

## **Chapter 1: Introduction**

### **1.1 Motivation and Problem Statement**

The evolution of the internet technologies had benefit people to accessing to the web easily. More and more services provide by this internet All of this can be virtualize thank to the technologies. Communication between people using the internet becomes part of their daily life. People used to communicate with each other's using the online chat system to transfer their messages. Traditionally, when people need to communicate with others they will have a face to face conversation to deliver the message, same goes to the education field. It is strongly encourage that student seeking for academic assistance from the lecturer when they face difficulties. Most often happening is when the exam are in the corner and or assignment due date. The traditional way to have a consultation is student make an email appointment with the lecturer and the lecturer accepted the appointment or lecturer is free and available at his room or lecturer consultation hour. However, this communication solution might be not convenient and not efficient due to some issue that happen before the consultation started. The consultation session can be realized in another similar way using an online solution.

There are few issue might be arise when the student want to have consultation with the lecturer using the traditional method which is making appointment and meet at a lecturer room to having the communication.

#### **1. Student or lecturer are not available in the school**

Sometime, the lecturer and student might have some personal issue and cannot come to school. For example the lecturer has to outstation for some important works and the outstation will take a period of time. Also student cannot come to school because they are not available in the area that near to the school. During this period of time, if a student wanted to have a consultation session with their teacher, they cannot meet the respective teacher in the university which causes the student delay their studies progress.

#### **2. Bunch of email have to filter**

Most often lecturer mailbox will be full of emails. Lecturer have to manually filter out which emails is regarding the consultation requests. It will require the lecturer efforts in filtering the emails. In the filtering process, lecturer might be missed out the emails regarding the consultation.

**3. Queue issue regarding the appointment**

Sometime human error can be easily made by the lecturer when scheduled the consultation session. The lecturer need to be extremely caution when scheduled the consultation session. It is possible lecturer forget and scheduled two consultation session at the same time. If it is happen, two students will came at the same time. Based on first come first serve, the second student has to wait for the first student to finish his consultation session then finally his turn. It is possible the second student have to wait for a longer time if the first student requires lot of times with the lecturer.

**4. Fail to recall about the consultation**

It is human nature that forgets some stuff from time to time. It is possible that either student or lecturer have totally forgotten about the consultation. If it is happen, these will caused another parties to wasting time and wait. Besides that, it is possible that either student or lecturer have forgotten what is the context or topic will be discuss in the later appointment. This is because student or lecturer has totally forgotten about there will be a consultation session later.

Therefore, the above issue can be enhanced using an online consultation solution.

## **1.2 Project Scope**

This project will be developed in web based. The project is planned to introduce an online web chat system solution for the student and lecturer. The project included an appointment system that will be handling all the appointment between the student and the teacher effectively. Furthermore, a real time communication chat system will be included as the feature in the project to make a face to face communication channel between the student and the teacher. There are additional feature included in the project such as the screen sharing that share the current screen of the user, file sharing and text chat. Besides

that, a SMS reminder system will be developed to inform the student or teacher about they will having the consultation session later. However, the real time communication web application will not be cover on the mobile site.

### **1.3 Project Objectives**

This project aim to build an online chat system that has the capability to have real-time communication using web browser. WebRTC approach will be implemented for achieve this real time communication. WebRTC approach does not require any additional plug in to the browser with the constraint that the browser support webRTC approach such as Google Chrome. However the screens sharing in the WebRTC approach require install a plugin.

#### **1. To provide an online consultation solution**

The traditional way to have consultation might face some issues such as availability of the student and lecturer in the school. Furthermore, if the student need to show their work on their computer they can show in here because some student might only have a desktop not a laptop it is not convenient to bring the entire desktop to the school. Although external hard drive can be used to store and show the work in the teacher computer but what if the work require some specific software to be open that why it is not convenient. Therefore, an alternative online solution is proposed to help to resolve some issue such as the location issue and making the consultation more convenient.

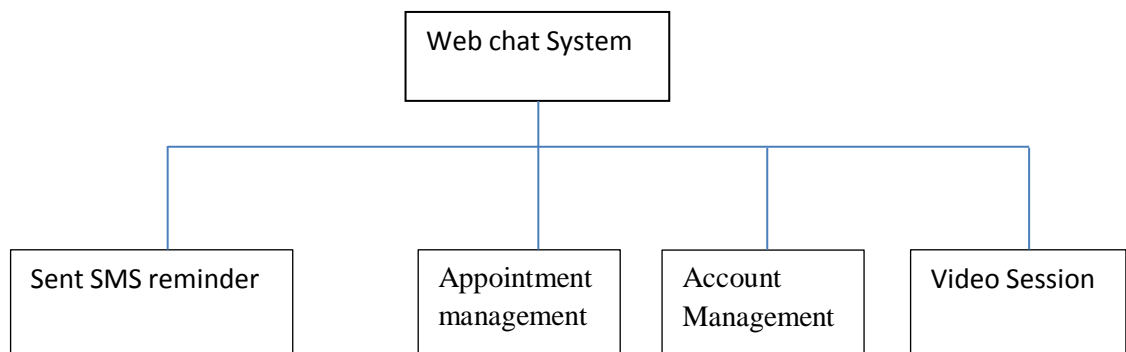
#### **2. To provide a more systematic appointment system**

A systematic appointment system will help in manage the appointment more effective and efficient. This appointment system is convenient because a template is provided to completed and sent. Both student and teacher can check the appointment template including checking the status of the appointment whether is accepted, rejected or pending. If the current timeslot already book by another student, the appointment made will not be successful. This also will minimize the lecturer effort to filter email regarding the consultation in their mailbox. Therefore, a systematic appointment system will help to reduce the effort in managing the appointments.

### **3. To provide a SMS reminder to remind the lecturer about the consultation**

A SMS reminder will be sent to both parties, lecturer and student to remind them about they will going to have a consultation later. This will ensure that both parties will not miss out the appointments that have been made before. If student or lecturer have forgotten about their appointment, when the SMS reminder come to their phone, he can login into the system and check what is the discussion will be having later in the consultation session. This also gives sometimes for the teacher and student in preparation of the material that need to be discuss in the consultation session later. Therefore, a SMS reminder system will help to increase the punctuality for the consultation.

#### **1.4 Proposed approach**



**Figure 1-4-F1 System Functionality**

#### **Appointment management**

The system enable user to make an appointment. If the user is a student, he or she only able to make an appointment with the lecturer. In similar way, if the user is a lecturer, he or she only able to make the appointment with the student. Furthermore, the user can view the detail and time for the appointment. Besides that, the user able to delete the appointment that has been made previously. Moreover, the user also able to accept or decline the appointment that has been requested. Next, the user also can view the appointment that have been rejected and delete the appointment.

#### **Sent SMS reminder**

The system will automatically schedule an SMS after the appointment have been accepted by the user. The system will sent the SMS reminder to remind the both student and lecturer that a consultation session will be held later.

### **Video Session**

The system enabled the user to have a video conferencing and text chat with another user. Moreover, the system also allowed the user to share his current computer screen with another user. Next, the system also allowed the user to share the file with another user.

### **Account Management**

The system allowed the user to register and create an account to login into the system. If the user wanted to edit the information of the account, the system allowed the user update the account information. The systems also enabled the user the change the old password to a new password but the user must able to enter the old password correctly for the confirmation.

### **1.5 Impact, significant and contribution**

The traditional method of consultation have a constraint with is the location. Both side either student or teacher may have issue regarding meeting at a specific location. With this online web portal, this will going to benefit the student and teacher having the real time communication using the internet at any location. This project is developing using webRTC which can simulate face to face communication through browser to browser. With the use of webRTC, both student and teacher no need install additional plugin when start their video session.

Students can always search for the teacher that they wanted to have consultation. If the teacher has registered in the system, the student can add the teacher in the list. After the student add the teacher in the list, student can always make an appointment with the teacher in this web portal. There is an appointment template to let the student select the date and time with additional remarks can be filled to inform the teacher precisely about this appointment is focus on what topic. The appointment template included a dropdown list that let the student chose the teacher that they added to the list to make an

appointment. With this appointment feature, it will benefit the teacher in having additional time for preparation of the consultation materials.

Teacher can always check the appointment list that show which student that have sent an appointment. Teacher can choose whether accepted or rejected the appointment made by the student based on the date time and reason for the appointment. Either accepted or rejected by the teacher, student can know their status of the appointment after the teacher click the button of accept or reject. Traditionally, the teacher have to check the appointment email every day ,sometime there maybe happen that the teacher miss out some of the appointment email made by the student for consultation. This can reduce the needs for the teacher to check which appointment email is and the teacher may have to replay the email for few times before the consultation can be settled down. The SMS reminder will be sent to remind both student and teacher about the appointment.

## **1.6 Project Background**

Sometime, student will have issue with understanding certain topic and content. With the growth of the internet, Student will generally go to the internet and search for the solution. In worst cases, they might get some incorrect information that result in misunderstanding concept the entire topic. The best solution for this issue is to consult the academic illustrators. Teacher can provide the correct information and give clearly explanation make the student fully understand the topic.

Traditionally before the student went for the consultation, there are basic several procedures they might goes through. Firstly, the student probably has to make an appointment with the teacher. The appointment can be made using call, sent message, email and so on. The appointment content normally contain this few basic details such as date, time, venue, and the purpose for making this appointment. After the appointment approved by the teacher, the student can proceed to consultation.

In this project, SMS reminder is implemented to help both user to remember they will have a consultation session later. Furthermore, webRTC will be implemented to simulate the face to face communication. W3C have developed a communication standard which is the webRTC that have similar to the standard that delivered by the IETF which is the

RTCWeb. The idea of WebRTC came when Google Chrome team realize that no satisfactory solution for real time communication when they working on functionality discrepancies between web and native desktop. In the old generation, RTC required either flash or plugin. Flash require server listen to run and low in quality in term of outcome. Plugin is created by the developer and the users have to install it before using it. Plugin is hard to maintain due to the different version of the browser or different operating system. Incompatibles real time communication can be solve by the webRTC. The general idea of the webRTC is the web browser can act as an application where the communication between browsers can be realized with no additional required of download and install software. All the requirements for supporting the standard are embedded in the browser themselves. The ability for audio and video communication, text communication and file sharing can be realized in the webRTC.

## **Chapter 2 Literature Review**

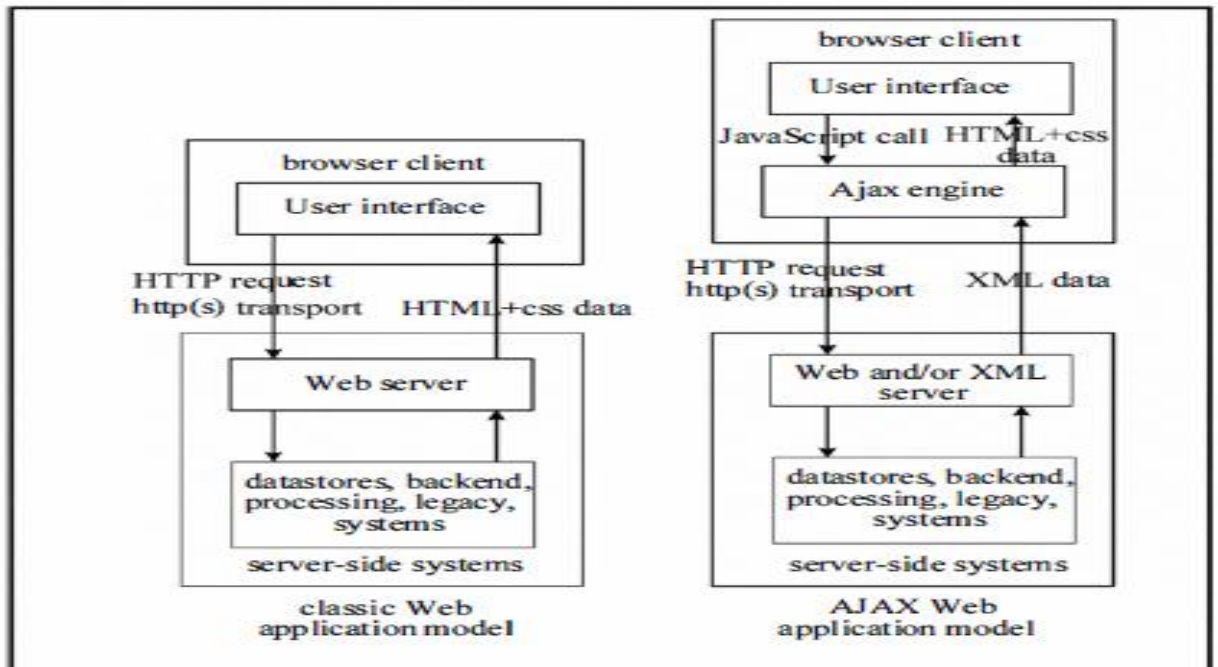
The approaches and techniques to develop the online consultation solution are crucial in this project. This chapter will be more focusing on the relevant literature review or relevant works that previous done by others as a reference to assist in building this online consultation solution.

### **2.1 The Web Asynchronous Communication Mechanism Research Based on Ajax**

This Paper is talking about the Ajax Web Communication mechanism technical feature. Client and Server Communicate directly by sending the service request through the browser is the classical model of Web application. Client is in the wait state when HTTP service request is processing. Furthermore, after the server handles the request completely it does not preserve the user information. This causes the service request to reprocess and regenerate the same page framework. There are three disadvantages of this classical model such as the server will processed all the service requests, transmission of same data will repeated on the network, and the information that the client truly needed will reload slowly.

Ajax engine is the middle layer removing the coupling between the client and server in the Ajax Web application model. The Ajax engine can asynchronously communicate with the browser and the server. The data sent by the client will be processes by the Ajax engine before sent to the server and only when new data return from the server will reload the changes required parts in the client page, otherwise the client page will be unchanged. This help to reduce the server burden and decrease the waiting time of the client. One of the main technical features of the Ajax is using the XHTML together with the CSS. Furthermore, DOM data is implemented to display the dynamic and interactive of the webpage. Moreover, the operation and exchange of data is using the XML and XLST. Next, the asynchronous data management is using the XmlHttpRequest object. Besides that, JavaScript is implemented for all the operation of the data. Below is the diagram that illustrated the different between the traditional model for web application and the Ajax model for web application.





**Figure 2-1-F1: Comparison of traditional model and Ajax model web application**

## 2.2 Node.js: Using JavaScript to build High-Performance Network Programs

Node.js is a server-side JavaScript environment. Node.js also called as Node. The aptly call as “V8” engine is based on Google’s runtime implementation. The purpose of node is to support long running server process and V8 engine used to support mainly the JavaScript in the browser.(Tilkov,Stefan &Vinoski,Steve ,2010) Node is based on an asynchronous I/O event model to support concurrency process. Event model at language level is supported by Node. This can be realized because the JavaScript contain functionality to support callback events, therefore it is excellent to implement in this approach.

Efficiency, scalable alternative offered by the event-driven programming gives much more control over the application activities switching that benefit the developers. The event handling is through the application register certain event and the notification system will notified. The advantage of using event-driven programming with asynchronous I/O is because of the prevention of the application blocked while waiting an I/O event. Despite there is advantages using the asynchronous I/O, there will have

disadvantage. One of the disadvantages that using asynchronous I/O is event notification facilities is not suitable to be implemented by all inter process-communication approaches. Furthermore, the second disadvantage is the sheer complexity of writing application in certain programming when handling the event and asynchronous I/O due to the different types of events. The different actions in different context have to be used to handle different types of events.

Stefan and Steve believe that there are number of factors that make server-side JavaScript enable use for all situation of web-based distributed application.(Tilkov, Stefan& Vinoski, Steve ,2010 ) One of the factor is the advent set of “HTML 5” technologies reduce the appeal of alternative client side platform. To create rich interface for user, JavaScript knowledge is important. JavaScript can be used to defined data view and filter criteria for those non-SQL type databases. Some dynamic programming languages become acceptable choice for server-side development such as Ruby and Python.

Node’s I/O approach is rule. Function take function as parameter, there are case that convenient function implemented for work synchronously.

```
var sys = require("sys"),
    http = require("http"),
    url = require("url"),
    path = require("path"),
    fs = require("fs");

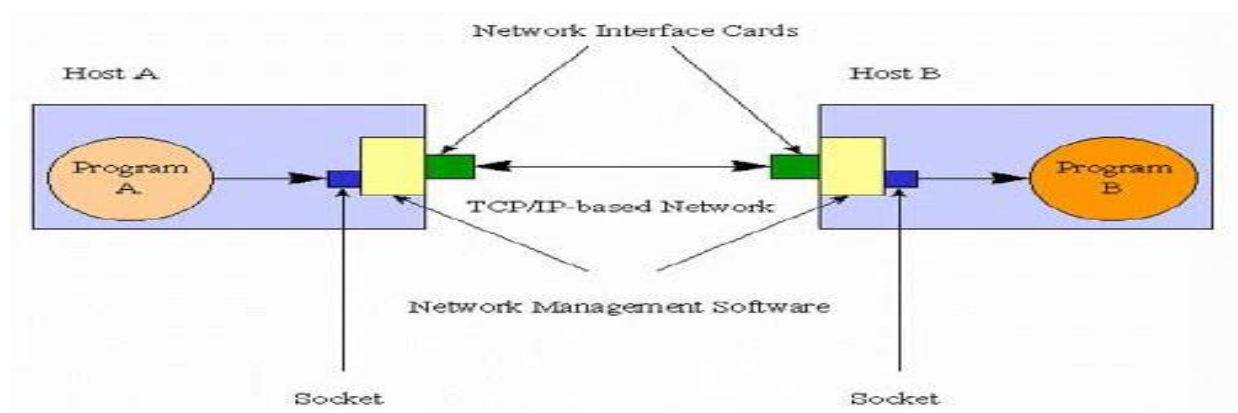
http.createServer(function(request, response) {
  var uri = url.parse(request.url).pathname;
  var filename = path.join(process.cwd(), uri);
  path.exists(filename, function(exists) {
    if(exists) {
      fs.readFile(filename, function(err, data) {
        response.writeHead(200);
        response.end(data);
      });
    } else {
      response.writeHead(404);
      response.end();
    }
  });
}).listen(8080);
sys.log("Server running at http://localhost:8080/");
```

**Figure 2-2-F1: Code of simple file server running on the node**

A node server process generally call by command and runs as a single thread. The main business logic is not execute parallel with Node process despite the operating system have the capability to do it. Node supports server-side JavaScript development framework and environments. A whole ecosystem of libraries has been created by the community for the Node. Node-MySQL or node-couch dB is tools that help to perform the asynchronous interaction with relation and NOSQL data store. The Node package manager is the way to for installation the libraries and their dependencies. Connect and Express is the example of frameworks that provide full feature of web stack.

### 2.3 The Design Method of Network Chat System Based on Socket and Cloud Computing

This paper is talking about way to enhance the communication chat in terms of the communication data stored more securely and efficiently. The implementation approach is using socket technology together with the cloud technology. Socket technology is refer to two program located within one communication link endpoint. The socket will bring the request sent by a program to another socket using the TCP/IP based network. (CAI, MinQiang, 2012) Below is the diagram show the socket key technology.

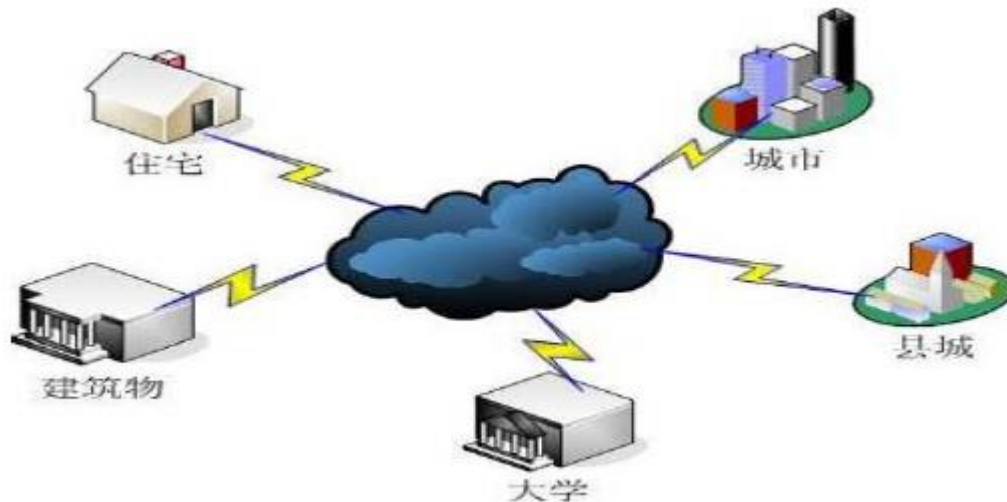


**Figure 2-3-F1: Communication between Sockets**

Java contains Socket-type and Server Socket-type for the implementation of the Socket communication to manage the user request and response. The use of the socket is to set

up the network connection between the sockets. An instance of Socket will be create on both ends by the program when the connection is success.

The combination of distribution processing, parallel processing and grid computing is the cloud computing. The analysis of the distributed computer is the fundamental of the cloud calculative because the process of the center of data is work similar as Internet. The application that required the resources switch on have to access to the storage of the computer system. Cloud computing has the blue print of vividly protrayed because it only need to depend on the internet service which use for tranmission of the data. The cloud computing enable the world work together because every member that have the access can use the certain cloud. Below diagram show the cloud computing concept.



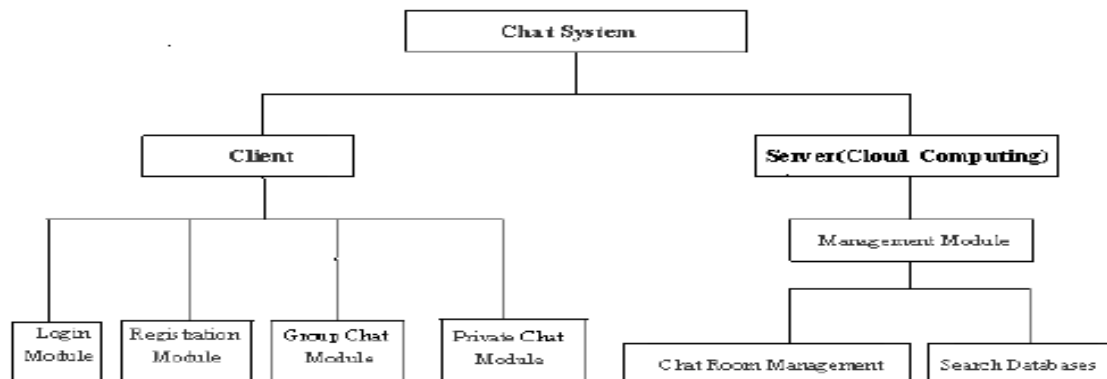
**Figure 2-3-F2: Cloud storage of cloud computing**

The cloud will act as storage and some computing task can be done on it. With this feature of the cloud, everyone just required an internet device that can connect to internet without consider about the amount of storage or computing and people would not worry about the data or information will loss cause they are already store inside the cloud.

The system design is using the java component which is AWT and swing which introducing the multithreading that executed the concurrency situation by implementation connection to the database using JDBC database connection and managed the database.

There are few function requirement of the system. The first functional requirement is the

function of user login which is let the user to login using their user name and password and chose the chat room to enter. The second functional requirement is the user registration which adding account by fill in all the particular information to the database. The third is the function of chat that categories by 2 which is group chat and private chat. The last function is the warning and kicking function. The administrator can warn and kick the user from the chat room. The system structure design of the chat system is divided to two parts. The first part of the chat system design is the client side. The client side handles login module, registration module, group chat modules, and the private chat module. The second part of the chat system design is the Server side. The server was design using the cloud computing. The server will handle the management module of the chat system such as the chat room management and the search database. Below is the diagram showing the system structure design.



**Figure 2-3-F3: Client and server system structure diagram**

## **2.4 Web-based Interactive Virtual Classroom using HTML5-based Technology**

This paper is talking about web-based interactive virtual classroom with new approach. This online learning approach will include face to face video conferences that allows student and instructor communication online. Various systems like presentation tools, online whiteboard, real time video and screen sharing have been included in this virtual classroom and because of the framework for real time participation tools some interactive

feature can be extended such as quiz system and group system. The system also support in offline situation.

The system is run on web-based application. Some of the web technologies have been implemented in the system. Firstly is the Node.js and Websocket. According to Nattha Buasri, Tanasak lanpan, Uarn Yamborisut, Damras Wongsawang, the system is fully written in JavaScript language because we utilize the node.js to create server side system. (Buasri N, Janpan T ,Yamborisut U ,Wongsawang D ,2014)Node.js is a software platform that in used to build scalable networking (server side) application. ( Ryan Lienhart Dahl, 2009) Node allow non-blocking I/O and event driven model to handle high throughput and performance.( Ryan Lienhart Dahl, 2009)The system have several functions that implemented using Node.js in server side such as web application framework, AJAX and MySQL database. Node.js is well supported with WebSocket.( 1. Fette, A. Melnikov, 2011) Web socket have been implemented to support real time approach on the system. Improvement of request/response model and full-duplex communication can be enable are because of the design of Websocket. ( 1. Fette, A. Melnikov, 2011) Socket.io has been used to apply Websocket and Node.js.

Furthermore, the project implemented new technology known as WebRTC to create video conferences system. WebRTC is an API definition being drafted by theW3C to enable browser-to-browser applications for voice calling, video chat, and peer-to-peer (P2P) file sharing without external plugins. (Sam Dutton, 2013) Moreover, two of the web service also implemented to integrate to support online material system which is Google Drive and YouTube. Google Drive API gives an efficient way and is reliable to create file management and sharing pool of file such documents and presentation without external storage. YouTube API provides a high performance system for video sharing. API (Application program interface) is widely used in web development.)

Web-based interactive Virtual Classroom (WebIVC) is the approach of the e-learning and virtual classroom, multiple users can access on device and web based software without additional plugin or tools. (Buasri N, Janpan T ,Yamborisut U ,Wongsawang D ,2014) The system contains of five components such as authentication, classroom, conferences, participation tools, and lastly is summary report. Below is the diagram showing the

relationship and interaction of Web-based interactive Virtual Classroom (WebIVC) system.



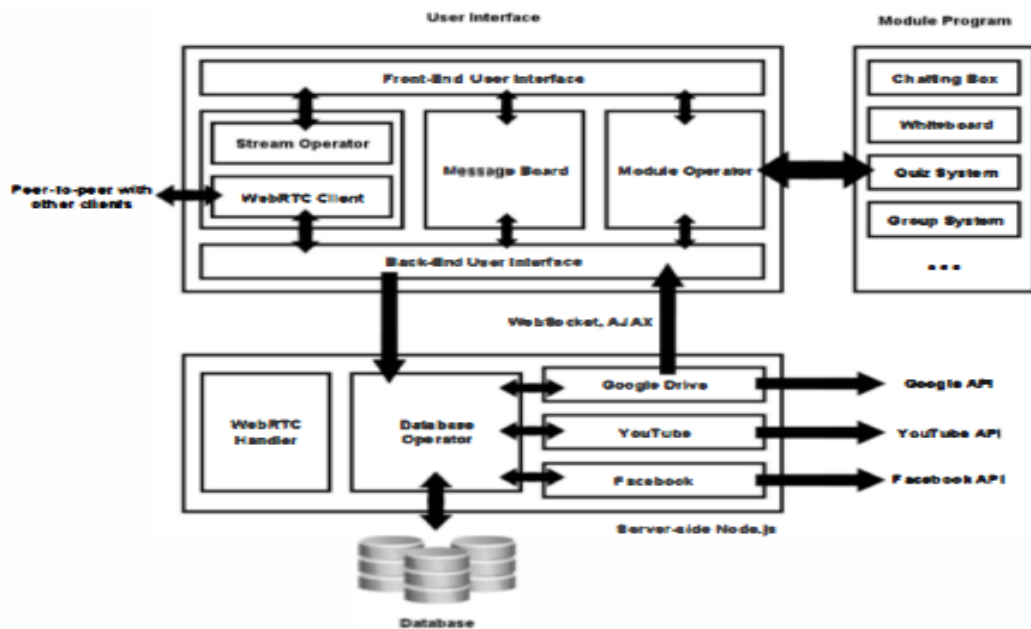
**Figure 2-4-F1: Flow diagram for WebIVC**

The authentication component is used to check whether the user is logged into the system. User account is the key to log in to the system. Because of the social site API provided fast and efficient for simple authentication system, they implement and utilize the social site API for this authentication process. (Buasri N, Janpan T, Yamborisut U, Wongsawang D, 2014) The second component is the classroom. In this classroom component, it contains some of the functionality such as classroom management, Participant management, message board, Google Drive, YouTube. It allows the user to manage the classroom information and provide role to each participant. Furthermore, the user can post various media on the message board in the classroom component. The AJAX technique has been implemented for the Google Drive authentication. Registration to Google Developer console is needed when using the Google Drive API for obtaining the credential information. However, YouTube player API does not need to register, except uploading.

The third component is the Web Conference. The virtual classroom and the various real-time media is virtualized under this component. This component supports multiple face to

face video conferences. The webRTC only support two peers when sharing the conference. To support multiple conferences, additional algorithms have to be implemented such as A communicated with B, B communicate to C and C communicated to A. In order to exchange the communication information, a web server is required. This communication process calls JavaScript Session Establish Protocol and implemented through the Session Description Protocol. The STUN/TURN server has been used if the interactive connectivity establishment (ICE) method has been fail. ICE is introduced to solve the problem of the different IP address for data exchange. For the screen sharing, the web server must support with the SSL because it only work on the SSL domains. The next component is the Participant tool. This component functional contains the user interactive features such as whiteboard, real-time chatting, whiteboard, emotion, and group system between the users. The last component is the summary report. This component introduces to record the WebIVC activities into a statistical report purpose for analysis. Some data such as visitor rate, student ranking attendance checking is recorded in this report.

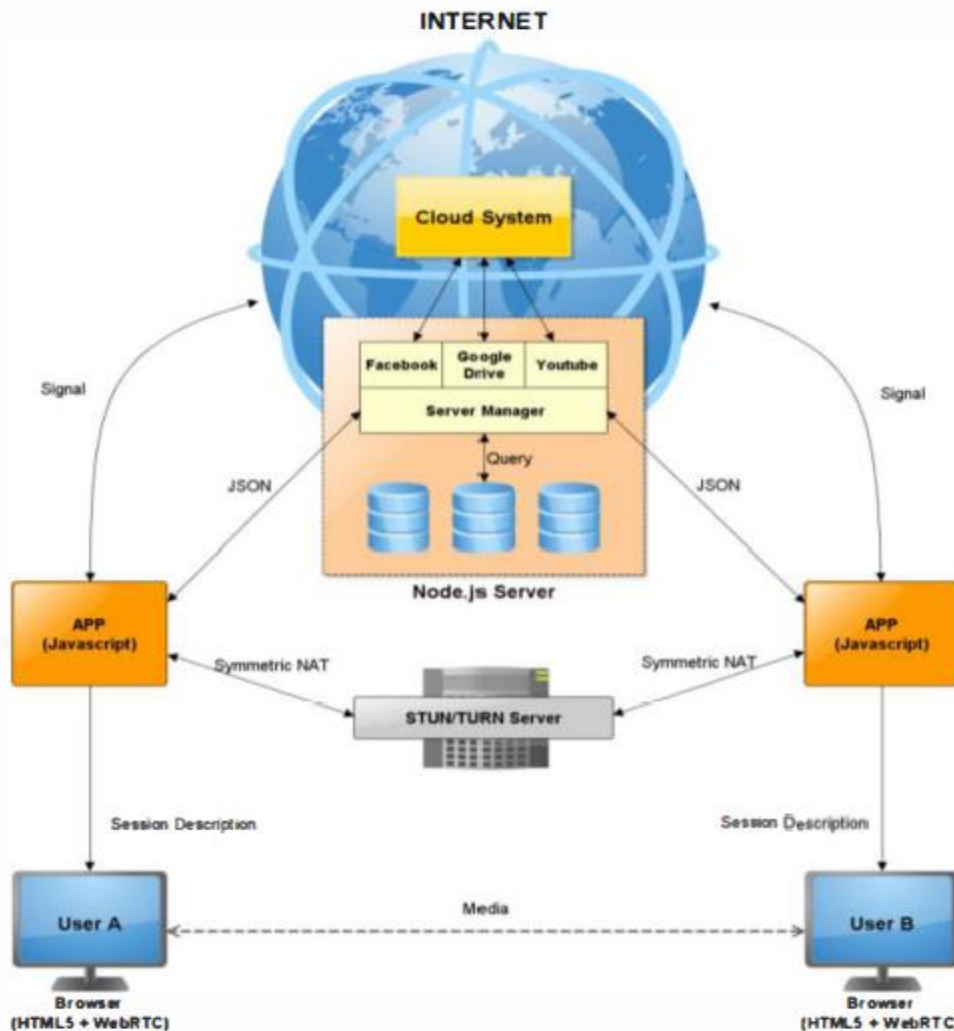
The application is design based on a system model. The module program is a collection of features and interactive tool which are control by the module function and can be extended .





**Figure 2-4-F2 System design of WebIVC**

This software is developed using the JavaScript for both client side and the server side. JQuery is the common library that has been implemented to develop with JavaScript, HTML and CSS. While in the server-side, Node.js is implemented for responsibility to create a connection between the web server and the database. The Node.js also supported the API service that mention above. The WebRTC API is implemented for creating the web conference. The JSON become the common data format for the data exchange between the client and the server. The template and session management are implement through the Express.js which is a external library of the Node.js. The webs-based design and the application layout are enhanced by using the stylus.



**Figure 2-4-F3: Architectural diagram of WebIVC**

## Chapter 3: System Design

### 3.1 Use Case

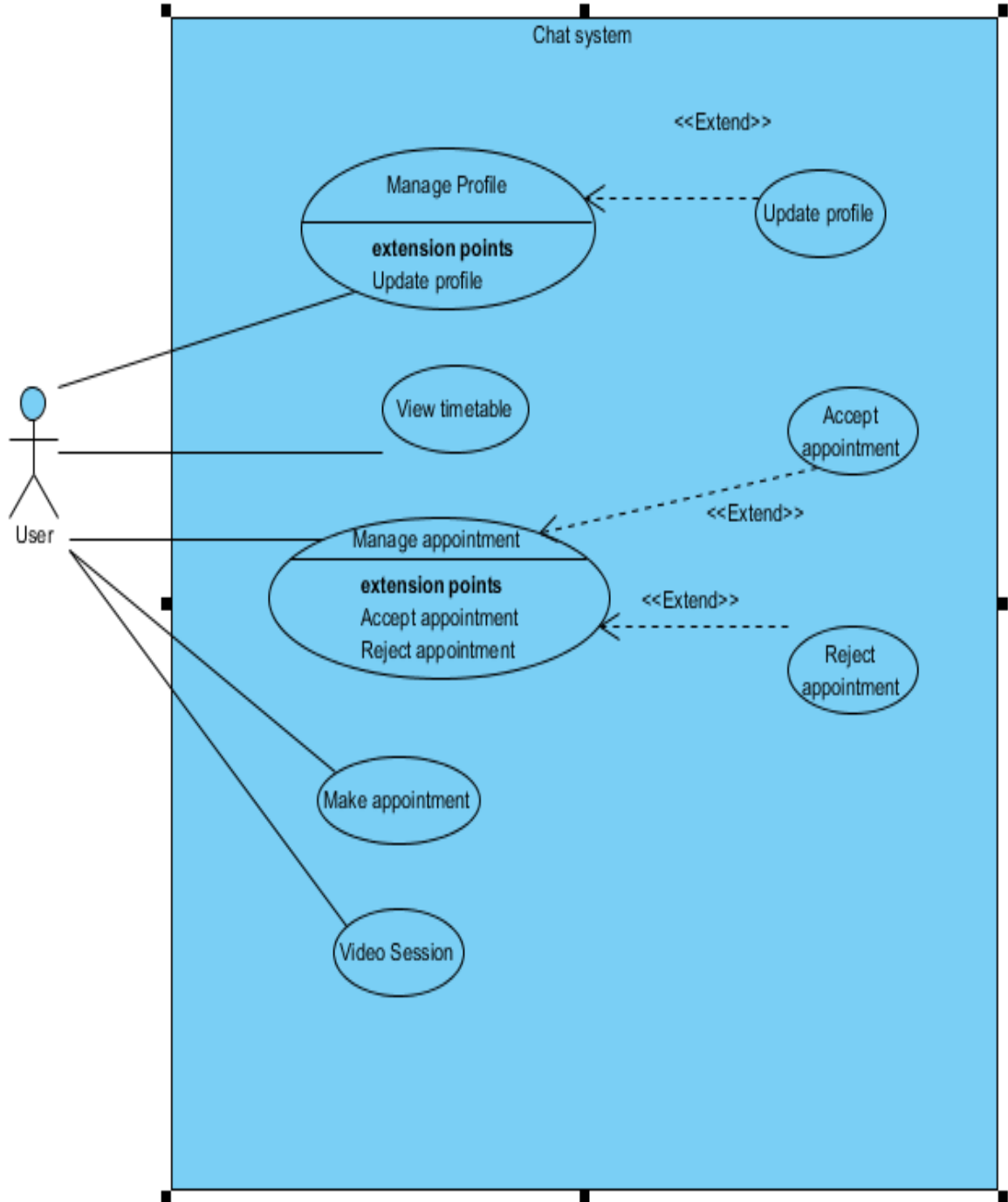


Figure 3-1-F1: Use Case

### 3.2 Use Case Description

<b>Use case ID</b>	UC001	<b>Version</b>	1.0
<b>Feature</b>	Manage user profile		
<b>Purpose</b>	To enable user to manage the user profile		
<b>Actor</b>	User		
<b>Trigger</b>	User press on the account information tab in the menu or user just login into the system		
<b>Precondition</b>	The user must login into the system		
<b>Scenario Name</b>	<b>Step</b>	<b>Action</b>	
<b>Main Flow</b>	1	The user click on the account information tab in the menu.	
	2	The system will show the user about the user information.	
	3	User click on the change password button to change password.	
	4	User can change any field except the username which is not editable.	
	5	User click on update button to update the user account.	
	6	System verifies the all the input of the user information.	
	7	System will update the user account.	
	8	System will display the message to inform user that the user account has been successfully updated.	
<b>Alternate Flow- User wish to change a new password</b>	3.1	System will open a change password popup window.	
	3.2	User enter the old password.	
	3.3	User enter the new password	
	3.4	User re-enter the new password for confirmation.	
	3.5	User press submit to submit to update the password.	
	3.5	System will validate the old password.	
	3.6	System will validate the new password and confirmation password.	
3.7	System update the new password display change password successfully.		

	3.8	System will close the popup and return to account information page.
<b>Alternate Flow- Invalid old password</b>	3.5.1	User enter the password not same with the old password.
	3.5.2	System will display the entered password not same with the old password.
<b>Alternative Flow- New password not same with the confirmation password</b>	3.6.1	User enter the confirmation password that not similar to the new password.
	3.6.2	System will display the new password and confirmation password not same.
<b>Rules</b>	User must login into the system	
<b>Author(s)</b>	Lee Joon Hoe	

<b>Use case ID</b>	UC002	<b>Version</b>	1.0
<b>Feature</b>	User can view the timetable		
<b>Purpose</b>	To enable user to view the timetable for the consultation session.		
<b>Actor</b>	User		
<b>Trigger</b>	User click on the timetable tab in the menu.		
<b>Precondition</b>	User must login into the system.		
<b>Scenario Name</b>	<b>Step</b>	<b>Action</b>	
<b>Main Flow</b>	1	User click on the timetable tab in the menu.	
	2	The system will show the user about the detail of the upcoming appointment.	
<b>Rules</b>	User must login into the system.		
<b>Author(s)</b>	Lee Joon Hoe		

<b>Use case ID</b>	UC003	<b>Version</b>	1.0
<b>Feature</b>	Manage appointment		

<b>Purpose</b>	To enable user to manage the appointments	
<b>Actor</b>	User	
<b>Trigger</b>	User click on the appointmentlist tab in the menu.	
<b>Precondition</b>	The user must login into the system.	
<b>Scenario Name</b>	<b>Step</b>	<b>Action</b>
<b>Main Flow</b>	1	The user click on the appointmentlist tab in the menu.
	2	System will show 3 panels to the user which is pending request, pending reply and rejected appointment.
	3	User click on the pending request panel.
	4	User click on the pending reply panel.
	5	User click on the Rejected panel.
<b>Alternate Flow- User click on the pending request panel</b>	3.1	System will display the pending requested
	3.2	User delete the appointment
	3.3	The system delete the appointment requested.
	3.4	System will display successfully deleted message.
<b>Alternate Flow- User click on the pending reply panel</b>	4.1	System will display the appointment that has been requested.
	4.2	System will prompt user either accept or reject the appointment requested.
<b>Alternate Flow- User accepted the appointment</b>	4.2.1.1	User press accept button to accept the appointment.
	4.2.1.2	System will update the appointment status as accepted.
	4.2.1.3	System will schedule a SMS reminder to be sent later.
	4.2.1.4	System will generate a video session link to be used as consultation later.
<b>Alternate Flow- User rejected the appointment</b>	4.2.2.1	User press reject button to reject the appointment.
	4.2.2.2	System will update the appointment status as rejected
<b>Alternate Flow- User click on the rejected panel</b>	5.1	System will display the rejected appointment to the user.
	5.2	User can delete the appointment that have been rejected.

<b>Rules</b>	The user must login into the system.
<b>Author(s)</b>	Lee Joon Hoe

<b>Use case ID</b>	UC004	<b>Version</b>	1.0
<b>Feature</b>	Make appointment		
<b>Purpose</b>	To enable user to make appointment		
<b>Actor</b>	User		
<b>Trigger</b>	User click on the make appointment tab in the menu.		
<b>Precondition</b>	The user must login into the system.		
<b>Scenario Name</b>	<b>Step</b>	<b>Action</b>	
<b>Main Flow</b>	1	The user click on the makeappointment tab in the menu.	
	2	System will display an appointment template for the user to fill in.	
	3	System will display a list of user.	
	4	User need to fill in all the appointment detail in the template.	
	5	User submit the appointment template	
	6	System will verify the input of the user.	
	7	System will display appointment successfully made.	
<b>Alternate Flow- User is a student</b>	3.1	System will only display the lecturer option.	
<b>Alternate Flow- User is a student</b>	3.2	System will only display the Student option.	
<b>Alternate Flow- The time table clashing</b>	7.1	The input and the timetable of either the user or the appointment user have been clash.	
	7.2	System display the appointment made not successfully.	
<b>Rules</b>	The user must login into the system.		
<b>Author(s)</b>	Lee Joon Hoe		

<b>Use case ID</b>	UC005	<b>Version</b>	1.0
--------------------	-------	----------------	-----

<b>Feature</b>	Video session	
<b>Purpose</b>	To enable user to have online conferencing.	
<b>Actor</b>	User	
<b>Trigger</b>	User click on the video session tab in the menu.	
<b>Precondition</b>	The user must login into the system.	
<b>Scenario Name</b>	<b>Step</b>	<b>Action</b>
<b>Main Flow</b>	1	The user click on the video session tab in the menu.
	2	The system will show the video session that are available.
	3	The user must click on the ready button to inform another user that he or she is available.
	4	System enabled the start button to start the video session.
	5	User click on the start button.
	6	System will open a new popup for the video session.
	7	User can start the video session but click the start button.
	8	System prompts for the user video and audio input.
	9	User enabled the audio and video input.
	10	User click on the screen sharing button.
	11	User click on the share file button.
	12	User can click on the text chat to sent message.
<b>Alternate Flow- No audio or video input</b>	1.1	User does not have an audio or video input device.
	1.2	System will unable to start the session.
<b>Alternate Flow- No Screen Sharing plug in</b>	5.1	User does not have the screen sharing plug in to share the screen.
	5.2	System will unable to start the screen sharing.
<b>Rules</b>	The user must login into the system.	
<b>Author(s)</b>	Lee Joon Hoe	

### 3.3 System Flow

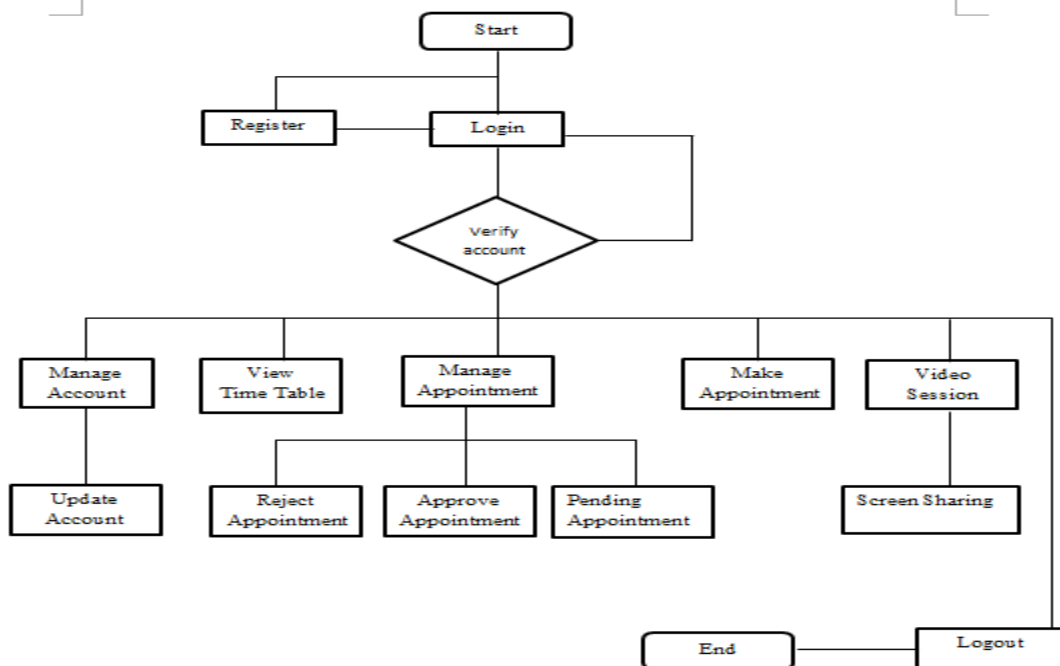


Figure 3-3-F1: System Flow

### 3.4 Flow description

#### 1. Register

A new user has to register an account in order to use the chat system. Some particular that user have to fill in such as the Username, password, first name, last name, select the gender and role, email. The Username register for the system must be unique. After register it will direct to the login page.



## **2. Login**

The user has to fill in the username and the password in order to login to the system. If the username or password is incorrect, the system will inform the user the incorrect input and request user to type in again.

## **3. Manage Account**

After successfully login into the system, the first page occur is the profile page of the user. The user can chose to edit their account information such as changing the password or update others detail into the system.

## **4. View TimeTable**

User can view all the upcoming consultation session that have been approved.

## **5. Manage Appointment**

The user can check their appointment here. There are table that show the sent appointment, the pending appointment, accepted appoint and rejected appointment. User can chose to delete the appointment that have been made and haven't been approved. User can accept appointment or reject appointment that been requested. User can delete the appointment that have been rejected.

## **6. Make appointment**

User can chose to make an appointment by filling the template provided. Some criteria have to full in are the data, time, appointment receiver, and the remarks is optional. After full in the template, the appointment can be sent.

## **7. Video session**

User can have a video session. Some additional features such as screen sharing feature, text chat and file sharing to benefit the student have understanding toward the problem.

## **8. Logout**

User can chose to logout from the system.

### 3.5 Entity Relationship Diagram

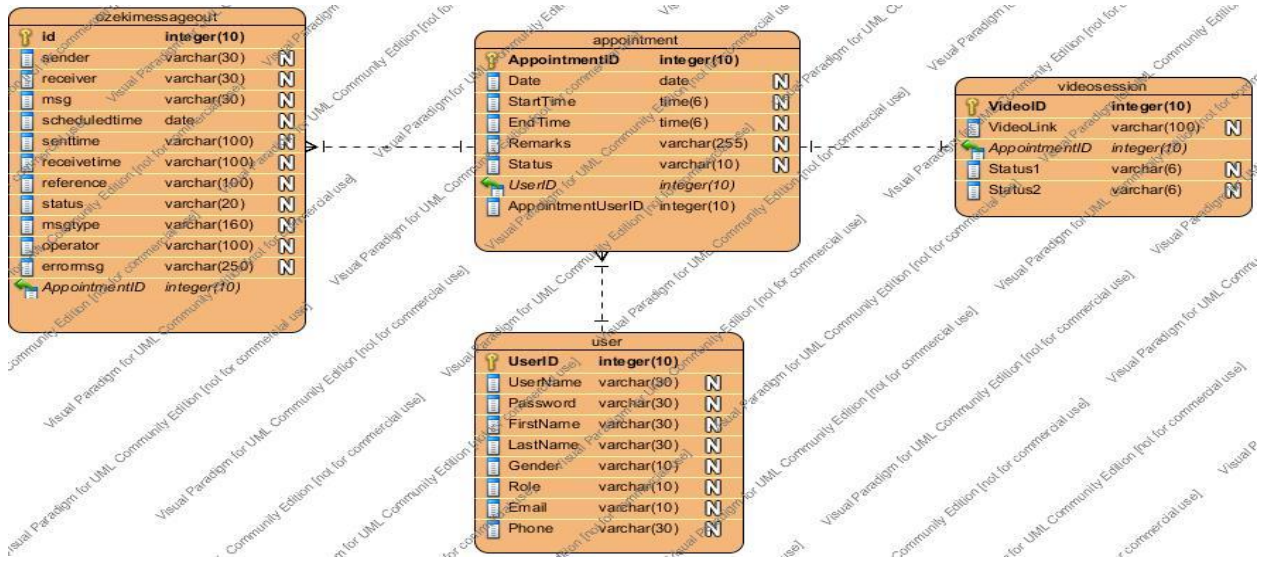


Figure 3-5-F1: Entity Relationship Diagram

### 3.6 SQL to create table

Table	SQL
appointment	<pre> CREATE TABLE `appointment` (   `AppointmentID` int(10) NOT NULL   AUTO_INCREMENT,   `Date` date NOT NULL,   `StartTime` time(6) NOT NULL,   `EndTime` time(6) NOT NULL,   `Remarks` varchar(255) DEFAULT   NULL,   `Status` varchar(10) NOT NULL,   `UserID` int(10) NOT NULL,   `AppointmentUserID` int(10) NOT NULL,   PRIMARY KEY (`AppointmentID`) );                 </pre>
user	<pre> CREATE TABLE `user` (                 </pre>

	<pre> `UserID` int(10) NOT NULL AUTO_INCREMENT, `UserName` varchar(30) DEFAULT NULL, `Password` varchar(30) DEFAULT NULL, `FirstName` varchar(30) DEFAULT NULL, `LastName` varchar(30) DEFAULT NULL, `Gender` varchar(10) DEFAULT NULL, `Role` varchar(10) DEFAULT NULL, `Email` varchar(30) DEFAULT NULL, `Phone` varchar(13) NOT NULL, PRIMARY KEY (`UserID`), UNIQUE KEY `UserID` (`UserID`), UNIQUE KEY `UserName` (`UserName`) ); </pre>
<b>videosession</b>	<pre> CREATE TABLE `videosession` ( `VideoID` int(10) NOT NULL AUTO_INCREMENT, `VideoLink` varchar(100) NOT NULL, `AppointmentID` int(10) NOT NULL, `Status1` varchar(6) NOT NULL, `Status2` varchar(6) NOT NULL, PRIMARY KEY (`VideoID`) ) </pre>
<b>ozekimessageout</b>	<pre> CREATE TABLE `ozekimessageout` ( `id` int(11) NOT NULL AUTO_INCREMENT, `sender` varchar(30) DEFAULT NULL, `receiver` varchar(30) DEFAULT NULL, </pre>

	<pre> `msg` text, `scheduledtime` datetime NOT NULL, `senttime` varchar(100) DEFAULT NULL, `receivedtime` varchar(100) DEFAULT NULL, `reference` varchar(100) DEFAULT NULL, `status` varchar(20) DEFAULT NULL, `AppointmentID` int(10) NOT NULL, `msgtype` varchar(160) DEFAULT NULL, `operator` varchar(100) DEFAULT NULL, `errmsg` varchar(250) DEFAULT NULL, PRIMARY KEY (`id`), KEY `id` (`id`) ); </pre>
--	---

**Table 3-6-T1 : SQL to Create Table**

### 3.7 Data dictionary

Attribute	Data Type	Description	Key
AppointmentID	INTEGER	ID of the appointment	Primary Key
Date	DATE	Date of the appointment	
StartTime	TIME	Start time of the appointment	
EndTime	TIME	End time of the appointment	
Remarks	VARCHAR	Additional information of the regarding the appointment	
Status	VARCHAR	Status of the appointment	
UserID	INTEGER	ID of the User	Foreign Key

AppointmentUserID	INTEGER	ID of another appointment user	Foreign Key
-------------------	---------	--------------------------------	-------------

**Table 3-7-T1 : appointment Table**

This is an appointment table that store the information of an appointment such as AppointmentID, Date, StartTime, EndTime, Remarks, Status, UserID, AppointmentUserID. The status of the appointment table will only consist of 3 values which is accepted, pending and rejected. The AppointmentID is the person that the user that want to make appointment with.

Attribute	Data Type	Description	Key
UserID	INTEGER	Id of the user	Primary key
UserName	VARCHAR	Username of the user	
Password	VARCHAR	Password of the user	
FirstName	VARCHAR	First Name of the user	
LastName	VARCHAR	Last Name of the user	
Gender	VARCHAR	Gender of the user	
Role	VARCHAR	Role of the user	
Email	VARCHAR	Email of the user	
Phone	VARCHAR	Phone number of the user	

**Table 3-7-T2 : user Table**

This is the user table that store the information of a register user such as a unique UserID, unique UserName, Password, FirstName, LastName, Gender, Role, Email, Phone number to identify the unique user.

Attribute	Data Type	Description	Key
VideoID	INTEGER	ID of the video session	Primary Key
VideoLink	VARCHAR	Link of the video session	
Status1	VARCHAR	Status of the user	
Status2	VARCHAR	Status of the another user	
AppointmentID	INTEGER	ID of the appointment	Foreign Key

**Table 3-7-T3: videosession Table**

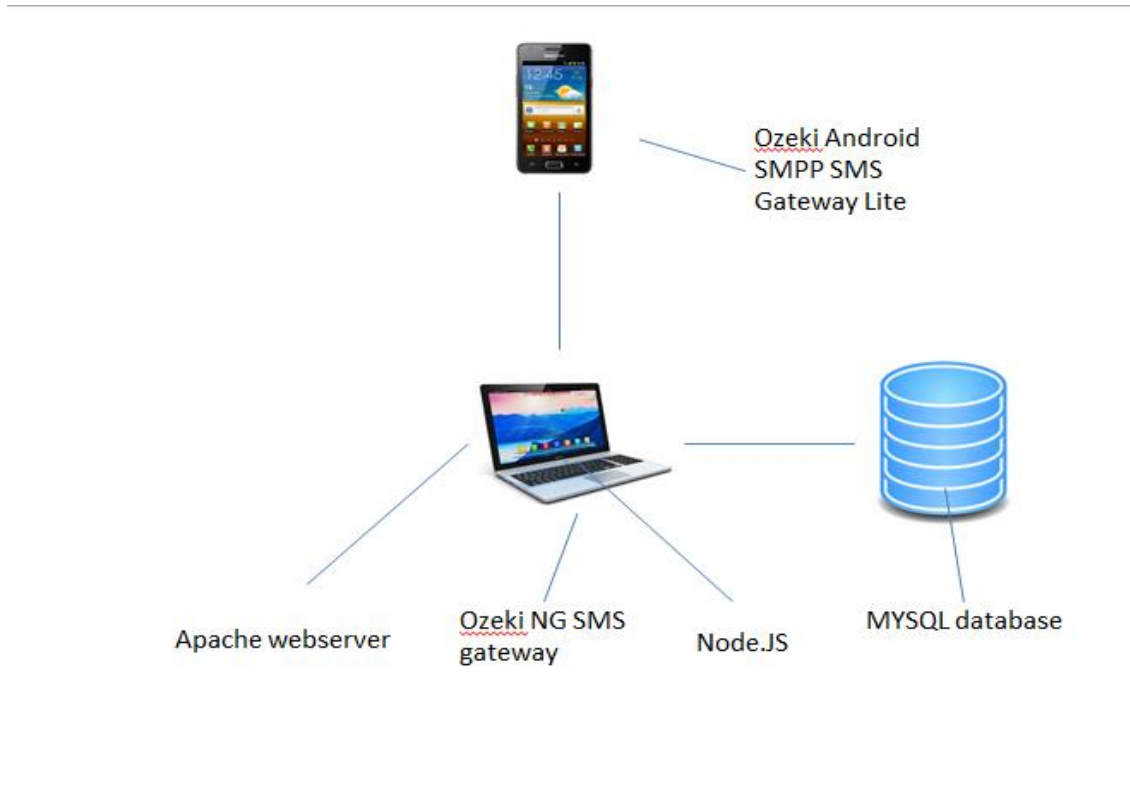
This is a videosession table that store the information for the video session such as the videoID, videolink, and the appointmentID. The Status 1 and Status 2 is used to check whether they user are available and ready for the video session. The video session can only be start when both user are ready. The appointmentID field is to trace the video session is link to which appointment that have been made.

Attribute	Data Type	Description	Key
id	INTEGER	Id of the message	Primary Key
sender	VARCHAR	The sender phone number	
receiver	VARCHAR	The receiver phone number	
msg	TEXT	The message type	
scheduledtime	DATETIME	The scheduled time for SMS	
senttime	VARCHAR	The SMS sent time	
receivetime	VARCHAR	The SMS received time	
reference	VARCHAR	Reference of the SMS	
status	VARCHAR	The status of the SMS	
msgtype	VARCHAR	The type of message	
operator	VARCHAR	Operator of the SMS	
errormsg	VARCHAR	The error message for the SMS process	
AppointmentID	INTEGER	ID of the appointment	Foreign Key

**Table 3-7-T4: ozekimessageout Table**

This is a table used for the OZEKI Ng SMS gateway to send the scheduled SMS from the database. The table store attribute such as id, sender, receiver, msg, scheduledtime, senttime, receivetime, reference, status, msgtype, operator, and errormsg.

### 3.8 Architecture Diagram



**Figure 3-6-F1: Architecture diagram**

## Chapter 4: Methodology

### 4.1 Methodology of system development life cycle

The system development life cycle has been implemented to deliver this project. The system development life cycle consist of 4 phase which is planning, analysis, design and implementation. The model that has been implemented for the development of this project is the prototype model. The prototype modeling is an incomplete system to use to show the user before the final product system has been deployed. There are some advantages using prototype such as early to detect if there are bugs or errors occurrence, lack of functionality can be detected earlier, help to explain the concept to the user easy.

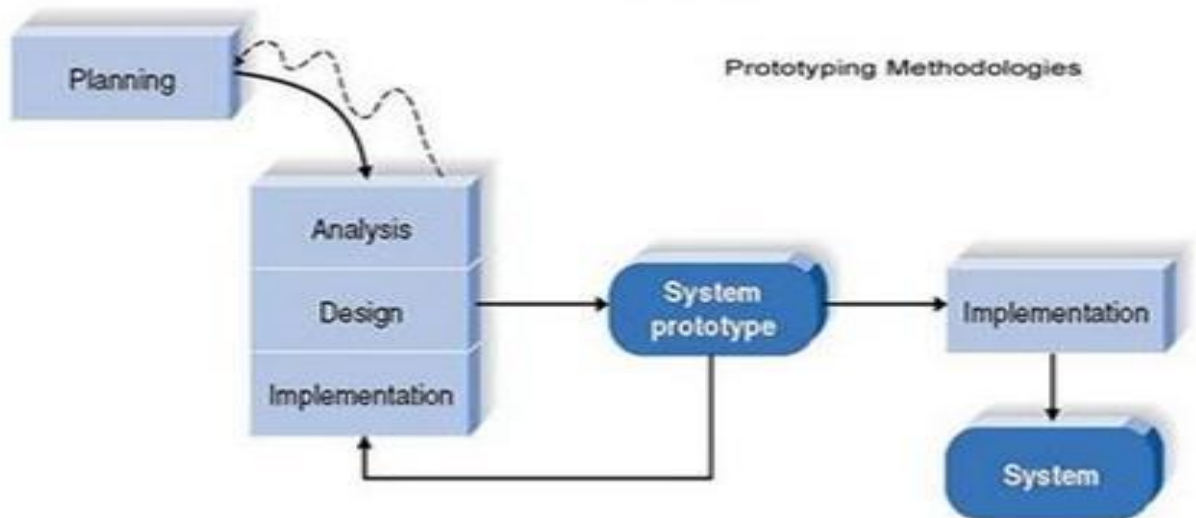


Figure 4-1-F1: Prototype model

Phase	Description
Planning	In this planning phase, the problem statement is defined. After the problem statement is defined, the solution to solve the problem is determined. The problem statement of this project are both student and teacher have to meet at specific location if they want to have consultation



	<p>and teacher and student have to check stack of email and to arrange the queue for the student consultation. The solution to solve the problem is developed an online chat system to provide an online solution and to give a systematic appointment system for making appointment.</p>
<b>Analysis</b>	<p>In this analysis phase, the requirement and functional of this project is defined. One of the requirements for the online chat system is have a video session to simulate face to face conversation. After analysis requirement, some features have to be added in order to fulfill the requirement. In order to fulfill the requirement, webRTC is introduced for to perform the video session along with some additional feature such as the screen sharing and the file sharing.</p>
<b>Design</b>	<p>In the design phase, the system will be consult based on the requirement and the functional from the analysis phase. Some of the diagrams are deliver to illustrate the system such as the flow chart to show the system flow, the entity relationship diagram to show the database design, the architectural diagram to show the architectural design and the use-case diagram show what the user can do in the system.</p>
<b>Implementation</b>	<p>In this implementation phase, the system</p>

	<p>will be coding and develop following the design phase. This project will implement the webRTC on the node server. The web portal will be implemented on apache web server. Some information data will be store in the database for reference.</p>
--	--

**Table 4-4-T2: The SDLC methodology**

### **System prototype**

Before the final system is delivered, the prototype will be developed. Prototype is use to check the system is fulfill the requirement of the project. Prototype also uses to test the system behavior. If the system give the output is not the expected result, the system contains bugs and errors. In this project the system prototype will be develop for testing whether the video session is working according to the plan.

## **4.2 Tools**

### **Apache Web server**

Apache is one of the web servers to run for this project. Apache has built in to support for a wide range programming language including PHP. Furthermore, Apache is free and very portable because it can run on many operating systems. The web portal will be run on this apache web server.

### **MySQL**

MySQL will be used to develop this project. MYSQL is an open source relational database management system that uses to manipulate the database. The purpose of use MySQL is because this project required storing some information such as the user information, appointment form MySQL also can use along with PHP. MySQL also support the framework for develop this project which is the Net Bean IDE.

## **Node.js**

Node.js is implemented to develop this project. Chrome JavaScript runtime used to build the platform for the Node.js purposely for creating a quick and scalable application. The event driven with non-blocking I/O model is introduced by the Node.js purposely for significant, effective and enhanced the real-time application data intensive that will executed in different distributed devices. This project will be using node as another server. Node server will be building the some feature for this project such as the webRTC, screen sharing, and file sharing feature.

## **Hypertext Markup Language (HTML)**

HTML will be the markup language to create this project website. HTML is use to organize and display the content of the webpage.

## **Cascading Style Sheet (CSS)**

CSS is the style sheet language that will be used to along with HTML in this project to provide better view. CSS can provide some feature such as adding animation, styling the font, providing effects, adding background images and so on.

## **JavaScript**

JavaScript is the client side language that will be used to develop this project. Some processes do not need to go through the server and can be handling by JavaScript such as simple validation or button event by writing function. JavaScript will be used for the node server as well in develop this project.

## **Asynchronous JavaScript and XML(Ajax)**

Ajax is chose to implement the system because it has a better interactivity. Ajax allows easier and quicker interaction between user and website as page are not reloaded for content to be displayed. Ajax application can be built allow easier navigator to user in comparison to using the traditional back and forward button on the browser. Ajax is compact because several multi-purpose application and feature can be settled using a single webpage without the need of clutter with several web pages.

## **Hypertext Preprocessor (PHP)**

PHP will be selected as server side language for developed this project. The reason PHP is selected because PHP can work together with MySQL and can run on apache webserver. PHP is use to process the request from the client side reply the result to the client side.

## **OZEKI NG SMS Gateway**

OZEKI Ng SMS gateway is used as a SMS gateway because it free for trial. OZEKI also help to automatically sent the schedule SMS from the database. Furthermore Ozeki Ng SMS gateway also can connect with MYSQL database with the help of the ODBC drivers.

## **RtcMulticonnection WebRTC library**

RtcMulticonnection API is used because it is open source. It offered API that easy to learn and used. Furthermore, it also able to support and build the screen sharing.

## **4.3 User requirement**

**-Google Chrome Browser**

**-Bandwidth 50kb/s**

**-Screen sharing plug -in**

## Chapter 5: System Implementation

### 5.1 Setup and configuration

#### 5.1.1 XAMPP Control Panel

In order to host the website, apache server and MySQL database is need. XAMPP control panel included the apache server and MySQL and can executed using this control panel.

First of all download and install the XAMPP Control Panel from

<https://www.apachefriends.org/download.html> .

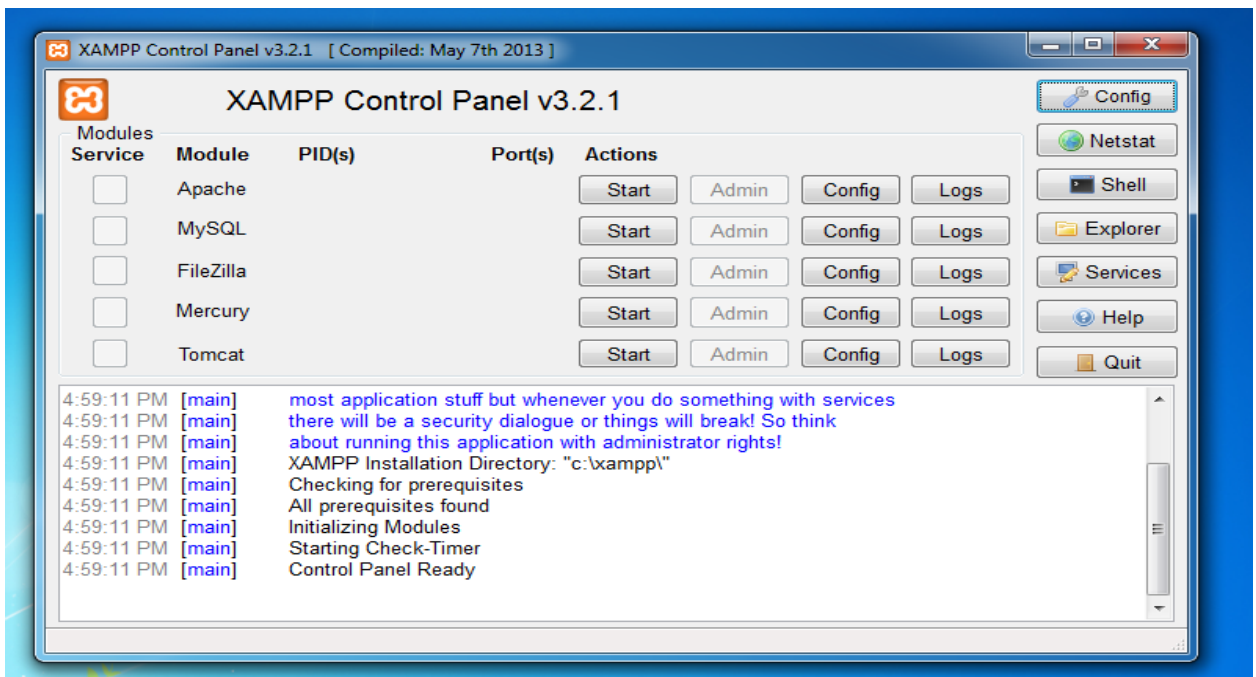
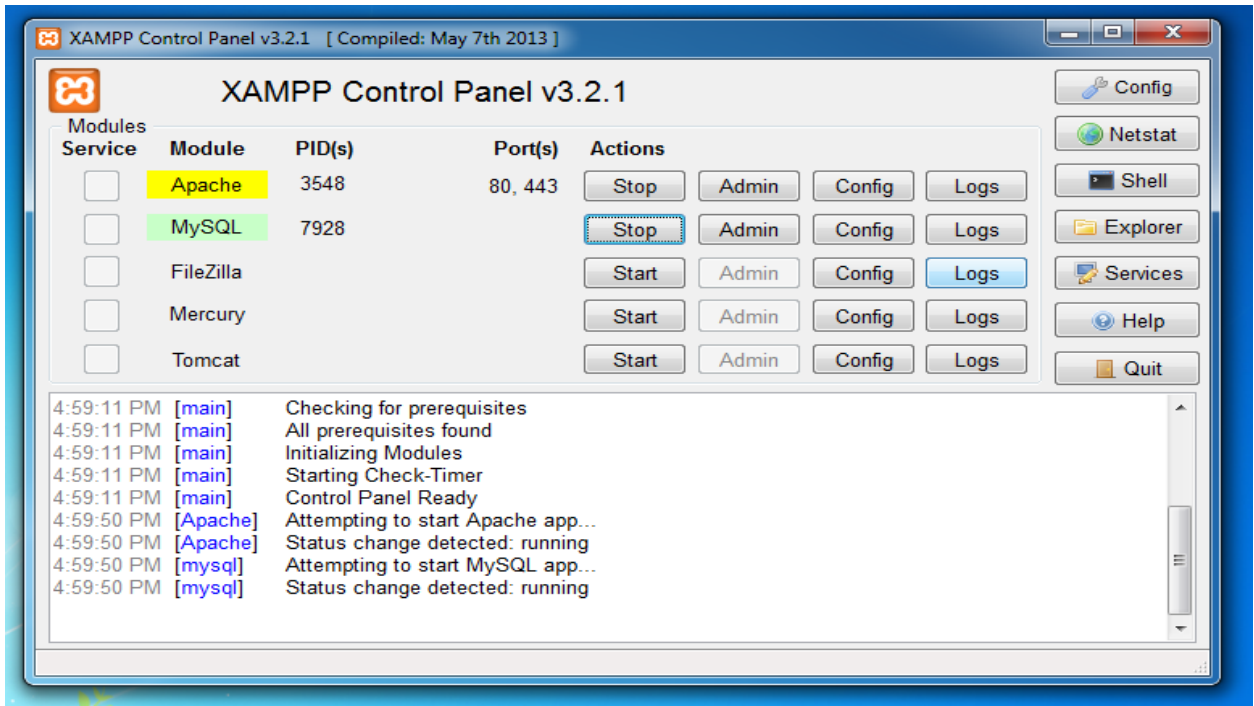


Figure 5-1-1-F1: XAMPP Control Panel

After the installation done, the control panel should be display as shown in the figure above



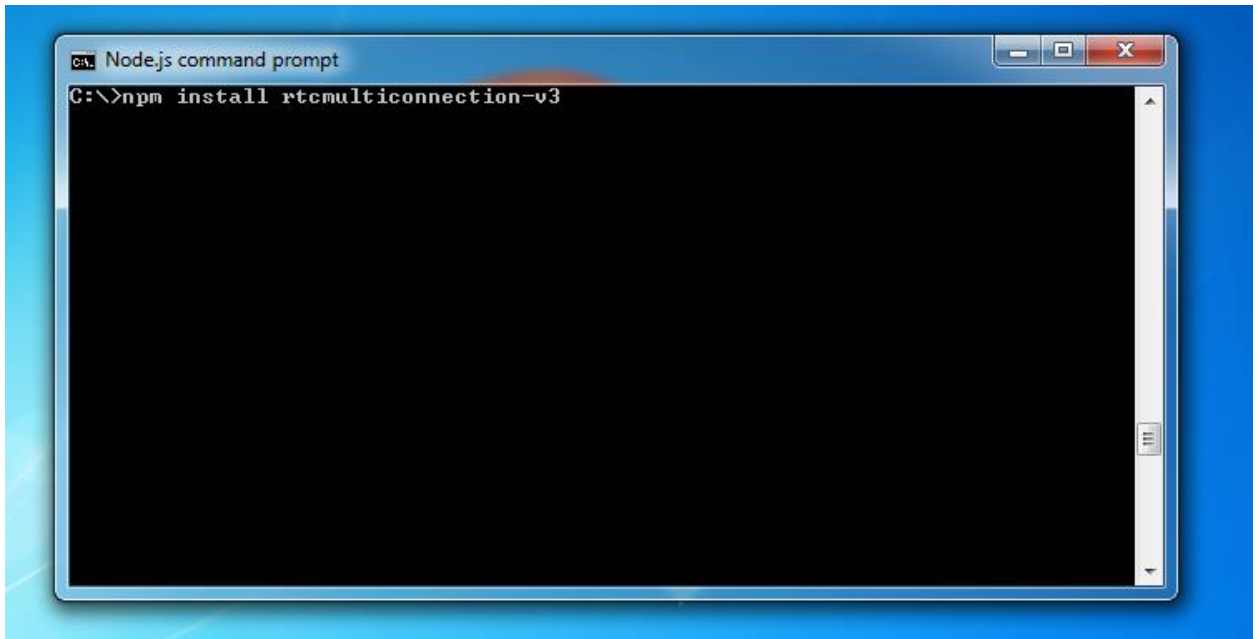
**Figure 5-1-1-F2: XAMPP Control Panel start service**

Click on the start on the Apache and MySQL to start run the Apache server and MySQL database.

### 5.1.2 node.js

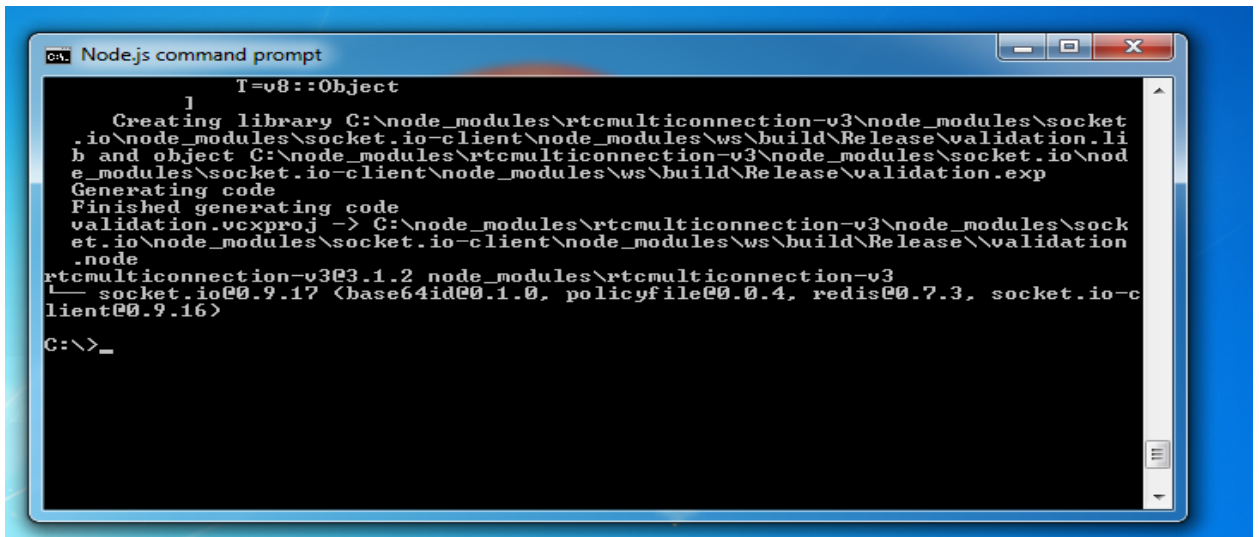
In order to do the video session, nodejs is needed. Download and install the nodejs from <https://nodejs.org/download/>

### 5.1.3 RTCMulticonnection webrtc javascript library



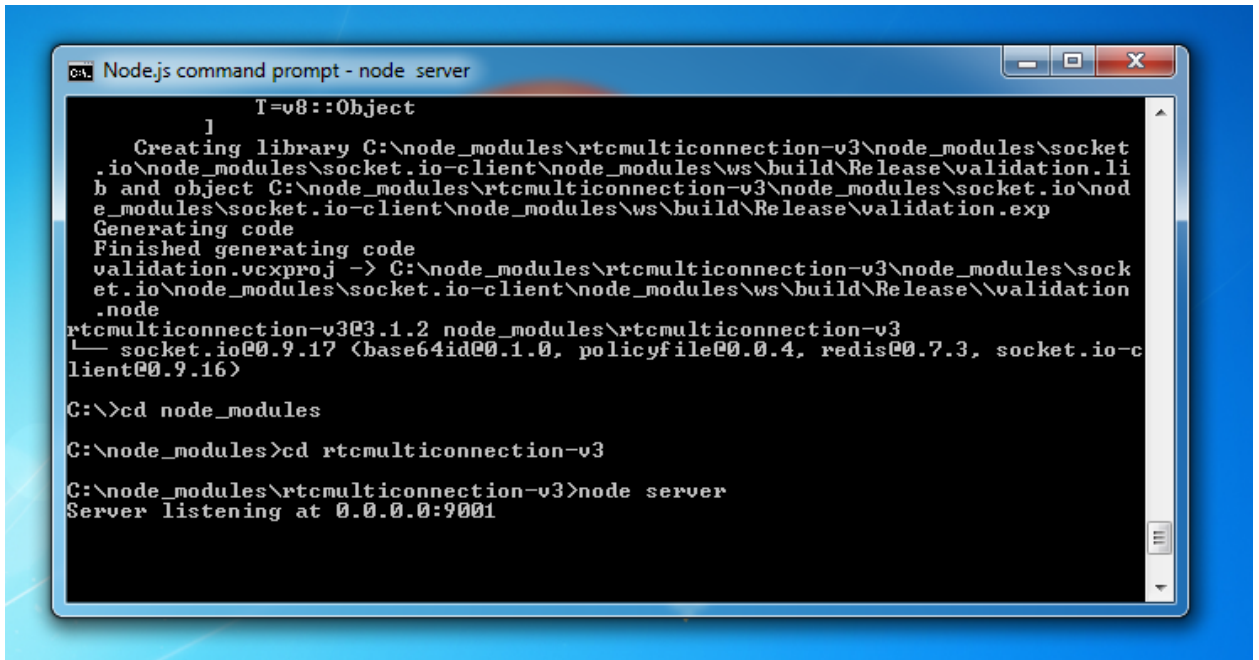
**Figure 5-1-3-F1: Node js command prompt**

RTCMulticonnection API is needed to build the video session. Open the node.js command prompt. Open the directory that want to library to be installed using command `cd "directory"`.



**Figure 5-1-3-F2: Rtcmulticonnection library installed**

The library successfully installed.



```
Node.js command prompt - node server
T=v8::Object
]
Creating library C:\node_modules\rtdmulticonnection-v3\node_modules\socket
.io\node_modules\socket.io-client\node_modules\ws\build\Release\validation.lib
and object C:\node_modules\rtdmulticonnection-v3\node_modules\socket.io\nod
e_modules\socket.io-client\node_modules\ws\build\Release\validation.exp
Generating code
Finished generating code
validation.vcxproj -> C:\node_modules\rtdmulticonnection-v3\node_modules\sock
et.io\node_modules\socket.io-client\node_modules\ws\build\Release\validation
.node
rtdmulticonnection-v3@3.1.2 node_modules\rtdmulticonnection-v3
├─ socket.io@0.9.17 (base64id@0.1.0, policyfile@0.0.4, redis@0.7.3, socket.io-cl
ient@0.9.16)
C:\>cd node_modules
C:\node_modules>cd rtdmulticonnection-v3
C:\node_modules\rtdmulticonnection-v3>node server
Server listening at 0.0.0.0:9001
```

**Figure 5-1-3-F3: Rtdmulticonnection Start Service**

In order to run the node server the script is needed. The server script is inside the library that installed previously. Go to the directory that the library installed. Execute command “node server” to run the server javascript. The bottom message show that the node server is running and listening to port 9001.



### 5.1.4 NetBeans IDE

Download the NetBean IDE from <https://netbeans.org/downloads/>

**NetBeans IDE 8.0.2 Download** 8.0.1 | 8.0.2 | 8.1 Beta | Development | JDK9 Branch Build | Archive

Email address (optional):

Subscribe to newsletters:  Monthly  Weekly  
 NetBeans can contact me at this address

IDE Language:  Platform:

Note: Greyed out technologies are not supported for this platform.

**NetBeans IDE Download Bundles**

Supported technologies *	Java SE	Java EE	C/C++	HTML5 & PHP	All
NetBeans Platform SDK	•	•			•
Java SE	•	•			•
Java FX	•	•			•
Java EE		•			•
Java ME					•
HTML5		•		•	•
Java Card™ 3 Connected					•
C/C++			•		•
Groovy					•
PHP				•	•
Bundled servers					
GlassFish Server Open Source Edition 4.1		•			•
Apache Tomcat 8.0.15		•			•

Download buttons: Java SE (Free, 90 MB), Java EE (Free, 186 MB), C/C++ (Free, 63 MB), **HTML5 & PHP (Free, 63 MB)**, All (Free, 205 MB)

**Figure 5-1-4-F1: NetBeans IDE**

Choose the HTML5&PHP to download. After finish download the installer, execute the installer to install the IDE.

### 5.1.5 ODBC MySQL connector

Download the ODBC MySQL connector from

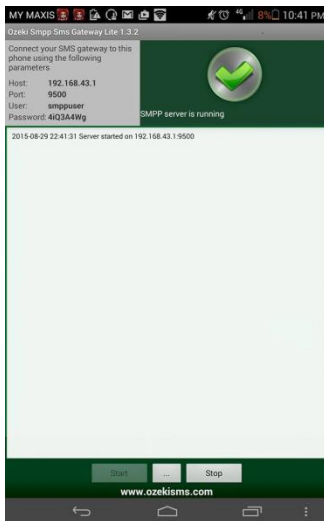
<https://dev.mysql.com/downloads/connector/odbc/5.1.html>. It is needed to connect the Ozeki NG SMS Gateway with the MySQL database.

### 5.1.6 OZEKI NG SMS Gateway Lite

OZEKI NG SMS Gateway Lite is a mobile apps that can get from Google play store and installed in the android phone. It is used to act as a GSM modem for send SMS.



**Figure 5-1-6-F1: OZEKI SMPP SMS gateway lite apps from Google play store.**



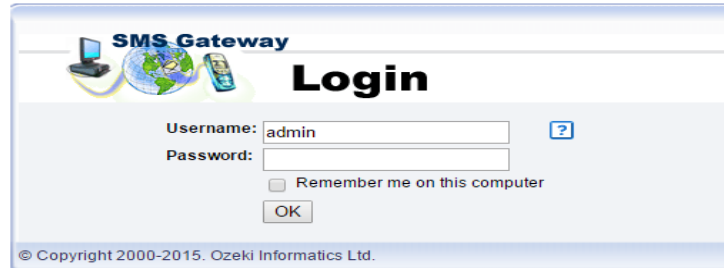
**Figure 5-1-6-F2: OZEKI SMPP SMS gateway lite apps.**

The top left corner information needed to use to configure at the SMS system later.

### 5.1.7 OZEKI NG SMS Gateway

Download and install the OZEKING SMS Gateway from

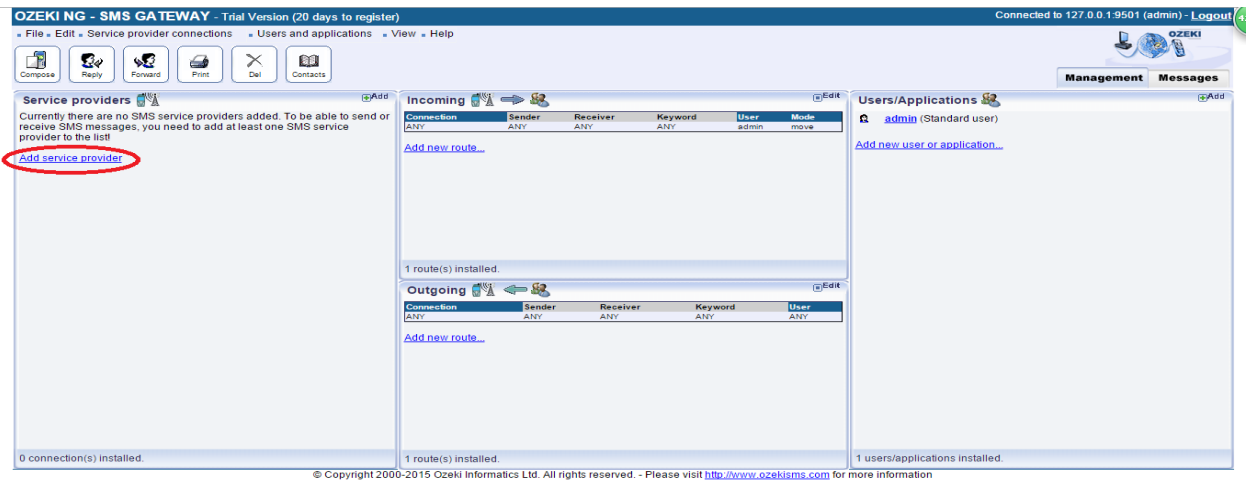
<http://www.ozekisms.com/index.php?owpn=112>.



Trial Version (20 days to register)  
We recommend you to use **Mozilla Firefox** or **Internet Explorer** to configure the software!

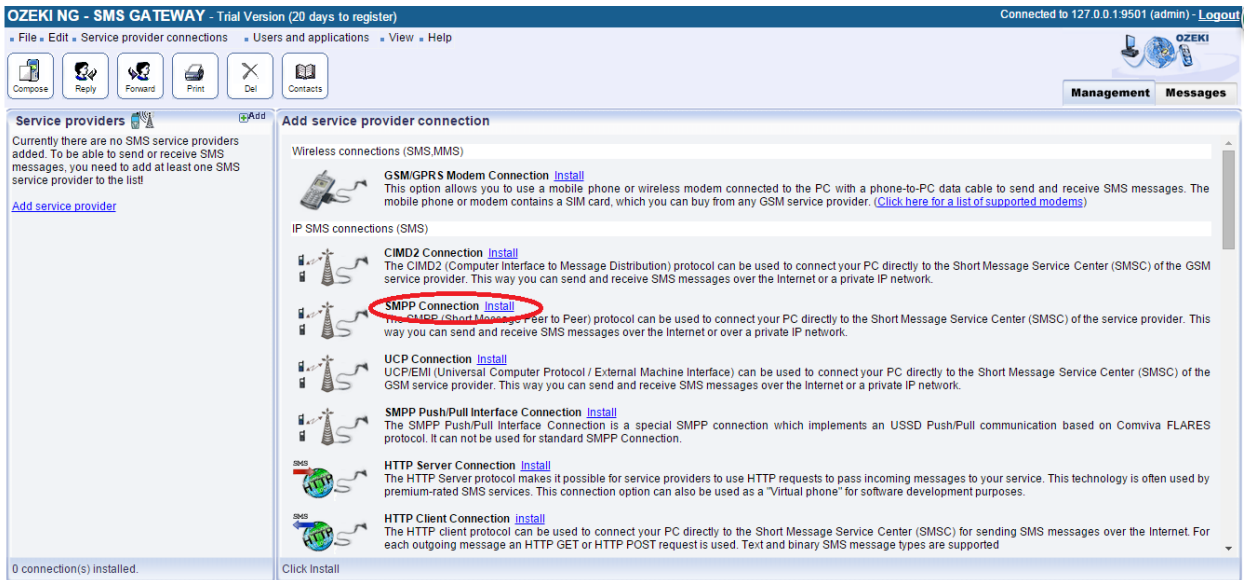
**Figure 5-1-7-F1: OZEKI SMS gateway page**

After installed open browser with [127.0.0.1:9501](http://127.0.0.1:9501). Figure 5.1.6.1 show the login page of the SMS gateway. Next, login into the system.



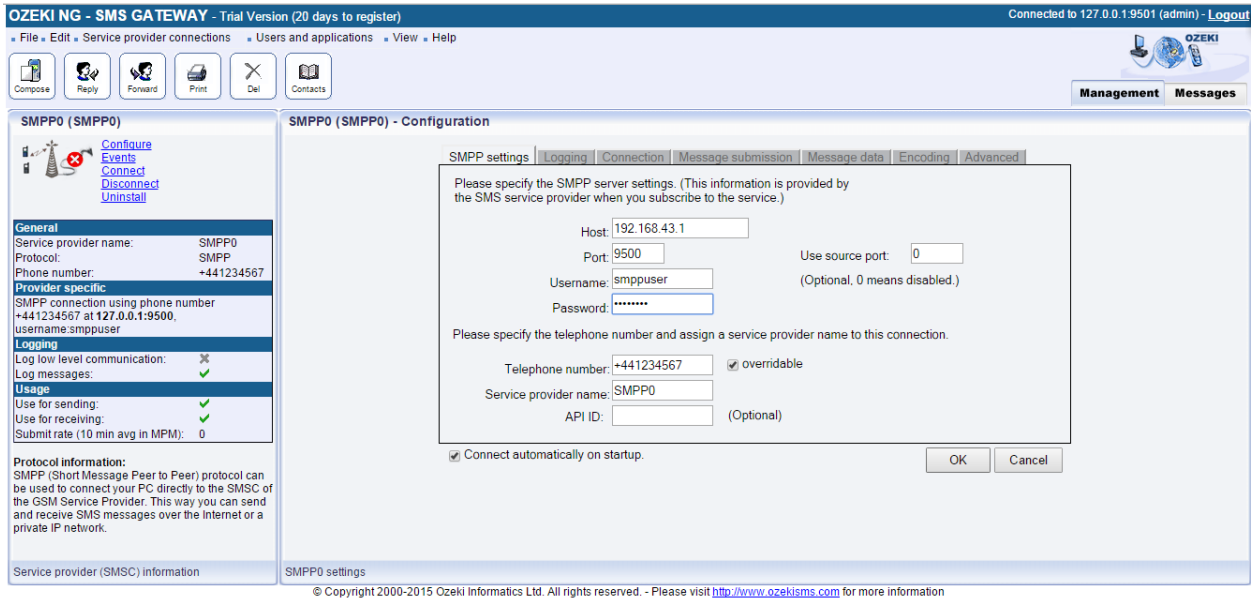
**Figure 5-1-7-F2: OZEKI SMS main page**

Click on the add service provider at the left hand side to configure the connection to the GSM device.



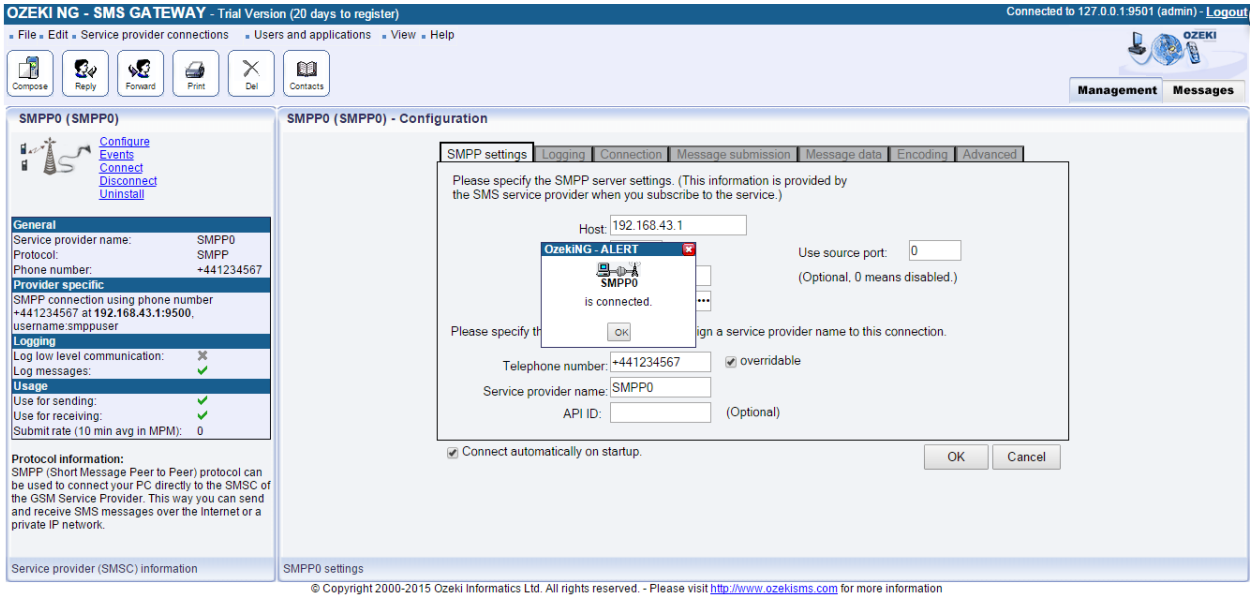
**Figure 5-1-7-F3: service provider connection page**

Select the SMPP connection and then click install.



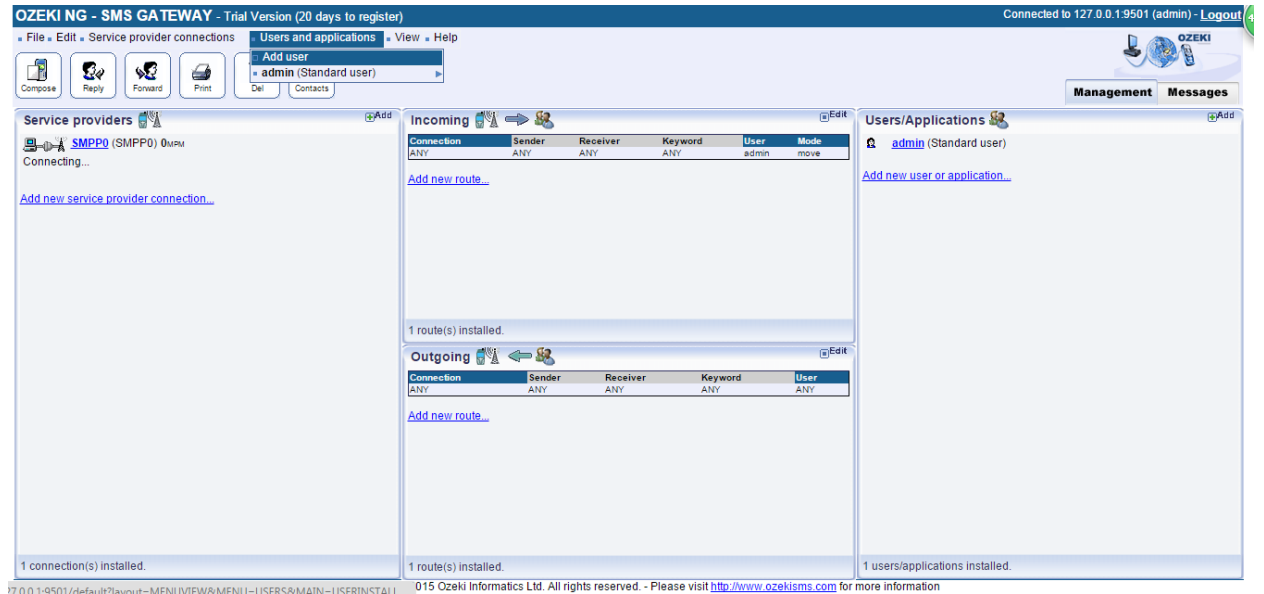
**Figure 5-1-7-F4: configuration page**

The information need to same with the OZEKI SMS gateway lite.



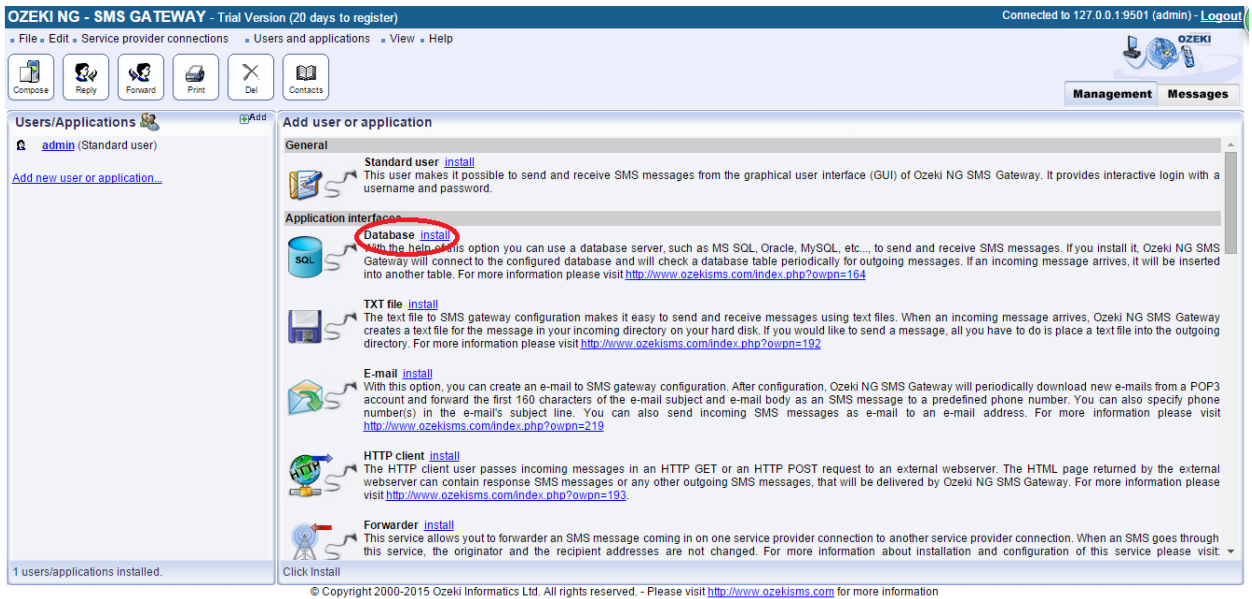
**Figure 5-1-7-F5: configuration page (successful)**

The configuration is successful.



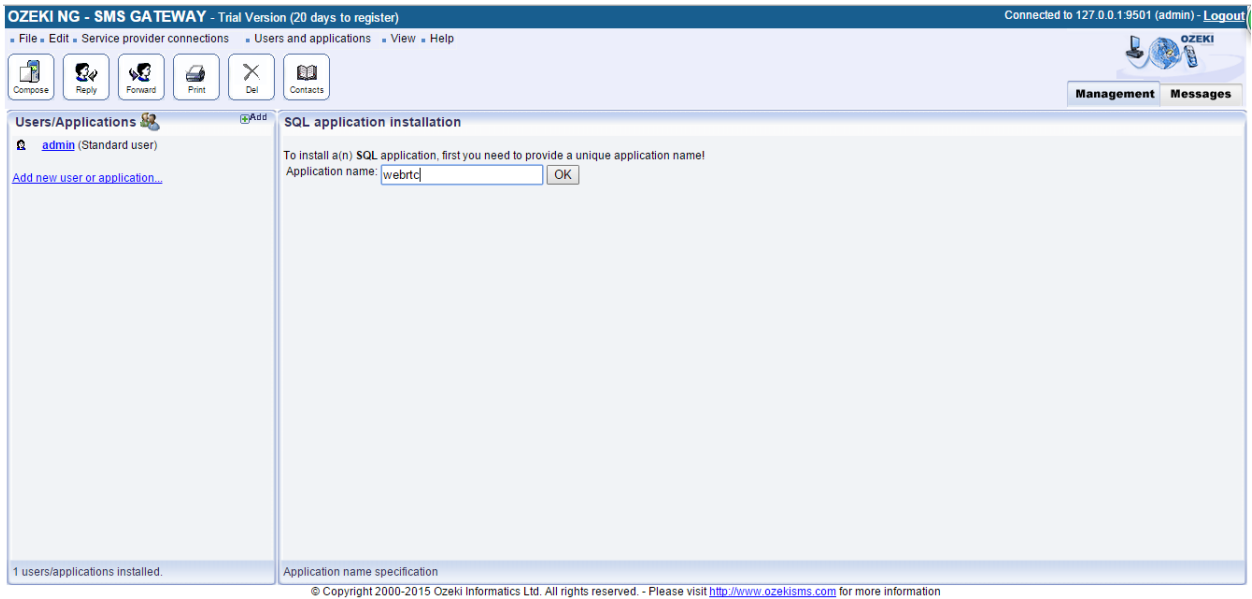
**Figure 5-1-7-F6: configure schedule SMS**

Click on the users and applications tab. Click Add user.



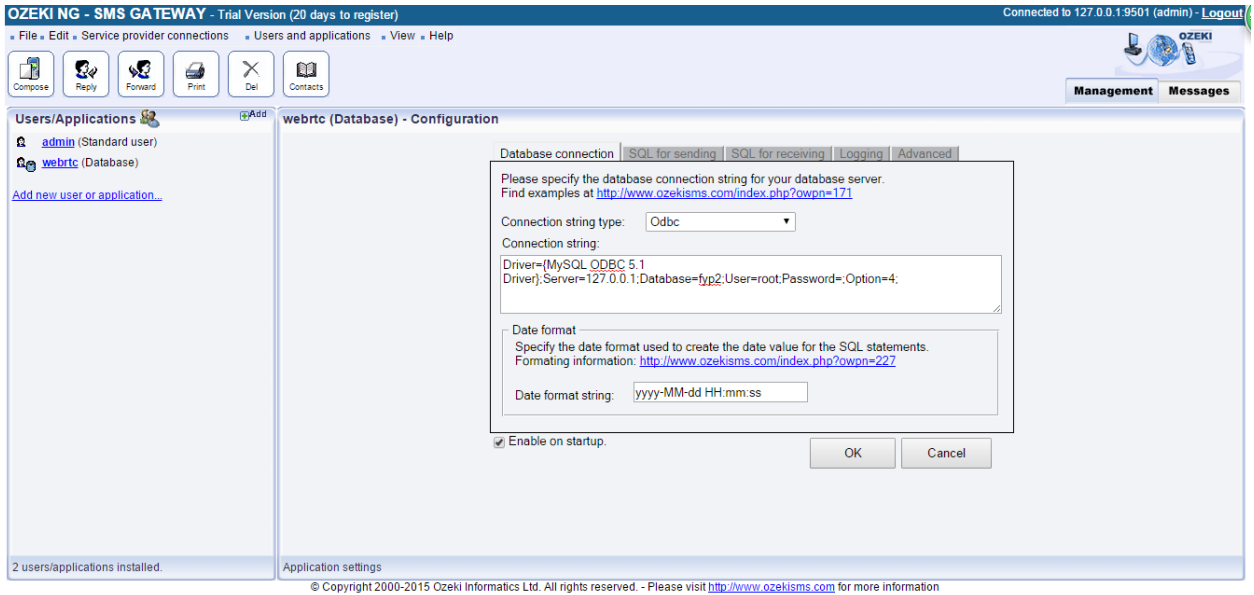
**Figure 5-1-7-F7: add user or application page.**

Select the database and click install.



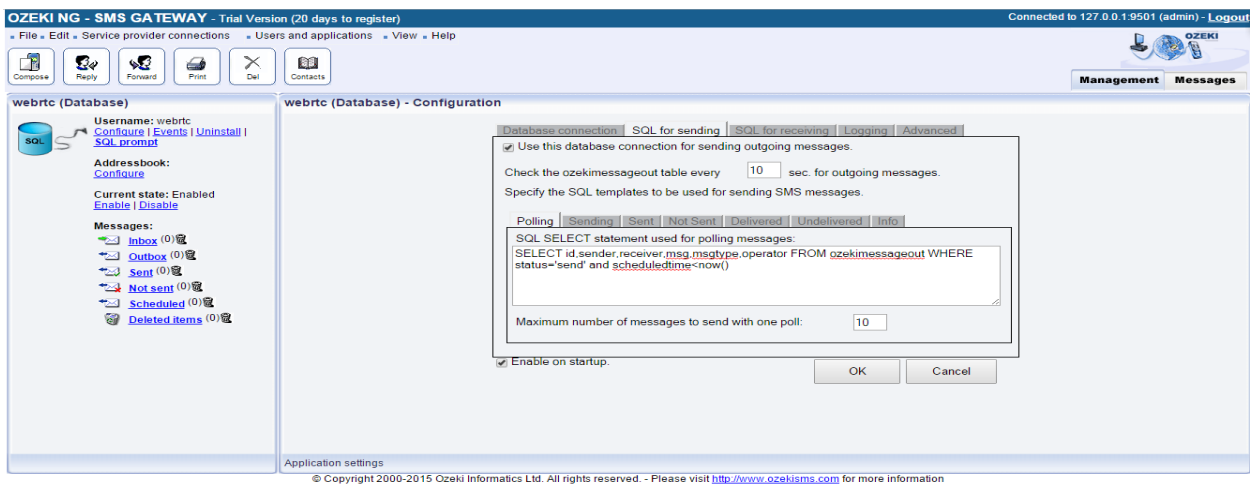
**Figure 5-1-7-8: SQL application installation page**

Type a name for the application.



**Figure 5-1-6-F9: Database configuration page**

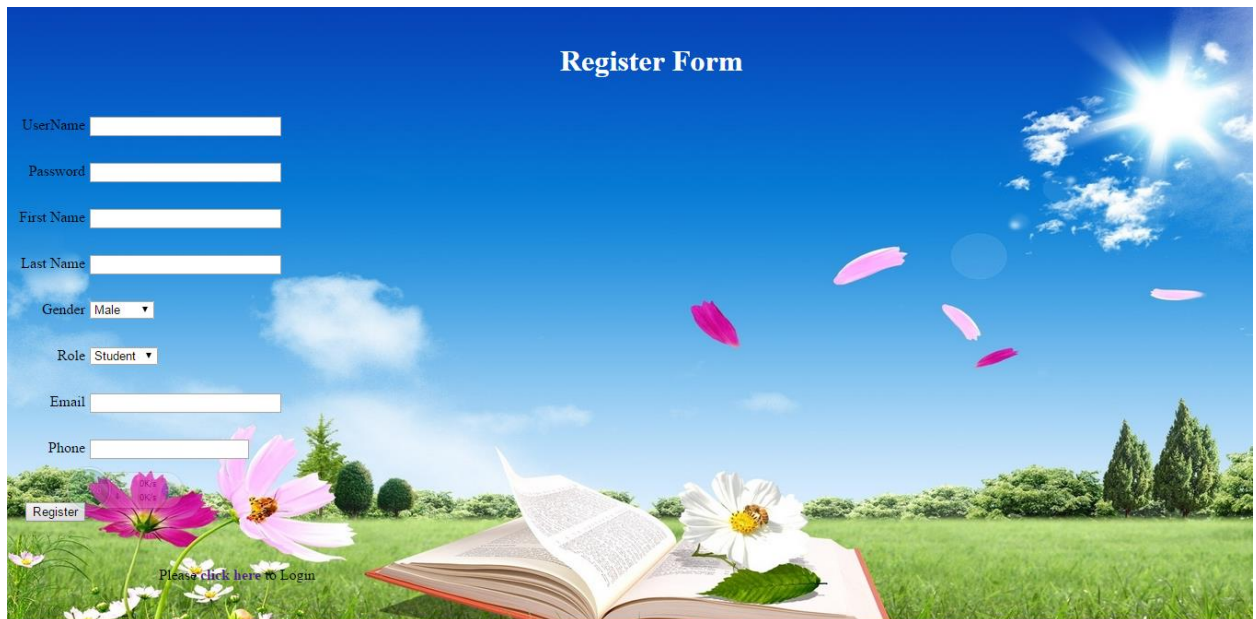
The connection string type chose the ODBC. The connection string type in the value shown in the Figure 5.1.6.9 to connect to the MySQL database.



**Figure 5-1-7-F10: Database configuration page**

Use a SQL query to select the scheduled SMS to be sent. The SQL query shown figure5.1.6.10 to select the schedule SMS from the ozekimeesageout table and sent. Press ok if configuration is complete.

## 5.2 Screenshot and Decription



The screenshot shows a web page titled "Register Form" set against a vibrant background of a blue sky with a bright sun, white clouds, and a green field with pink and white flowers. An open book lies on the grass in the foreground. The form fields are as follows:

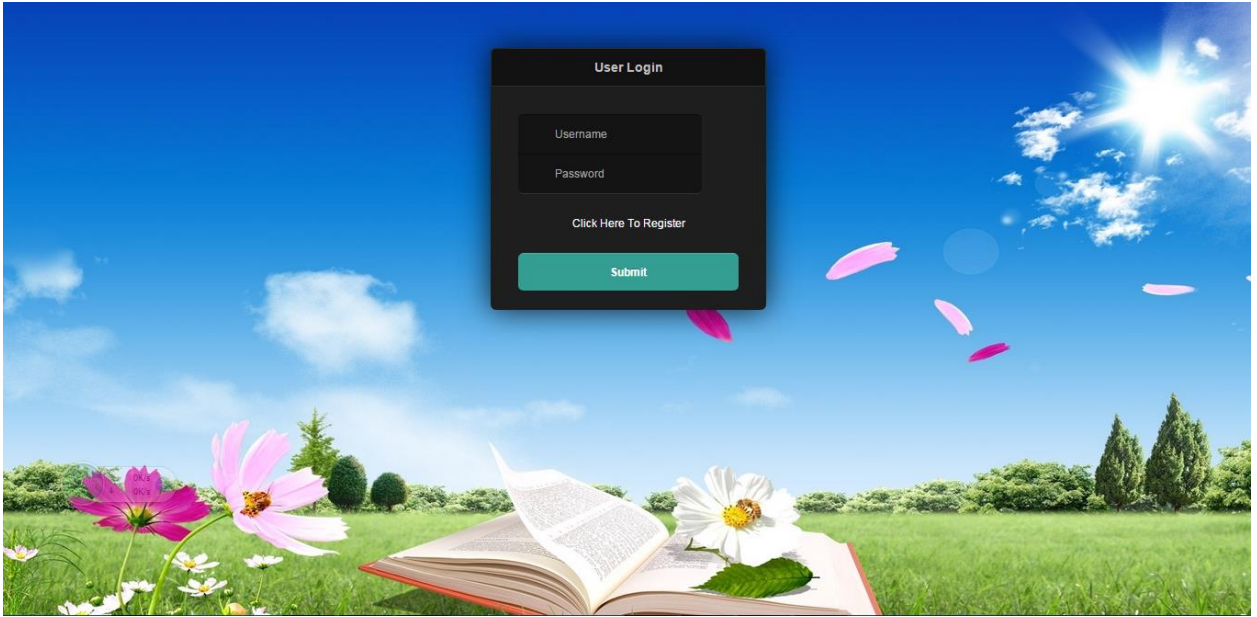
- UserName:
- Password:
- First Name:
- Last Name:
- Gender:
- Role:
- Email:
- Phone:

At the bottom left, there is a "Register" button and a link that says "Please click here to Login".

**Figure 5-2-F1: Register Page**

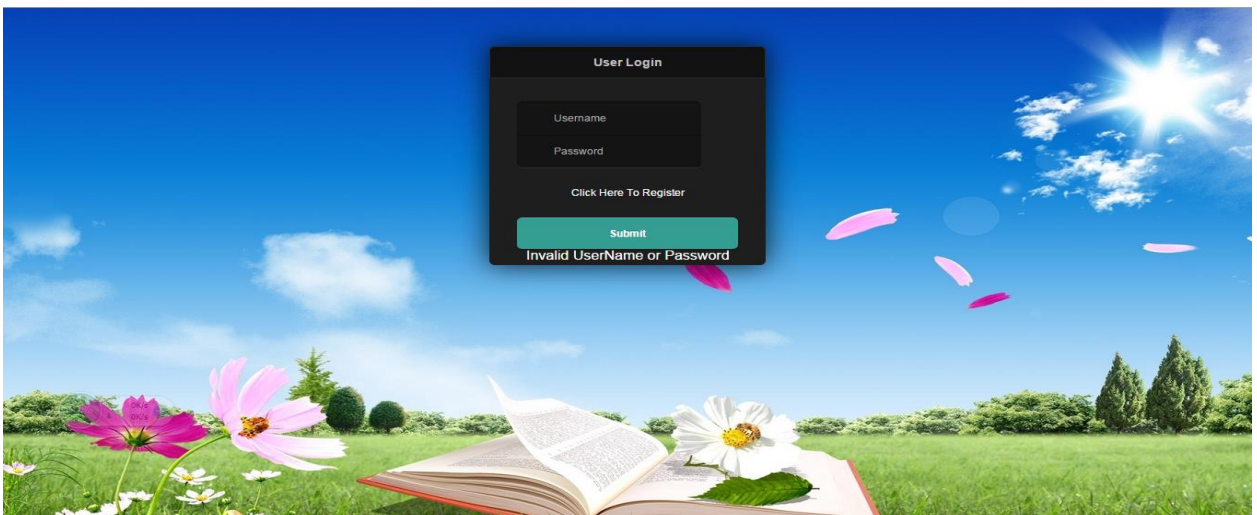
This is the register page for the user to register an account for the system. The user has to fill in all the details in order to register for an account. After user submitted the detail, the system will verify the data of the user entered. If every data is correct the system will display successful register account message to the user. User can return to the login page to login into the system. The username must be unique also cannot same with the username that already exist in the database, the system will display error message and request user to key in another username.





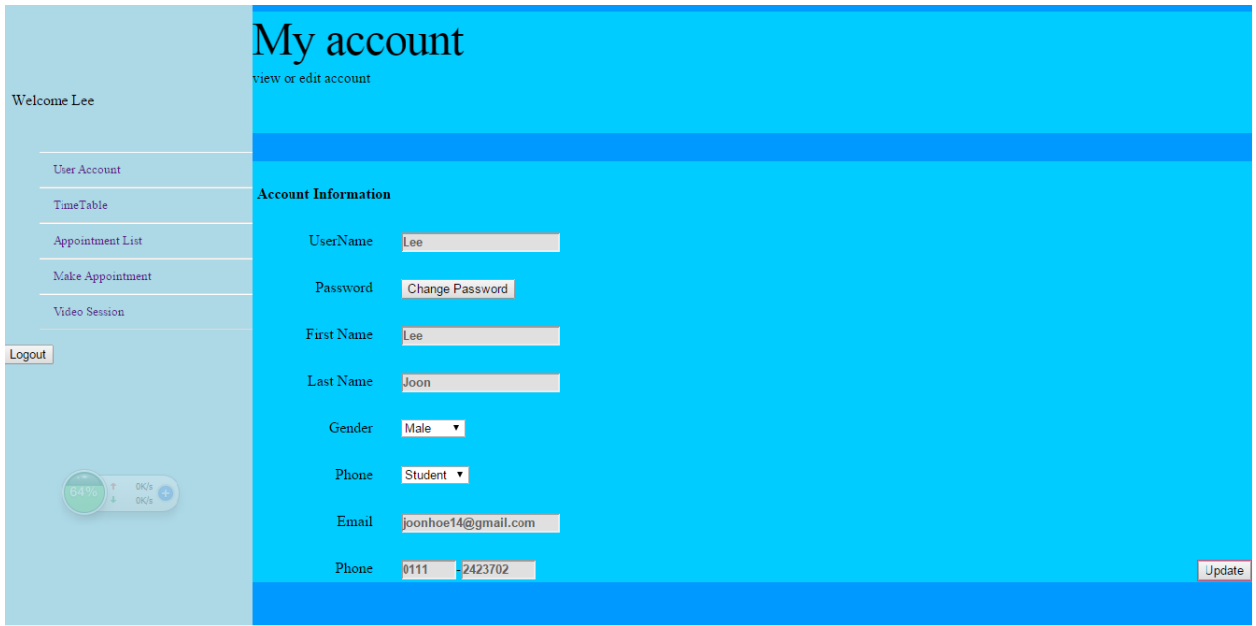
**Figure 5-2-F2: Login Page**

This is the login page for the web chat system. The user has to enter their username and password in order to login to this portal. If the user does not have a user name he can go to register page to register for an account. After user fill in the detail he can login into the system. The system will verify the form data with the database.



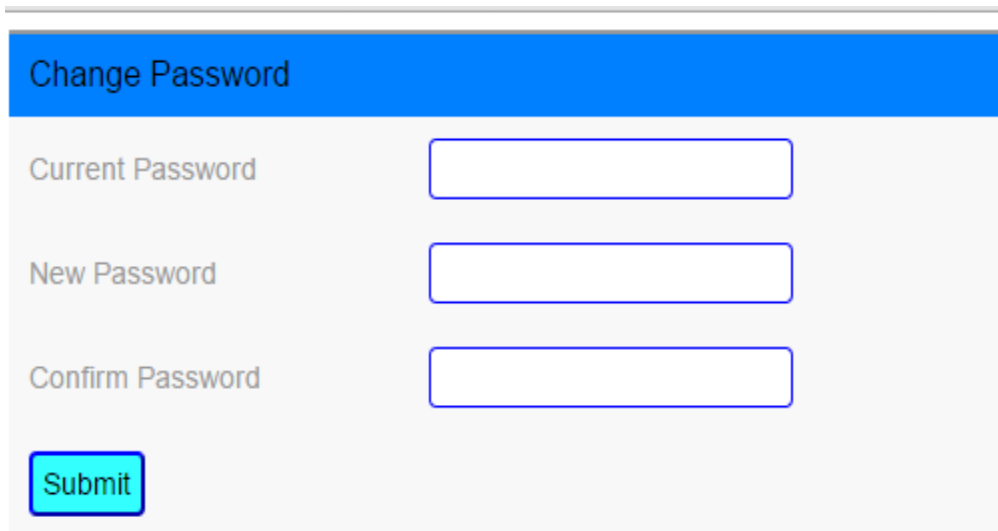
**Figure 5-2-F3: Register Page(Invalid Login)**

If the username or password is incorrect or not exist in the database, the system will display error message to user. The user has to reenter the data to login to the system.



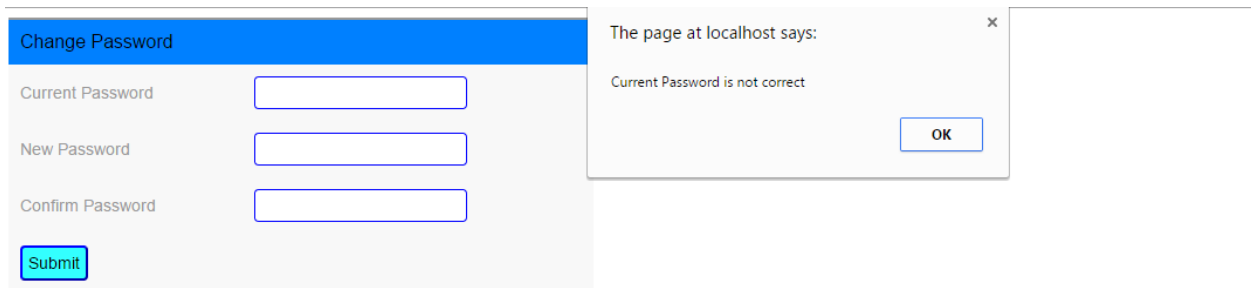
**Figure 5-2-F4: User Account page**

After the user successfully login, the first page that the user enter will be the user account page. The user can view his account information here. If the user wish to edit his account information he can enter the data into the field which is editable except for the username that are not editable. If the user which to change his password, user can click on the change password button to change the password.



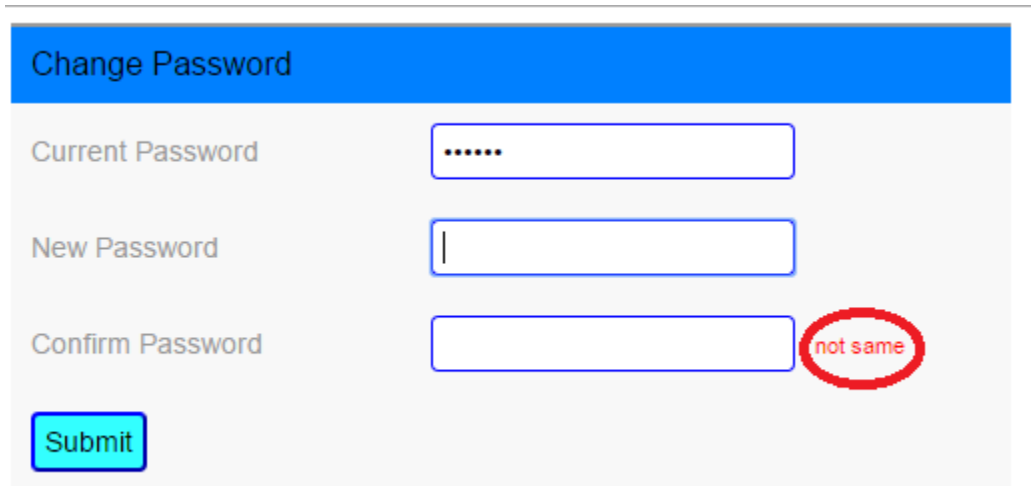
**Figure 5-2-F5: Change Password PopUp**

A pop up window will be show for the user that wish to change the password. Firstly the user have to reenter the current password. After that the user need to enter the new password. Lastly the user have to reenter the new password to reconfirm. If the input is correct the user will tell the user the password have already successfully change and will close this pop up and return to the account information page.



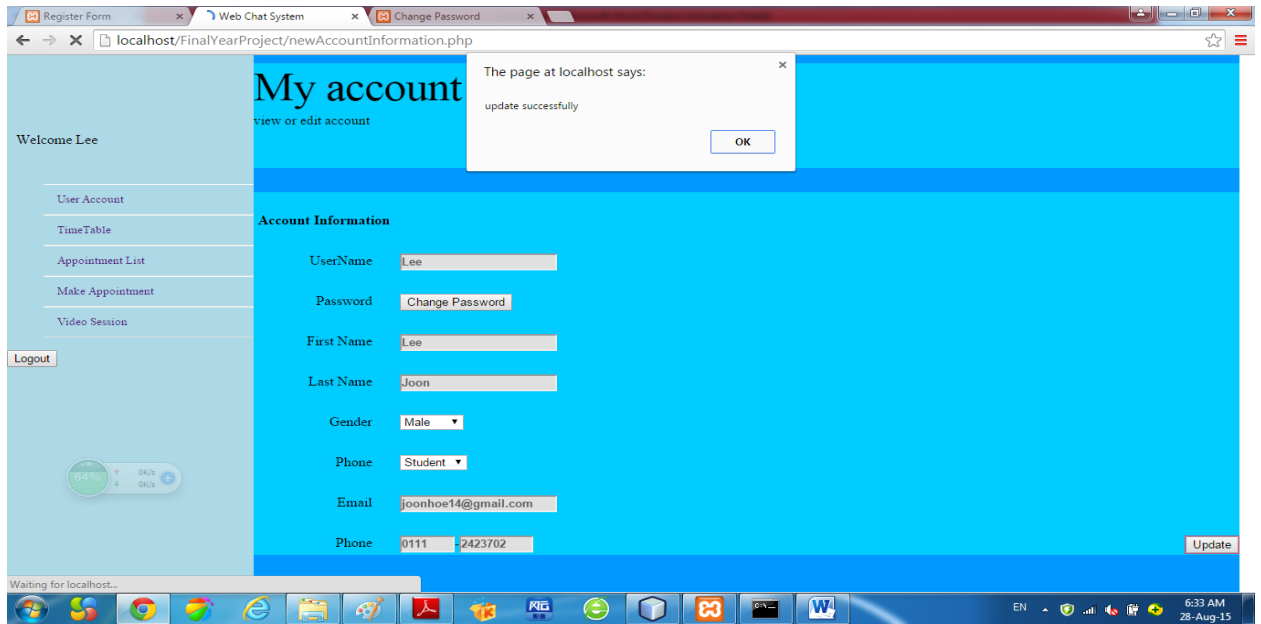
**Figure 5-2-F6: Change Password PopUp(Incorrect current password)**

If the current password is not same with the password in the database system will display error message.



**Figure 5-2-F7: Change Password PopUp(New and confirmation password not Same)**

If the new password and the confirm password are not same, the system will display not same message.



**Figure 5-2-F8: User Account(Update Successfully)**

User edited the information and click on the update button. System will show the update successfully message.



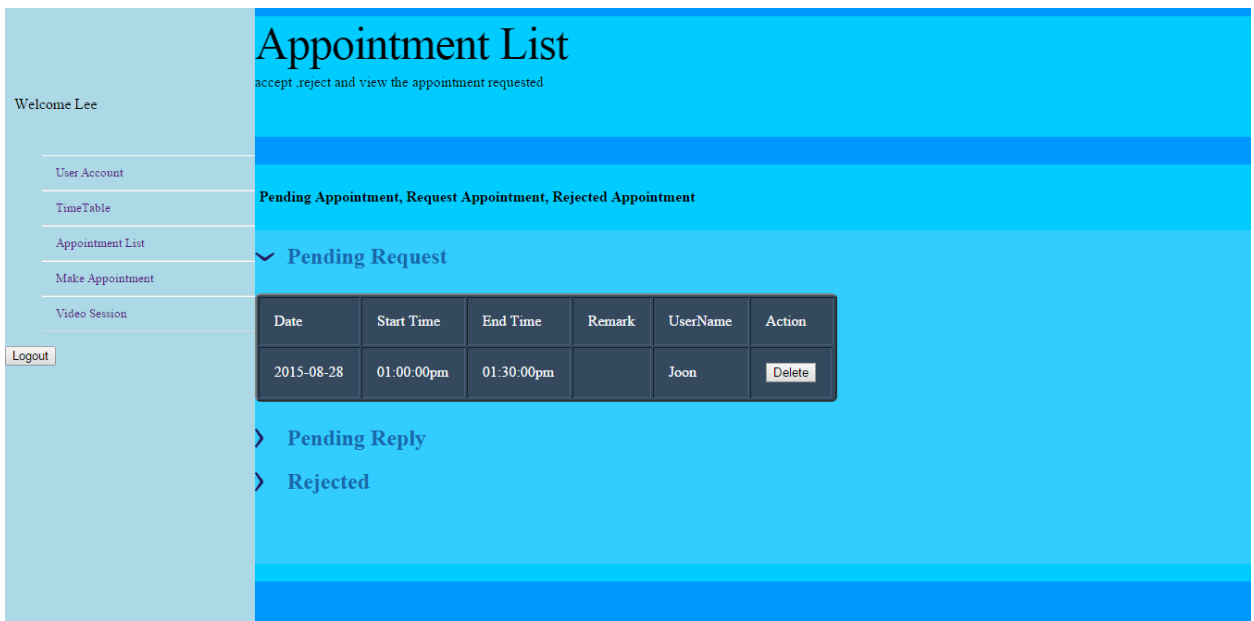
**Figure 5-2-F8: Timetable Page**

This is the timetable page that shows the user about the information of the scheduled upcoming appointments that have been made and accepted.



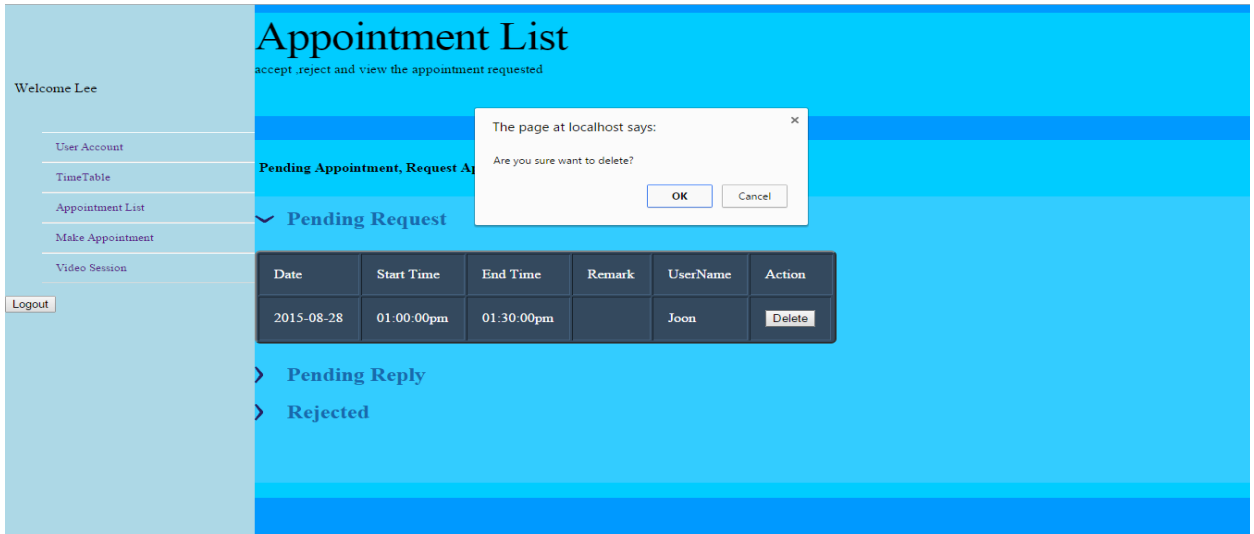
**Figure 5-2-F9: Appointment List**

This page will show the status of the appointment. There are 3 panel of the status appointment will be shown which is the pending request, pending reply, and rejected.



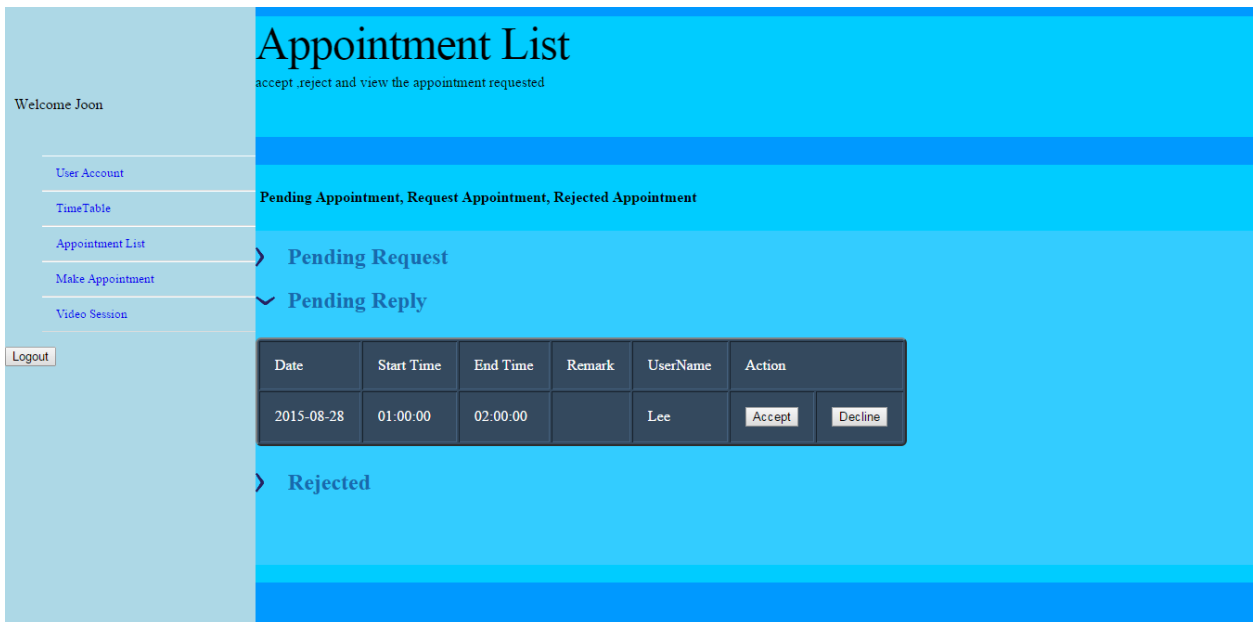
**Figure 5-2-F10: Appointment List(Pending Request)**

The pending request panel is show the information of the appointment that has been made and did not reply yet.



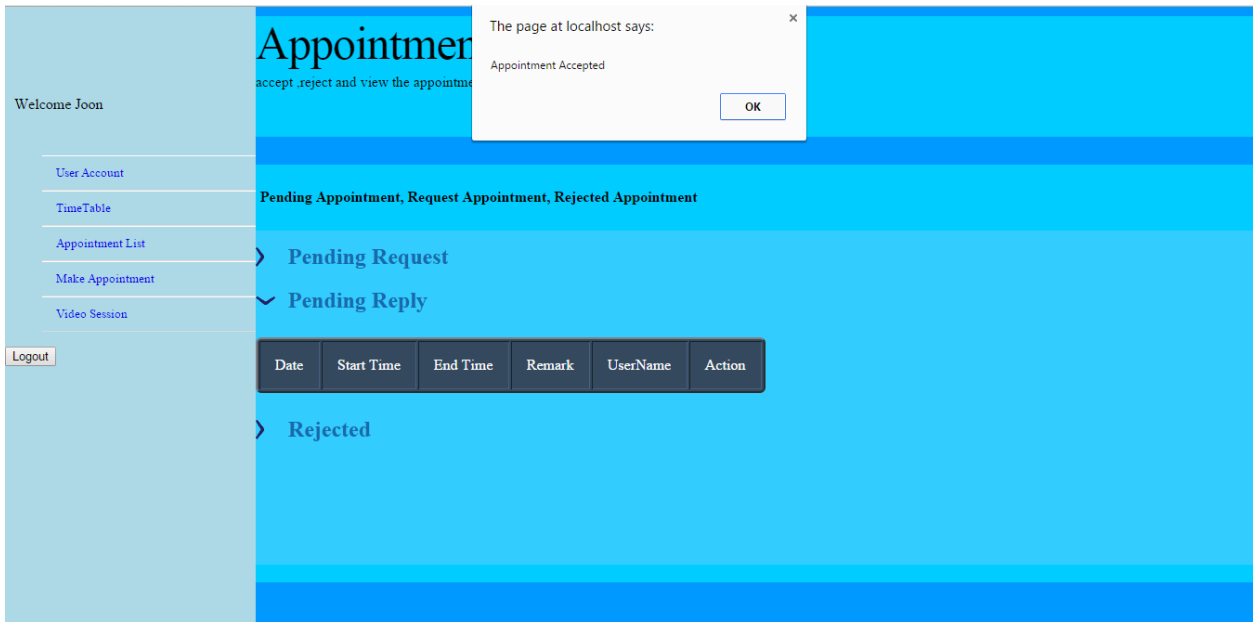
**Figure 5-2-F11: Appointment List (Pending Request Delete Appointment made)**

If the user wanted to delete the appointment that have been made and yet not responded, user can click on the delete button to delete the appointment. A confirmation box will pop to ask the user want to delete the appointment or not. If the user pressed OK, the appointment will be deleted.



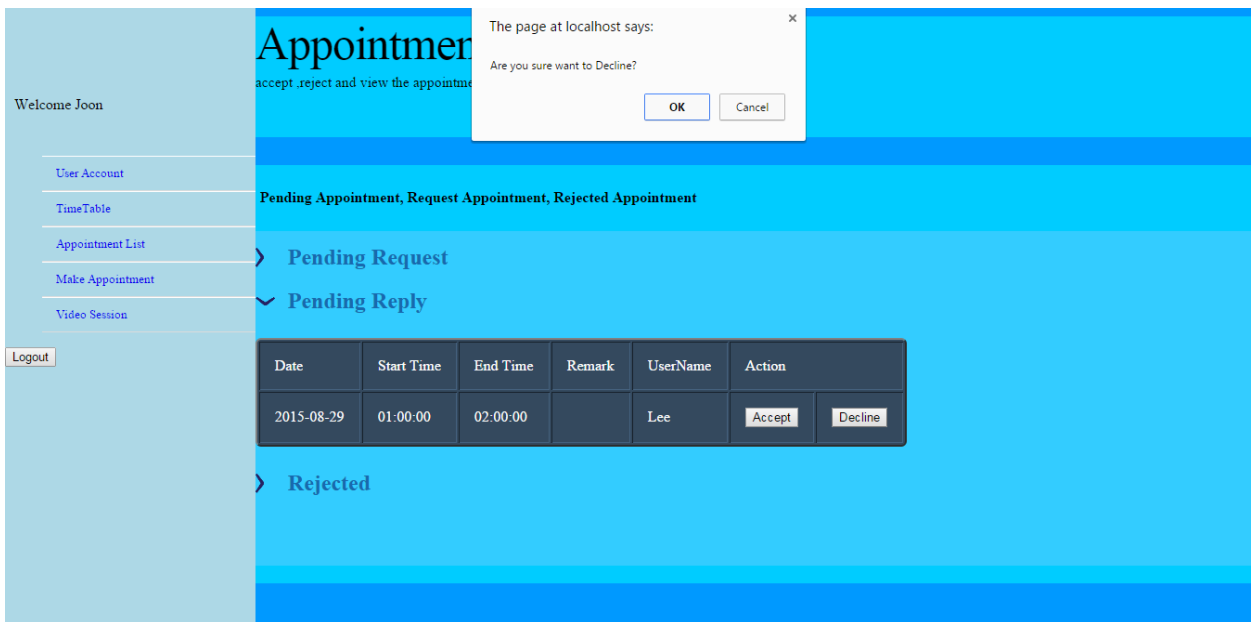
**Figure 5-2-F12: Appointment List (Pending Reply)**

The pending reply panel will show the information of the appointment that request by others users. The user can make an action whether reject or accept the appointment.



**Figure 5-2-F13: Appointment List (Pending Reply Accept Appointment)**

If the user pressed on accept button, the system will show the user the appointment have been successfully accepted.



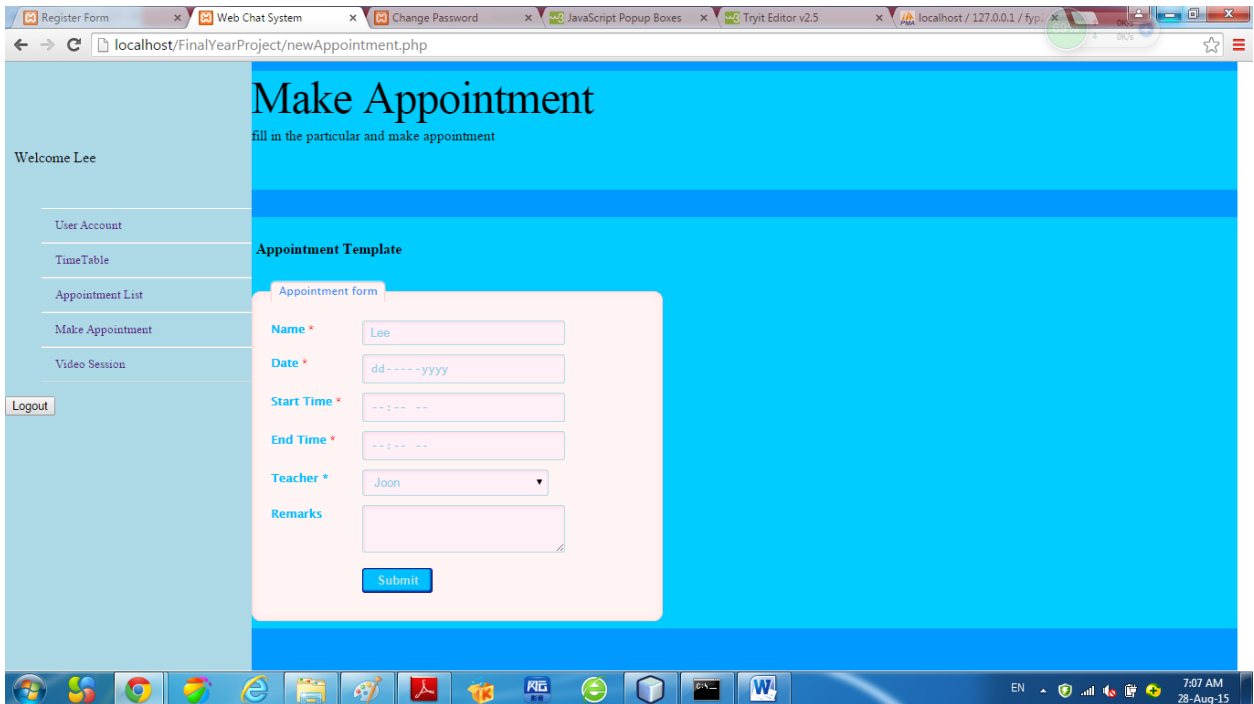
**Figure 5-2-F14: Appointment List (Pending Reply Decline Appointment)**

If the user wanted to decline the appointment, he can press the decline button. The system will prompt user for confirmation. If user pressed ok the appointment will be rejected.



Figure 5-2-F15: Appointment List (Rejected Appointment)

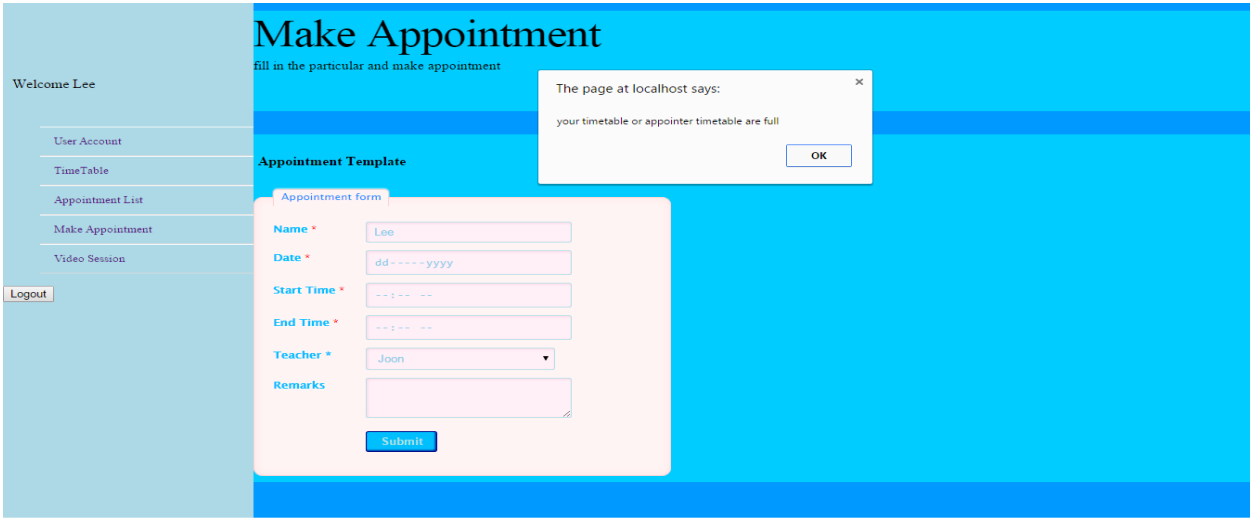
The rejected panel will show the information of the appointment that have been made and rejected.





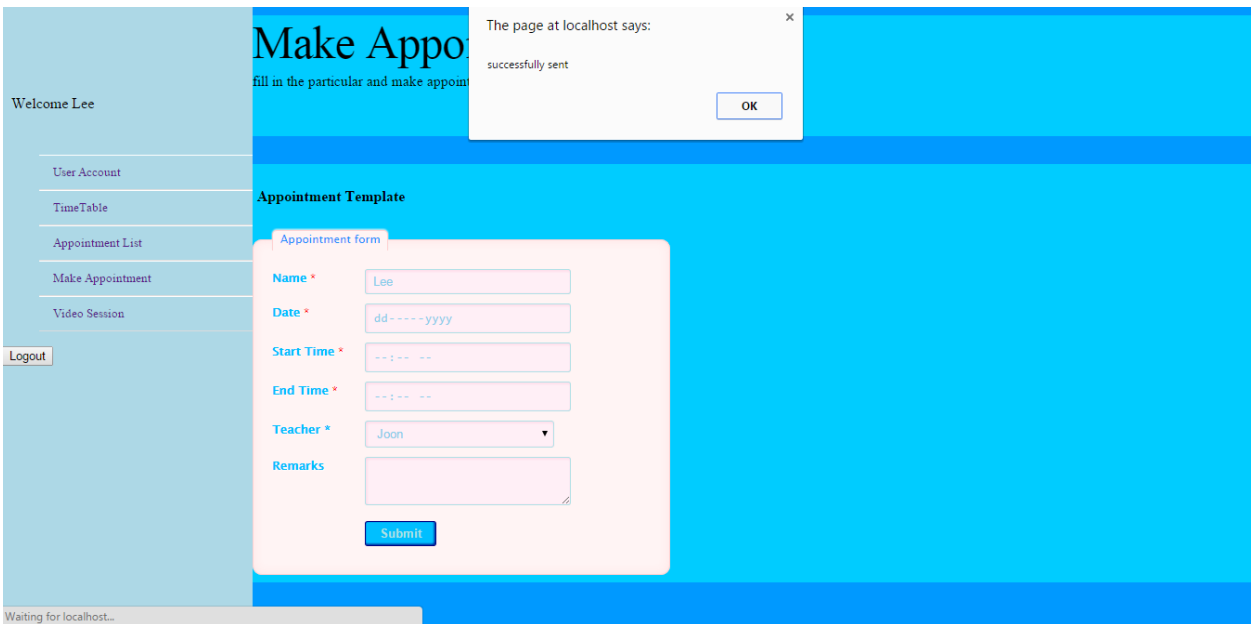
**Figure 5-2-F16: Make Appointment**

This page enabled the user to fill in the template and make an appointment with others user.



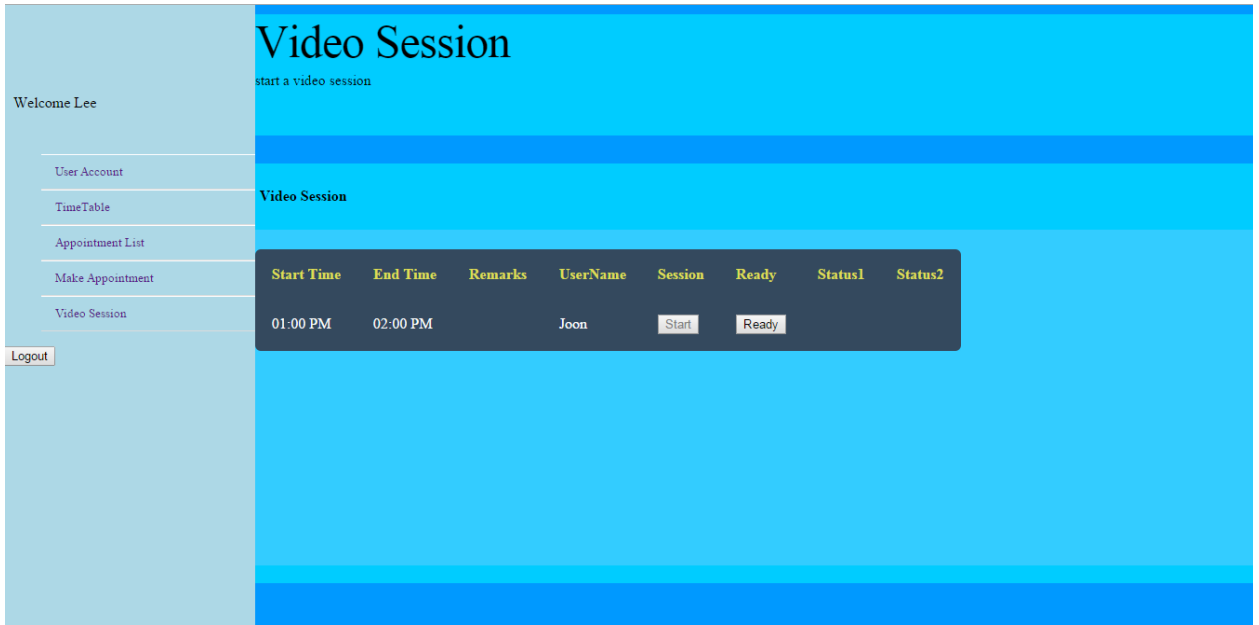
**Figure 5-2-F17: Make appointment (timetable clashing)**

If user or another users timetable clash then the appointment will not be made and error message will be display.



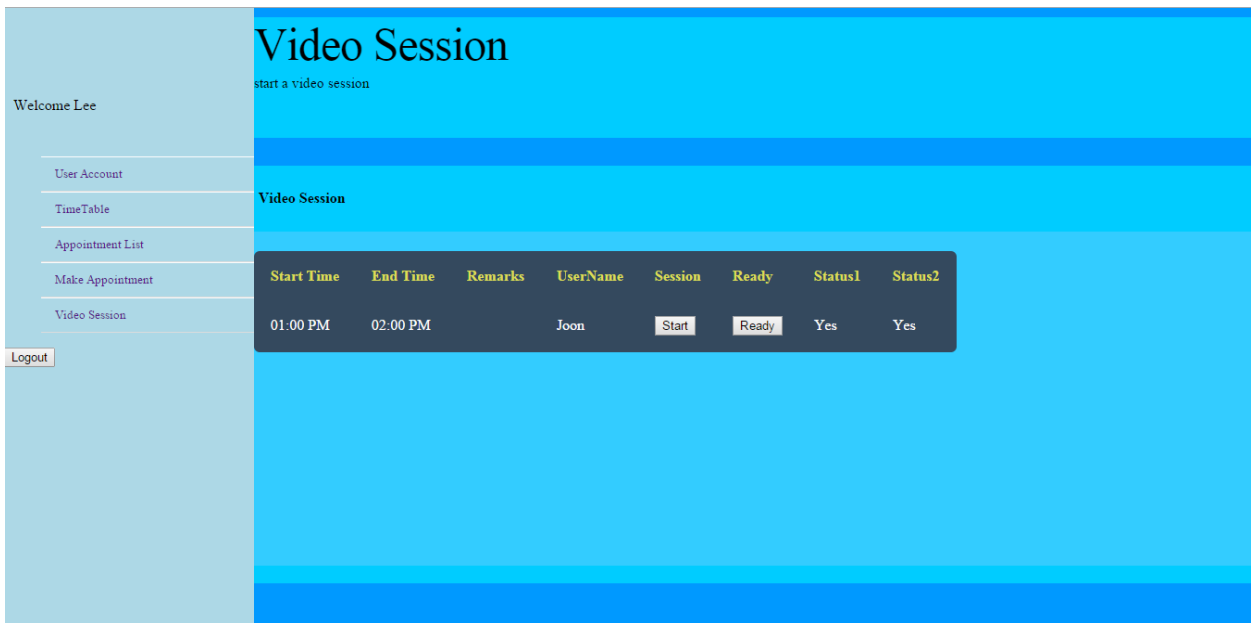
**Figure 5-2-F18: Make appointment (successful)**

If the time table does not clash, the appointment will be made and sent successfully.



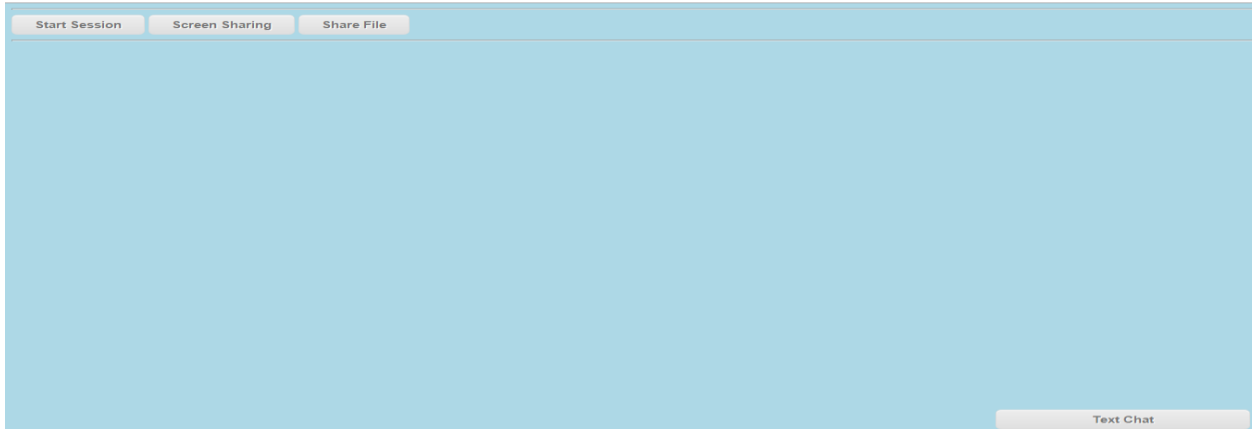
**Figure 5-2-F19: Make appointment (successful)**

This page enabled the user to open video session page with another appointment user. It will display the appointment that has been accepted.



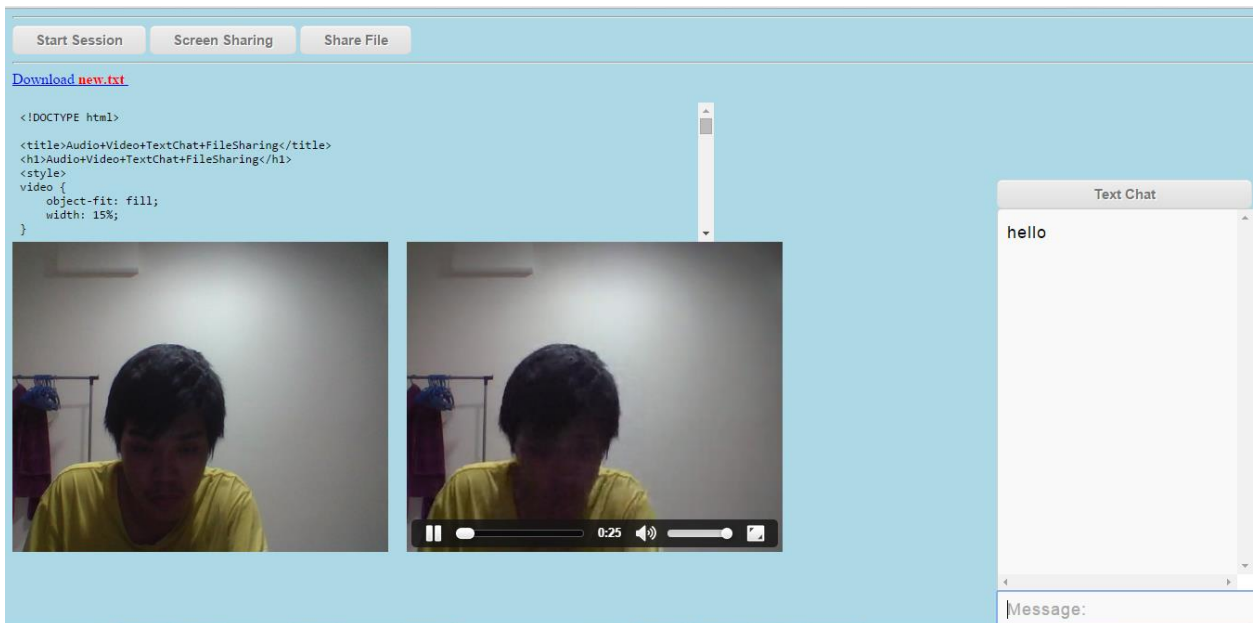
**Figure 5-2-F20: Video Session**

If the users are ready, they need to click the ready button in order to inform each other that are ready. After both status1 and status2 become “YES”, the start session button will be enabled. Click on the start button a new window will be pop.



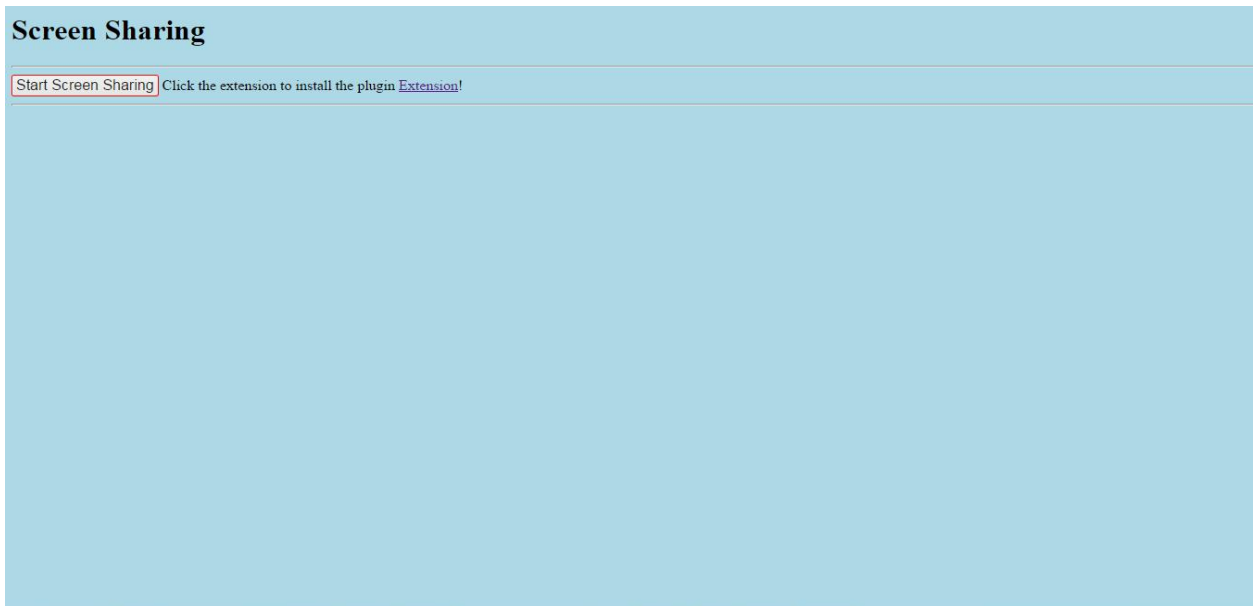
**Figure 5-2-F21: WebRTC page**

This page is the video session page for the users to start the conversation. The user click on the start session button to start the webcam of the computer. Another user also needs to click on the start session button.



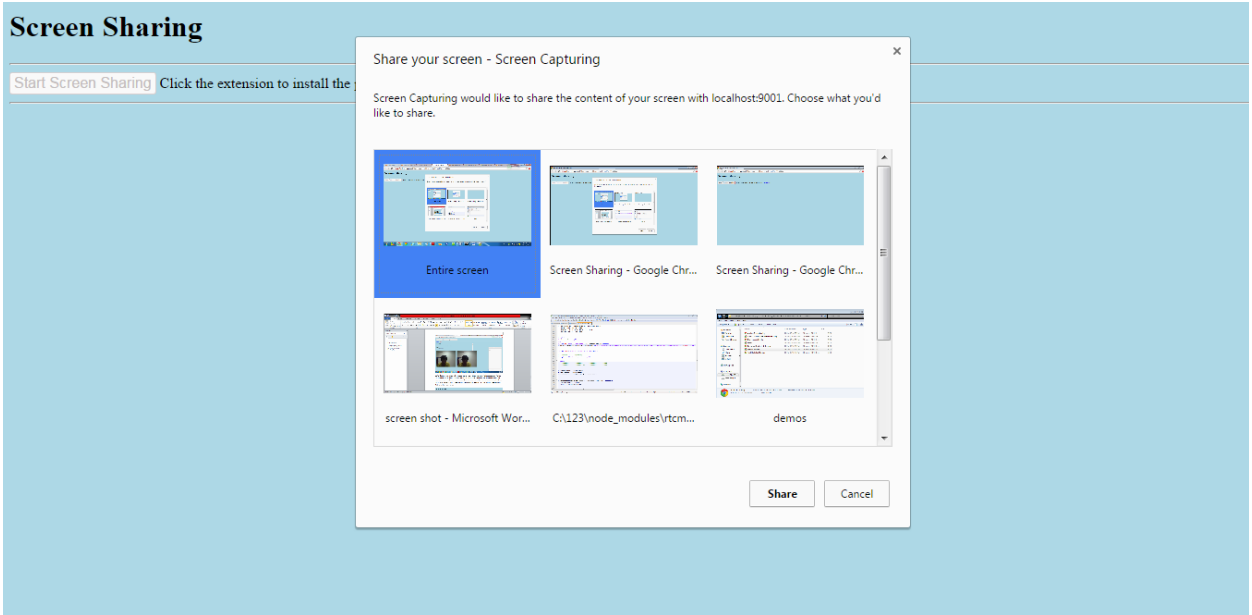
**Figure 5-2-F22: WebRTC page(connection establish)**

After that the connection will be open. The system will prompt user for the permission for the audio and video enabled. Users have to allow the browser capturing the audio and video. The user can to have text chats conversation. The user also can to share the files among each other's. If the user wishes to have screen sharing to share his screen he can click on the screen sharing button to share his screen.



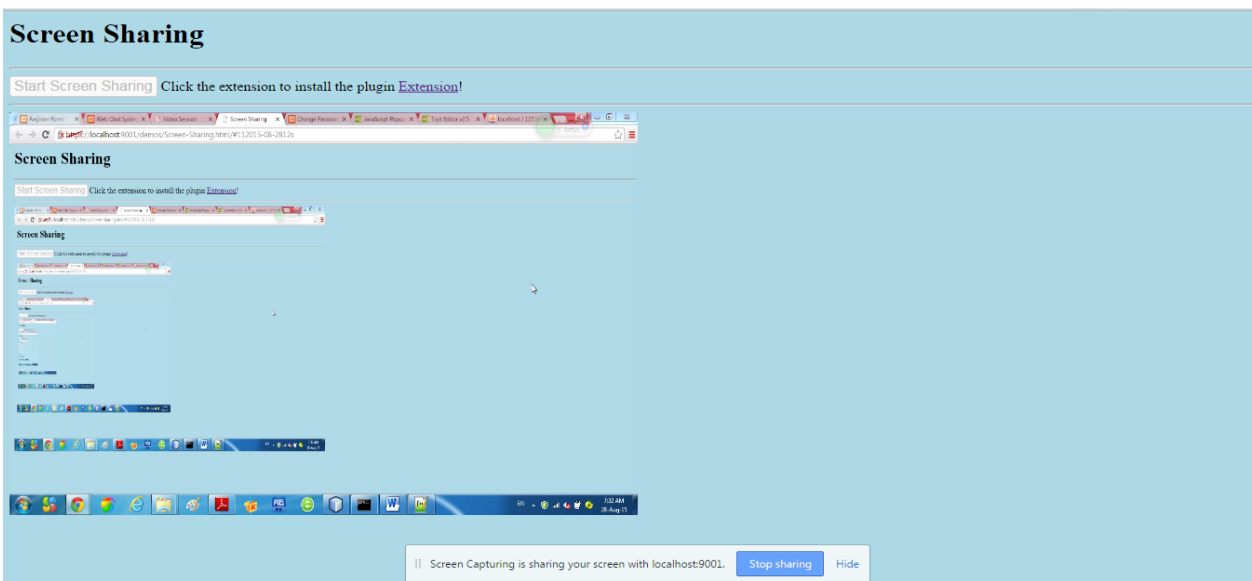
**Figure 5-2-F23: Screen Sharing page**

This page enabled to share his screen to another user. In order to share the screen, the user must have the screen capturing plug in installed. If the user does not install yet, he can click on the link above to install the plug in. make sure the browser have enable the plugin.



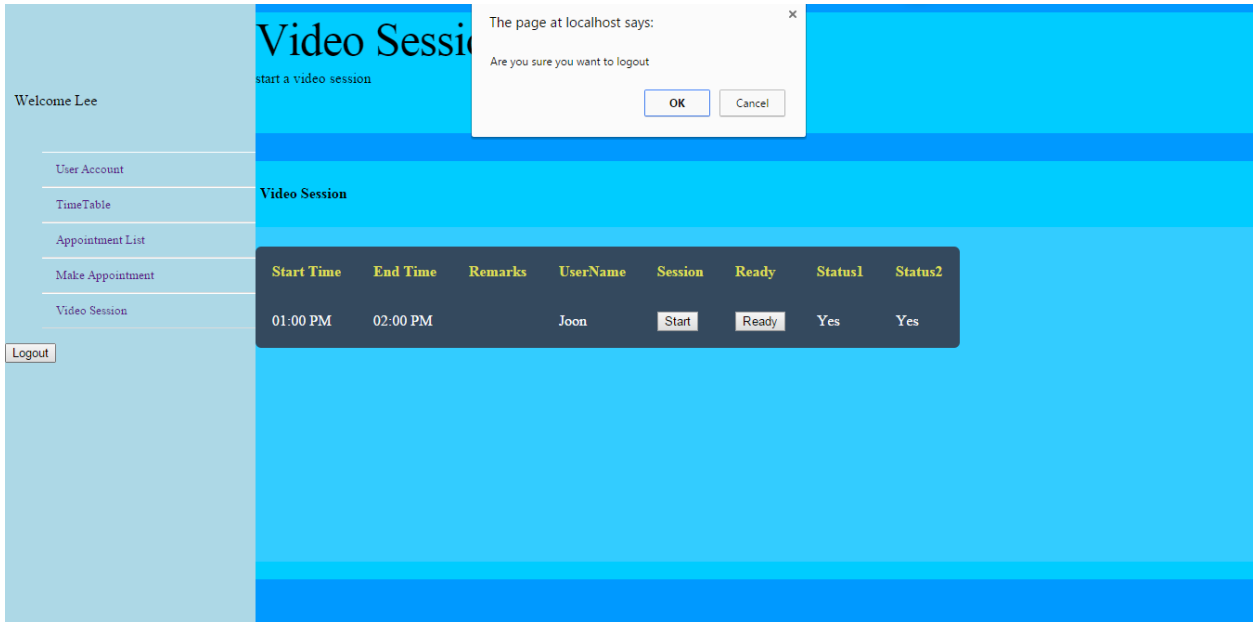
**Figure 5-2-F24: Screen Selection**

The browser will prompt for the user for which screen user want to share.



**Figure 5-2-F25: Screen sharing pagescreen sharing start)**

The user is sharing the screen with another user.



**Figure 5-2-F26: Logout**

User click on the logout button to logout from the system. The system will pop confirmation to ensure the user wanted to logout from the system.

## **Chapter 6: Conclusion**

### **6.1 Project Review, Discussions and Conclusions:**

This project is talking about develop a chat system for the teacher and student. The motivation of develop this project is because the communication through internet become commonly in now daily life. Communication using internet in the academic field might help the student and teacher communicate more effective and efficient. The transitional consultation method may give several problems. The first problem is student or lecturer is not available in the school. With the online consultation solution, location is not a concern as long as have internet. Furthermore, if the student need to show their work on their computer they can show in here because some student might only have a desktop not a laptop it is not convenient to bring the entire desktop to the school. The second issue is bunches of email have to filter by the lecturer. A systematic appointment system will help in manage the appointment more effective and efficient. The student or teacher can manage their appointment easily using the appointment system. The third issue is arrangement issue regarding the appointment. Lecturer might mistakenly schedule the appointment. The appointment system will detect that if the time slot is available and ensure that no 2 consultation session is schedule at same time. If the timeslot either lecturer or student have clashing the appointment will not be made. The last issue is failed to recall about the consultation. It is human nature that forgets some stuff from time to time. SMS reminder system developed helped to remind the lecturer about the consultation session. In order to develop the WebRTC, RTCMulticonnection have been implemented. Thanks to Muaz Khan for developing this awesome API. The documentation also explained very detail about how to use the library.

### **6.2 Future Work:**

As time go on, the system should be improved to a bigger scope. There are few recommendation can be used in the future development. Firstly, the online consultation solution which is the WebRTC can be built on the mobile browser since today era lot of

peoples are afford to have a smartphone. Secondary, the appointment system can added 1 more modules which is make the teacher set a period of availability time. When the student make the appointment with the teacher, if the system detect the time is not in range of the availability time, it will display error message. Thirdly, if a group of student that having the same topic of question. The lecturer manage to edit the appointment can be joined by how many person. After that the online consultation will be perform one to many broadcasts for the group of students. Last but not least, the entire online consultation session can be recorded as the reference for the student. If the student wanted to watch again the consultation session, he can replay the video.