
6.4	Requirement Specification	44
6.4.1	Requirement 1: User Login	44
6.4.2	Requirement 2: LectComm Compose	45
6.4.3	Requirement 3: LectComm Reply	46
6.4.4	Requirement 4: Forum Create Topic.....	47
6.4.5	Requirement 5: Forum Post Reply	48
6.4.6	Requirement 6: Map Location	49
6.4.7	Requirement 7: To-Do List	50
6.4.8	Non-Functional Requirements.....	51
6.5	Monthly Journals.....	52
6.6	Survey Answers.....	56

Executive Summary

The idea for my project derives from my personal experiences with lectures in NCI, along with my experiences with education in general. While communication with majority of lectures are great, there are others where lack of communication has been the difference between an honours and a pass for myself and fellow classmates. Even though we are in third level education, I feel as though people still take somewhat of an embarrassment if they don't understand something their fellow peers do. I know that on multiple occasions there have been times where at the end of lectures, lecturers will give an opportunity for students to ask questions or ask if they want to go over anything again. The majority of the time students will not ask questions or interact in anyway, and at the same time these students are asking for extra classes at the end of semesters and complaining that the lecturer wasn't good enough or offering enough help, when this is not necessarily the case.

"LectComm" will allow students and lecturers to interact with each other at a more optimal level, by using a timestamped messaging service that allows the lecturer to view messages at his/her own leisure. This will also lead to less interruptions being taken away from valuable class time. I believe that this applications concept is easy to use and when completed will have a very user friendly interface and provide a beneficial service to all users.

1 Introduction

1.1 Background

During a lecture, some students can become overwhelmed by the amount of information that is being covered in such a short period of time. The lecturer may also be rushing through the content at a faster pace, expecting to go over things students didn't understand at the end. Unfortunately, both of these issues tend to never be resolved due to the lack of communication between students and lecturers.

It is very common for a student not to ask for assistance during or after class. This is because people do not want to look as if they don't understand something that everyone else does. These problems are mostly associated with younger students, as they are more likely to form more judgemental friend groups. Even if a lecturer encourages feedback, it is still very rare that they will get any.

Ironically the same students that do not ask for help during or after lectures, and claim to understand the course content, are the same students that seek extra classes at the end of the semester and claim that it is the lecturer's fault for their bad grades.

There are certain times of the year where students discuss the course content amongst each other. Usually around the time of exams and CAs. This usually takes place on a Facebook page created by the class representatives. This however, is a flawed system as many students do not use Facebook regularly and some do not use it at all

As I am a student myself I have both witnessed and succumbed to most of, if not all of these claims. My initiative was to develop this application to help aid students and lectures overcome these issues and for both to have a more enjoyable experience in third level education.

I decided to build my application using both Android because, not only do I have experience in creating Android applications, I find that I have a greater

understanding of XML and Java, compared to the languages Apple and Microsoft use for their operating systems. It is also beneficial to me that one of my fourth year modules (Multimedia and Mobile Application Development) provides me with helpful information when dealing with Android.

1.2 Aims

The purpose of my final year project is to develop an application that allows students to interact with lecturers and also each other, at a more optimum level. This app will appeal to the eye of the user and offer functionality that will be beneficial to both students and lecturers. The application will allow users to sign in using pre-existing login credentials provided to them by NCI. Student users will be then able to send

During or proceeding lectures, students will be entering the “Lecture Communication” section of the app. They will be presented with a button, labelled “HELP!” Students will be able to interact with this button anytime throughout the duration of the lecture. By interacting with this button it will send a notification to the lecturer to notify them that one or more students have failed to grasp what the lecturer is trying to explain. Students will also be able to type a brief description pointing out exactly what part or parts of the lecture they were not comfortable with or did not understand.

With all of this information at the end of the lecture, the lecturer will be presented with a list of students with the following information; their student number, the issue they had, and a timestamp of when they had said issue. If there is a common problem involving multiple students the lecturer may be able to address this issue by organising an extra class or taking a few minutes out of the next lecture to go over the said issues. However, if the problems about the lecture are isolated to say one or two students, the lecturer will be able to organise a “one to one” session to discuss the issues the student has had.

In regards to discussion between students about upcoming exams and CAs, there will be forum functionality that will allow students to discuss amongst themselves in a real time and on a more structured system. This will be accessible to every student not just those who use social media sites such as Facebook to communicate.

1.3 Technologies

This application is being developed using HTML, PHP, Android and MySQL. I am using HTML and PHP to develop the web app that lecturers will be using during and after lectures. Through Android Studio I will be using both Java and Android XML along with the others mentioned above.

For the user interface of my project I will be using Android XML, which will clearly define all items displayed within the application. I find Android XML very easy to use and due to it being part of my curriculum for one of my final year modules, it is beneficial for me to be using it as said module is aiding me in fully grasping all the functionality that is available to me.

I have used Java in my application to add functionality to my XML pages which are the core part of this app. In regards to buttons and events Java provides activities for each one. Due to Java being the main programming language I have studied throughout the years I have been in college, it was the perfect language to choose for developing my application.

I used MySQL to create, build and design the database used within my app. I chose MySQL as it is the only query based language I have had previous experience with. By using Byet.Host, I was able to store Lecture Communication's database online for free, by using phpMyAdmin hosted on their user panel.

PHP files were used to connect my application to the database hosted on Byet.Host. PHP was used as a general purpose programming language to define the methods of retrieving and creating data stored in my Byet.Host database, such as log in information, messages sent to and from students and lecturers, and categories, topics and replies on the forum section.

I chose to develop with android as I have experience dealing with android and I am more familiar with using Java, instead of the languages other companies use for their mobile operating systems.

2 System

2.1 Requirements

2.1.1 Functional requirements

A functional requirement is a statement that identifies what the system must do. They indicate processes that the system has to perform. In this section I will discuss the functional requirements associated with my application.

LectComm/Forum Login: The user logs in for both the LectComm and the Forum section, share the same table of data that contains the specific user login information. The table is small and only contains the user id, username, and password. These accounts are created external from the app to prevent the miss use of the app itself. Users will only be provided with login details if they are a lecturer or student of the college. This is similar to student email accounts provided to each student when they begin semester one of first year.

Student Compose: Students will be able to interact with lecturers during or after class, by sending the lecturer a message in regards to what issue they had. The user will create the message then send it by pressing the big “Help” button located below where they created their message.

Student/Lecturer Inbox: Both a student and lecturer will have an inbox, the student will just use his inbox to view replies from a lecturer, these replies will be how they will try to resolve the issue you had understanding the content they were covering. The lecturer’s inbox however will be filled with the issues students have in regards to the course content being covered. Each message will have the username, subject and a timestamp of when the message was sent.

Student/Lecture Reply: Whenever a message is sent to another user they will also have the option to reply to said message. The must simply click the link labelled “Reply” and will be brought to another page where they can compose their response.

Student/Lecturer Outbox: With the outbox functionality users can see every message they have ever sent. It is similar to the outbox where the messages are displayed with the username, subject and timestamp.

Forum Topic Creation: Once a user is within a category and successfully logged in they will be able to create topics within the category itself. They will be asked to provide a subject line and then fill in the rest of the topic content.

Forum Post Reply: If a user interacts with a topic they will be given the option to reply to the topic in question. The user may simply post their response and it is then displayed directly under the main topic in the thread.

Distance from NCI: A user will be able to use the Maps section of the LectComm app to see how far away they are from NCI at their current location. They are also provided with the option to check the time it will take them to get there, by car, train or bike.

To-Do List: A user will be able to create a to-do list with tasks they set for themselves. They will also be able to remove each task when they are completed.

2.1.2 Data requirements

SQL Requirements: This application will store all user and non-user data on a server which will be hosting my SQL database. I will be using PHP files which will also be hosted on the same server, too access the information contained in different tables within the database.

2.1.3 User requirements

In this section, I will talk about the user requirements. These are the requirements that are necessary for the user to have in order to use both the mobile and web applications.

Be a Student: In order to gain access to lecturer's modules you will have to attend an institution of third level education. The lecturer will have to approve a student before they gain access to a module.

Be a Lecturer: In order to gain access to lecture permissions, you will have to be. An admin will have to approve that the user is in fact a lecturer.

Android Phone: To use the mobile app the user must possess a phone that uses the Android operating system.

Browser Access: To use the web app the user must have access to a web browser on either a PC or a Smartphone.

Android 6.0 Marshmallow: The user should preferably have Marshmallow installed, but the application is not limited to Marshmallow and will be backwards compatible.

Internet Access: The user will need Internet access in order to download the app.

2.1.4 Environmental requirements

In this section, I will talk about the environmental requirements. These are the vital requirements that must be present when developing the application.

Windows: This application was developed using Dell Windows laptop with Android Studio as the Android development IDE.

Notepad++ (Initially): I used Notepad++ to create the HTML and PHP files for my project before they were hosted on Byet.host.

Android Studio: I used Android Studio to build the application from the ground up.

Byet.host: I used Byet.host to host the PHP and HTML files and to host the applications database, with the help of phpMyAdmin.

Android Device: An android device was needed in order to run and test the app during the development stages.

Internet Access: Internet access was needed in order to test out the user functionality between two separate devices.

2.1.5 Usability requirements

This section will highlight usability requirements. These will provide concrete objectives during the interface and design process. These usability requirements evolved from the initial meetings I had with target users as described above.

Easy to Understand: The user interface of the application will be easy to follow and understand.

Easy to Operate: The actions should be consistent and free flowing. Any errors that may occur during the use of the application should explain where and why there was an error.

Ease of Use: The system should be easy to use; users should not have to spend time trying to figure out how to perform an action.

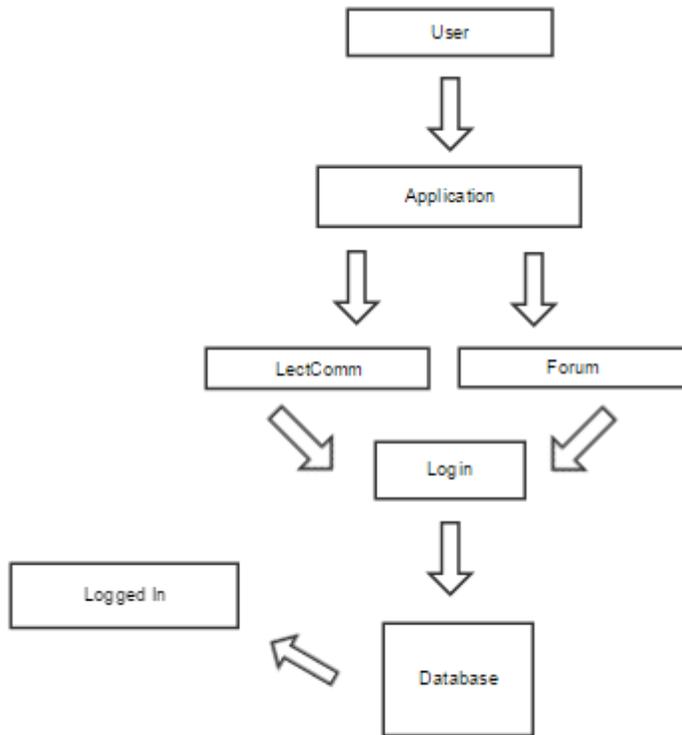
Good Presentation: The layout of the application will be presented in an attractive way that is able to catch the eye of the user. The colour scheme of the application will stand out and make the app an enjoyable experience for the user.

2.2 Design and Architecture

This application is built primarily using Android XML with Java along with HTML and due to the app containing quite a few functions that needed access to an SQL database, PHP was used to process any requests to the database. I chose PHP over other languages due to my familiarity with the language itself and how reliable it is in regards to interacting with databases hosted online. Upon inspection of the requirements and functions that were to be developed, I personally believed that PHP, would produce an application that would “catch the user’s eye”. The XML pages were designed in such a way that when both students and lecturers go through multiple pages on the app, that it would appear to be a smooth transition between the various pages. This was implemented in the design of the app, as the same formatting and style is present on each page.

2.3 Use Case Diagrams

The following diagram shows the process of both the user Registration and the user Login. To begin the Lecturer/Student must interact with the application the user will be prompted to sign in to their existing account. If they are not an existing user, they will have to click the option to. If the user selects registration then they are brought to the registration page where they will enter their email, username and password etc. to be able to gain access to the application LectComm. Once the user is successfully registered, the user's information is stored in an SQL database where their details can be retrieved at any time. If the user selects the option to log in, they are brought to the LectComm login in screen where they required to fill out the necessary information. Upon logging in, the system connects to the database to retrieve the required information and check for authentication. Once this process is completed and the login authorization is confirmed, the user is then logged in and brought to the user's home page.



This diagram illustrates the process of communication within the application. If a student wishes to send a message to their lecturer, they must first interact with the application. They will then be presented with list of modules associated with their account. These are taken from the user's information in the database. Once the user clicks on a module they will be presented with a button that says "HELP". When the user interacts with the button they will then have the option to submit their issue to send to the lecturer. Once the lecturer receives the issue he can reply to the user directly and attempt to resolve said issue.

2.4 Implementation

2.4.1 Creating Database Tables

I used MySQL queries to create tables within the main database for the user log in and for messages sent using both the LectComm section of the app and the forum section. For a full view of all the queries I used see Appendix 6.1.

Connecting Databases to Project

In order to connect my files to the database stored online I first had to create a connect.php file by which included where the database was being hosted and my login information for that said database I used the following php code to do so.

```
$host = "sql201.byethost10.com";  
  
$username = "b10_18022027";  
  
$password = "testroot123";  
  
$db = "b10_18022027_data";  
  
mysql_connect ($host, $username, $password) or die  
(mysql_error ());  
  
mysql_select_db($db);
```

Login – The user logins for both the LectComm and the Forum section, share the same table of data that contains the specific user login information. The table is small and only contains the user id, username, and password. These accounts are created externally from the app to prevent the miss use of the app itself. Although the table of information was the same the way I used it was slightly different for both. For the Forum section of the app I allowed users the view the main categories before logging in, but not allow the user to create a topic within said categories until they did.

```
$cid= $_GET['cid'];  
  
if (isset($_SESSION['uid'])) {
```

```

$logged = " <a href='create_topic.php? cid= ". $cid."> Create A  

Topic. </a>";

} else {

$logged = " Please log in to create a topic for this forum.";

}

```

For the LectComm section of the app the user has no access at all until they sign in correctly with their provided login credentials.

2.4.2 LectComm

In order to create the LectComm section of the application I first made three separate areas the compose, the inbox and the outbox. Once the user is logged in they can navigate through all three seamlessly through the one page. I used php to parse and direct information from other files to the main message file that was displaying the information.

```

if (! empty($_GET['id'])) {
    $pages = scandir ($pages_dir, 0);
    unset ($pages [0], $pages [1]);

    $id = $_GET['id'];

    if (in_array ($id.'. inc.php', $pages)) {
        include ($pages_dir.'/' . $id.'. inc.php');
    }

```

2.4.3 Forum

Once a user is logged into the forum they can create a topic within a set within a specific category, to allow this I created a topic page and made a subsequent parse page that drew the information from where the topic was created and placed it within the main topic file. I also used this parse functionality when creating the reply function.

```

$cid = $_POST ['cid'];

$title = $_POST ['topic_title'];
$content = $_POST ['topic_content'];
$creator = $_SESSION ['uid'];

$sql = "INSERT INTO topics (category_id, topic_title,
topic_creator, topic_date, topic_reply_date) VALUES (" . $cid . ", " .
$title . ", " . $creator . ", now (), now ())";

$res = mysql_query($sql) or die (mysql_error ());
$new_topic_id = mysql_insert_id ();

$sql2 = "INSERT INTO posts (category_id, topic_id,
post_creator, post_content, post_date) VALUES (" . $cid . ", " .
$new_topic_id . ", " . $creator . ", " . $content . ", now ())";

$res2 = mysql_query($sql2) or die (mysql_error ());

$sql3 = "UPDATE categories SET last_post_date=now (),
last_post_date=" . $creator . " WHERE id=" . $cid . " LIMIT 1";

$res3 = mysql_query($sql3) or die (mysql_error ());

```

2.4.4 Maps

For the maps section of the app I had to first find the coordinates for NCI then set a permanent marker on it so it could be our focal point.

```

LatLng nci = new LatLng (53.348865, -6.243115);

```

I then proceeded to create marker for the user's current position using the geolocation within google maps.

```

If (ActivityCompat.checkSelfPermission(this, Manifest.permission.
ACCESS_FINE_LOCATION)! =PackageManager.PERMISSION_GRANTED
&&
ActivityCompat.checkSelfPermission(this, Manifest.permission.
ACCESS_COARSE_LOCATION)! =
PackageManager.PERMISSION_GRANTED) {
return;

```

2.4.5 To Do List

To create the to-do list I made a prompt box that would appear to allow the user to create their task. The prompt box would appear by using an `AlertDialog` and ask the user to create a task. The information was then stored in an SQLite database within located within Android Studio.

```
AlertDialog dialog = new AlertDialog.Builder(this)  
    . setTitle ("Add a new task")  
    . setView(taskEditText)  
    . setPositiveButton ("Add", new  
DialogInterface.OnClickListener() {  
        @Override  
        public void onClick (DialogInterface dialog, int which) {  
            String task = String.valueOf(taskEditText.getText());  
            SQLiteDatabase db = mHelper.getWritableDatabase();  
            ContentValues values = new ContentValues ();  
            values.put (TaskContract.TaskEntry.COL_TASK_TITLE,  
task);
```

2.5 Testing

Testing was an ongoing experience throughout this whole project. Every time I would run my project I would test it for errors, and every time I got an error I would take note of it. I would then scan through my code and locate the error to see where I could have gone wrong and fix it. I found this to be beneficial as opposed to creating an entire project without running the code once and in the end being presented with multiple of errors. The later you leave fixing the error the more difficult it can become to fix as it can start a ripple effect. If piece of code relies on the code with an error in it, and the code with the error is fixed, that original piece of code may have needed that code to run properly.

2.5.1 Database Connection Test

Test: If the database is connected to the application.

Expected Outcome: The database is successfully connected to the application and data can be transferred from the database to the php files.

Actual Outcome 1: Error: Data is not being retrieved from the database.

Test Result: Test Failed

Actual Outcome 2: The messages are visible to the lecturer in the correct format.

Test Result: Test Successful

Comment: In the connect.php file I had declared an old database and old password, I rectified the error by inserting the up to date database and password and now the database connects successfully.

2.5.2 LectComm Login Test

Test: If users can login to the LectComm section of the application.

Expected Outcome: Users will be able to use their login credentials that were given to them to login and gain access to the LectComm section of the application.

Actual Outcome 1: Error: User was not permitted access to the app.

Test Result: Test Failed

Actual Outcome 2: User was granted access to the app.

Test Result: Test Successful

Comment: I forgot to close a curly bracket in the login.php file this caused the file to compile with errors.

2.5.3 LectComm Compose Test

Test: If a student can compose a message to send to a lecturer

Expected Outcome: A student will be able to compose a message addressed to the lecturer of their choice, it will be comprised of a subject line and the message content.

Actual Outcome 1: The user is successfully able to compose a message to send to their lecturer, comprised of the subject line, message content and a timestamp.

Test Result: Test Successful

2.5.4 LectComm Inbox Test

Test: If the lecturer is able to see the message sent to him/her by the student

Expected Outcome: The messages sent by students to lecturers can be seen by the lecturer with the Username, Subject and a timestamp of when it was sent.

Actual Outcome 1: Error: Timestamp is not showing with the message in the inbox.

Test Result: Test Failed

Actual Outcome 2: The messages are visible to the lecturer in the correct format.

Test Result: Test Successful

Comment: The error was located in compose.inc.php file “Y-m-d H: i: s” was incorrectly typed with semi colons instead of colons.

2.5.5 LectComm Reply Test

Test: If a Lecturer can reply to a student after a message has been sent.

Expected Outcome: The lecturer is able to send a reply to a student by clicking the reply button. The reply section is similar to compose section as the three inputs that are needed are the username, subject line and message content.

Actual Outcome 1: The lecturer is successfully able to compose a private reply to the student that sent them the message in the first place. It is comprised of the reply title, reply content and a timestamp.

Test Result: Test Successful

2.5.6 Forum Login Test

Test: If Users can login to the Forum section of the application.

Expected Outcome: Users will be able to use their login credentials that were given to them to login and gain access to the LectComm section of the application.

Actual Outcome 1: Error: User was not permitted access to the app.

Test Result: Test Failed

Actual Outcome 2: User was granted access to the app.

Test Result: Test Successful

Comment: The error was located in the login_parse.php file I left out a semi colon and this lead to my application compiling with errors.

2.5.7 Category View Test

Test: If categories are visible to users.

Expected Outcome: The categories created within the database are able to be viewed and interacted with by the user.

Actual Outcome 1: Error: Categories aren't visible when user is logged in.

Test Result: Test Failed

Actual Outcome 2: Categories are now visible to students when they are logged in.

Test Result: Test Successful

Comment: Error was located in the view_category.php file, I forgot to insert a \$ beside the sql on the line :

```
$sql = "SELECT id FROM categories WHERE id='".$cid.'" LIMIT 1";
```

This lead to the file compiling with errors the \$ was put in and file ran successfully.

2.5.8 Topic Creation Test

Test: If student can compose a topic within a category.

Expected Outcome: A student is able to create a topic within the category that contains the topic title, topic content and a timestamp.

Actual Outcome 1: The user is able to successfully create a topic within a category containing topic title, topic content and a timestamp.

Test Result: Test Successful

2.5.9 Views Test

Test: If a topic created by a user can be viewed by another user.

Expected Outcome: Topics that are created by students should be visible within a category in the topic view section.

Actual Outcome 1: User is successfully able to view all the topics created by other users within the topic view

Test Result: Test Successful

2.5.10 Reply Compose Test

Test: If a user can compose a reply.

Expected Outcome: A student is able to create a reply within a topic that contains the reply title, reply content and a timestamp and post it underneath the main topic.

Actual Outcome 1: The user is successfully able to compose a reply comprised of the reply title, reply content and a timestamp.

Test Result: Test Successful

2.5.11 Reply View Test

Test: If a user can view all of the replies.

Expected Outcome: Students should be able to view all replies to topics that lie beneath the main topic.

Actual Outcome 1: Error: Replies aren't appearing at all let alone underneath the main topic.

Test Result: Test Failed

Actual Outcome 2: Students are now able to view all replies to topics, with the reply title, content and timestamp lying beneath the main topic.

Test Result: Test Successful

Comment: The error was located in the view_topic.php file. I had named the incorrect php connect file for this project I was using a file called connect.php and by accident I had misnamed it to dbconnect.php which was a connector I used on a previous project.

2.5.12 Maps Test

Test: If a users' location can be shown on the map in relation to nci.

Expected Outcome: A user should be able to view their current location in relation to NCI and from there get directions by clicking the directions button.

Actual Outcome 1: Error: Users current location is not appearing when they click on the maps section of the application as the whole map is not loading.

Test Result: Test Failed

Actual Outcome 2: The map is now loading and user's current location is successfully plotted on the map and can be viewed in relation to NCI, user is also able to get directions and eta.

Test Result: Test Successful

Comment: Error was located in the maps gradle on android studio I had inserted a API key from a previous project I was working on and this lead to the file compiling with errors and not running.

2.5.13 To Do List Test

Test: If a user can create a task

Expected Outcome: A user is able to create a task and add it to his to-do list, if they wish to delete the task they simply press done.

Actual Outcome 1: The user is successfully able create a task when they click the create button, they are also able to post the task to their to-do list successfully.

Test Result: Test Successful

2.6 Graphical User Interface (GUI) Layout



LectComm Homepage

This is the launch page for the LectComm app. You are provided with four different options. "LectComm", which brings you to the main functionality of the app that allows you to communicate with lecturers. "Forum", which brings you to the forum page that allows you to communicate with other students through topics and posts about various aspects of the course you are doing. "Map" which shows you where in relation to your college you are, and provides you with directions on how to get there. "To Do List" a simple to-do list that provides a user with a list of things they have set themselves to do.



Login

LectComm - Login

This is the user login for the LectComm section of the application. By entering the login credentials (i.e. username and password) provided to you by the college, and clicking login, you can gain access to said part of the app.

Welcome, **x12378906!** [Logout](#)
[Compose](#) | [Inbox](#) | [Outbox](#)

To:

Subject:

Message:



LectComm - Compose

After the user is logged in they are presented with three options, the first one being Compose where mainly the student would pick the lecturer they would like to send their message to. The message should include the subject and very brief description of the issue they had with the lecturer.

Welcome, **PaulHayes!** [Logout](#)
[Compose](#) | [Inbox](#) | [Outbox](#)

From: Subject: Time:
x12378906 [Slides 21-25](#) 2016-05-10 03:25:23

LectComm - Inbox

After compose the user has their Inbox where for students they can see their replies from lecturers and for lecturers they can see the list of students that have an issue with the current lecture topic. By clicking on the subject the user will be brought to the full message view of the issue sent. Example can be seen below

LectComm – View Message

From: x12378906
Subject: Slides 21-25
Message: I can't seem to get my head around the content covered on slides 21-25.
Time: 2016-05-10 03:25:23
[Reply](#)

This message view allows the user to see the full message sent by another user, rather than just the subject line. It provides the Username, Subject, Message and Timestamp. There is also an option to click reply which allows the user to send a reply message.

Welcome, **PaulHayes!** [Logout](#)
[Compose](#) | [Inbox](#) | [Outbox](#)

To:

Subject:

Message:

LectComm - Reply

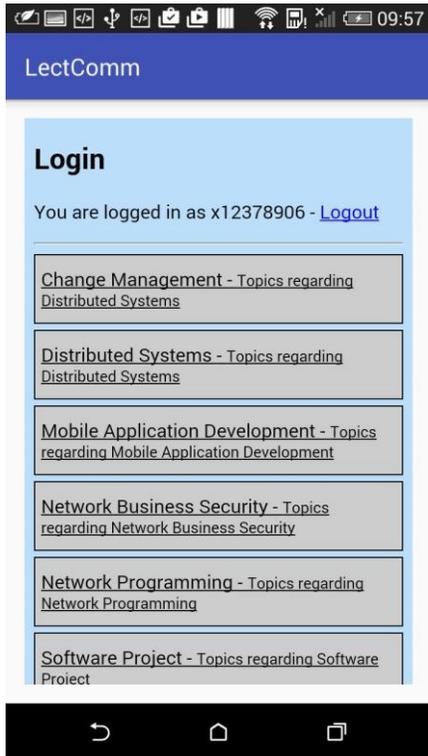
When a user clicks the reply link on the message view page they will be directed to this page. It is mainly used for lecturers toward students, in order to arrange a meeting or hold an extra class. It has the same functionality as the compose section.

Welcome, **x12378906!** [Logout](#)
[Compose](#) | [Inbox](#) | [Outbox](#)

From:	Subject:	Time:
PaulHayes	RE:Slides 21-25	2016-05-10 03:31:51

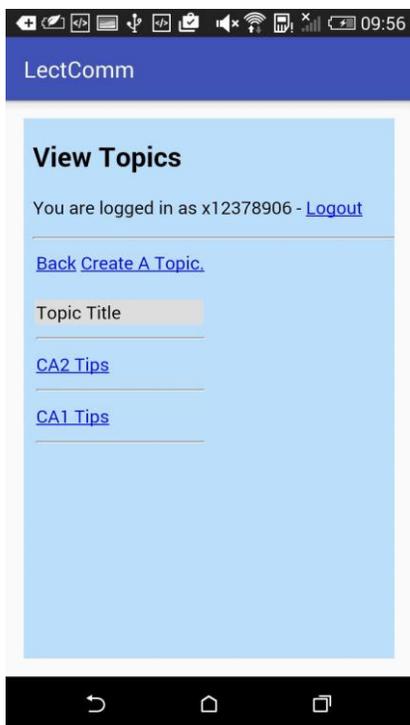
LectComm - Outbox

The user outbox provides the user with every message they have sent whether it be a reply or a main message. This will make it easier for users to keep track for what they sent and when.



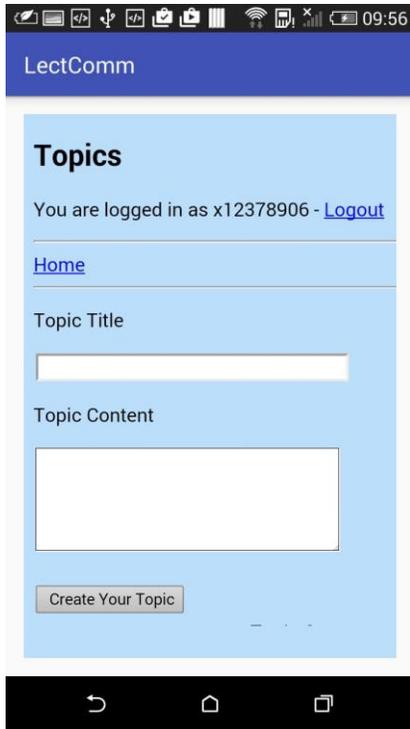
Forum - Homepage

The forum homepage contains two functionalities, one being the login and the other being the category view. The user login is the same as the login for the LectComm section of the app. The category view enables the user to view each module available to them in that academic year, and once a module is clicked they will be brought to the topic view page where they can view user's topics in regards to said module.



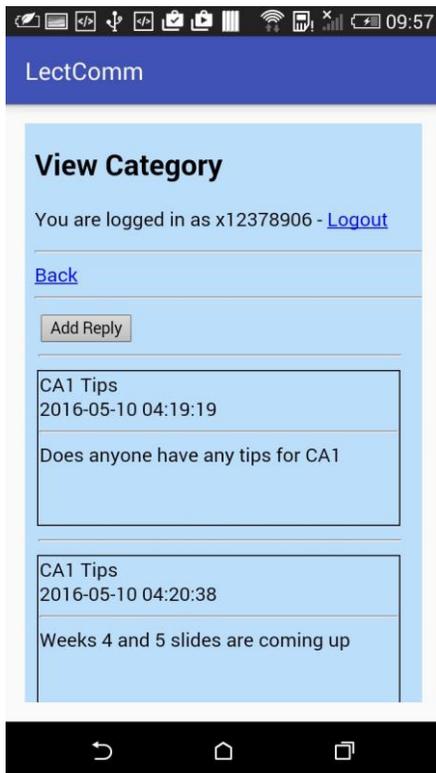
Forum – Topic View

The topic view page contains a list of user created topics that others users can interact with. It is also the page that provides a link to the topic creation page of the forum section.



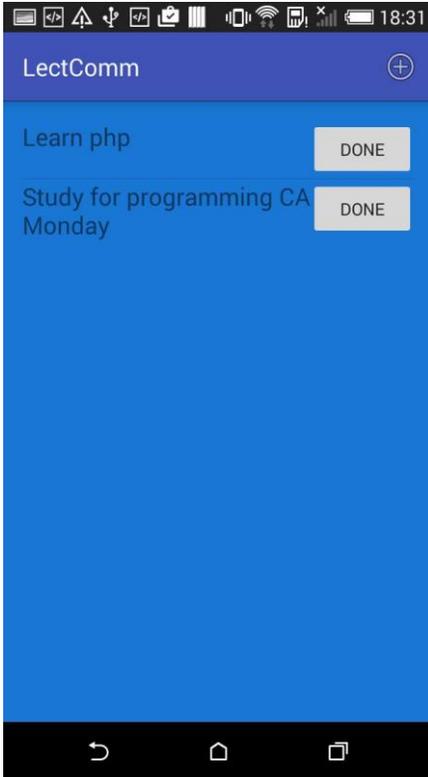
Forum – Topic Creation

Once the user has clicked on the “Create A Topic” link on the topic view page they will be presented with topic creation page. On this page you will be asked to provide the title of the topic and the fill the content section with more information on the topic that you want to be discussed. Once both of these have been filled the user must click “Create Your Topic” to make it viewable on the topic view page.



Forum - Post View

When a topic is clicked on the topic view page the user will be brought to the post view page. On this page every reply post that equates to the topic that was clicked, are listed top to bottom under the main topic. In order to post a reply, the user must first click “Add Reply”, then fill out the form on the reply page.

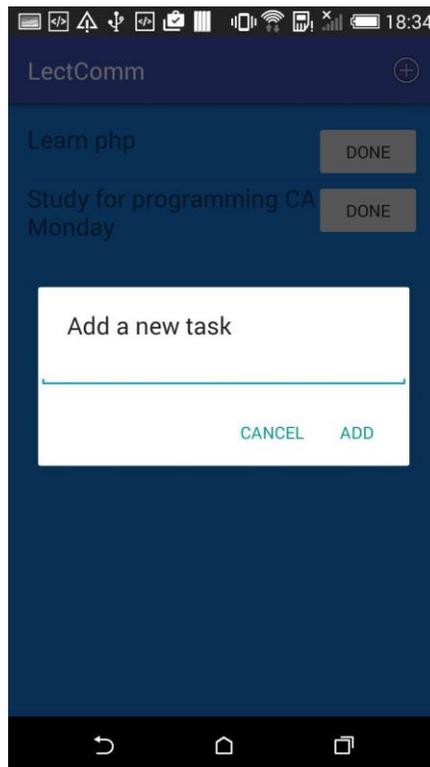


To-Do List

In the to-do list section of the app the user can create task they wish to complete in regards to their college studies. To create a task a user must click the plus button located at the top right of the screen. Once that is clicked the user is ushered to add a new task by typing it in the prompt box provided. It is then posted to the to-do list. If the user wants to delete a task they must simply click the “Done” button.

To-Do List

- New Task



Map Location



This function enables the user to find their current location in relation to the college they work and or attend, The National College of Ireland in this case. By clicking the “BLUE” arrow on the bottom right of the screen, will allow the user to find directions to the college and will provide them with estimated time of arrival depending on what mode of transport they are using (i.e. Car, Train, Bike, Walking)

2.7 Customer testing

For the first part of my customer testing process I decided to try and use the website Reddit.com. I chose to do so because, one, you can do it anonymously and two, send it to a larger subject group that is external to the people you know. Reddit's content is organised by areas of interest called "subreddits". Within these content areas you can post pictures, videos etc. as long as it is in relation to the given "subreddit". The subreddit I chose to post to was reddit.com/r/college. I included the project APK so it could be downloaded on to users' phones and a link to the survey monkey that I created in order to get their user feedback. Due to reddit.com/r/college being used primarily by other students, it was the perfect place to test my app.

As well as getting an external source of customer testing, I also sent the application to students in my year group and the year below. They provided me with more beneficial replies due to actually attending NCI. As well as providing me with a filled out survey monkey, they also gave me verbal feedback which can sometimes, provide for more information and lead to an overall better project.

Within the survey itself I asked the following questions:

What about the application would you change to improve it?

Do you like the look of the application?

Do you agree the application is easy to navigate?

What do you like most about this app?

What do you like the least about the app?

Would you use this app in a lecture?

Do you think this app benefits student and lecture communication?

The answers to the survey questions can be viewed in the Appendix 6.7.

3 Conclusions

I have really enjoyed developing this application. It has proved to be a great challenge and through my attempts to overcome those challenges, I have achieved a greater understanding of different programming languages and learnt new techniques in software development. I had very little knowledge of software development before, specifically in the area of SQL and PHP. I enjoyed working on an application that dealt with problems within the classroom of a 3rd level institution.

By getting user feedback during customer testing I found areas where my app had pros and cons. Pros including that people seemed to like the idea an anonymous way of talking to lecturers about the issues they have in class and the forum functionality that enables students to discuss upcoming CA's and exams. The cons that seemed common were, it was limited to android phones, students without smartphones wouldn't be able to use it, and that lastly be the style was too basic and didn't appeal to some people. There are vailed points on both sides and I used all advice to help make my application the best it could possibly be, in accordance with my ability.

I am however very pleased with the overall outcome of the final application, with its simple design and the variety of functions it offers its users. Despite any disappointments in the development process, overall I am very pleased with my experience with developing this application. I also believe this project has helped me develop a better sense of project management with the deliverables that needed to be handed in on a comfortable frequent bases.

Although I do agree my app is very limited to what can actually be expanded on, it all depends on permissions given to me by the college or lecturers themselves.

4 Further development or research

Firstly, I hope that in the near future to have the application running on iOS and Windows devices in order to broaden the number of available users.

I also believe my application has the potential to be used in more than just third level institutions. If it is found to be an effective app, it can be branched out for use in secondary schools, and the more mature year groups of primary schools. Due to the rise in the use of technology in secondary school classrooms (i.e. smart boards, tablets, laptops), this application can be easily implemented once it is monitored properly, and if users are responsible enough to use it in the correct way.

I hope to have a fully functional web app that provides everything the mobile version does and not just the student to lecture communication.

I would like to make the app a more student assistance app by adding functionality like a Timetable for users to keep track of their classes and studies, a chatroom for students to chat with each other in real time and a system like the student to lecture communication except for the students and the student's union.

5 References

Site Point. (2016). *Starting Android Development, creating a To-do App*. [online] Available at: <http://www.sitepoint.com/starting-android-development-creating-todo-app/> [Accessed 11 Feb 2016].

PHP Tutorial: Private Message System (Part 1/5). (2013). [video] <https://www.youtube.com/watch?v=9g1paLDWIZc&:phpclasses2013>.

PHP Series - Building A PHP MySQL Forum Tutorial Series Introduction (2011). [video] <https://www.youtube.com/watch?v=R37IITAFW1c:TimKippTutorials2011>.

6 Appendix

6.1 MySQL Queries

6.1.1 Users

```
CREATE TABLE `b10_18022027_data`.`users` (  
  `id` INT NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  `username` INT NOT NULL,  
  `password` INT NOT NULL  
) ENGINE = MYISAM;
```

6.1.2 Private Messages

```
CREATE TABLE `b10_18022027_data`.`private_messages` (  
  `id` INT NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  `from_user` VARCHAR (20) NOT NULL,  
  `to_user` VARCHAR (20) NOT NULL,  
  `subject` VARCHAR (400) NOT NULL,  
  `message` TEXT NOT NULL,  
  `date` DATETIME NOT NULL,  
  `read` BOOLEAN NOT NULL DEFAULT FALSE  
) ENGINE = MYISAM;
```

6.1.3 Topics

```
CREATE TABLE `b10_18022027_data`.`topics` (  
  `id` INT (11) NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  `category_title` TINYINT (4) NOT NULL,  
  `topic_title` VARCHAR (150) NOT NULL,
```

```
`topic_creator` INT (11) NOT NULL,  
`topic_last_user` INT (11) NOT NULL,  
`topic_date` DATETIME NOT NULL,  
`topic_reply_date` DATETIME NOT NULL,  
`topic_views` INT (11) NOT NULL DEFAULT '0'  
) ENGINE = MYISAM;
```

6.1.4 Posts

```
CREATE TABLE `b10_18022027_data`.`posts` (  
`id` INT (11) NOT NULL AUTO_INCREMENT PRIMARY KEY,  
`category_id` TINYINT (4) NOT NULL,  
`topic_id` INT (11) NOT NULL,  
`post_creator` INT (11) NOT NULL,  
`post_content` TEXT NOT NULL,  
`post_date` DATETIME NOT NULL  
) ENGINE = MYISAM;
```

6.1.5 Categories

```
CREATE TABLE `b10_18022027_data`.`categories` (  
`id` INT (11) NOT NULL AUTO_INCREMENT PRIMARY KEY,  
`category_title` VARCHAR (150) NOT NULL,  
`category_description` VARCHAR (150) NOT NULL,  
`last_post_date` DATETIME NULL,  
`last_user_posted` INT (11) NULL  
) ENGINE = MYISAM;
```

6.2 Project Proposal

Project Idea

The objective of my software project is to create an application that heightens the communication between both students and lecturers in higher education. The basic concept is during/before a lecture, students and lecturers will be able to sign in to the app using their “log-in” credentials. Students will be able to join their lecturer’s module in the app, and after the lecturer has granted students permissions, the said lecture will be presented with the list of students that have joined the class.

During the lecture students will be presented with a button, labelled “HELP!” students will be able to “tap” or “click” this button anytime throughout the duration of the lecture. By interacting with this button it will send a notification to the lecturer to notify that one or more people have failed to understand what the lecturer is trying to explain. Students will also be able to type a few lines pointing out exactly what part of the lecture they do not understand.

With all of this information at the end of the lecture, the lecturer will be presented with a list of students with the following information; Name, Student Number, Email Address and Issue. If there are multiple students having problems the lecturer may be able to address this issue by organising an extra class or taking a few minutes out of the next lecture to go over the said issues.

However, if the issues about the lecture are isolated to say one or two students, the lecturer may be able to organise a “one to one” session to discuss the issues the student has had.

Reasoning

The reason I chose this idea for my software project is due to my personal experiences in NCI. While communications with some lectures are great, there are others where lack of communication has been the difference between passing and failing grade for my fellow classmates. Although we are in third level education,

I feel as though people still take somewhat of an embarrassment if they don't understand something their fellow peers do. I know that on multiple occasions there have been times where at the end of lectures, lecturers will give an opportunity for students to ask questions or ask if they want to go over anything again. Almost 90% of the time students will not ask questions or interact in anyway, and at the same time these students are asking for extra classes at the end of semesters and complaining that the lecturer wasn't good enough or offering enough help, when this is not necessarily the case.

I feel as though my project idea will allow students and lecturers to interact at a more optimal level and not take away from valuable class time. I believe that this applications concept is easy to follow and when completed will have a very user friendly interface.

Technical Research

From the beginning I have thought that my idea was very unique, but just to be sure I did some researched and found that the only service that was slightly similar to mine is a web page called RateMyTeacher.com. Rate My Teacher gives students the opportunity to rate their teachers based on a star rating ranging from one to five. This service can seem like a great concept, but when I looked further into the rating system I found that it can be used a bullying tool towards teachers, as they do not have the opportunity to monitor their profile or reply to students. Students also have the option to post anonymously, therefore will feel free to say anything without consequence.

If you compare RateMyTeacher.com to my app, you will see that even though you are rating aspects of the lecturers teaching style, I think it's in a more constructive way. The rating isn't as public, it is not rating the lecturer themselves and the lecturer can have more control of their environment.

Technical Details

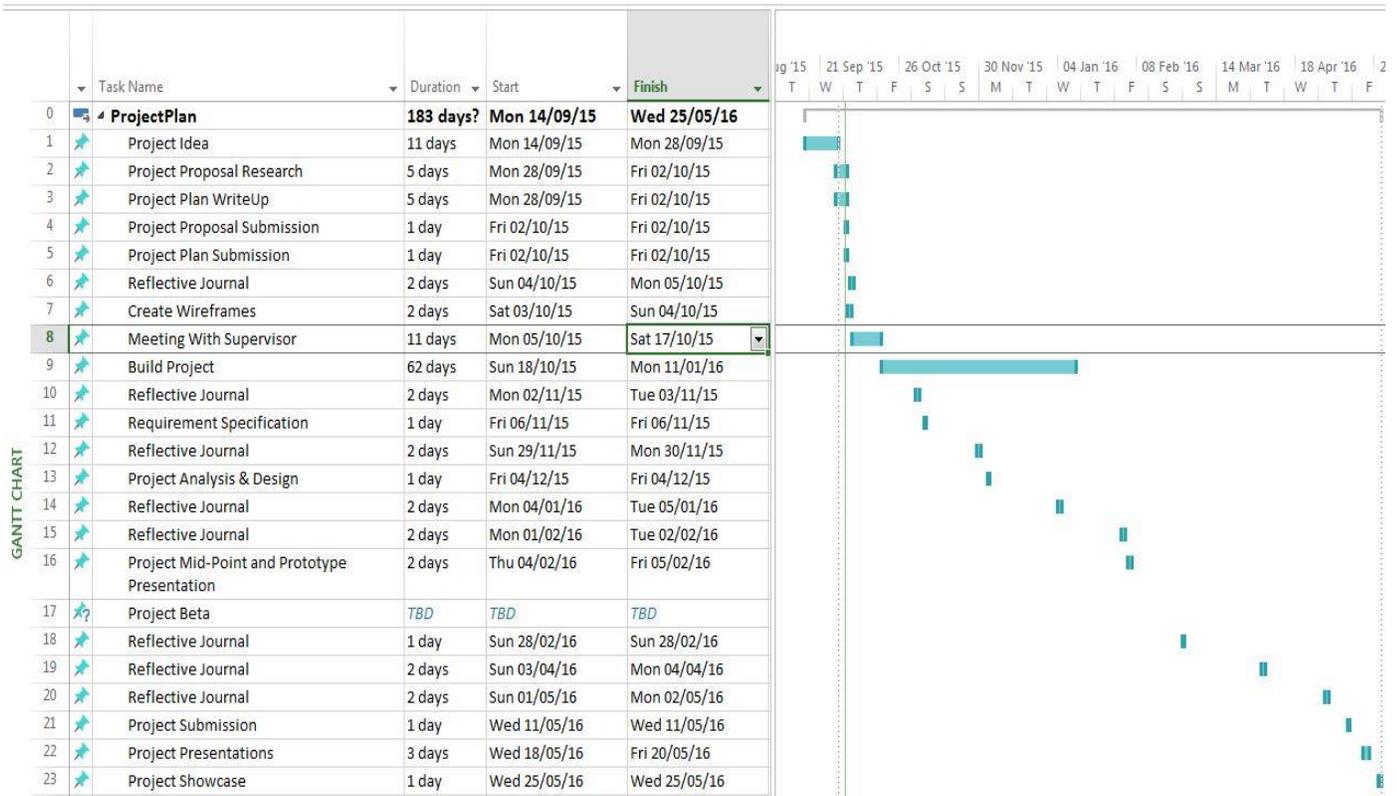
I will be using Notepad++ and Android Studio. The languages I wish to implement into my project are; HTML5, PHP, JavaScript, CSS, Java, Android XML, and MySQL. Firstly, I want to create a web application for my project so that it is available on all devices. When that is near completion I wish to start developing a mobile app for android users. I chose android as I have experience dealing with android and I am more familiar with using JAVA and other mobile operating systems.

The requirements needed to use this app will be an Android Smartphone preferably using Marshmallow 6.0 or a PC, to be a student or lecturer at an institute of 3rd level education, internet access (Wi-Fi or 4g)

Evaluation

The way in which I plan to evaluate my application is to perform a number of tests. I will be using unit testing, integration testing and functional testing. Unit testing will validate small components of my app ensuring they handle input and output correctly. Integration tests ensure that all components work well and are compatible with each other. Finally, functional tests will verify that the end-to-end lecturers and students will be dealing with will be smooth and easy to use. These tests will assist me in enabling my app to reach its greatest potential.

6.3 Project Plan



6.4 Requirement Specification

6.4.1 Requirement 1: User Login

Description & Priority

This requirement is "User Login". Student/Lecturer uses their login credentials to gain access to the mobile or web application.

Use Case

Precondition

Student/Lecturer must have a registered account and possess their correct login credentials.

Activation

This use case starts when a "Student/Lecturer" clicks the "Login" button on the main page of either the web app or mobile app.

Main flow

1. After clicking the login button user must enter their login credentials.
2. If login credentials are entered the user, then clicks button to log in.
3. If the credentials are correct the user will now have access to the application.

Alternate flow

1. User enters wrong credentials and does not gain access to the application.

Post condition

Student/Lecturer now have access to the app and proceed.

6.4.2 Requirement 2: LectComm Compose

Description & Priority

This is what allows student to interact with their lecturer. If a student has an issue with the course content being covered they can simply send their lecturer a message through the app to try and get it resolved.

Use Case

Precondition

Student must have an issue to bring up with the lecturer, either during or after the class has ended.

Activation

This use case starts when a Student wishes to communicate with a lecturer during or after a lecture.

Main flow

1. Student has issue with content that is being covered.
2. Student clicks the "Help!" button on the app.
3. Student then has option to enter exactly what his issue is.
4. Student enters the issue.
5. Student clicks send.
6. Student waits to hear back from lecturer.

Alternate flow

1. Student has no issue with the class, therefore has no need for interaction with lecturer.

Post condition

Student will successfully inform the lecturer of the issues they are having with the topics being covered, and the lecturer will deal with it as they wish.

6.4.3 Requirement 3: LectComm Reply

Description & Priority

This is the functionality that allows lecturers reply to a student who has an issue with the course content that is being covered at that moment in the semester. The Lecturer simply replies with an answer if he/she can solve the issue they are having.

Use Case

Precondition

Lecturer must have pre-existing messages from students in order to be able to reply to them.

Activation

This use case starts when a lecturer wishes to interact with a student that has already sent the lecturer a message with a problem

Main flow

1. Student sends issue with content being covered.
2. Lecturer receives issue.
3. Lecturer pinpoints the whereabouts of issue that has taken places.
4. Lecturer enters the reply to the student with his/her answer.
5. Lecturer clicks send.
6. Student receives information on how the lecturer will deal with the issue.

Alternate flow

1. Lecturer receives no messages from students therefore cannot reply to said students

Post condition

A lecturer will reply to a student, but only if the student has already sent a message. The lecturer will have hopefully come to a conclusion on how to aid said student.

6.4.4 Requirement 4: Forum Create Topic

Description & Priority

This requirement allows the student to create a topic within a category located within the forum functionality. This will allow the student to create a discussion thread and allow other students to reply to them.

Use Case

Precondition

Student must have a topic discussion they would like to post, in relation to college studies (i.e. CAs, Exams, Course Content)

Activation

This use case starts when a student has a topic they would like to have a discussion about, with other members of their year

Main flow

1. Student has a topic they would like to have a discussion about.
2. Student clicks the create a topic button.
3. Student provides a topic title and topic content.
4. Student submits topic.
5. Topic is located in the Category student selected.
6. Other students are able to see and interact with the new topic

Alternate flow

1. Student has no topic to submit, therefore has nothing to send to the topic view page.

Post condition

Student will successfully post their topic to the topic view page, and other students will be able to interact with the said topic.

6.4.5 Requirement 5: Forum Post Reply

Description & Priority

This requirement allows the student to reply a topic located within a category This will allow the student to create a response to the original topic to start a discussion.

Use Case

Precondition

There must be a topic within the topic view for a student to reply to.

Activation

This use case starts when a student wishes to reply to a topic that is currently in the topic view.

Main flow

1. Student sees a topic that peaks their interest.
2. Student clicks on the reply button.
3. Student composes a reply.
4. Student clicks send.
5. The reply is displayed underneath the original topic.

Alternate flow

1. Student has no interest in the topics people have posted and chooses to create his own topic and start a discussion of his own.

Post condition

Student will successfully compose a reply and post it underneath the topic and a discussion begins as more students add more and more replies.

6.4.6 Requirement 6: Map Location

Description & Priority

This requirement allows the user to find their location and directions in relation to the college they are attending.

Use Case

Precondition

User must have geolocation settings turned on in order to track the users' location in regards to the college.

Activation

This use case starts when a Student wishes to know how far away from the college they are.

Main flow

1. Student wishes to know how far away from the college they are.
2. Student selects the map tab on the app homepage.
3. Student must click the find my location button located at the top right of the screen.
4. Student's location is identified and shown in relation to the college they attend
5. Student must click on the direction button located on the bottom right of the screen.
6. Student is provided with directions on how to get to their destination.

Alternate flow

1. User doesn't have geolocation settings on their phone, user will not be able to see their current position on the map.

Post condition

Student will be provided with the most efficient way to get to college, with estimate time of arrival for Car, Train, Bike and by Foot.

6.4.7 Requirement 7: To-Do List

Description & Priority

This requirement allows the user to create a to-do list, made up of tasks created by said user on the app itself.

Use Case

Precondition

The user should have at least one task they wish to add to their to-do list.

Activation

This use case starts when a user wishes to add a task to their to-do list.

Main flow

1. User must have a task they wish to add.
2. User clicks the add task button.
3. User then enters the task they wish to add.
4. User then clicks the add button to add the task to the to do list.
5. User completes task.
6. User clicks done on the task.
7. The task deletes.

Alternate flow

1. A user has no tasks they wish to carry out so they do not add a task to the to-do list.

Post condition

A user will successfully add a task to the to-do list in order to be carried out.

6.4.8 Non-Functional Requirements**Availability requirement**

This application will only be available to a user if they have access to the internet. In regards to the mobile app, it will only be available if 1. The app is not yet downloaded on their mobile device the user will have to gain access to the google play store. 2. If the app is already downloaded the user will only need access to their smartphones interface.

Security requirement

This application will have a password guarded log-in. This will provide constant security of your personal information.

6.5 Monthly Journals

September

This month, my most notable achievement was coming up with my project idea and also being able to create a project plan. I feel that a massive weight has been lifted that I have finalised my idea. In my opinion one of the hardest parts of starting a project is deciding what do you yourself actually want to do and setting out a plan with methods in which you are going to use to complete said project. My contributions to the projects included; conceiving my project idea, researching to see if there were similar applications in the market, also researching to see if my idea was plausible, writing up a project proposal, devising a project plan and starting development on my app. I felt, it worked well to have confidence in myself in being able finalise an idea for my software project. Every idea I had prior to my final decision, I felt as though I either didn't have the skills required to complete them or the concepts just weren't original/complicated enough for a fourth year software project. Next month, I will continue to develop my application. Hopefully if everything goes to plan I will get some form of efficiently working background code for the main interface. I will also begin my Requirements Specification Report. I will also have to organise a meeting with my project supervisor, I will be looking forward to this meeting, as up until now only my colleagues have been able to give me feedback on my application. Even though they like the idea, I would appreciate more critical feedback from one of my superiors. After the meeting I know I will be on a more linear path towards the completion of my project

October

This month I had my meeting with my project supervisor. I explained to him what my project was all about and he gave me a positive go ahead. I then decided to work on the user registration. I started developing my project with the cloud based IDE "Cloud 9". This IDE was very helpful whenever I wanted to work on a project

in college or at home using different computers. The transition between the two was so smooth and reduced the hassle of sending myself files over Dropbox and resulting in errors. The user registration was completed, I created a database using SQL in the Cloud9 terminal, and used PHP to connect it to my existing HTML code for the interface. I feel like this month was very productive even though there was a lot of strenuous continuous assessments due and with the addition of class tests that were looming over my head. Next month, I will start developing the user login as I have the registration already complete I hope to get a fully functional user account system up and running.

November

This month I successfully linked up the user registration and the user login to create a functional user account system that was up and running. I feel like this month just like last month was a somewhat productive one, even though there were still the likes of continuous assessments to be completed and a one or two in class tests towards the middle and end of the month. I was happy I completed the user registration, but I feel a tad bit behind when I compare my project to others in the class. Next month, I will begin developing the main functionality of the application such as the communication between two users such as a lecturer and a student or a student and a student. I hope to get this completed as it would give me a boost in confidence in completing this project.

December

This month I continued to work on the main functionality of the application. I created a space where two users after logging in could communicate between one another. I feel as though I need to start focusing more on getting more of my main functionality completed, I do not feel as though my project is at the state it should be. I hope to rectify this within the coming month. I started to slack on the project work towards Christmas time and the days following I was studying for my January

exams. Next month, I do not intend on working on my project due to my stress levels being at an all-time high dealing with the pressure of studying for exams.

January

This month, I did not give any contributions towards my project due to my stress levels being at an all-time high dealing with the pressure of studying for exams. I feel as though I should have still dedicated more time to project during the holidays and coming up to the exams, as it would have been more beneficial to me coming up to the mid-point presentation. Next month, I will be preparing for my Mid-Point Presentation and finalising my project prototype for the presentation itself.

February

I finished my first prototype of my application for my mid-point presentation, finished the map integration and to-do list. I feel as though my midpoint did not go as well as I expected it to, but in hindsight it was all my fault as I failed to prepare a good enough project to present and my documentation wasn't at the stage it should have been. I intend to move it up a gear and start to balance college studies and my final year project more evenly because at the end of the day they all go towards my degree.

March

This month I finished the LectComm functionality and Forum functionality. This month I decided to get the two main functionalities of my app out of the way so I could focus on my exams and CA deadlines. I started by creating the forum functionality this was a difficult task as I changed how I wanted to go about doing it quite a number of times. This was also the month I decided to switch from using cloud9 to using an actual free hosting service, which I found in Byet.Host. I had to create all the database tables from scratch and create new source files for things

such as the log in functionality. I felt my project was coming together more as whole now that it was hosted on a registered domain.

April

This month I had my exams and put the finishing touches on my project and technical report. Not much else could have been done this month as I had exams and other module projects due in. But once they were finished I started to tidy up my project and the subsequent report, with things like adding better style to my app and for the report I spent time carrying out my customer testing using a survey monkey feedback survey.

6.6 Survey Answers

Q1: What about the app would you change to improve it ?

for it to be available on iphone

Q2: Do you like the look of the app ?

Like

Q3: Do you agree the application easy to navigate ?

Neither Agree or Disagree

Q4: What do you like the most about the app ?

the communication for students

Q5: What do you like least about the app ?

its not available on an iphone

Q6: Would you use this app in a lecture ?

Yes

Q7: Do you think this application would benefit the communication between both lecturers and students ?

Yes

Q1: What about the app would you change to improve it ?

To be able to edit the notes you create in the to-do list.

Q2: Do you like the look of the app ?

Strongly Like

Q3: Do you agree the application easy to navigate ?

Agree

Q4: What do you like the most about the app ?

To be able to communicate with a lecturer on a one to one with out having to deal with the formality of email.

Q5: What do you like least about the app ?

the way it looked wasn't appealing to me

Q6: Would you use this app in a lecture ?

Yes

Q7: Do you think this application would benefit the communication between both lecturers and students ?

Yes

Q1: What about the app would you change to improve it ?

The style of it could be improved.

Q2: Do you like the look of the app ?

Neither Like or Dislike

Q3: Do you agree the application easy to navigate ?

Strongly Agree

Q4: What do you like the most about the app ?

You can discuss topics with other students such as upcoming CAs

Q5: What do you like least about the app ?

The topics in the forum could be more structured.

Q6: Would you use this app in a lecture ?

Yes

Q7: Do you think this application would benefit the communication between both lecturers and students ?

Yes

Q1: What about the app would you change to improve it ?

I would allow the lecturer to be able to have control over who sends him/her messages.

Q2: Do you like the look of the app ?

Like

Q3: Do you agree the application easy to navigate ?

Agree

Q4: What do you like the most about the app ?

The ability to communicate one on one with a lecturer without others in the class knowing.

Q5: What do you like least about the app ?

That there is no user profile.

Q6: Would you use this app in a lecture ?

Yes

Q7: Do you think this application would benefit the communication between both lecturers and students ?

Yes

Q1: What about the app would you change to improve it ?

For it to be a fully functional website as well as a mobile application.

Q2: Do you like the look of the app ?

Like

Q3: Do you agree the application easy to navigate ?

Neither Agree or Disagree

Q4: What do you like the most about the app ?

The communication between the student and lecturer seems to be the main function.

Q5: What do you like least about the app ?

I don't have a smartphone.

Q6: Would you use this app in a lecture ?

No

Q7: Do you think this application would benefit the communication between both lecturers and students ?

Yes

Q1: What about the app would you change to improve it ?

i like the way it is

Q2: Do you like the look of the app ?

Strongly Like

Q3: Do you agree the application easy to navigate ?

Strongly Agree

Q4: What do you like the most about the app ?

its lectcomm and forum parts

Q5: What do you like least about the app ?

nothing really

Q6: Would you use this app in a lecture ?

Yes

Q7: Do you think this application would benefit the communication between both lecturers and students ?

Yes

Q1: What about the app would you change to improve it ?

The colour scheme

Q2: Do you like the look of the app ?

Dislike

Q3: Do you agree the application easy to navigate ?

Agree

Q4: What do you like the most about the app ?

Both the forum and "LectComm" sections stand out the most.

Q5: What do you like least about the app ?

The colour scheme.

Q6: Would you use this app in a lecture ?

Yes

Q7: Do you think this application would benefit the communication between both lecturers and students ?

Yes

Q1: What about the app would you change to improve it ?

Take out the to-do list it seems pointless.

Q2: Do you like the look of the app ?

Like

Q3: Do you agree the application easy to navigate ?

Agree

Q4: What do you like the most about the app ?

The core idea has a lot of promise.

Q5: What do you like least about the app ?

The to-do list.

Q6: Would you use this app in a lecture ?

Yes

Q7: Do you think this application would benefit the communication between both lecturers and students ?

Yes