ROOM RESERVATION AND SCHEDULING SYSTEM FOR IICS

A Capstone Project

presented to the

Department of Information Technology

Institute of Information and Computing Sciences

University of Santo Tomas

In partial fulfilment

of the requirements in the degree of

Bachelor of Science in Information Technology

Quitor, Christian Jeus D.P.

Plata, Tristan Miguel C.

Silvestre, Rich Carl Michael

Valdes, Paolo O.

Perol, Alma V.

March 2019

Table of Contents

List of Figures

Chapter 1: Introduction

Project Context

Room reservation requests in the Institute of Information and Computing Sciences are done through email approvals. The person requesting for a room must send an e-mail to the IICS office regarding their reservation request, which must be approved by the said office. However, plotting a vacant room is done manually, making it hard to keep track of which rooms are vacant at a specific time whenever someone makes a request. As for their current scheduling system, new schedules require manual input of data, and manual plotting of data entered, which requires a bit of effort from the admin. The team had come up with a solution: to make a web-based system that will serve as a basis for room scheduling and reservation system for the IICS.

The system involved databases for the room schedules of the IT, IS, and CS departments. The system required a user to log-in using his or her unique ID (student number for students, and employee number for faculty members). Then, a request to reserve a room for a specific span of time. The admin will either approve or deny the request. Lastly, an e-mail notification will be sent by the admin as a response to the request, whether or not the request has been approved or declined.

The system also involved a Calendar for the Scheduling module that can be used to create schedules for a section. The department heads have access to the Scheduling module of the system. The Calendar allowed the head to select a particular timeframe first, after which data can be entered for the timeframe selected.

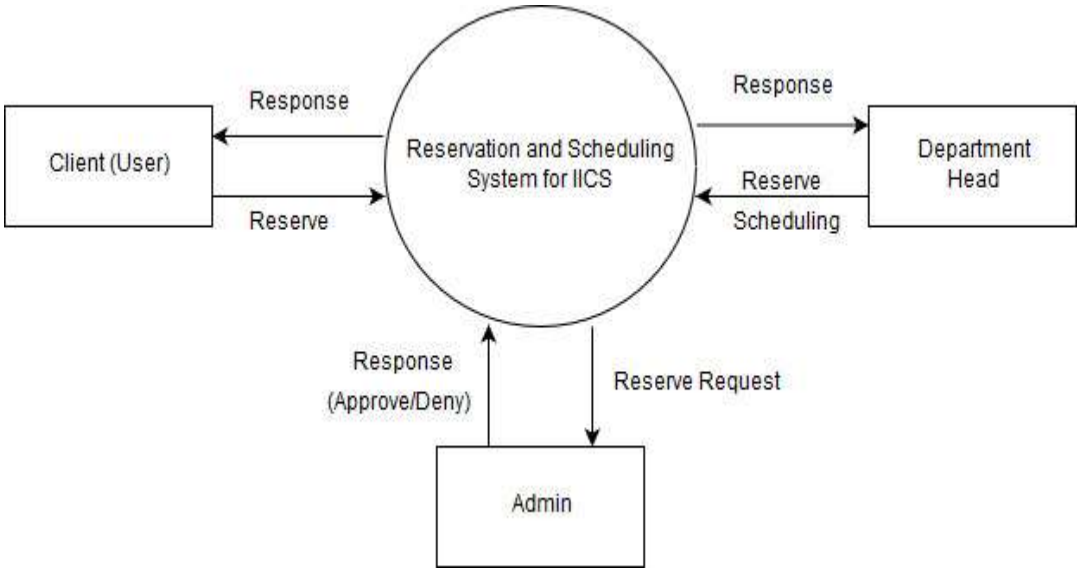


Figure 1. 1: Context Diagram

Purpose and Description

The applications benefit students, faculty members, and the administrators inside the IICS Office. For both students and faculty members, they will select a room and enter their desired time for their reservation. The administrators will then be responsible for the approval of the reservation requests. They can also deny the requests if there are problems encountered prior to the reservation request.

The reservation application allowed the students and faculty members to make reservations without the need to fill-out a reservation form. The application also allowed administrators to view the list of reservation requests.

Also, the scheduling application helped the department heads to create schedules for a particular section without having to manually enter data. The Scheduling module derived from data entered in the Maintenance module.

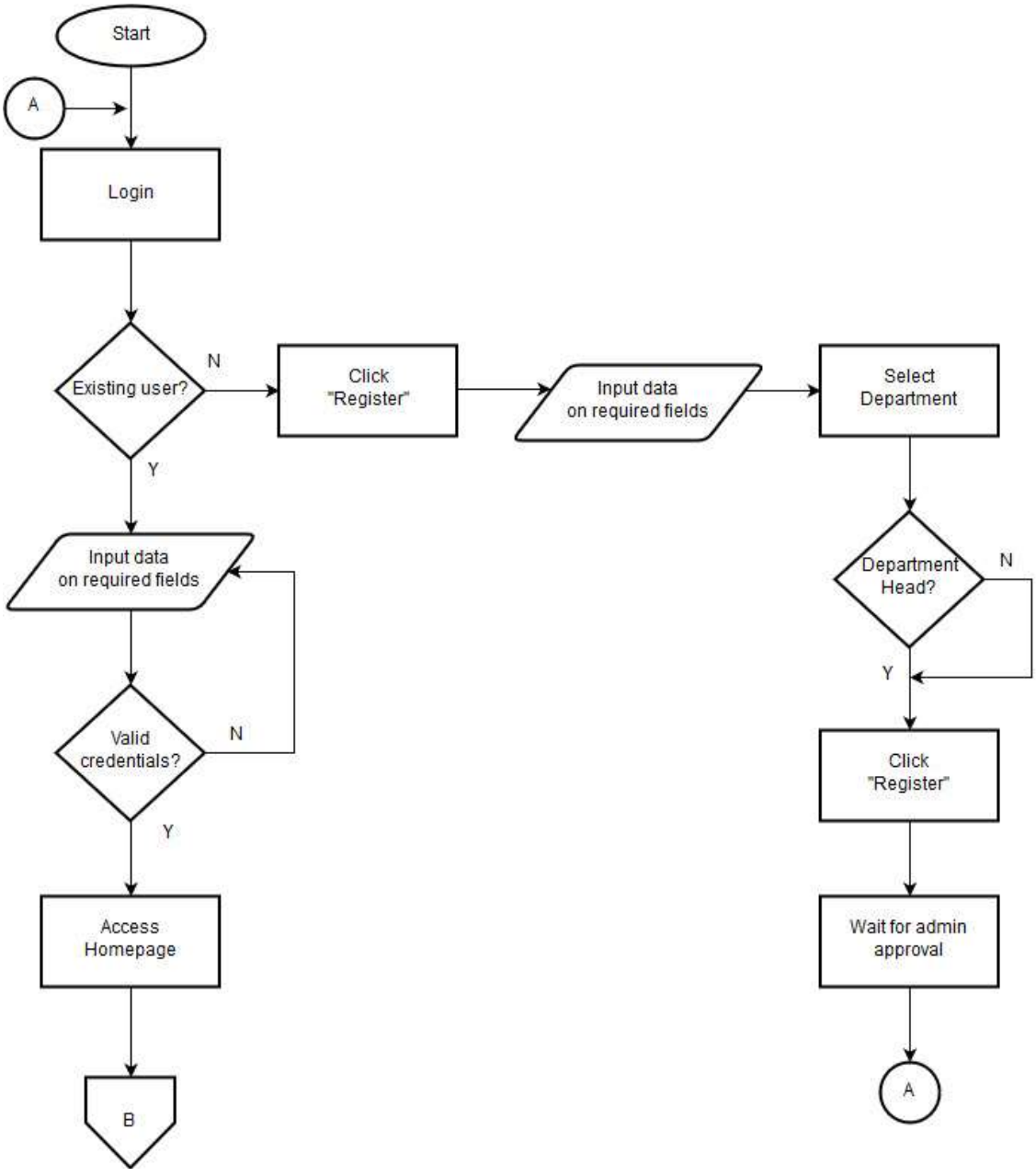


Figure 1.2.1: Client Flowchart – Login/Register

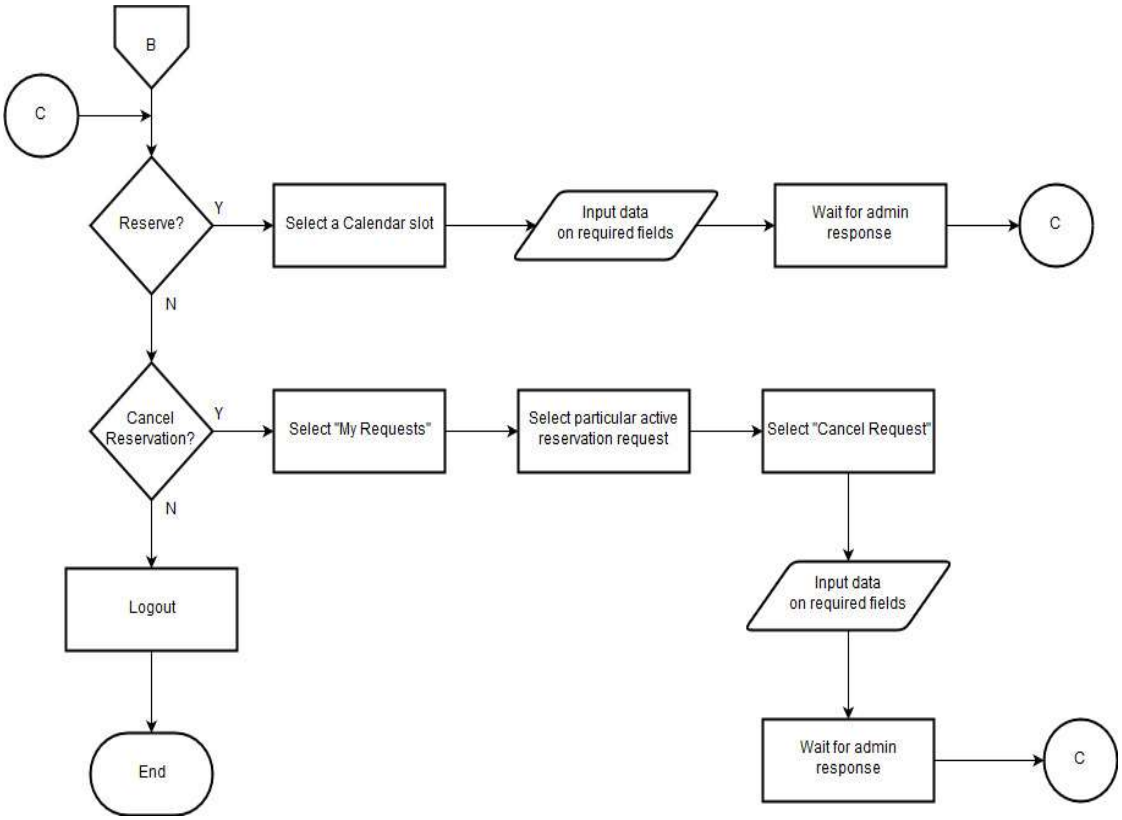


Figure 1.3.2: Client Flowchart – Room Reservation

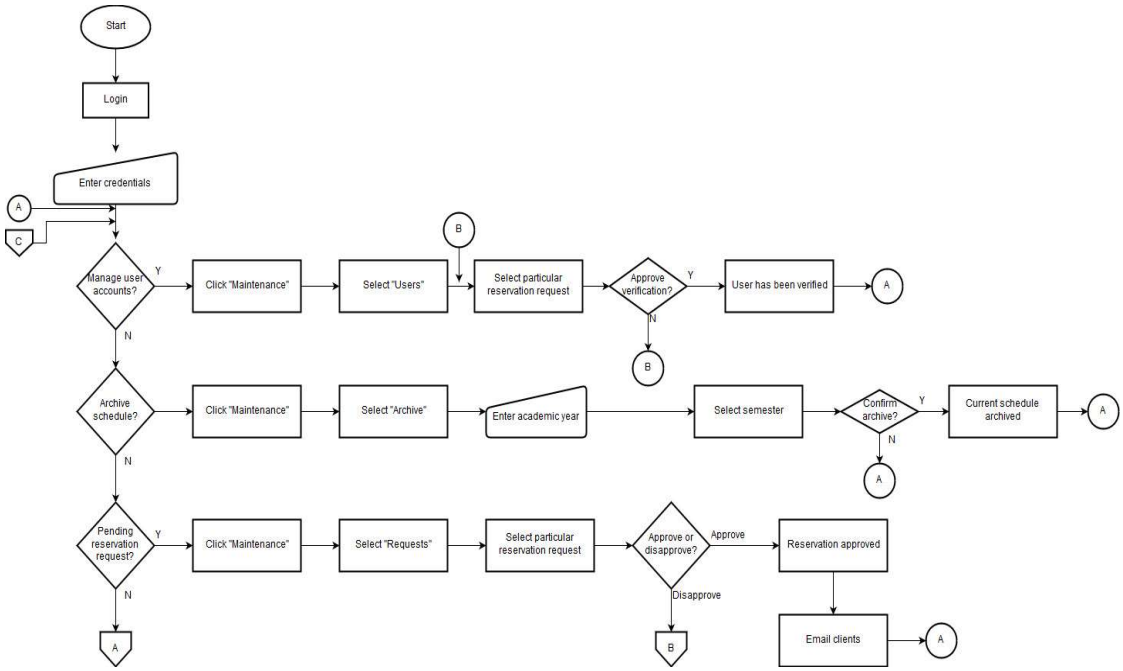


Figure 1.4.3: Admin Flowchart – Manage Accounts/Approval of Requests

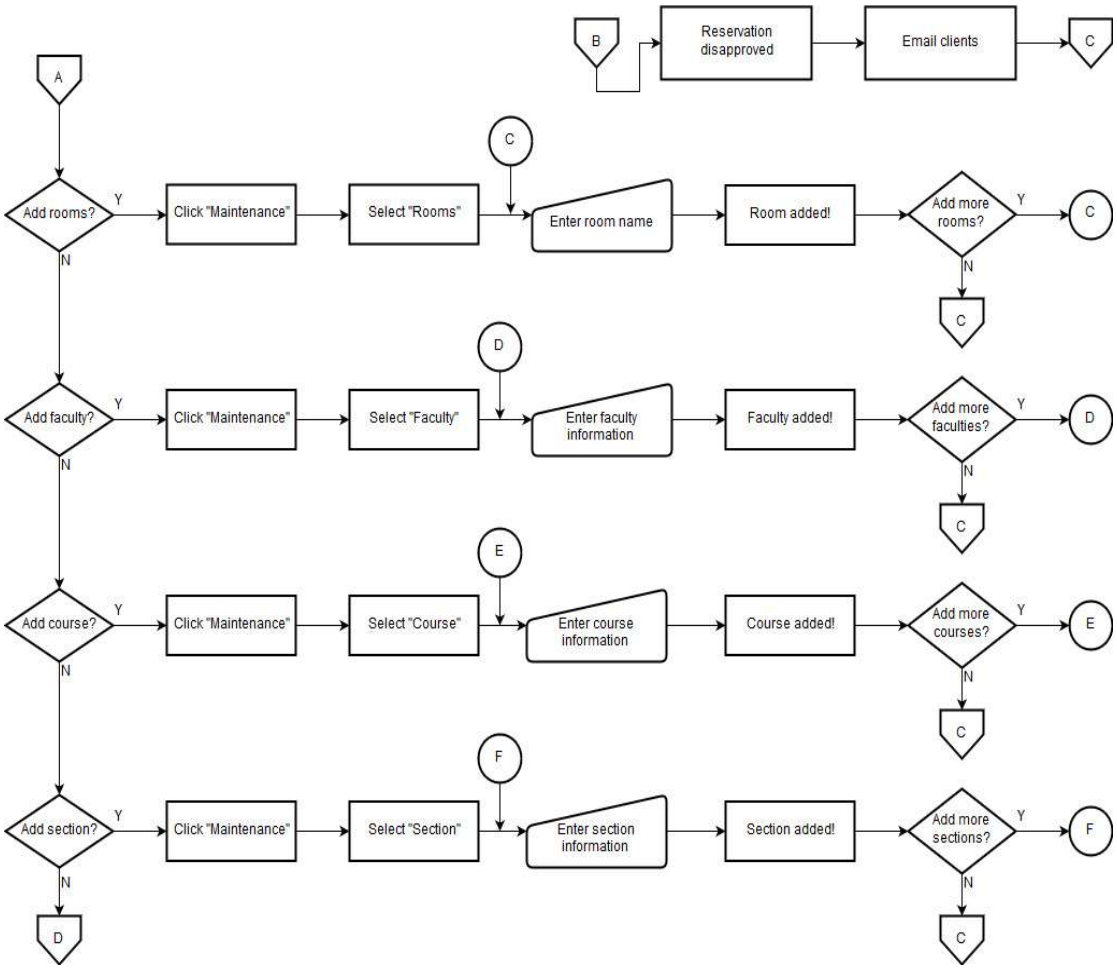


Figure 1.5.4: Admin Flowchart – Add Facility of Rooms/Faculty/Courses/Sections

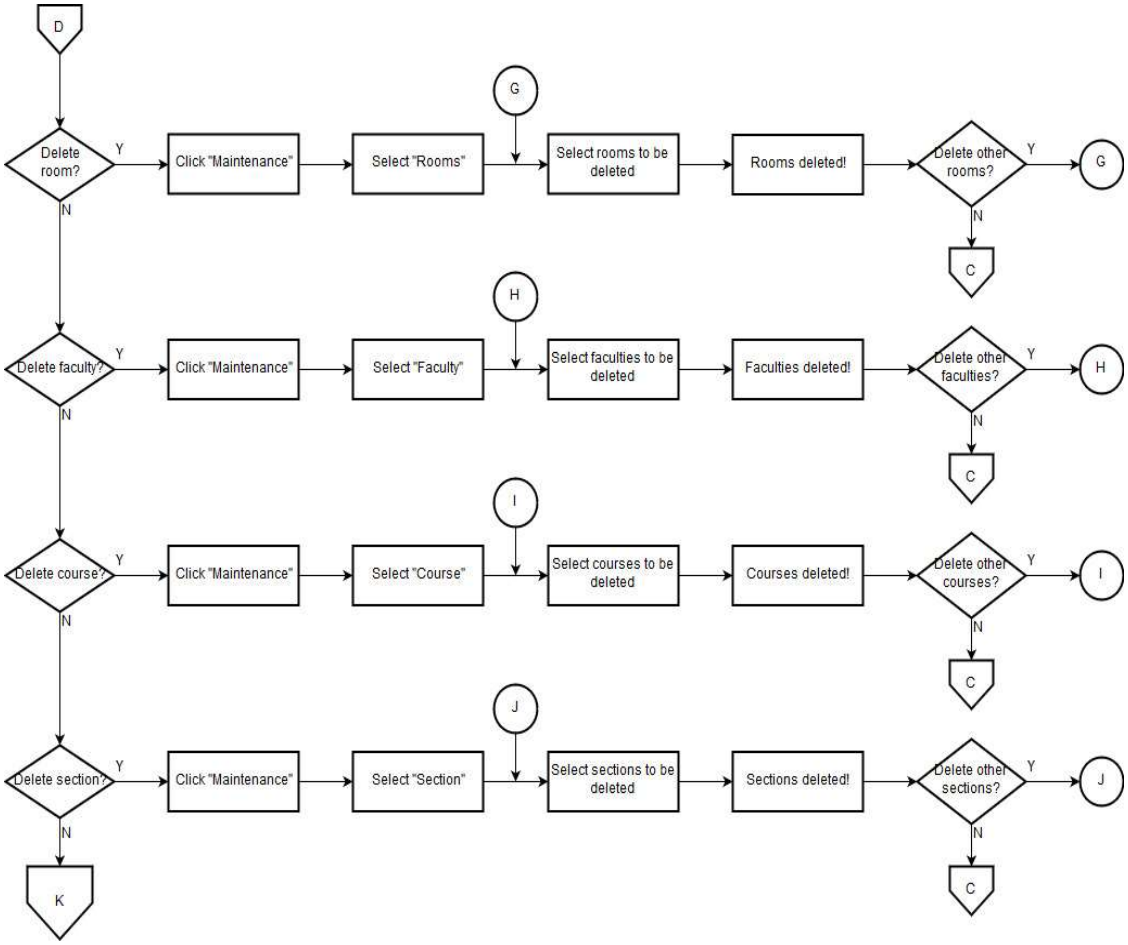


Figure 1.6.5: Admin Flowchart – Delete Facility of Rooms/Facility/Courses/Sections

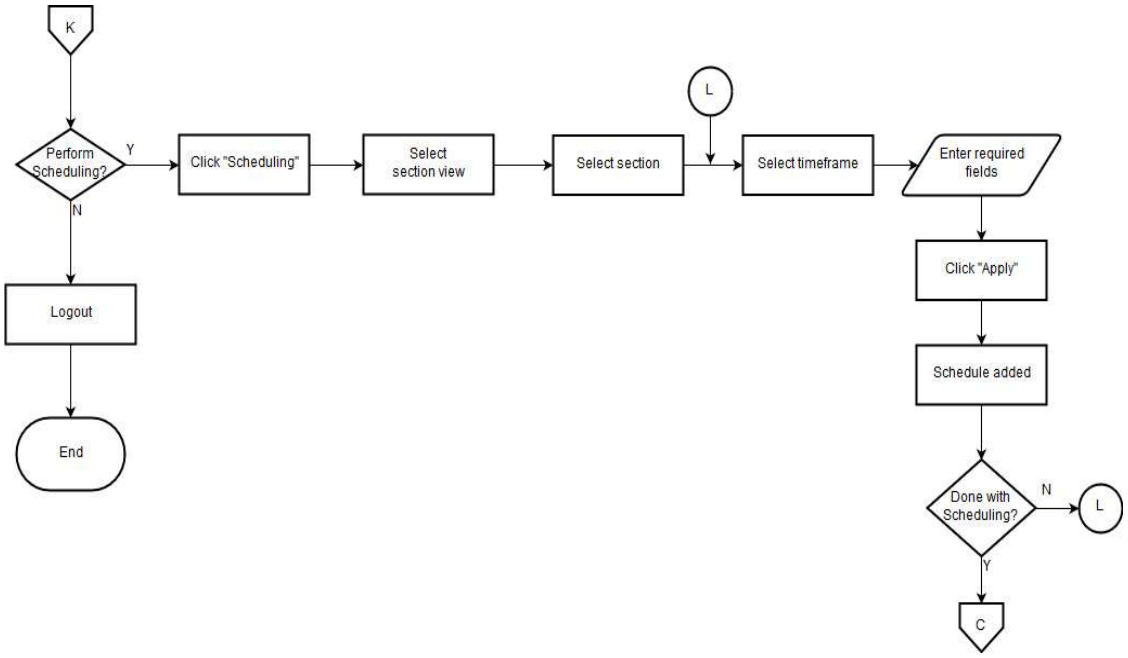


Figure 1.7.6: Admin Flowchart – Scheduling

Objectives

General Objective

The project aims to develop a web-based system that will integrate both room scheduling and reservation for the Institute of Information and Computing Sciences.

Specific Objectives

1. To view events, schedules, and reservation through a Calendar
2. To create a maintenance function for managing rooms, faculty, courses, sections, and users
3. To feature an Archive function that allows admins to create a record of previous schedules

4. To add a function that allows clients and admins to view the status of reservation requests
5. To create a module for creating a schedule of students, faculty, and rooms every semester, in consideration of the following constraints:
    - Rooms
    - Number of minimum and maximum of units of faculty members
    - Number of hours
    - Courses
    - Sections per semester
    - Special classes and petition classes
    - Request of faculty members
6. To create a module that will generate reports which will include the following:
    - Room Schedule
    - Faculty Schedule
    - Sections Schedule

Scope and Limitations

1. The system can only allow 4 types of Access Levels: User, New User, Department Head, and Admin.
2. The password complexity of the system includes an uppercase character, lowercase character, and number for the password.
3. The calendar feature of the system can only view months, days and weeks.
4. The maintenance module features batch encoding, and includes add and delete of the following:
    - Rooms
    - Faculty
    - Courses
    - Sections
5. The Archive function can be accessed only by the Admin.
6. The scheduling feature can only be accessed by the department head
7. The reports generation can only be accessed by the department head, and will include the following:
    - Room Schedule
    - Faculty Schedule
    - Section Schedule

Chapter 2: Review of Related Literatures, Studies, and Systems

Computerized schedules work very well only when all participants are committed to keeping their calendars current (*Ellis Davidson, 2018).* The proponents decided to produce a scheduling and reservation app due to the benefits of computerized applications compared to manual, “The pen-and-paper method is the old way of scheduling appointments for a workgroup, in which all scheduling is funneled through an appointments secretary with a central calendar(*Ellis Davidson, 2018)* and due to the nature of the proponents clients environment where technology is all around the clients and will be able to fully utilize the project.

Scheduling Systems plays a vital role in the industry in terms of productivity, “Scheduling systems are designed to help you keep on top of everything you need to get done. Rather than wasting valuable time writing and rewriting lists on a notepad, why not keep track online?” (Kayla Sloan, February 2nd, 2018)

The addition of a navigation bar is of great help to the clients due to the fact that it helps the client “With an ever-present guidepost fixed at the top or side of the screen, a site owner never need worry about his or her user flows being dammed up by confusion or immobility.” (Zack Rutherford, February 27, 2018)

The goal of the system in terms of UI is to reduce the clutter and unneeded functions in the system, thus maximising focus and productivity, as stated by Mr. Babich “A minimalist strategy in web design is one that seeks to simplify interfaces by removing elements and content that do not support user tasks.”(Babich, 2017)

“SuperSaaS”, an online appointment scheduling website, demonstrated the use of a calendar in displaying the events within a day, a week, or a month. The events come with

a description, and the duration of the event. Clicking on an empty slot will generate a popup window that allows users to enter data for their reservation. Our application would be integrated with the Google Calendar, which will then perform similar tasks done by the SuperSaaS calendar, especially making a reservation by clicking a slot on the calendar itself. However, our application will have a verification process per reservation request so as to avoid false reservations. ("Demonstration of a Meeting Room Reservation System", n.d.)

The use of technology also allows students and professors to be more organized. In terms of scheduling and making reservations, technology allows a simpler and faster processing of requests. It also allows students and professors to collaborate with each other to avoid conflicts in schedule. With the use of the Google Calendar, our application will allow users to see the events (including reservations) within a day, a week, or a month upon logging in, which makes checking vacant rooms more effective. (Andrade, 2015)

A simple UI (User Interface) for a website is also a major factor when it comes to measuring UX (User Experience). A straightforward design is preferred over having exaggerated interfaces, as users may have trouble looking for a particular function in the website. A simple interface implies better user experience, mainly when it comes to looking for and using the website's functions. Elements such as margin sizes, typography, and color guides also count. (Darling, 2016)

Our application comes with a verification process in the administrator side. Once a client makes a reservation, the administrator will first need to verify it. From there, the admin will either approve or disapprove the request. It is important to gather and validate

information about the clients making a reservation. This is to avoid having anyone, at any time, to make a false reservation. (Pandey, 2015)

Google Calendar is a web-based time-management and scheduling calendar developed by Google that can be used by teachers to assign events which can be shared to their students. Google Calendar can also be used by teachers to push out reminders to everyone by adding guests to the event. (Calhoon, 2014)

ORO Valley Bicycle, is an event viewing site, on which displays events within a week or month. The events also displays the title and time of the event. Upon clicking an event on the calendar it will show more details about the event which includes the whole duration of the event, the venue, and description of the selected event. (Calendar, n.d.)

Web applications can be accessed through any device which has different screen resolutions. Due to our project being a web application, Bootstrap framework will be used for our web application to have cross-browser capabilities, support for mobile user interfaces, and responsive web design capabilities. (Marah, 2018)

Web applications are applications that is based on web instead of the traditional desktop application. Web applications compared to desktop application are more cost effective since web applications doesn’t need to be developed for every operating system since it is accessed through the web. It can be accessed everywhere; web applications only need an internet connection and a browser in order to access it. It is easily customizable and is accessible for a range of devices, since web applications can be customized easily it can also adjust its contents depending on the device used. (The benefits of web-based applications, n.d.)

The page layout for a website handles how content appears on different webpages. Always check if your web pages are displaying your content the way you designed it. For example, is the navigation bar placed correctly? Are the images in line with the text area? Are the tables displayed in a proper and orderly fashion? Bad webpage layout makes a website look crowded and irritable to the user. It is imperative that your website has good layout so that users can find what they are looking for in your page with ease. Good layout is as important as any other element when designing a website. ("Components of a Website".n.d.)

The website will need to look great on all kinds of platforms and screens, whether you're viewing the page on a laptop, on a desktop, on a tablet, and most especially on smartphones. Design your website in a way that it's optimized for mobile viewing or, as we call it, responsive design. This enables your site to load faster, increased readability, and easy to navigate even on mobile interfaces, which is the way most web users access websites nowadays. (Kelly-Barton, 2018)

When creating a website, the back-end component is just as important as the front end for it to run smoothly. Therefore the proponents have decided to use PHP as the language for the job. A good benefit of using PHP is that it can interact with many different database languages including MySQL. Both PHP and MySQL are compatible with an Apache server which is also free to license. PHP can also run on Windows, Linux and Unix servers. ("Why do we use PHP". n.d.)

A good website must come with a good database management system. It enables the system to have a structured collection of data and store them in separate tables rather than putting all the data in a single storage. This enables the website to access these stored

information at a much faster rate and a more organized and flexible programming environment. With a well-designed database, your website never sees inconsistent, duplicate, out of date, or missing data. (“What is MySQL?”. n.d.)

PHPExcel (PowerKiKi, 2019) allows data to be put into an excel file using PHP. It is used for the scheduling module of the system. Once data has been entered into the “Add Schedule” modal, the PHPExcel handles the data and allocates them into the excel.

One of the most popular libraries for working with Excel spreadsheet in PHP is PHPExcel (Stapler, 2016). It can read, write, and create excel files or PDF files.

Unlike PHPExcel, PHPSpreadsheet can work with various excel file formats such as Open Document Format or OASIS. It is an improvement over the old PHPExcel (Gravelle, n.d.).

Real time computation using JavaScript allowed the detection of the current year in the computer. This helped in the Archive module of the system, where selecting an Academic Year “From” automatically follows with the succeeding year in the Academic Year “To” (Jacobs, 2016).

By using jQuery, the system was able to bind key presses and mouse clicks when selecting the academic years in the Archive module. When selecting an Academic Year “From”, the value in the Academic Year “To” field is automatically captured (Tom, 2011).

By using FullCalendar, recurring events were able to be implemented. It gets the data from the scheduling module, and applies it to the reservation module. It integrates the two functions. (slicedtoad, 2015)

Tooltips allow for a faster and simpler message conveyance. Instead of generating a popup message or a new window, generating tooltips on the spot will instantly alert users. In particular scenarios, such as leaving a required field blank in a Registration window, tooltips are the more preferred function to use. (Tooltips, n.d.)

It is possible to access the website with a mobile browser, but it requires the use of XAMPP. The purpose of this is to test how the website will look like and perform in a different environment. (Dunn, 2016)

The system also had a sorting function for modules that had data in columns. This made looking for a particular data organized. The Archive module of the system featured a dropdown selection in viewing data. It was also modified so that the user will see only a particular set of data based on the sorting view selected in the dropdown. (user3263266, 2014)

Synopsis

No.	Author	Title	Summary
1	Andrade, D,	The Benefits of Educational Technology	The use of technology improves productivity and organization.
2	Babich, N.	Minimalistic Design With Large Impact: Functional Minimalism For Web Design	The use of minimalistic design can lead to maximization of focus and productivity.
3	Calhoon, K. B.	Google Calendar and Gmail--Class Reminders	Google Calendar can help teachers forward events and reminders to students.
4	Darling, S.	UI Toolkit: The Benefits of User Interface Design	A simple user interface is preferred over having an exaggerated design.

5	Davidson, E.	What Is a Computerized Vs. Manual Scheduling System?	A Computerized scheduling system has an upper hand in productivity due to its ability to do task manual labor cannot.
6	Dunn, N	Accessing your Local Web Server from a Mobile Device using XAMPP	It is possible to use a mobile browser to access the website under XAMPP configuration.
7	Gravelle, R.	Making the Switch from PhpExcel to PhpSpreadsheet	PHPSpreadsheet is now user over PHPExcel as it is an improvement over the latter.
8	Jacobs, M.	Real time calculation with PHP & array	Real time calculation helps in capturing academic year for Archive.
9	Kelly-Barton, C.	10 Essential Components of A Successful Website	Responsive web design is important when creating content for users on different platforms and devices.
10	Marah, B. J.	Mastering Bootstrap 4: Second Edition	Bootstrap helps a web application have cross-browser capabilities, support for mobile user interfaces, and responsive web design capabilities.
11	No Author	Components of a Website	Web page layout is important for users to easily navigate the website and look for points of interest.
12	No Author	Demonstration of a Meeting Room Reservation System	Having a virtual calendar will help in viewing events and for making reservations.
13	No Author	ORO Valley Bicycle Calendar	A web-based calendar that helps viewing descriptions of events.
14	No Author	The benefits of web-based applications	Web-based applications has more advantages over desktop applications.
15	No Author	Tooltips	Tooltips are more effective than popup windows to convey messages in particular scenarios.

16	No Author	What is MySQL?	A good database management system is important for a successful website.
17	No Author	Why do we use PHP	It is important to have a means of communicating with the website's database while handling how pages interact with each other.
18	Pandey, R.	Data Verification is an Important Factor for Retaining Customers	It is important to have a customer's data verified before processing his or her requests.
19	PowerKiKi	PHPExcel	It allows data to be processed into an excel file.
20	Rutherford, Z.	Fixed Navigation Bars: Pros and Cons	Fixed navigation bars are used to maximize efficiency and to reduce client confusion.
21	slicedtoad	Simple Repeating Events	FullCalendar is able to have recurring events.
22	Sloan, K.	4 Benefits of Using a Scheduling System in Your Business	Scheduling systems are common in the industry and is used to reduce time constraints.
23	Staper, R.	PHPExcel Tutorial – How to install PHPExcel	PHPExcel is commonly used to create excel or PDF files.
24	Tom	jQuery KeyUp and Click	Allows key presses and mouse clicks to be bound for shortcuts.
25	User3263266	Want to show/hide div based on dropdown box selection	A dropdown selection can be used to display only a particular set of data.

Chapter 3: Technical Background

Functional Requirements

1. The system allowed the user to:
    1.1. Login with the right credentials
    1.2. Register an account
        1.2.1. Password complexity includes the following:
            1.2.1.1. Uppercase character
            1.2.1.2. Lowercase character
            1.2.1.3. Number
    1.3. View current events using Fullcalendar
2. The system allowed the Client and Department Head to:
    2.1. Make reservations
    2.2. Cancel reservations
    2.3. Check reservation status
3. In addition, the system allowed the Department Head to:
    3.1. Create schedules
        3.1.1. The following constraints are considered:
            3.1.1.1. Rooms
            3.1.1.2. Number of minimum and maximum of hours of faculty members
            3.1.1.3. Number of hours
            3.1.1.4. Courses
            3.1.1.5. Sections per semester

3.1.1.6. Special courses and petition classes

3.1.1.7. Request of faculty members

3.2. Generate reports

3.2.1. Reports will include the following:

3.2.1.1. Room Schedule

3.2.1.2. Faculty Schedule

3.2.1.3. Section Schedule

3.3. Single or batch encoding of the following:

3.3.1. Room maintenance

3.3.2. Faculty maintenance

3.3.3. Course maintenance

3.3.4. Section maintenance

4. The system allowed the Admin to:

4.1. Approve reservations

4.2. Disapprove reservations

4.3. Sort reservation requests

4.4. Single or batch encoding of the following:

4.4.1. Room maintenance

4.4.2. Faculty maintenance

4.4.3. Course maintenance

4.4.4. Section maintenance

4.5. Create an archive of the current requests

4.6. Manage user accounts

4.6.1. Features “Access Levels”

4.7. Check reservation statuses

5. The system allowed the user to logout.

Non-functional Requirements

1. Register

Reliability	The system will require all fields to be filled. Leaving at least one field blank will generate an error. If the username has already been taken, the system will notify the user.
Portability	The system has been applied with Bootstrap capabilities in order for the system to be optimized in commonly used browsers (e.g. Google Chrome, Mozilla Firefox, Microsoft Edge).
Security	The system will allow registration only when the entered data in the Password and Confirm Password fields are the same. Also, the password needs to follow a certain criteria. Otherwise, registration will fail.

2. Login

Reliability	The system requires all fields to be filled. When a field has been left blank, the system will notify the user. If the credentials do not match, the system is expected to generate an error.
Portability	The system has been applied with Bootstrap capabilities in order for the system to be optimized in commonly used browsers (e.g. Google Chrome, Mozilla Firefox, Microsoft Edge).
Security	The system will allow the user to login only when the credentials entered match. Also, the characters for the password are encrypted.

3. Reserve

Reliability	Once a reservation request has been made, it will automatically be recorded in the "Requests" window, wherein the clients can see their reservation requests, and their statuses.
Usability	The Calendar used in the system does not conflict with the other elements such as the color palette used for the system.
Portability	The system has been applied with Bootstrap capabilities in order for the system to be optimized in commonly used browsers (e.g. Google Chrome, Mozilla Firefox, Microsoft Edge).
Efficiency	The calendar integrated into the system provides the user with ample information about vacant times. Clicking on a vacant time in the calendar will allow the user to make a reservation request. Also, once the reservation has been approved, it will be reflected into the Calendar.
Security	The system requires the user to enter inputs. If the user does not input data, the system will not proceed with the reservation process. Also, the reservation will only be finalized if the admin approves of the request.

4. Maintenance

Reliability	Once new data have been added, they will be immediately reflected into the system. These data can then be used as basis for the Reservation feature of the system.
Usability	The "Add…" and "Delete…" sections of the Maintenance feature are properly labeled, and do not conflict with the background used.
Portability	The system has been applied with Bootstrap capabilities in order for the system to be optimized in commonly used browsers (e.g. Google Chrome, Mozilla Firefox, Microsoft Edge).
Efficiency	The Add and Delete features of the Maintenance function are put together in a single window, allowing the admin to manipulate data without having to move between windows.
Security	The system requires the admin to enter inputs while adding data. If the admin does not enter anything, the system will not

	proceed with the addition of information.

5. Scheduling

Reliability	The system can identify if the user did not input a value in the fields, if the user did not input a value, the system will not proceed.
Usability	The Room, Courses, Section and Faculty buttons are well arranged and easily identified.
Portability	The system has been applied with Bootstrap capabilities in order for the system to be optimized in commonly used browsers (e.g. Google Chrome, Mozilla Firefox, Microsoft Edge).
Efficiency	The system is efficient because the user will no longer type the value but rather only select a predetermined value that was created by the admin via the maintenance function of the system.

System Requirements

Minimal Requirements

Operating System: Windows 7

Browser: Google Chrome, Microsoft Edge, Mozilla Firefox

Optimal Requirements

Operating System: Windows 10

Browser: Google Chrome, Microsoft Edge, Mozilla Firefox

Methodology

Development Methodology

Waterfall Model

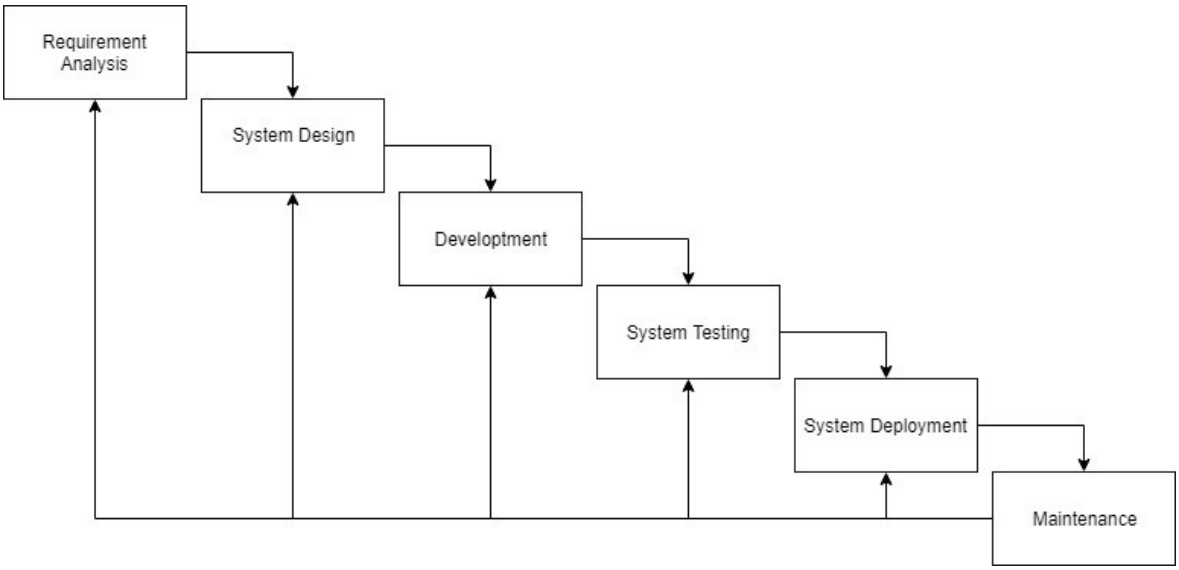


Figure 3.1: Waterfall Lifecycle Model

The figure above exhibits the processes involved in the Waterfall Model. The procedures involved are requirement analysis, system design, implementation, system testing, and system deployment.

In the requirement analysis phase, the proponents gathered all the requirements. Once the requirements have been captured, the team did brainstorming in order to gain more ideas about the project. After brainstorming, the team verified if the requirements were feasible or not.

After the requirement analysis phase, the system design phase came next. In this phase, the team created the design for the system. The hardware and software requirements

of the system were also identified in this phase. The team documented the designs and the requirements in the end.

After the system design phase, the development phase came next. In this phase, the developers started to create the codes. For the following phase, these codes were to be used. The team also conducted a testing for each function in this phase.

After the development phase, the system testing phase came next. In this phase, the functions created were tested in an integration testing, where the functions must work as expected. It is also where the team saw if the system still met the requirements by testing the functional and non-functional activities of the system. The team documented the test results at the end.

After the system testing phase, the system deployment phase came next. Once the environment where the system will be deployed has been set up, the team can deploy the system. Once deployed, the team must once again check each and every function in the system to ensure that nothing had broken.

The last phase of the Waterfall Model is the System Maintenance phase. In this phase, the application must be running in its environment. If the system encounters errors, the team must immediately apply the fixes. Over time, the application will also be enhanced via versioning. The system maintenance is turned over to the IICS Office.

System Design

Activity Diagrams

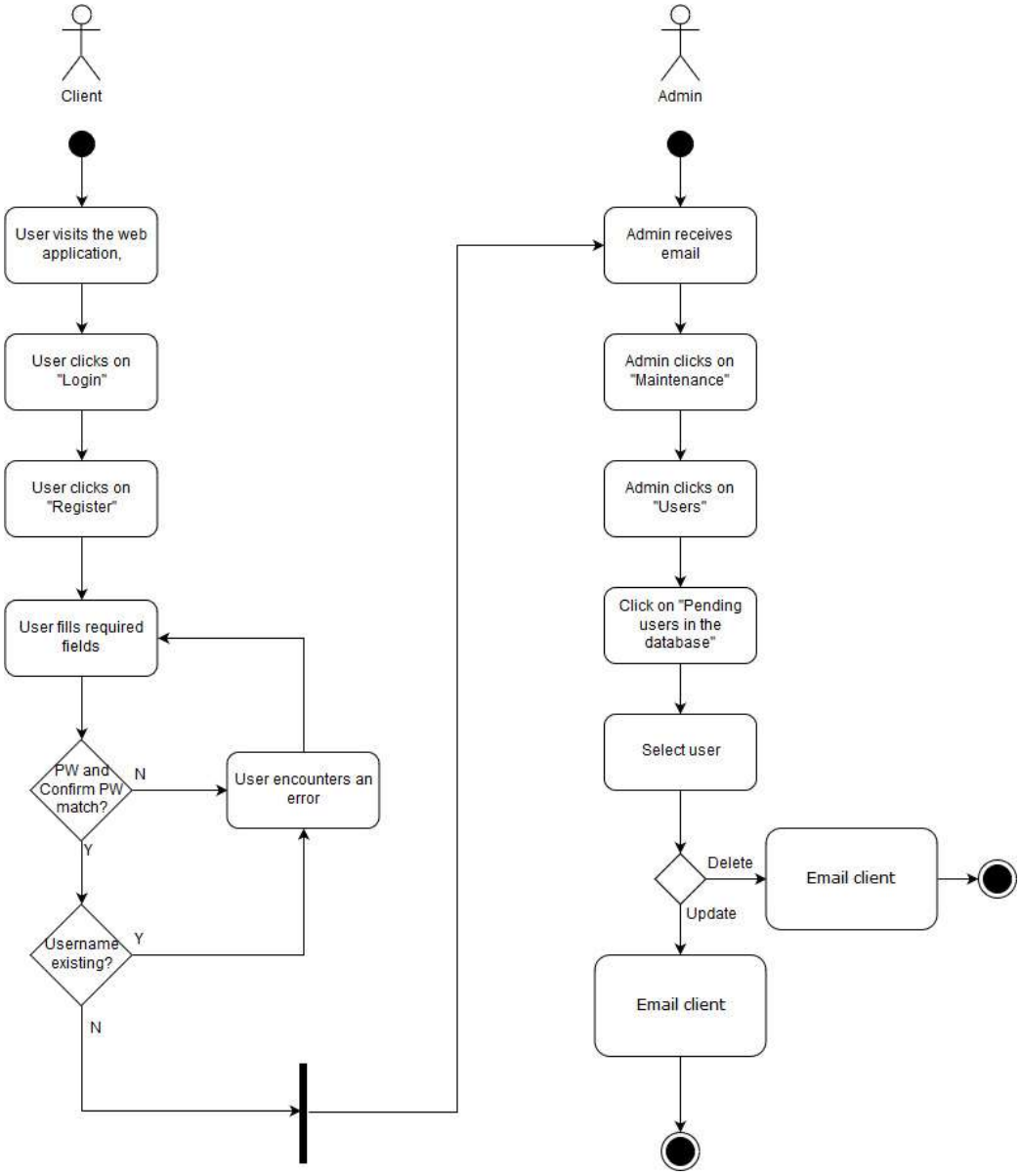


Figure 3.2.1: Creating a user account

The user must first visit the web application. Then, the user must click on the Login function on the upper-right portion of the window. Doing so will make a popup window appear. In this popup window, the “Register” button is just below the “Login” button. The user must click the Register function. The popup window will generate text fields for the Email Address, Username, Password, and Confirm Password. It will also have a selection for the Department, and a checkbox for the Department Head. The password must follow a certain complexity. If the data entered for the password and confirm password fields do not match, the user will encounter an error. Also, if the username entered has already been taken, the user will encounter an error, stating that the user must enter another username. If the data entered for the password and confirm password fields match, and if the username entered is still available, the system will send an email to the admin for the verification request.

On the admin side, if he/she chooses to verify the account, the client will receive an email about the confirmation, and the client will be granted access to the system. If deleted instead, the client will also receive an email containing the reason why their request was removed.

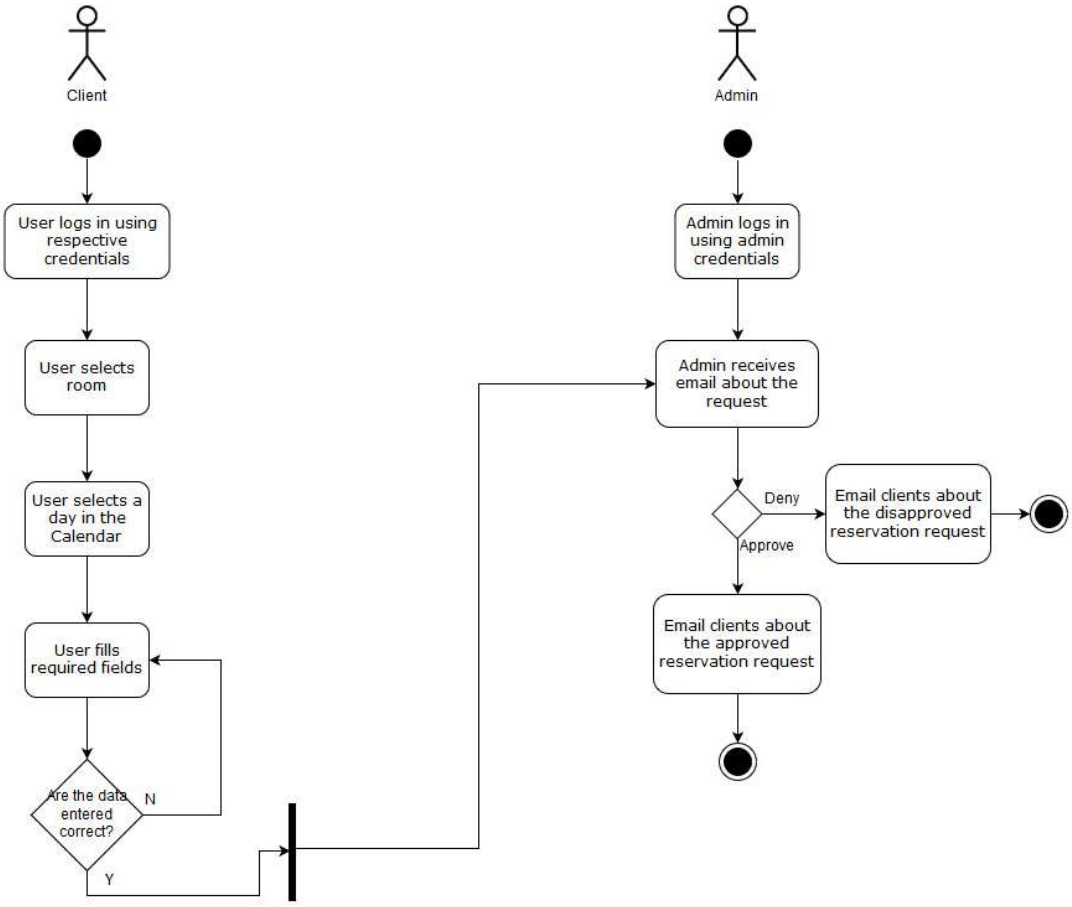


Figure 3.3.2: Reservation

The user will first login using their unique ID (e.g. student number). After logging in, the user must select a room, then a day in the calendar. Then, they will need to fill-up required fields such as time, and title of the request. After entering data, the application will allow the user to verify their inputs. If the user selects “No”, they will be redirected into the input screen. If the data entered is valid, the application will then detect if the desired date and time is vacant. If taken, the application display a message saying that the room is unavailable, after which the user must select a different room. Once the reservation request is done, the administrator will receive an email. The administrator will then approve

or disapprove the reservation request. Clients will then receive an email regarding whether their request has been approved or disapproved.

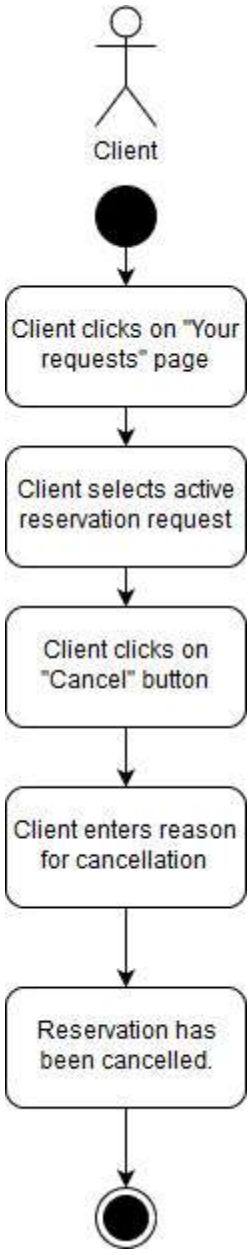


Figure 3.4.3: Cancel Reservation

If the user wants to cancel their reservation, they will first need to login. Then, the user must select the “Your requests” function. The window will display the list of reservations from the user, and their statuses. The user can then select an active reservation request, and click on “Cancel”. If the user clicks on the button, a reason for cancelling the request is required to be made. After stating the reason, it will be sent to the admin as record and the reservation will be cancelled.

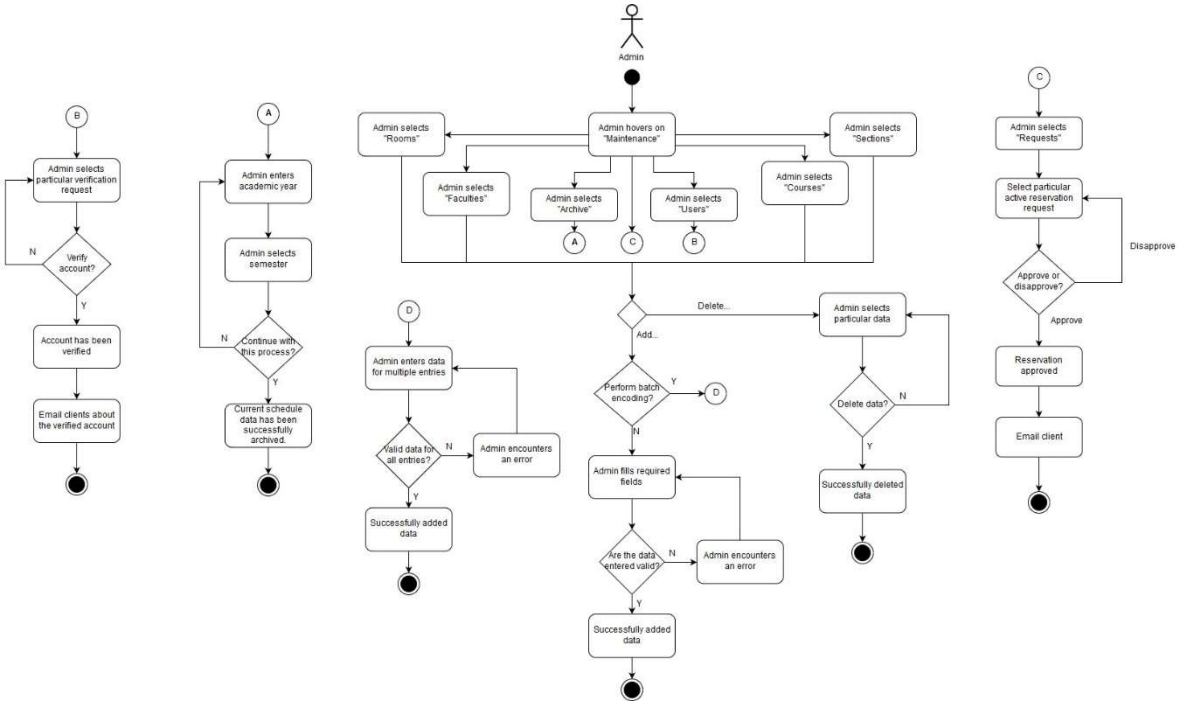


Figure 3.5.4: Maintenance

The administrator first needs to hover the cursor to the Maintenance function. After that, four submenus will appear, namely “Rooms”, “Faculties”, “Courses”, “Sections”, “Archive”, “Users”, and “Requests”. Excluding Archive and Users, the admin then needs to click one of these functions in order to be directed into the Add or Delete window.

In the “Add…” portion, the admin will enter required fields such as room number and time for “Rooms”, employee number for “Faculties”, course code for “Courses”, or name of section for “Sections”. The module also features batch encoding for multiple data entry at once. If there is an invalid data entered, the admin will encounter an error, and must enter data again. If the data is valid, the new room, faculty, course, or section will be created.

In the “Delete...” portion, the admin can select multiple data, and then click the “Delete” button to delete it from the system.

If the admin selects “Archive”, the admin will be taken to the Archive window. In this window, the admin must enter the academic year. After that, the semester must be selected. When the admin finally clicks on “Archive Semester”, a popup window will appear, asking for confirmation from the admin. If the admin clicks on “No”, the popup window will disappear. If the admin clicks on “Yes”, the current semester information will be successfully archived.

If the admin selects “Users”, the admin will be taken to the Manage User Accounts window. In this window, the admin will see the requests for account verification from users. Pending requests will have the access level “newuser” on their requests. The admin can then select these requests and verify it. Once verified, the account will become a “User”, and will be granted access into the system. The client will also receive an email once their request has been accomplished.

If the admin selects “Requests”, the admin will be taken to the Requests Management window. In this window, pending reservation requests from clients will be displayed. The admin can approve a reservation request by choosing “Approve” on the

particular request. Similarly, a request can be disapproved by selecting “Deny”. Once the admin responds to a reservation request, its respective client will be sent an email regarding the update about their request.

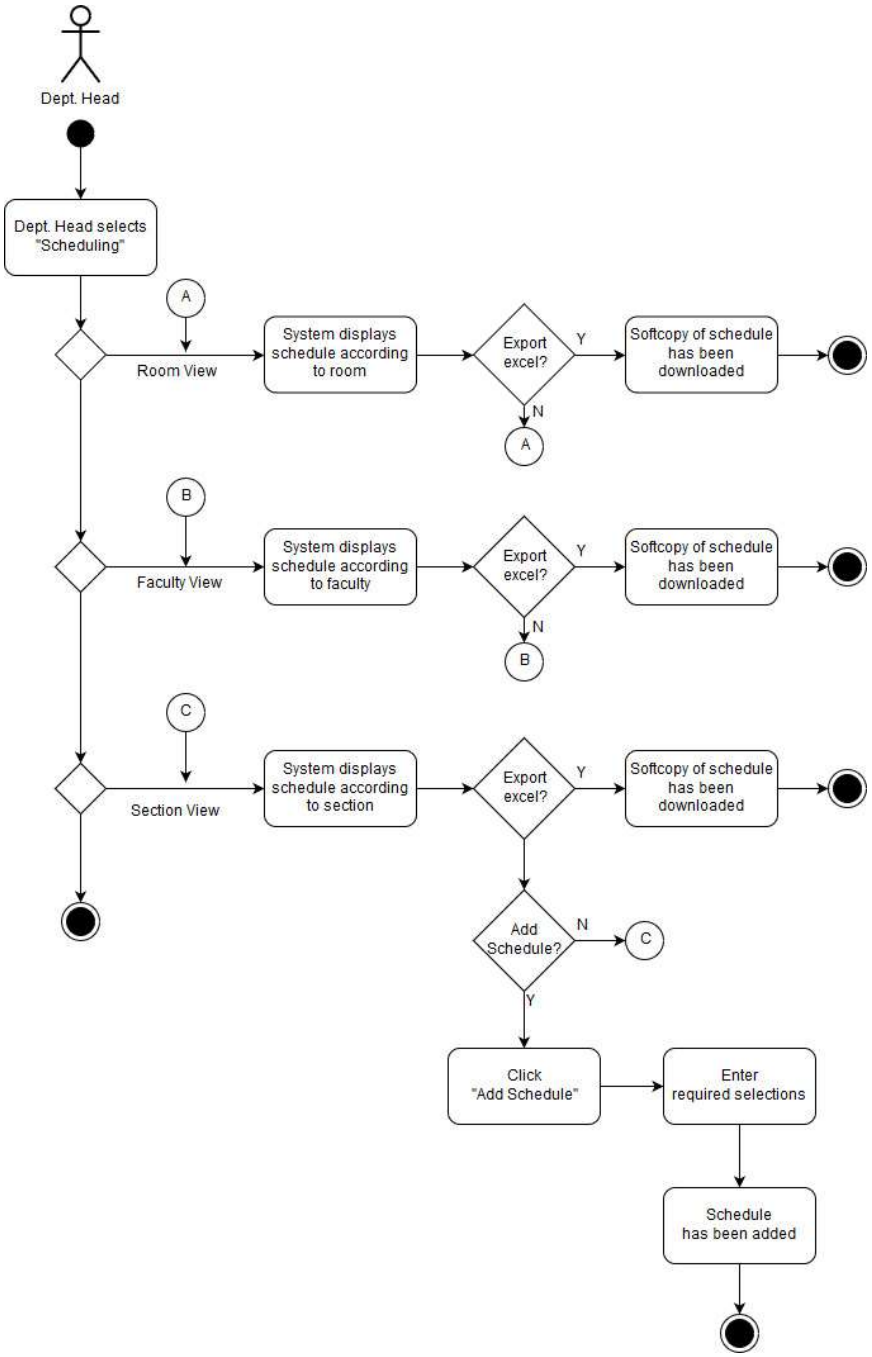


Figure 3.6.5: Scheduling

If the Department Head selects “Scheduling”, they will be taken to the Scheduling Window. First, the system will display three types of views: Room View, Faculty View, and Section View.

If Room View is selected, the system will display schedules according to room number. The department head can only view the schedule in Room View. In the Section View, if a room has been allotted, the schedule will also reflect the room information along with the course and section. Also, the “Export Excel” function can be used to download an excel file of the selected schedule.

If Faculty View is selected, the system will display schedules according to faculty. The department head can only view the schedule in Faculty View. In the Section View, if a faculty member has been assigned to a particular course, the schedule will also reflect the course information, along with the room number. Also, the “Export Excel” function can be used to download an excel file of the selected schedule.

If Section View is selected, the system will display schedules according to sections. The department head can create schedules while in Section View. Once “Add Schedule” is clicked, the head must then enter data for the Course, Room, Faculty, Section, Day, Start Time, and End Time. Once added, the data will be reflected in the schedule. Also, the “Export Excel” function can be used to download an excel file of the selected schedule.

Use Case Diagrams

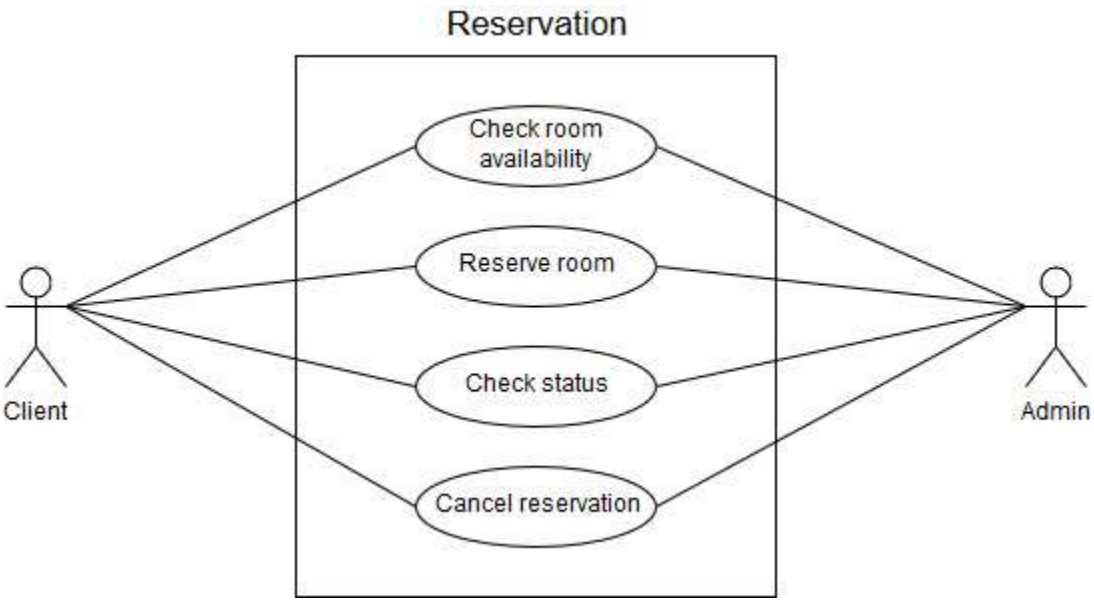


Figure 3.3.1: Reservation Use-Case Diagram

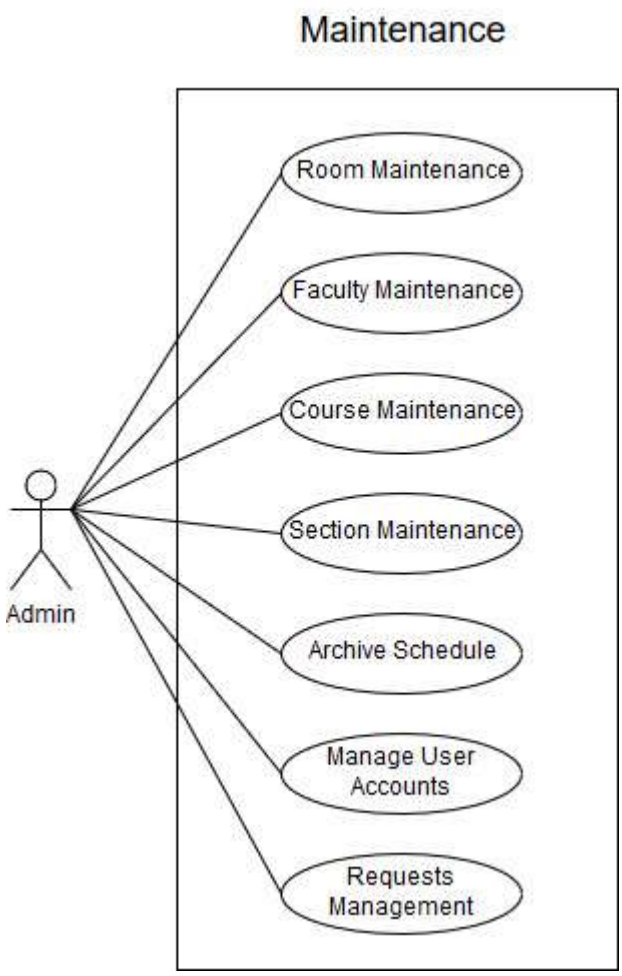


Figure 3.3.2: Maintenance Use-Case Diagram

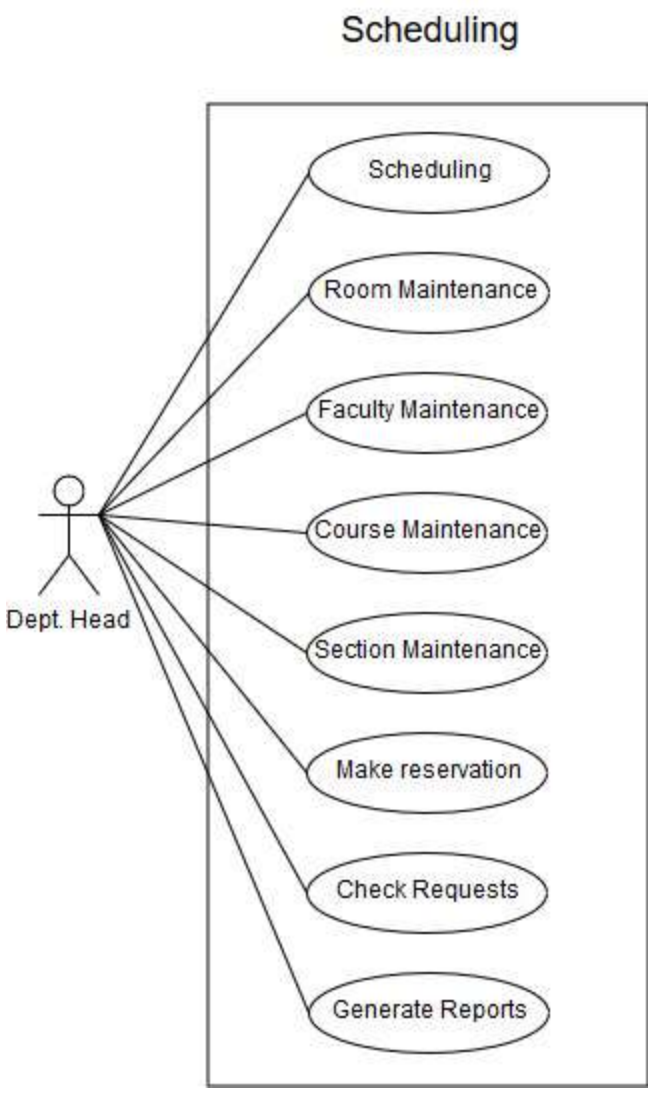


Figure 3.3.3: Scheduling Use-Case Diagram

Development Plan

Register Window

The user must enter their email, username, and password. A criteria for the password must also be followed. The user must then select their department. The user can also click on the checkbox for department head identification. After that, the user must wait for the admin to verify their account.

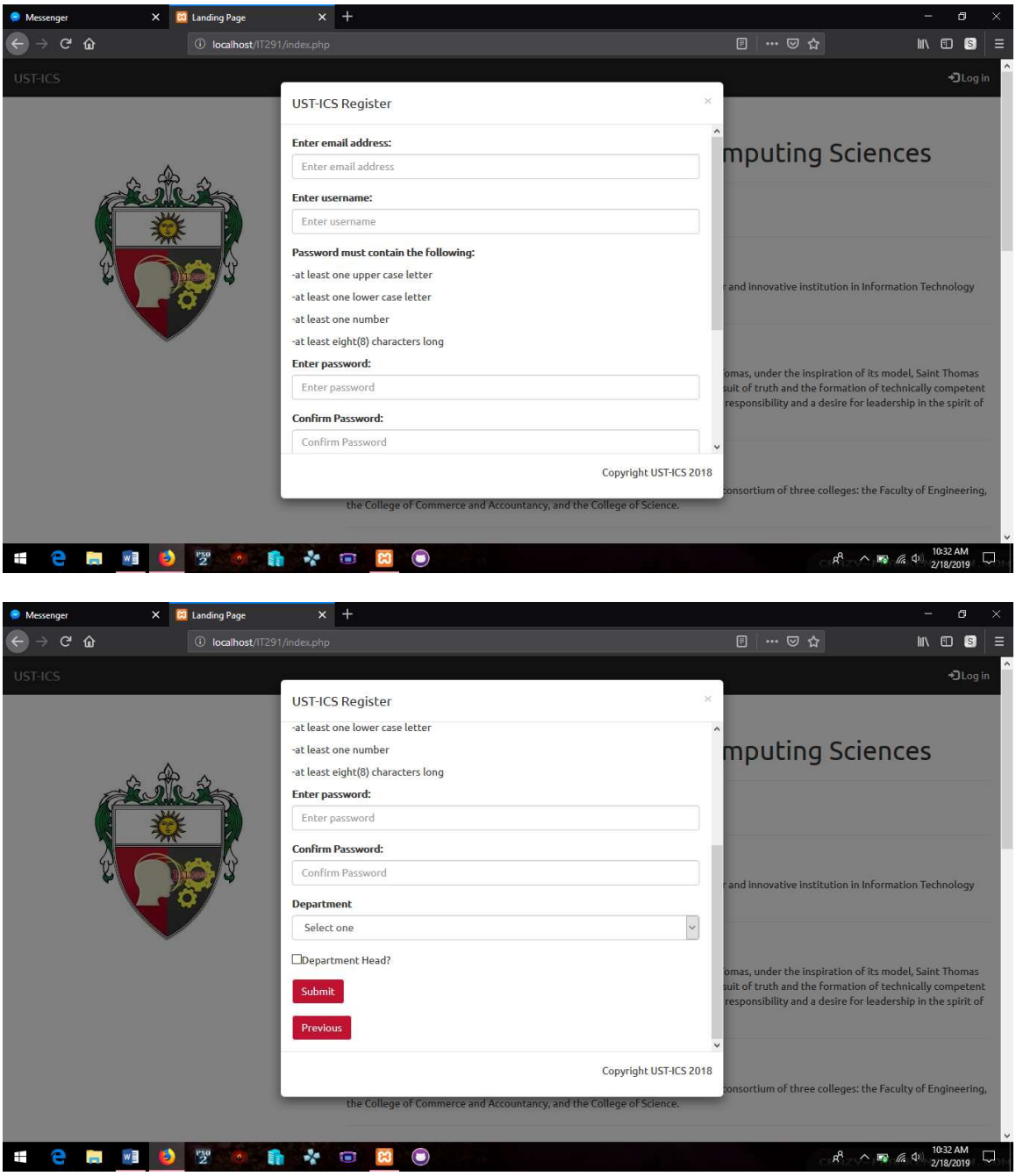


Figure 3.4.1: Register Window

Login Window

Once verified, the user can enter their username and password to access the system.

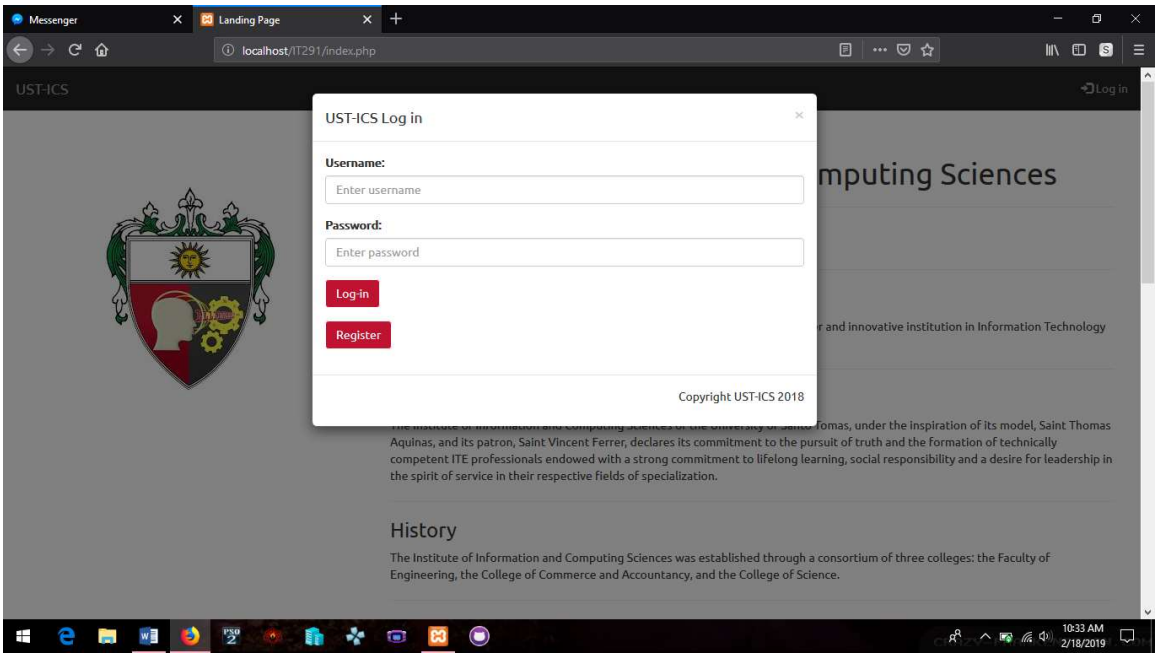


Figure 3.4.2: Login Window

Homepage and Event Viewing

The calendar will allow users to view the current events in a day, week, or month.

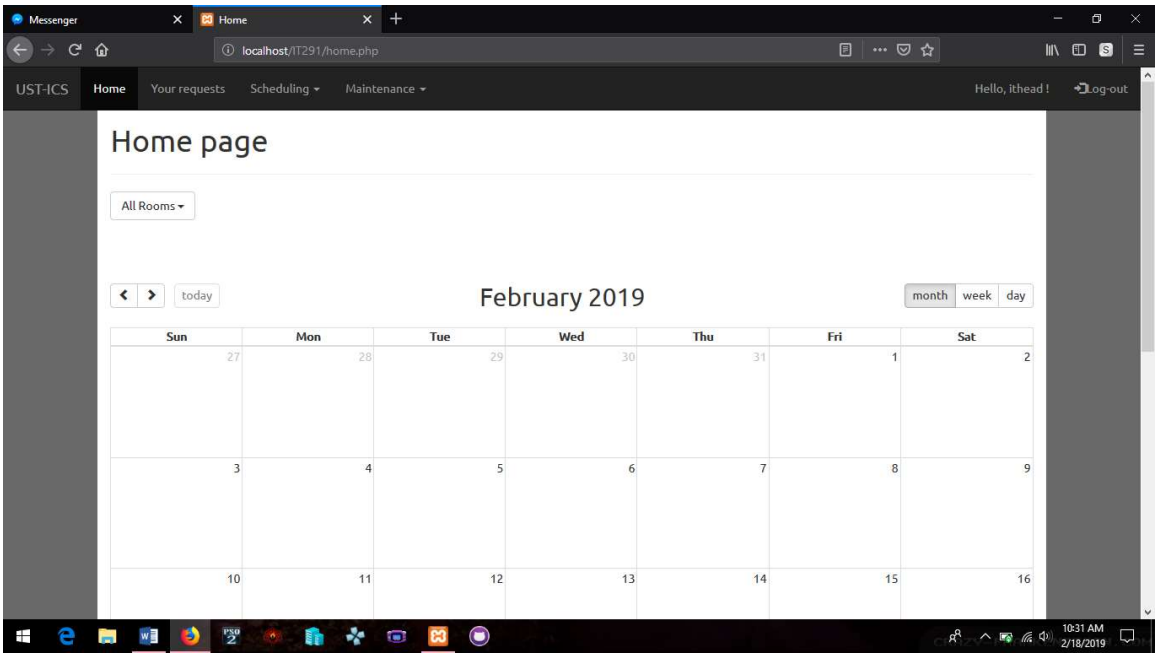
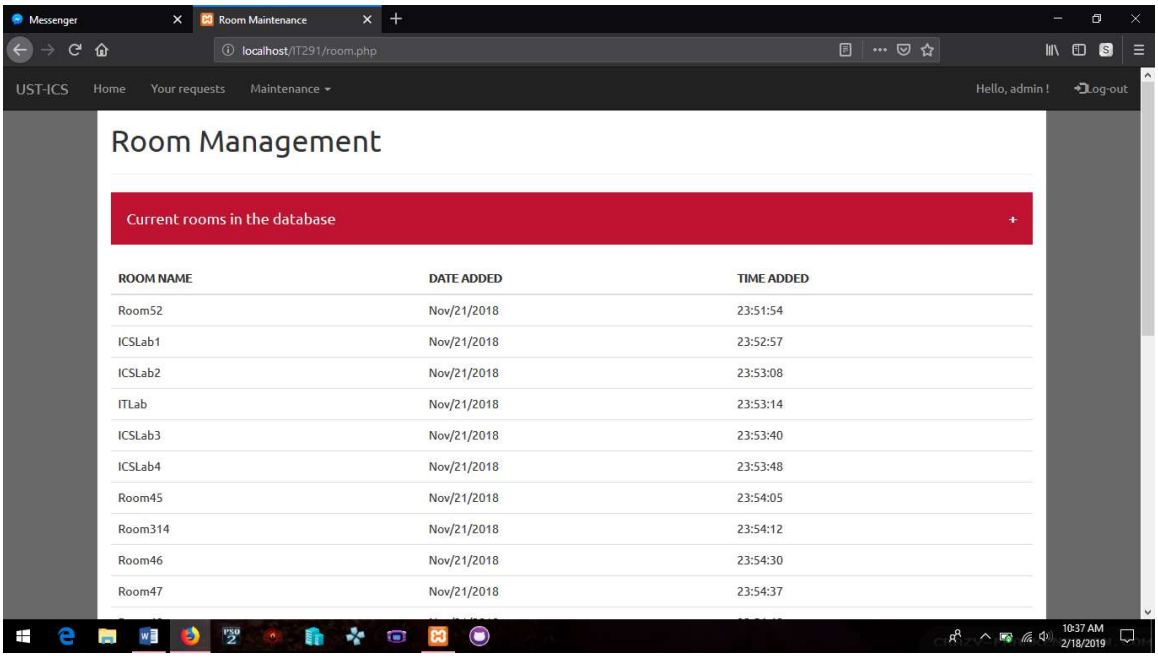
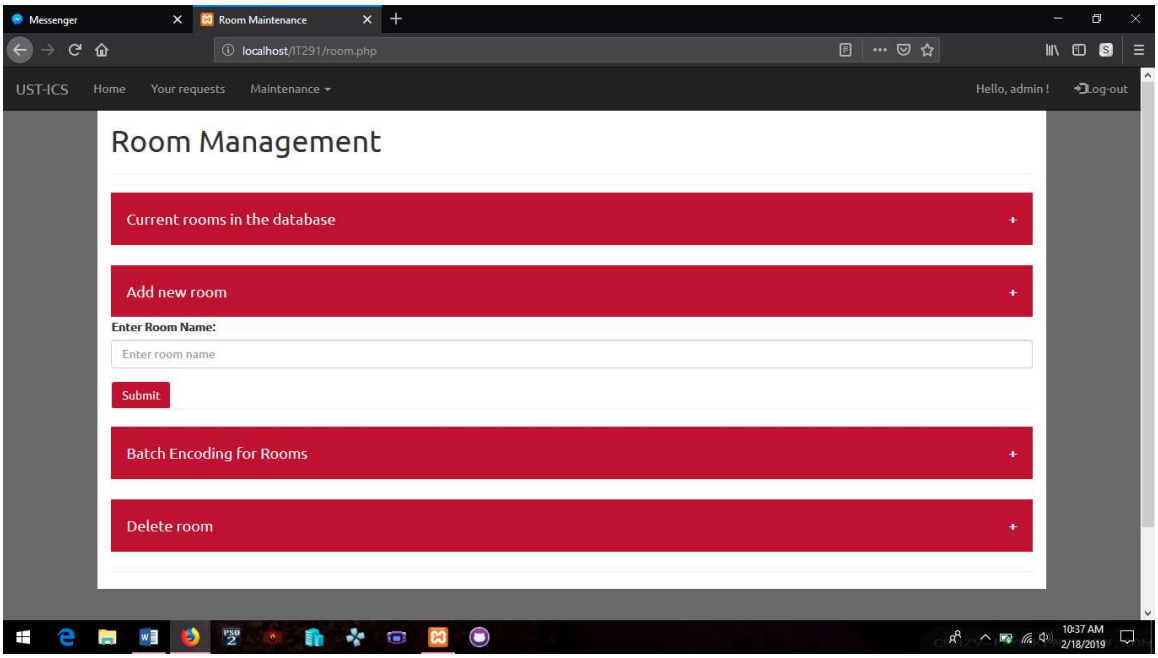


Figure 3.4.3: Homepage

Room Maintenance

The admin must enter the room name, after which it will be displayed immediately in the current rooms. Batch encoding can also be used to enter multiple data at once. The admin can also select multiple rooms for deletion. The admin can also click on the room name to sort the rooms by ascending or descending order


Messenger
Room Maintenance
localhost/IT291/room.php
UST-ICS
Home
Your requests
Maintenance
Hello, admin !
Log-out
Room Management
Current rooms in the database
ROOM NAME
DATE ADDED
TIME ADDED
Room52 Nov/21/2018 23:51:54
ICSLab1 Nov/21/2018 23:52:57
ICSLab2 Nov/21/2018 23:53:08
ITLab Nov/21/2018 23:53:14
ICSLab3 Nov/21/2018 23:53:40
ICSLab4 Nov/21/2018 23:53:48
Room45 Nov/21/2018 23:54:05
Room314 Nov/21/2018 23:54:12
Room46 Nov/21/2018 23:54:30
Room47 Nov/21/2018 23:54:37
10:37 AM
2/18/2019


Messenger
Room Maintenance
localhost/IT291/room.php
UST-ICS
Home
Your requests
Maintenance
Hello, admin !
Log-out
Room Management
Current rooms in the database
Add new room
Enter Room Name:
Enter room name
Submit
Batch Encoding for Rooms
Delete room
10:37 AM
2/18/2019

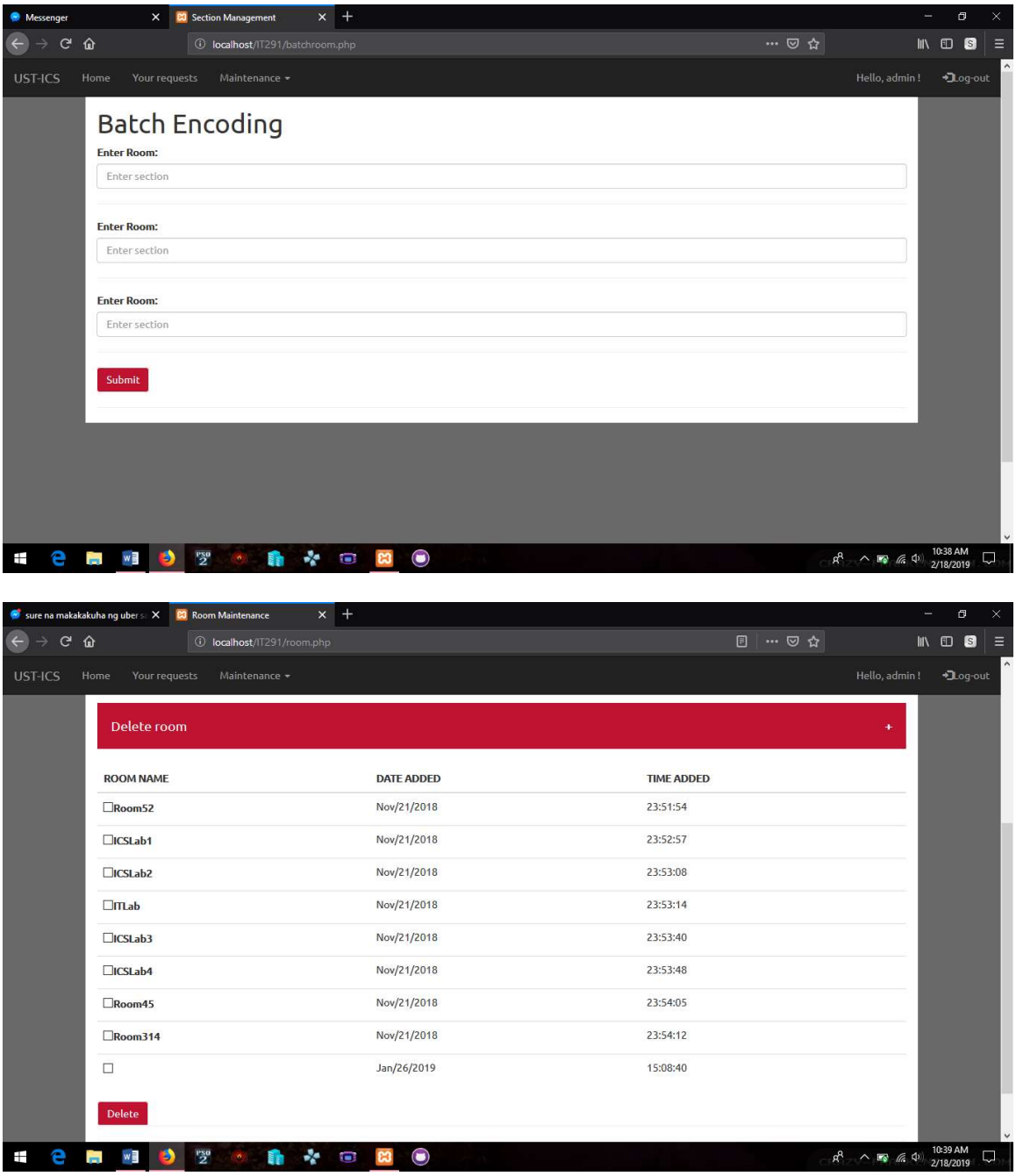
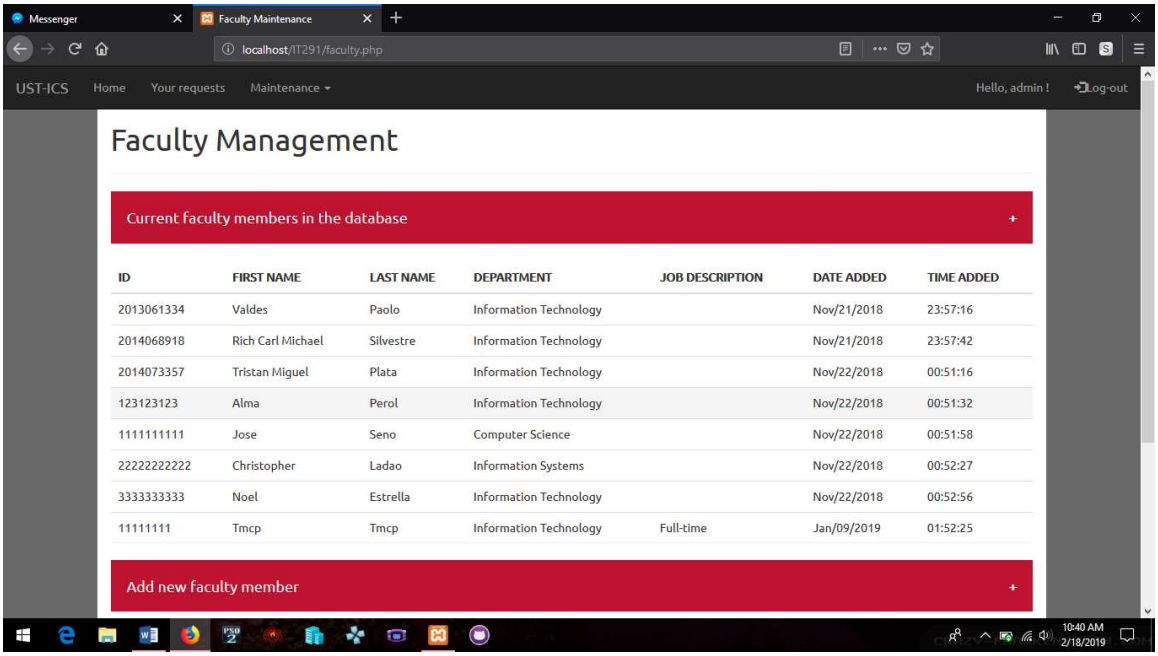
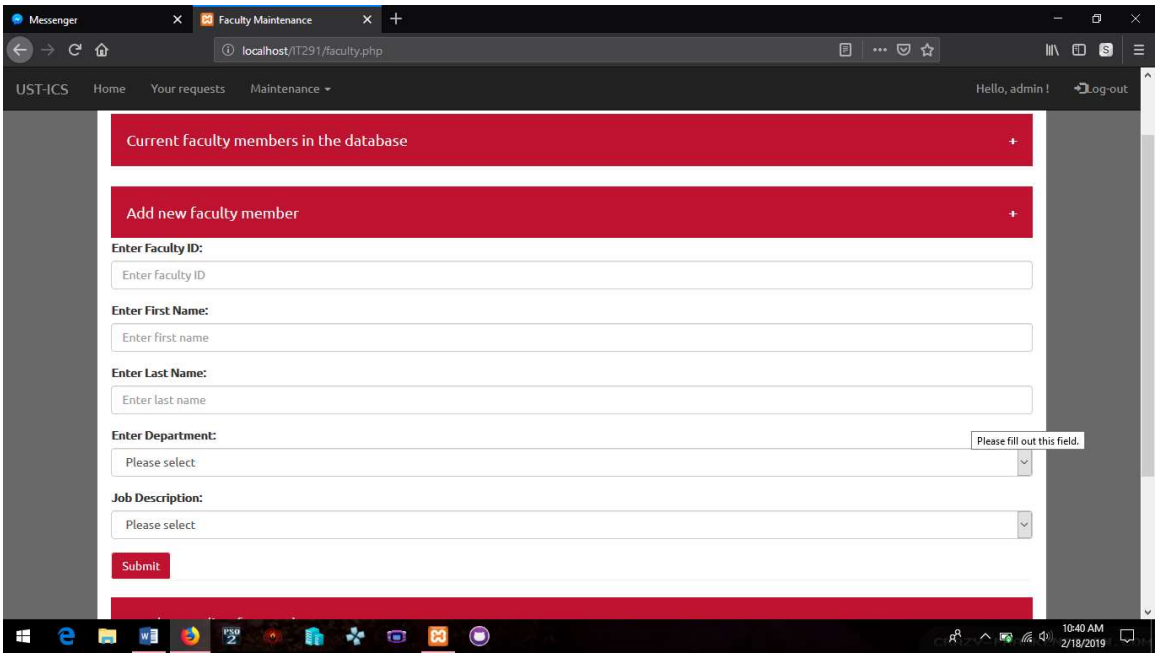


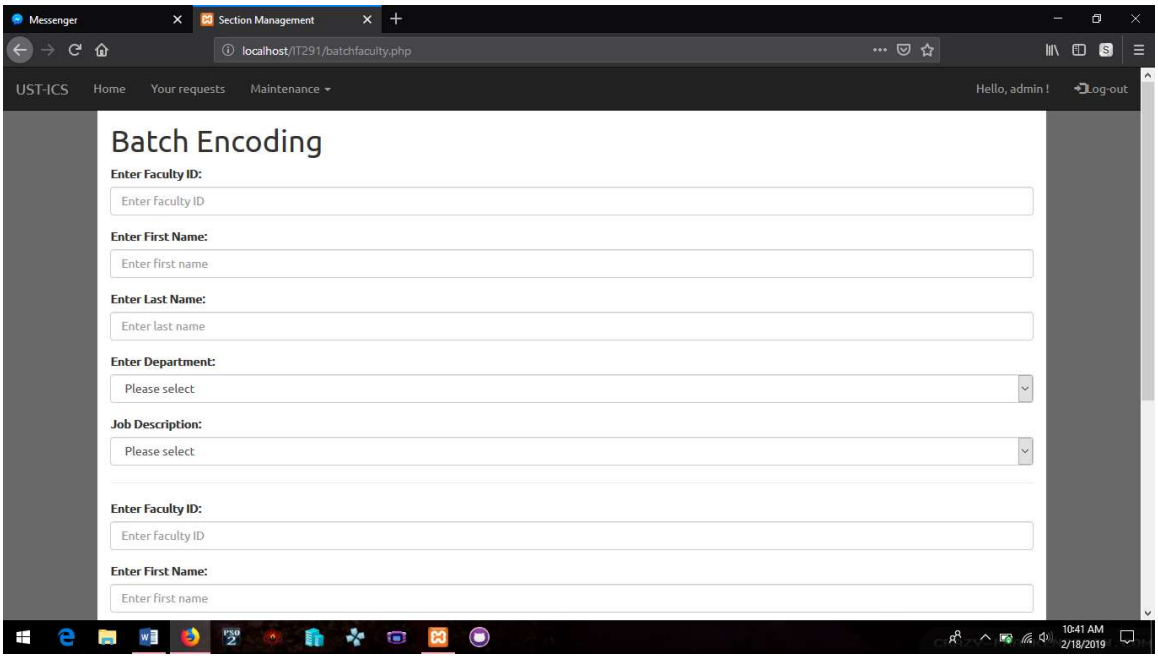
Figure 3.4. 4: Room Maintenance

Faculty Maintenance

The admin must enter the Faculty ID, First Name, Last Name, Department, and Job Description for a particular faculty member. It will be displayed immediately once added. Batch encoding can also be used to enter multiple data at once. The admin can select multiple records for deletion. The admin can also click on the room name to sort the rooms by ascending or descending order.




Messenger
Faculty Maintenance
localhost/IT291/faculty.php
UST-ICS
Home
Your requests
Maintenance
Hello, admin !
Log-out
Current faculty members in the database
Add new faculty member
Enter Faculty ID:
Enter faculty ID
Enter First Name:
Enter first name
Enter Last Name:
Enter last name
Enter Department:
Please fill out this field.
Please select
Job Description:
Please select
Submit
10:40 AM
2/18/2019


Messenger
Section Management
localhost/IT291/batchfaculty.php
UST-ICS
Home
Your requests
Maintenance
Hello, admin !
Log-out
Batch Encoding
Enter Faculty ID:
Enter faculty ID
Enter First Name:
Enter first name
Enter Last Name:
Enter last name
Enter Department:
Please select
Job Description:
Please select
Enter Faculty ID:
Enter faculty ID
Enter First Name:
Enter first name
10:41 AM
2/18/2019

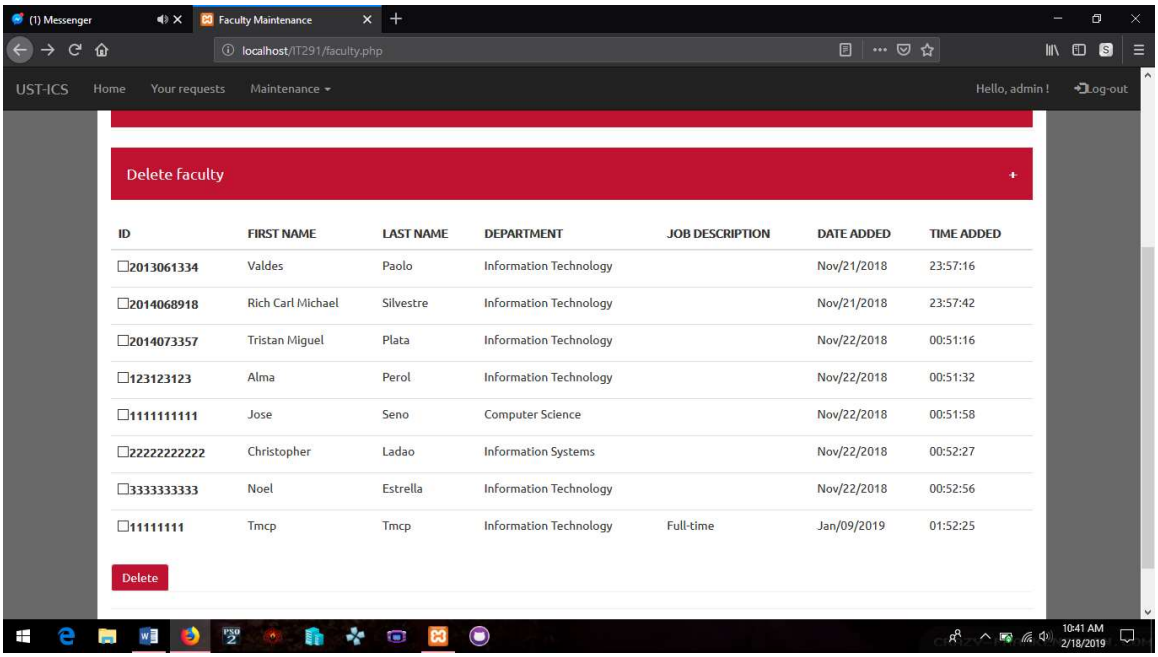
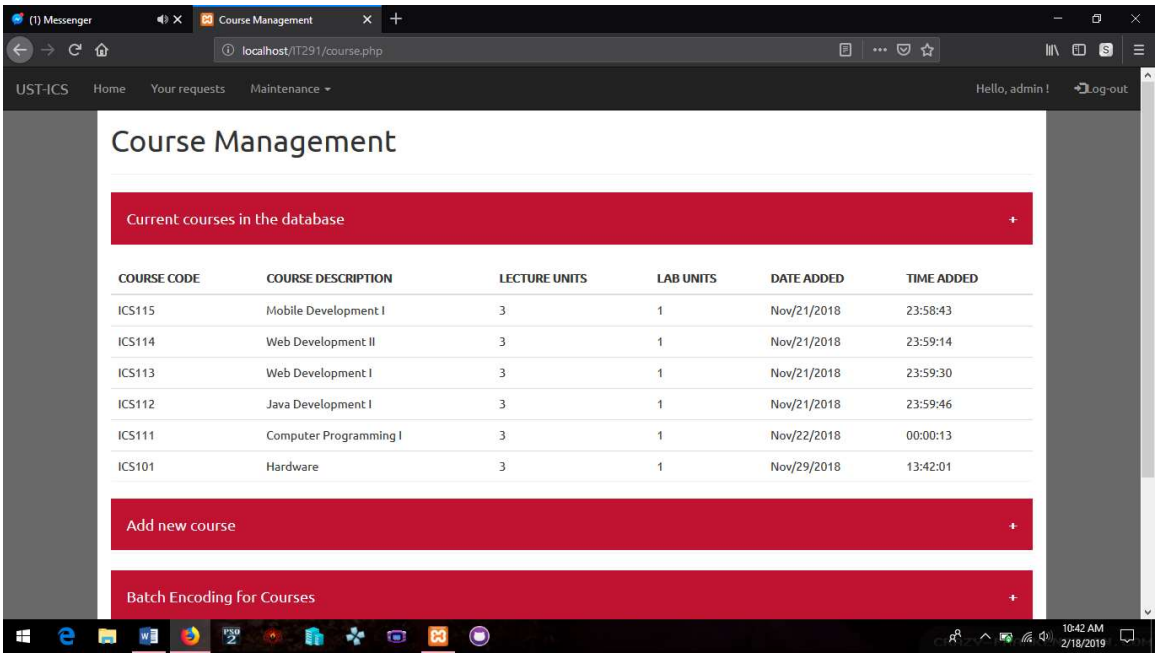
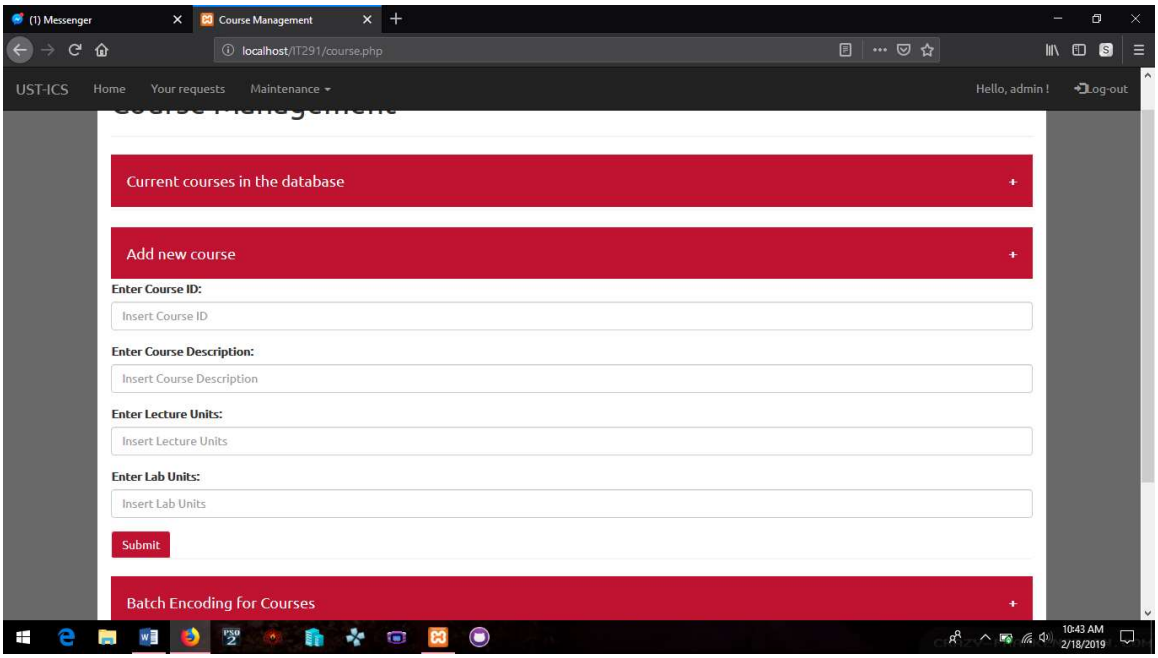


Figure 3.4.5: Faculty Maintenance

Course Maintenance

The admin must enter the Course ID, Course Description, Lecture Units, and Lab Units for a particular course. It will be displayed immediately once added. Batch encoding can also be used to enter multiple data at once. The admin can also select multiple records for deletion. The admin can also click on the room name to sort the rooms by ascending or descending order.


(1) Messenger
Course Management
localhost/IT291/course.php
UST-ICS
Home
Your requests
Maintenance
Hello, admin !
Log-out
Course Management
Current courses in the database
COURSE CODE
COURSE DESCRIPTION
LECTURE UNITS
LAB UNITS
DATE ADDED
TIME ADDED
ICS115 Mobile Development I 3 1 Nov/21/2018 23:58:43
ICS114 Web Development II 3 1 Nov/21/2018 23:59:14
ICS113 Web Development I 3 1 Nov/21/2018 23:59:30
ICS112 Java Development I 3 1 Nov/21/2018 23:59:46
ICS111 Computer Programming I 3 1 Nov/22/2018 00:00:13
ICS101 Hardware 3 1 Nov/29/2018 13:42:01
Add new course
Batch Encoding for Courses
10:42 AM
2/18/2019


(1) Messenger
Course Management
localhost/IT291/course.php
UST-ICS
Home
Your requests
Maintenance
Hello, admin !
Log-out
Current courses in the database
Add new course
Enter Course ID:
Insert Course ID
Enter Course Description:
Insert Course Description
Enter Lecture Units:
Insert Lecture Units
Enter Lab Units:
Insert Lab Units
Submit
Batch Encoding for Courses
10:43 AM
2/18/2019

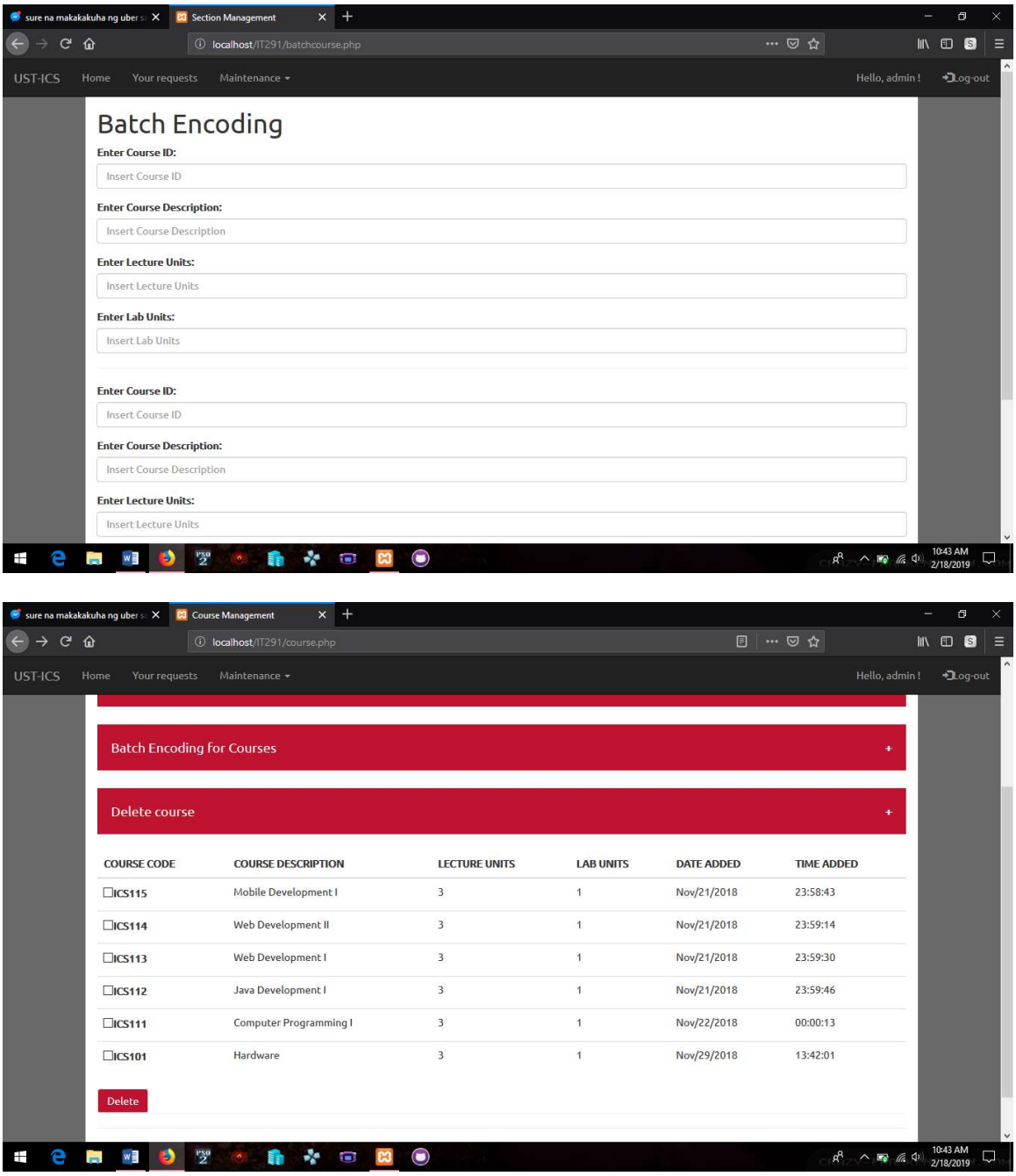
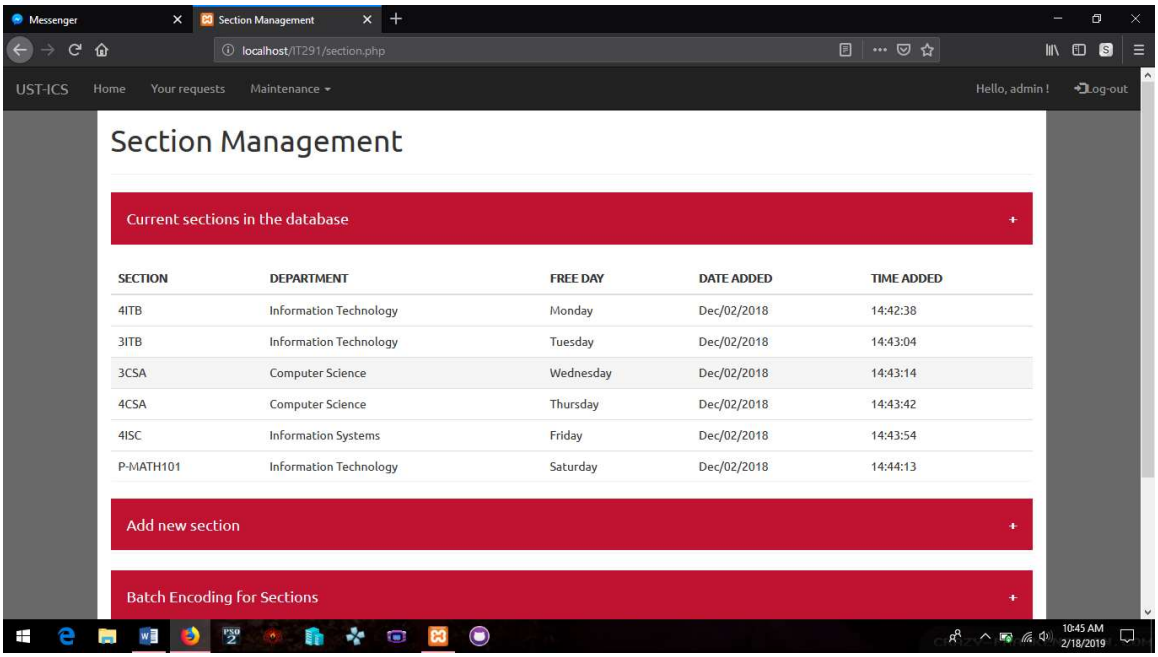
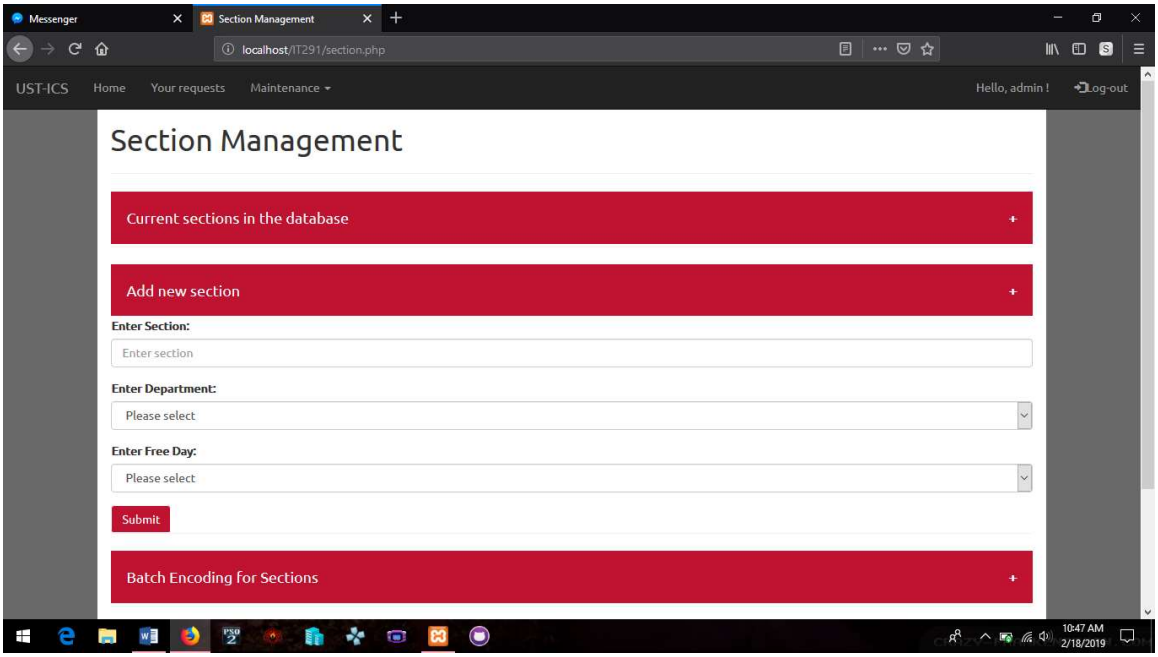


Figure 3.4.6: Course Maintenance

Section Maintenance

The admin must enter the Section, and Department, and Free Day for the particular section. It will be displayed immediately once added. Batch encoding can also be used to enter multiple data at once. The admin can also select multiple records for deletion. The admin can also click on the room name to sort the rooms by ascending or descending order.





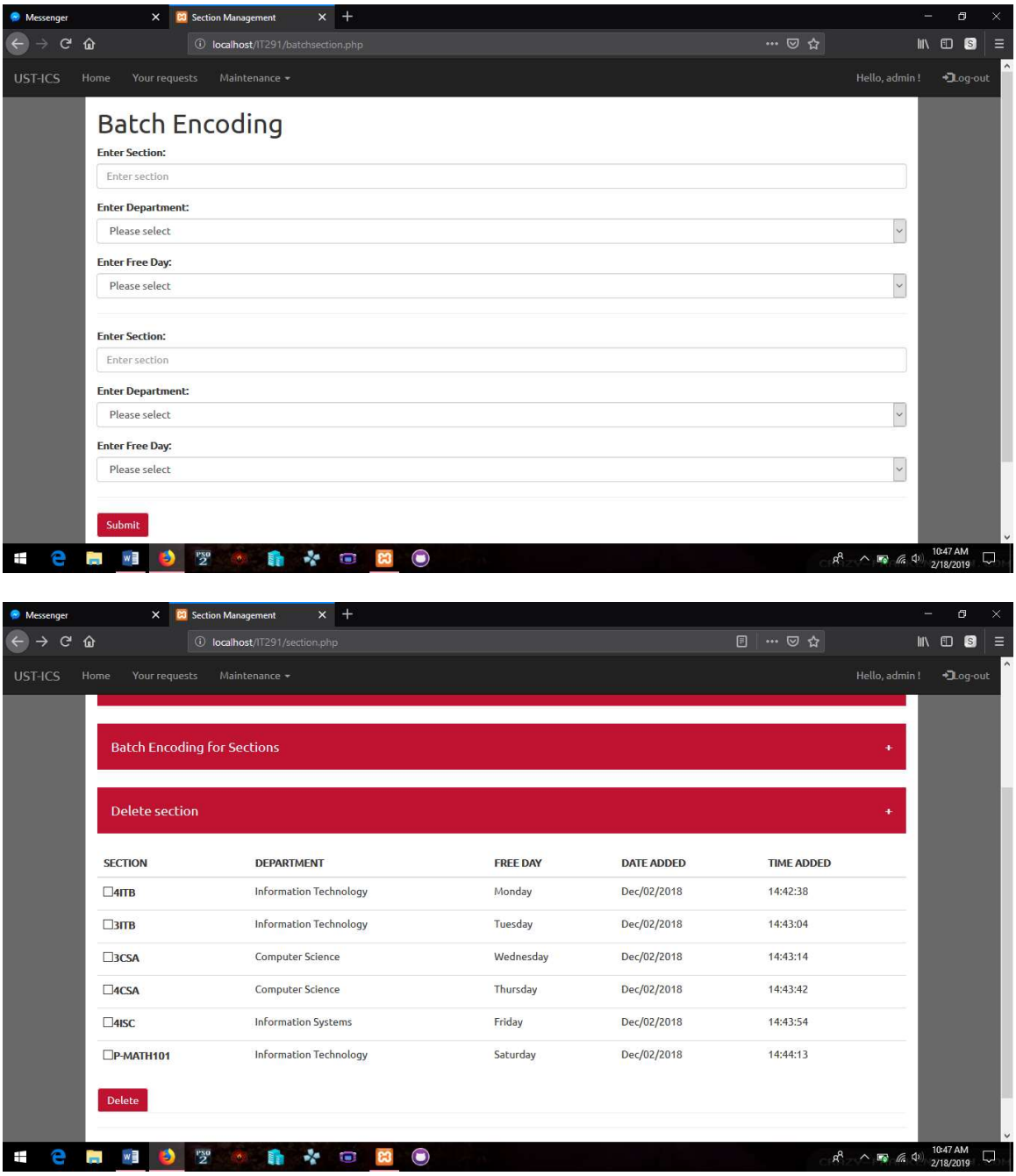
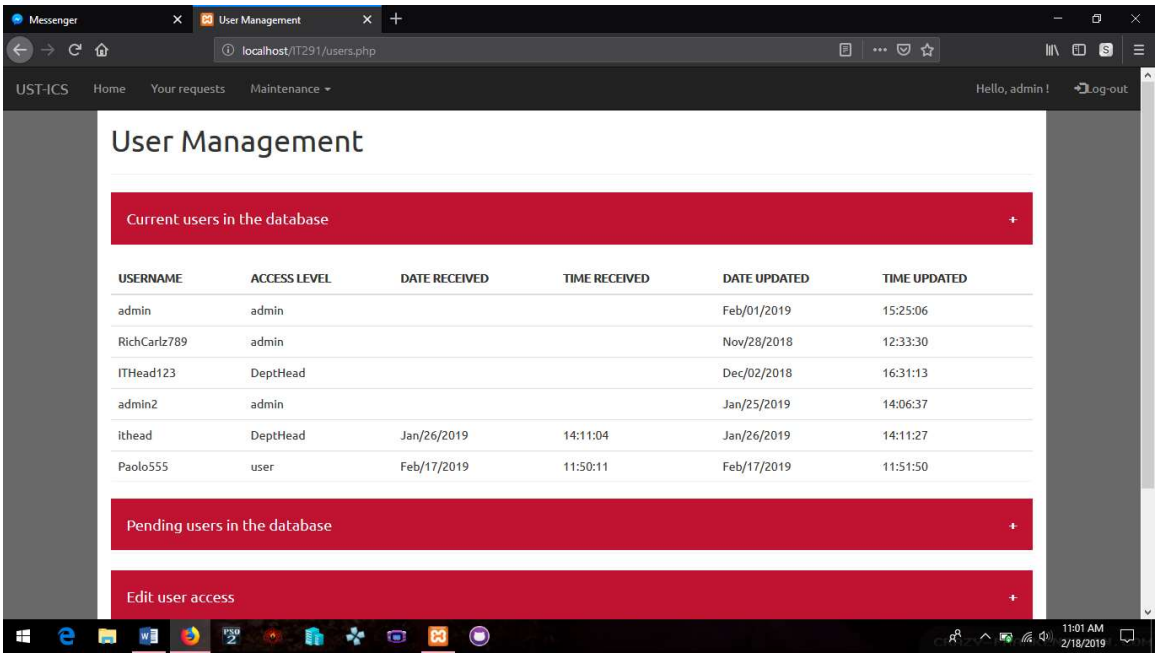
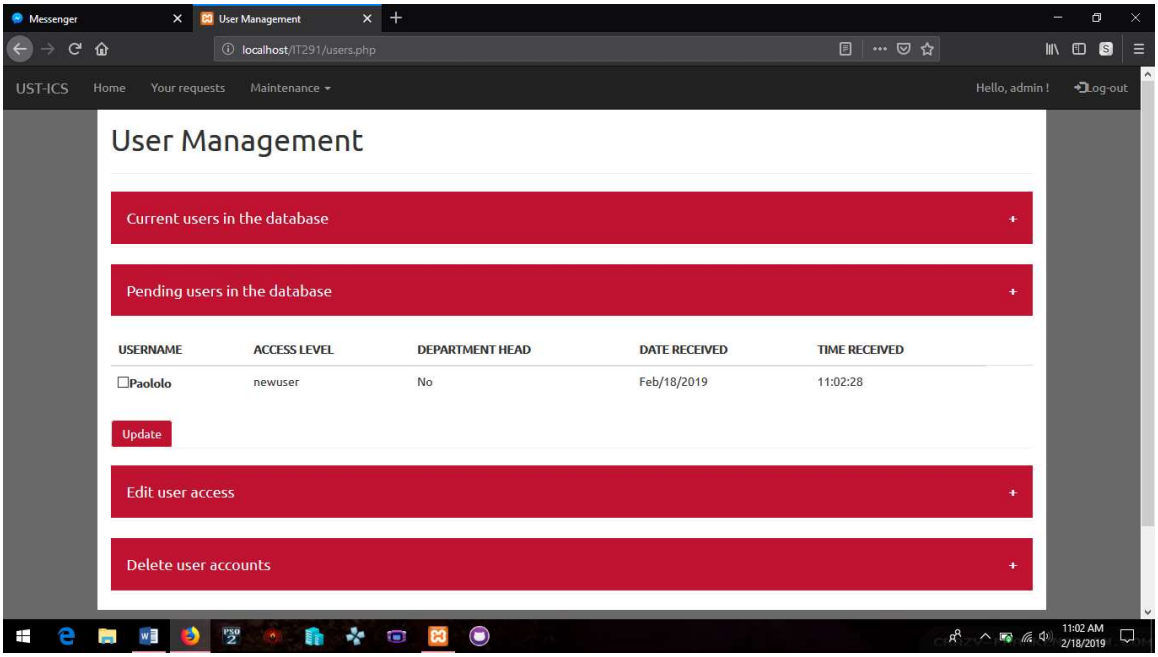


Figure 3.4.7: Section Maintenance

Manage Users

The window will display current users and pending verification requests. The admin can select a “newuser” and verify it to allow the respective user to gain access into the system. The admin can also click on the username to sort the users by ascending or descending order.





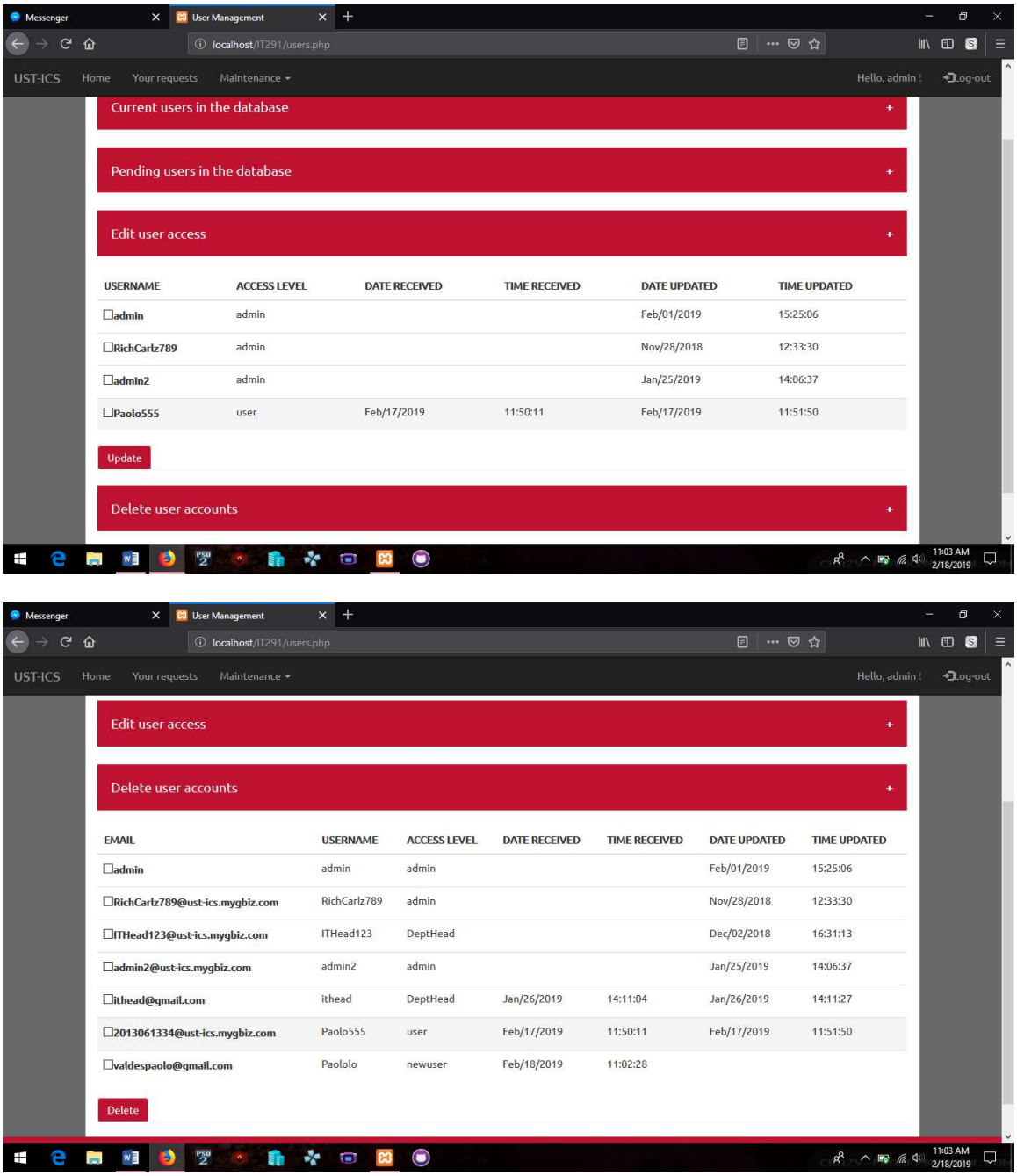


Figure 3.4.8: Manage Users

Reservation

The client must fill out the reservation form after selecting a vacant timeframe in the calendar. Once done, the request will be sent to the admin for approval.

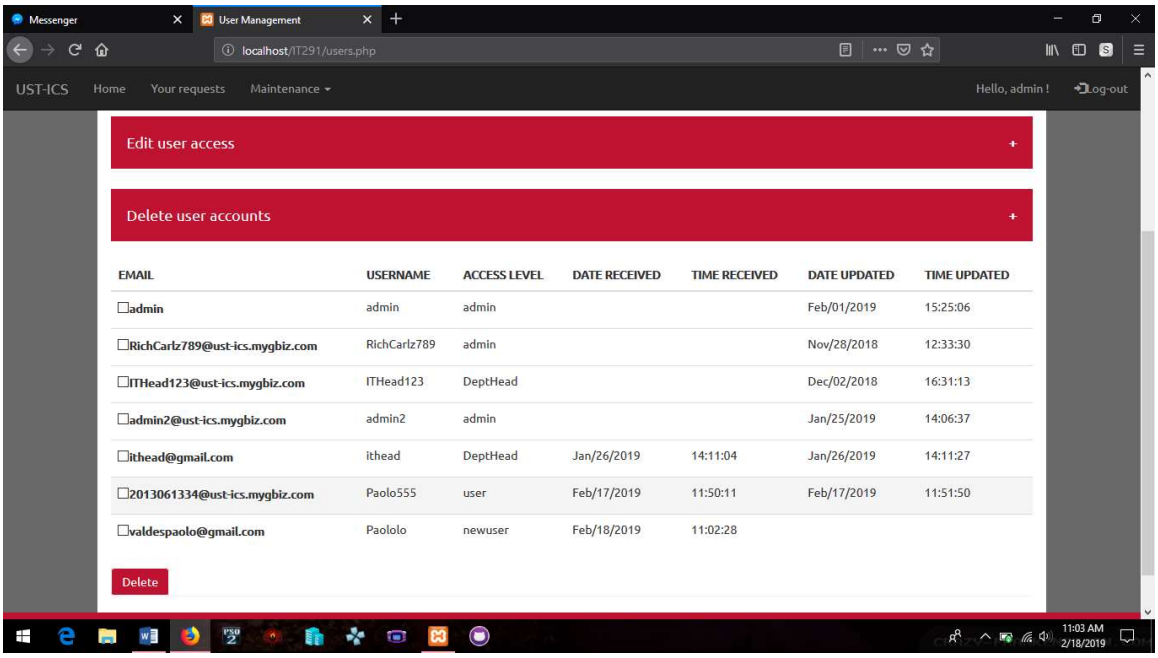


Figure 3.4.9: Reservation

Archive

The admin must enter the academic year, and select the semester. A record of the current schedule will be immediately archived into the system. The admin can also view archived schedules by choosing which academic year and semester.

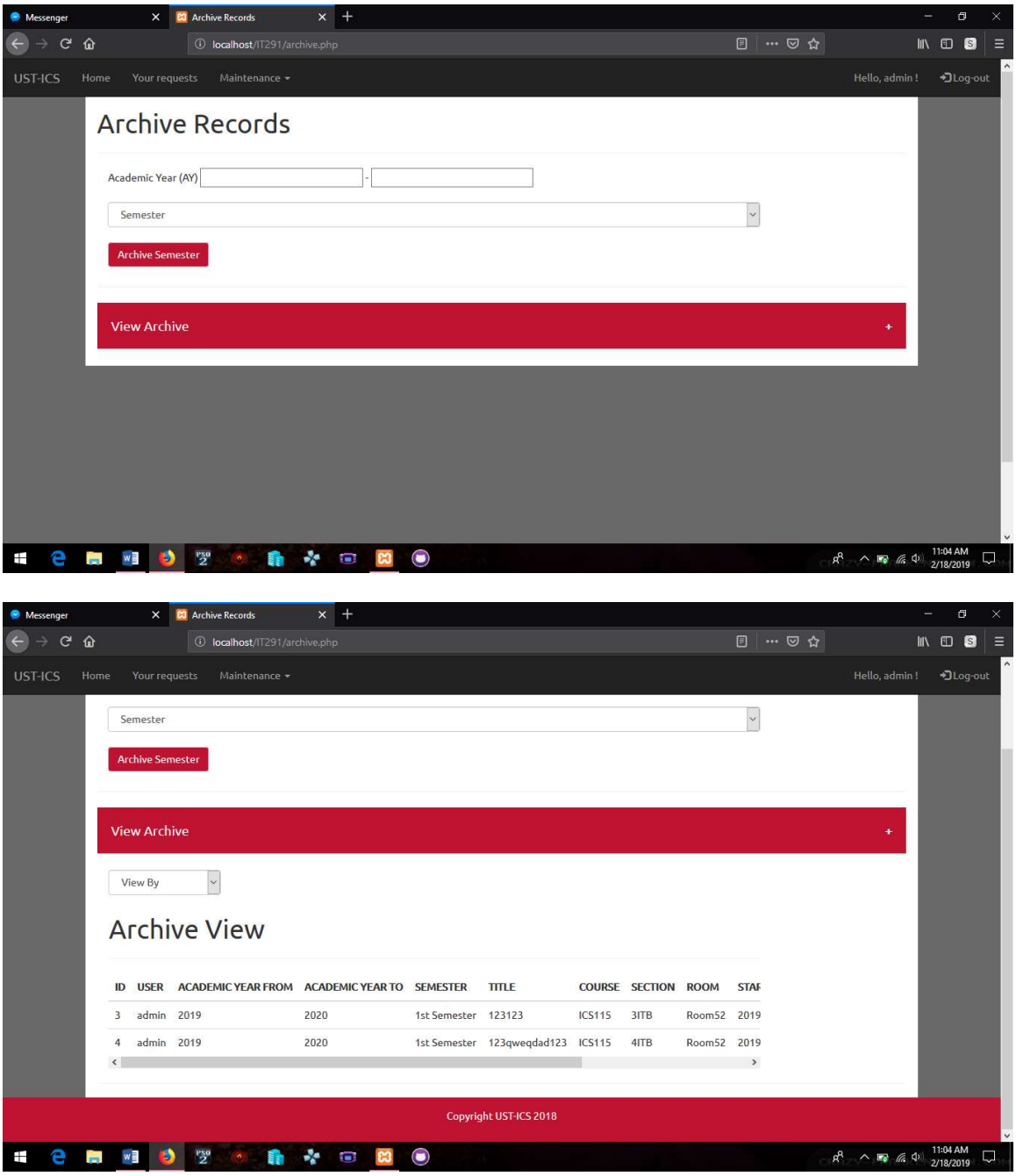


Figure 3.4.10: Archive

Room View for Scheduling

The admin can view the current schedule inline with the rooms, the admin can also add the faculty, course, and sections. The data that can be used in the Room View Scheduling is the same data that the admin made in the maintenance function of the system.

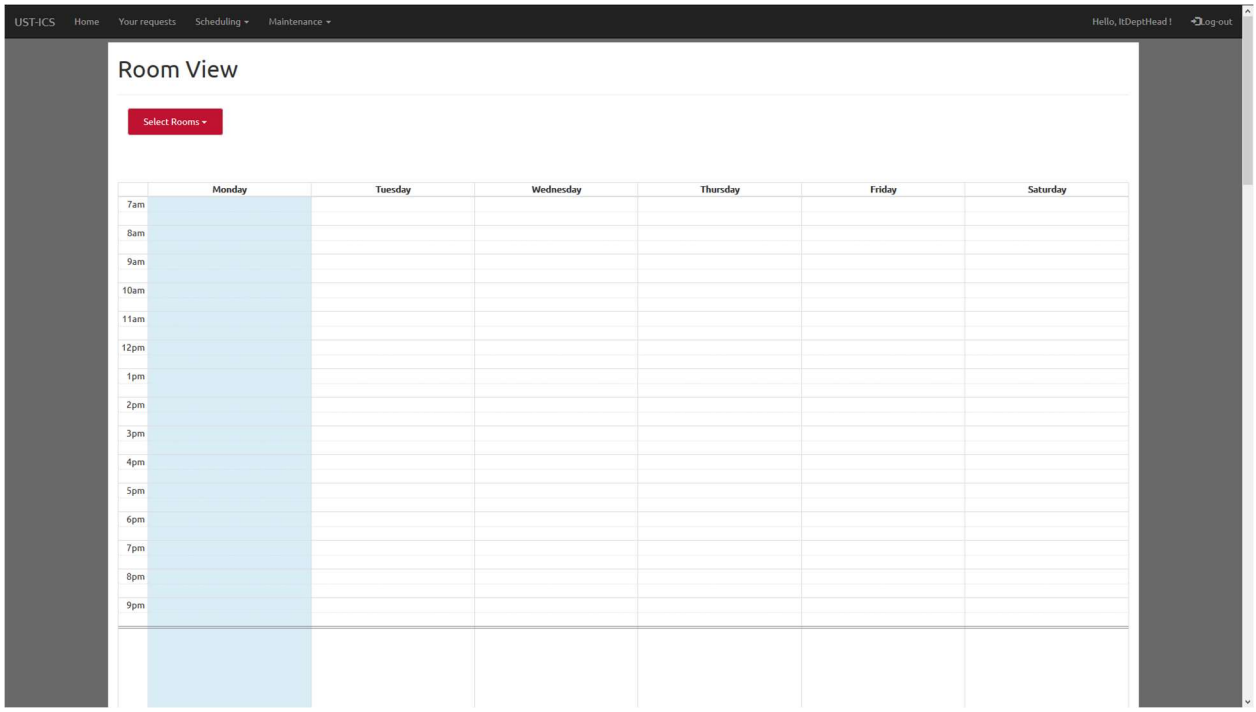


Figure 3.4.11: Room View for Scheduling

Faculty View for Scheduling

The admin can view the current schedule inline with the faculty, the admin can also add the room, course, and sections. The data that can be used in the Faculty View Scheduling is the same data that the admin made in the maintenance function of the system.

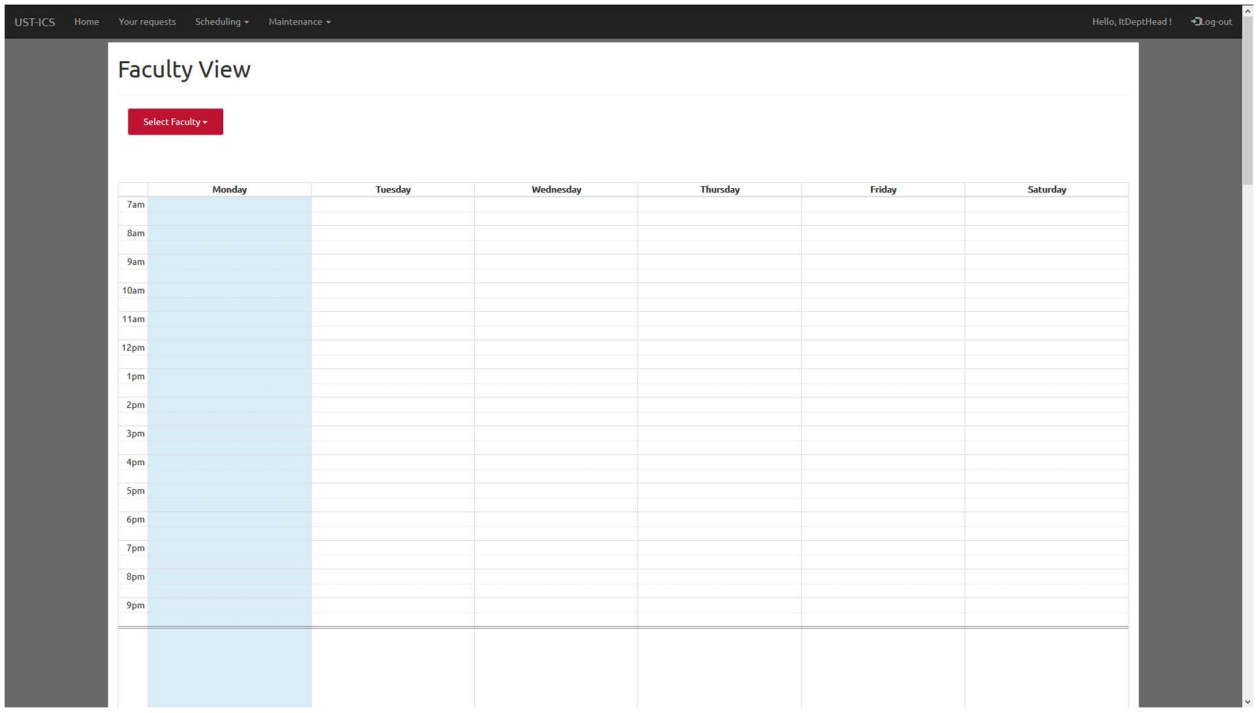


Figure 3.4.12: Faculty View for Scheduling

Section View for Scheduling

The admin can view the current schedule inline with the section, the admin can also add the room, course, and faculty. The data that can be used in the Section View Scheduling is the same data that the admin made in the maintenance function of the system.

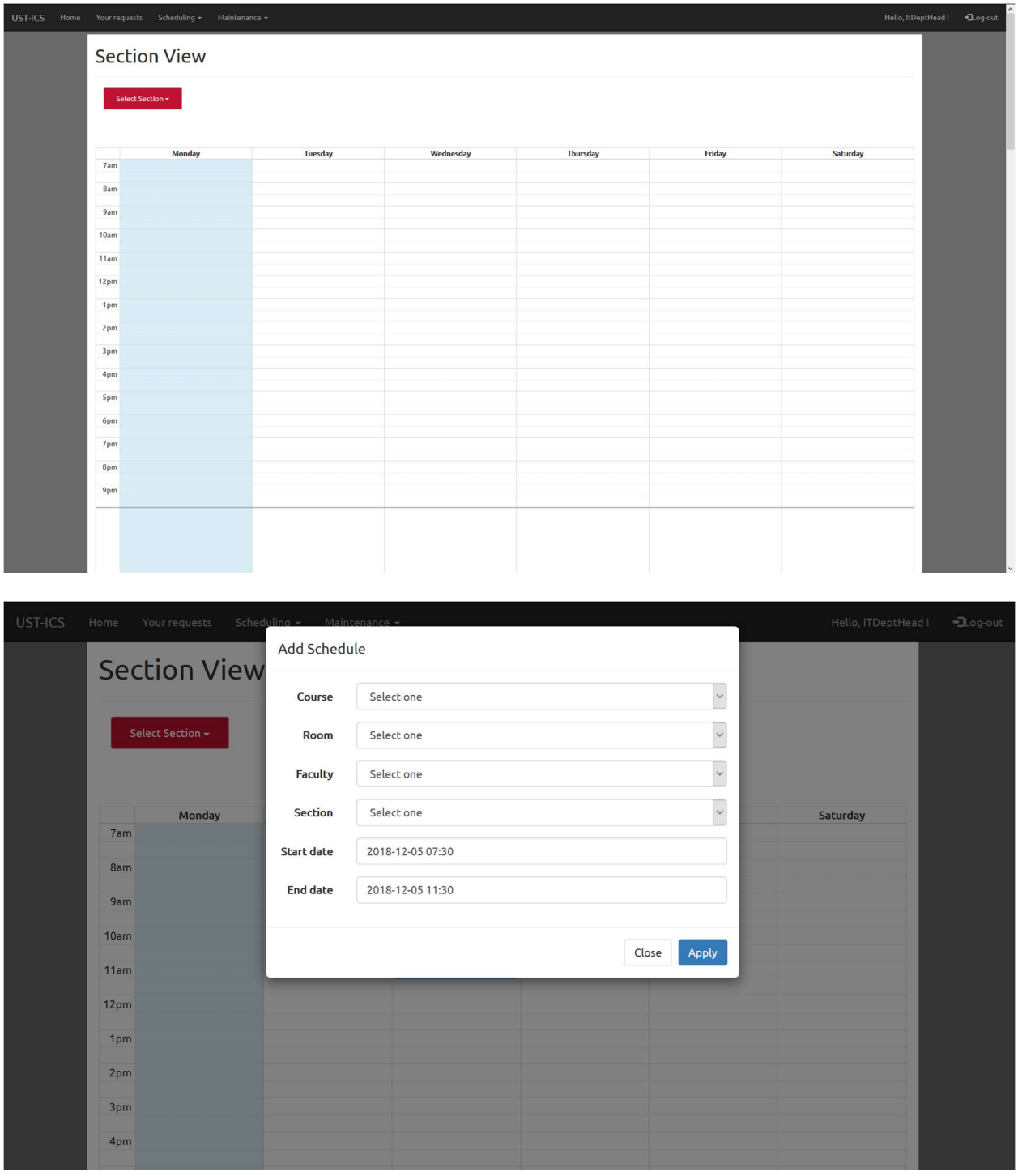
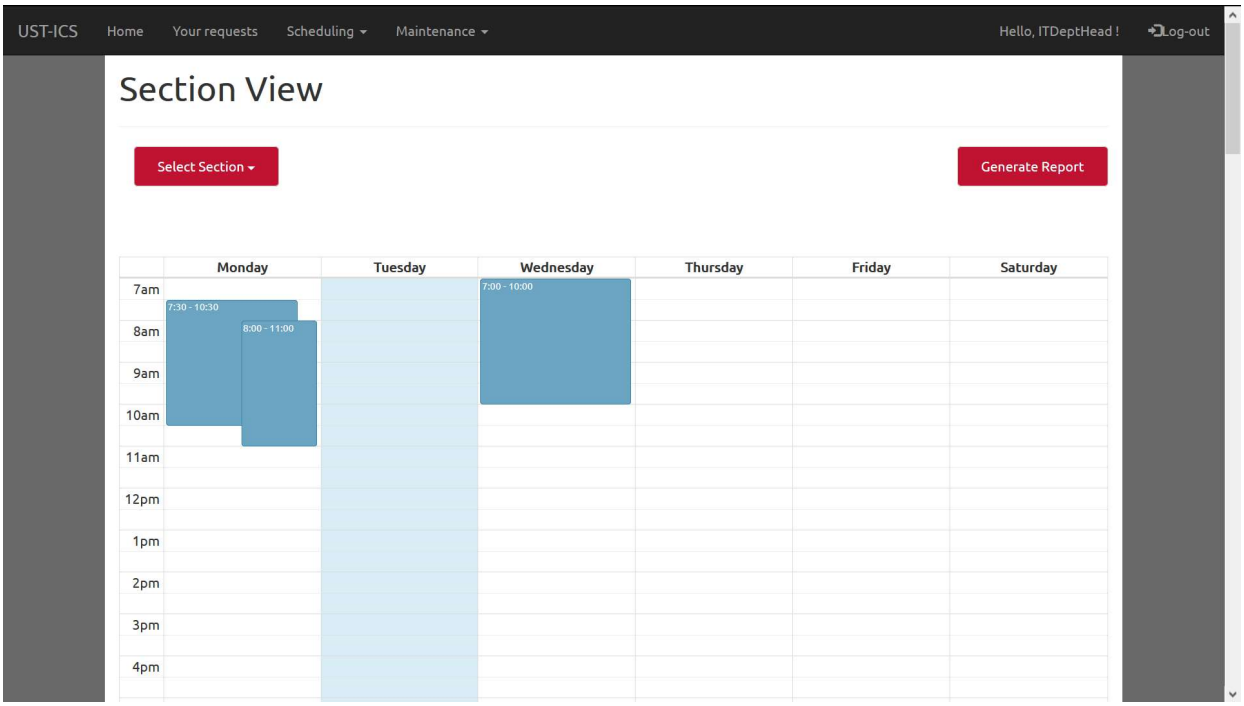


Figure 3.4. 13: Section View for Scheduling

Reports

The department head must select the “Generate Report” button in order to create a printable copy of the desired schedule.



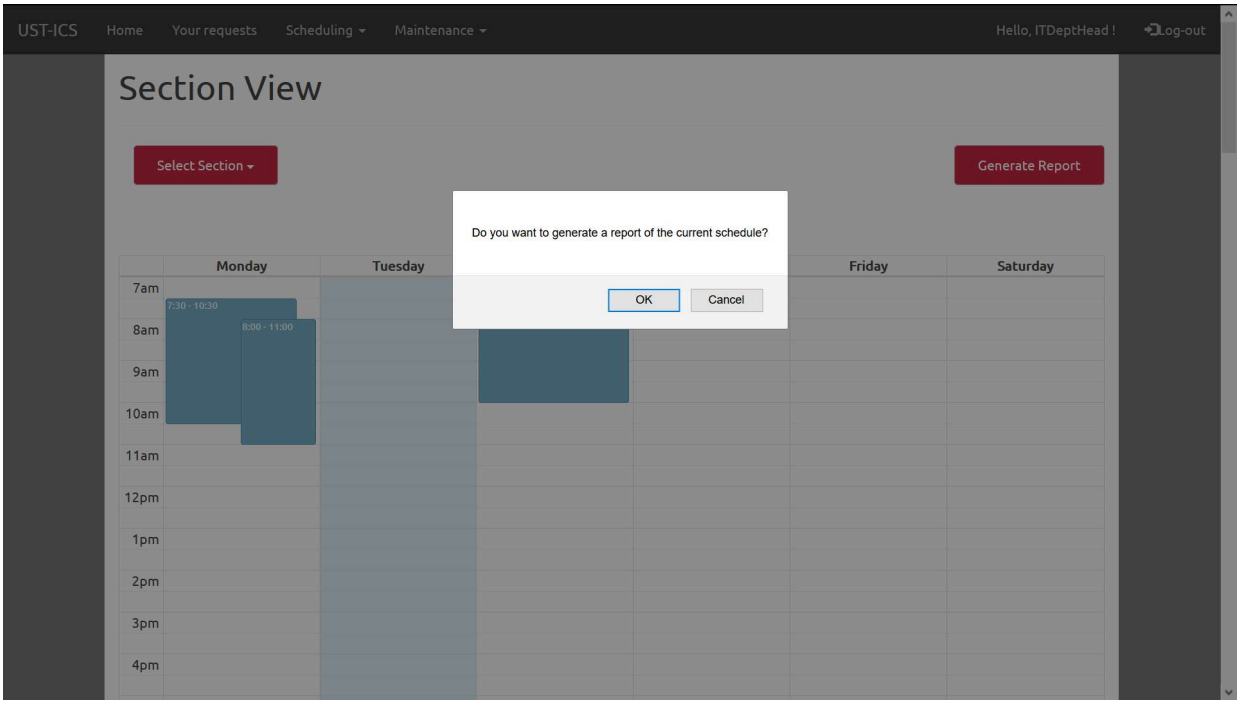


Figure 3.4. 14: Generate Report

Your requests

The list of pending, approved, and denied requests will be displayed.

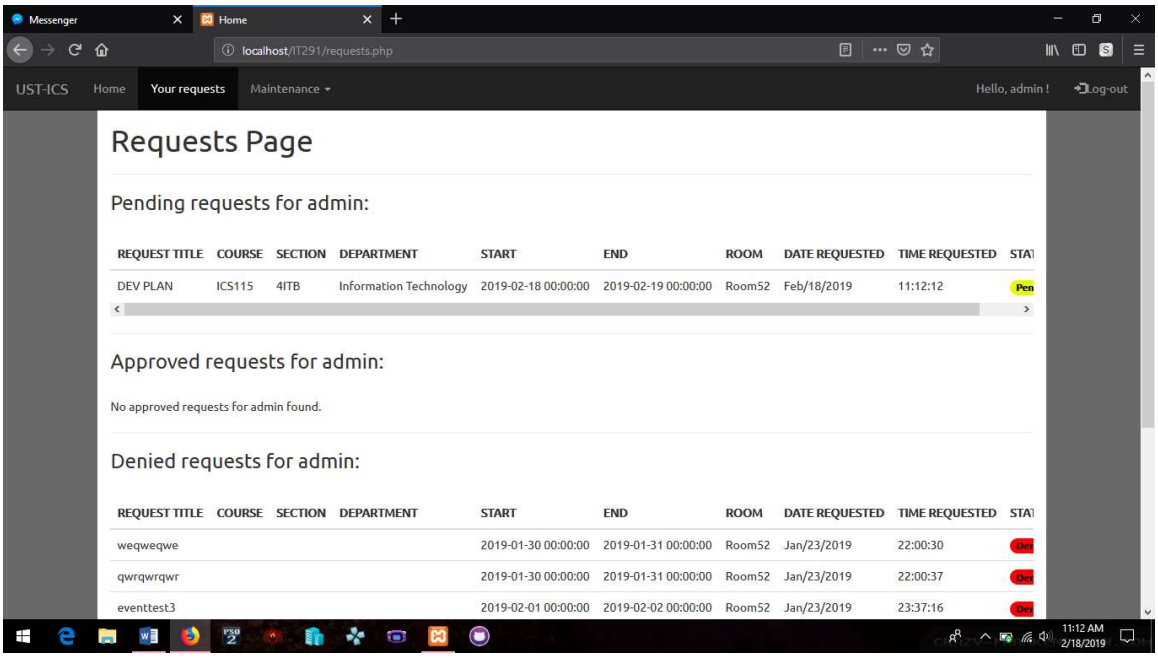


Figure 3.4. 15: Your requests

Requests Management

The list of pending requests will be displayed. The administrator can approve or deny these requests. Once an action has been made, the client will be emailed.

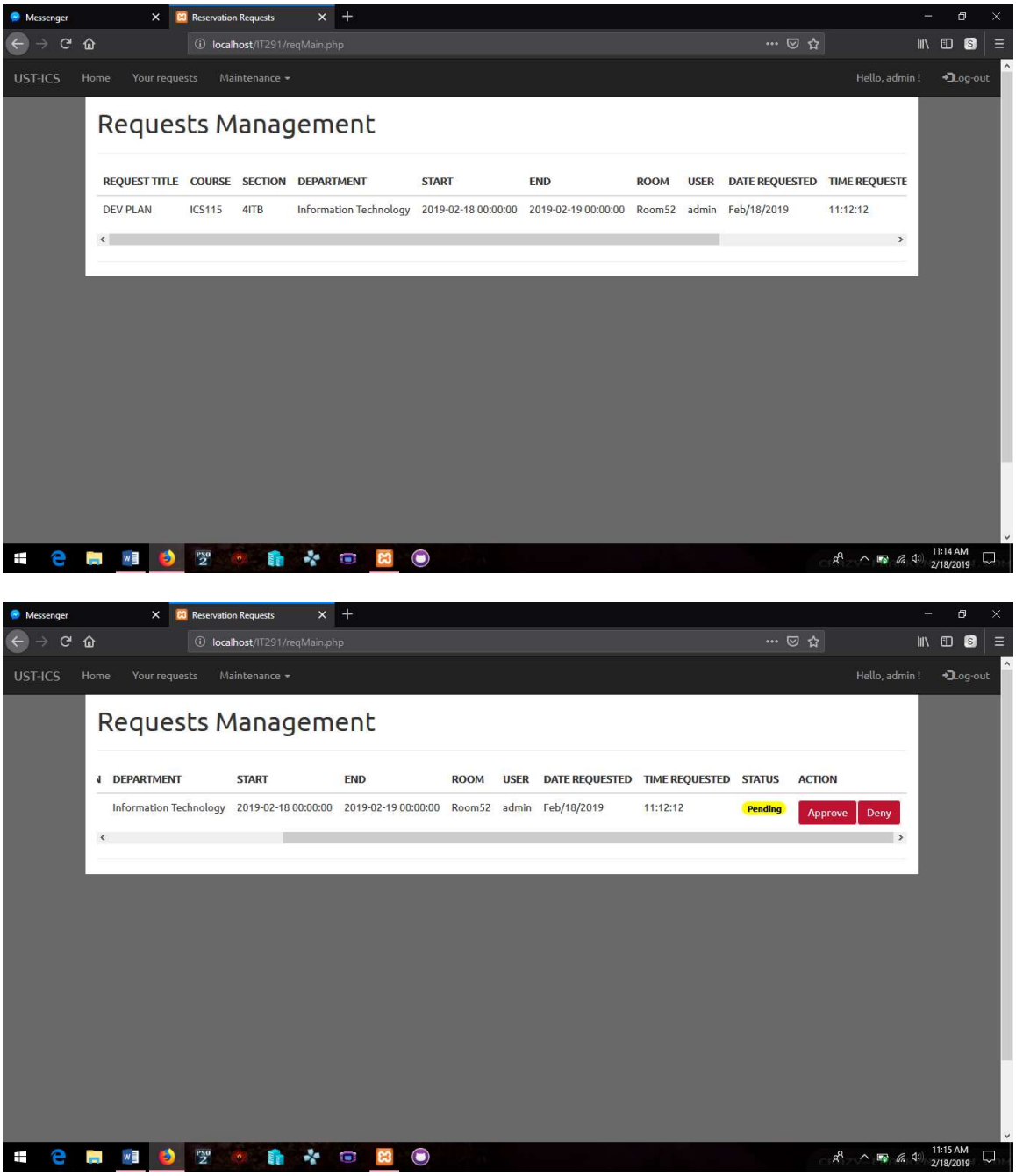


Figure 3.4. 16: Requests Management

Test Plan

Master Plan

The Institute of Information and Computing Sciences' reservation system is manually done; the requester must fill-out a form and have it signed by the administrator in the IICS Office. Their scheduling system also features manual entering of data. In order to alleviate this problem, the team will develop a system that will merge the capabilities of an online reservation and scheduling applications into one whole system.

Particular functions of the system, such as the Login function, will be applied with security capabilities, such as encrypting the characters when the user enters data into the password field. Another security-applied function is the Register function, where an admin must first verify the one registering before the user is granted access into the system.

The design of the system will be kept simple in order to avoid difficulties in understanding the functions of the system. When a user opens the system, the login page appears. If the user is registered, they can enter their credentials and access the system. The accessible functions once logged in for non-admins are limited. If the user is not registered, they can create an account. The system requires the user to enter password using a required criteria in order to increase security. Once they have entered data, their registration request will be sent to the admin as email for verification. Only when the admin has verified their registration request will they be able to login.

Once logged in, for non-admin users, they will be able to see a Calendar in the homepage. This calendar shows the events for the day, the week, or the month. This is also where reservations or events can be made. The user must select a vacant timeframe in the calendar. Once selected, a popup window appears, wherein the user must enter a

description for the reservation. Once the request has been made, it will be sent as an email to the admin. The admin will be notified every time a reservation request has been done. From there, the admin will approve or disapprove of the request. Either way, the client will be notified of an update about their request. If their request has been approved, the calendar will reflect the reservation on the chosen timeframe.

Non-admin users can also keep track of their requests via the "Your requests" function. In this function, the current and previous reservation requests are recorded. This is also where the client can check the status of their reservation requests. Also, the records can be sorted accordingly in order track particular records easier.

Department Heads also have control over the maintenance function of the system. However, they cannot use the Archive and Manage Users function. Also, they are the only ones who can make use of the Scheduling function. The Scheduling function is responsible for the creation of schedules for rooms, faculty, and sections. It also has a filtering feature in the form of a Room View, Faculty View, and Section View. In the Section View, the department head can select a certain timeframe on the integrated calendar, after which they need to enter the course, room, faculty, and section. Once the data has been applied, the calendar will reflect the newly added schedule. The newly added schedule is also reflected in the Room and Faculty View. Also, the department head can generate a report of a particular schedule, which can also be printed.

Administrators have additional privileges once logged in. Admins are notified of reservation requests. They are then in charge of approving or disapproving the requests. Once they approve or disapprove a request, the clients are notified via email. Administrators can also manage user accounts. When a client registers, their account first

needs to be verified. Admins will also be notified of the verification requests, and once they verify an account, its respective owner will be notified via email.

Administrators also have access to the Maintenance function of the system, which allows them to manage rooms, faculties, courses, and sections. Each maintenance module features batch encoding, allowing the entry of multiple data at once. In adding rooms, the admin only needs to enter the room's name. Once added, the system will immediately reflect the new data in the respective window. The admin can also delete multiple rooms. After deleting the rooms, they will be instantly removed from the system.

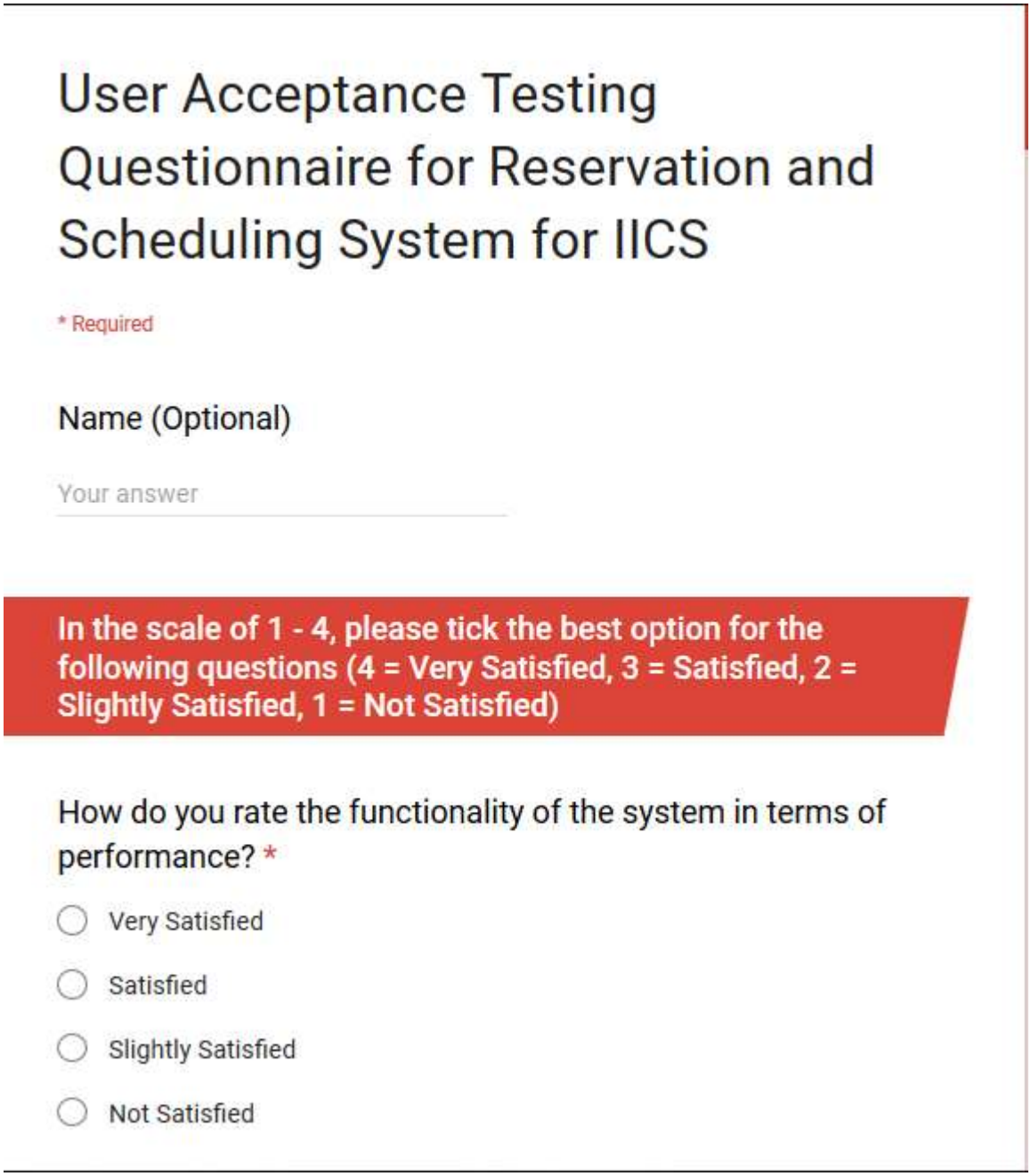
In adding faculties, the admin is required to enter the Faculty ID, First Name, Last Name, and Department of the particular faculty. Once added, the new data is immediately reflected from the system. The delete function also has checkboxes to allow multiple deletion of data.

In adding courses, the admin is required to enter the Course ID, Course Description, Lecture Units, and Lab Units. Once added, the new data is immediately reflected from the system. The admin can delete multiple course also, which are expected to be removed from the system once deleted.

In adding sections, the admin is required to enter the Year Level, and Department for the section. Once added, the new data is immediately reflected from the system. If the admin deletes data, they are immediately removed from the system.

The administrator can also use the Archive function. The admin is required to enter the Academic Year and Semester. Archive will create a record of the current schedule or calendar, which can be used as a basis for the next schedule. Once archived, the current calendar will be emptied.

User Acceptance Testing



User Acceptance Testing Questionnaire for Reservation and Scheduling System for IICS

* Required

Name (Optional)

Your answer

In the scale of 1 - 4, please tick the best option for the following questions (4 = Very Satisfied, 3 = Satisfied, 2 = Slightly Satisfied, 1 = Not Satisfied)

How do you rate the functionality of the system in terms of performance? *

- Very Satisfied
- Satisfied
- Slightly Satisfied
- Not Satisfied

Figure 3.6.1: User Acceptance Test Part 1 of 2

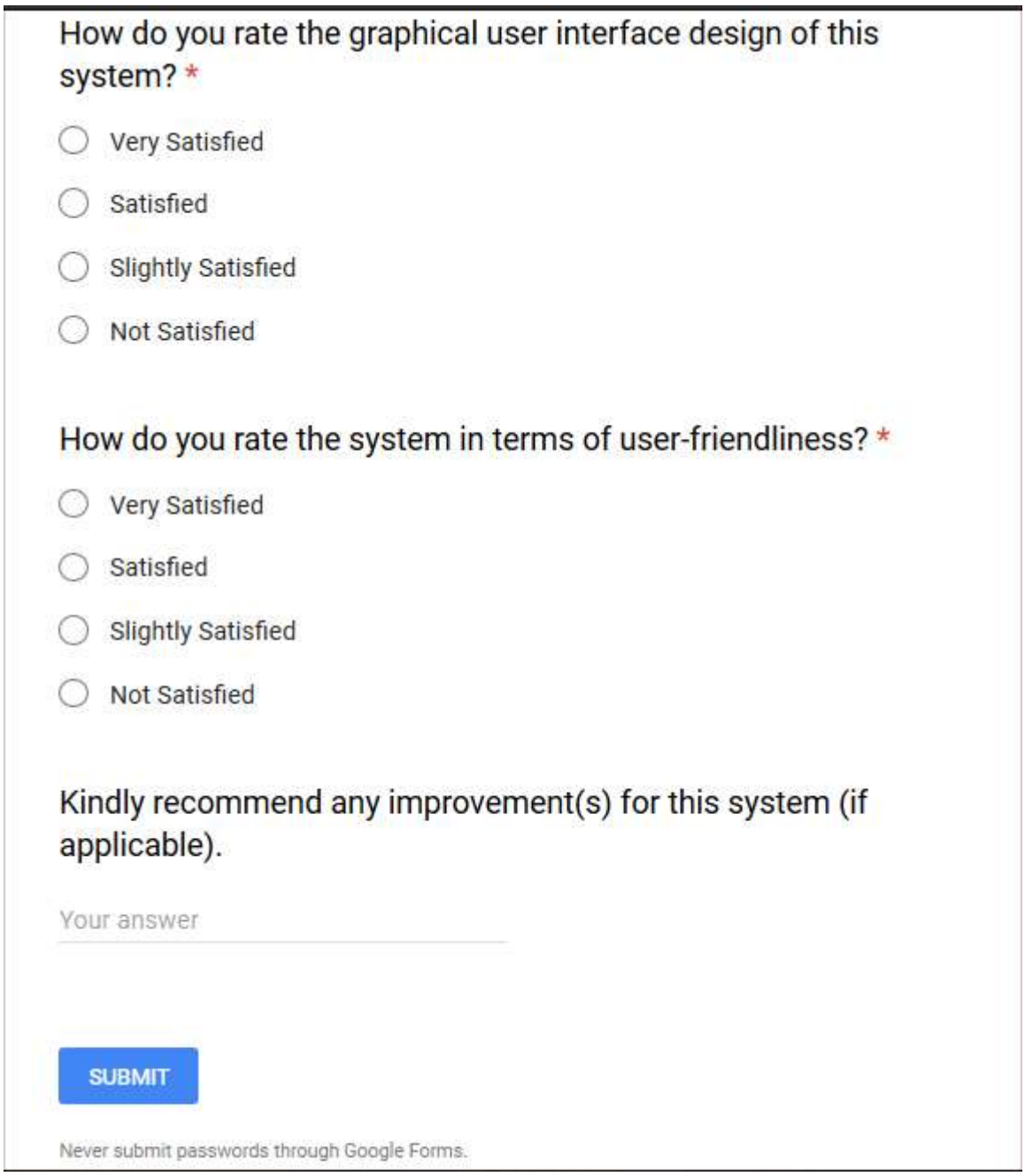


Figure 3.6.2: User Acceptance Test Part 2 of 2

Acceptance Plan

The team will be using test cases for each function. It becomes acceptable only when the expected result matches the actual result. A survey will also be used to help validate the acceptability of the system.

Test Cases

Login

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Valid Login	1. UN/PW match	1. Enter UN 2. Enter PW 3. Click "Login"			
2. Invalid Login	2. UN/PW mismatch	1. Enter UN 2. Enter PW 3. Click "Login"			
3. Blank Login	3. UN/PW blank	1. Click "Login"			

Register

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Valid Register	1. PW/Confirm PW match, follows password complexity	1. Enter UN 2. Enter PW 3. Enter confirm PW 4. Click "Register"			
2. Invalid Register	2. PW/Confirm PW match, does not	1. Enter UN 2. Enter PW 3. Enter confirm			

	follow password complexity	PW 4. Click "Register"			
3. Blank Register	3. UN/PW, or Confirm PW blank	1. Click "Register"			

Reserve

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Blank Reserve	1. Left all fields blank	1. Select calendar date 2. Click "Request"			
2. A field has been left unsatisfied	2. All fields satisfied except some	1. Select calendar date 2. Enter Summary 3. Click "Request"			
3. Successful request	3. All fields satisfied	1. Select calendar date 2. Enter Summary 3. Select Department 4. Enter Section 5. Enter Course 6. Enter Purpose 7. Click "Request"			

Add Room

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Add Success	1. Entered room name	1. Enter room name 2. Click “Add”			
2. Add Blank	2. Room name left blank	1. Click “Add”			

Delete Room

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Delete Room	1. Selected rooms to be deleted	1. Select rooms 2. Click “Delete”			

Add Faculty

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Add Success	1. Entered data into the fields	1. Enter faculty ID 2. Enter First Name 3. Enter Last Name 4. Select Department 5. Click “Add”			
2. Invalid Add	2. Left a field unanswered	1. Enter faculty ID 2. Enter First Name 3. Enter Last Name 4. Click “Add”			

3. Add Blank	3. Left all fields blank	1. Click “Add”			

Delete Faculty

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Delete Faculty	1. Selected faculties to be deleted	1. Select faculties 2. Click “Delete”			

Add Course

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Add Success	1. Entered data into the fields	1. Enter Course ID 2. Enter Course Description 3. Enter Lecture Units 4. Enter Lab Units 5. Click “Add”			
2. Invalid Add	2. Left a field unanswered	1. Enter Course ID 2. Enter Course Description 3. Enter Lecture Units 4. Click “Add”			
3. Add Blank	3. Left all fields blank	1. Click “Add”			

Delete Course

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Delete Course	1. Selected courses to be deleted	1. Select courses 2. Click "Delete"			

Add Section

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Add Success	1. Entered data into the fields	1. Enter Year Level 2. Select Department 3. Click "Add"			
2. Invalid Add	2. Left a field unanswered	1. Enter Year Level 2. Click "Add"			
3. Add Blank	3. Left all fields blank	1. Click "Add"			

Delete Section

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Delete Section	1. Selected sections to be deleted	1. Select sections 2. Click "Delete"			

Archive

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Archive_ Success	1. Admin entered data into all fields	1. Hover on Maintenance 2. Click “Archive” 3. Enter Academic Year 4. Select Semester 5. Click “Archive Semester” 6. Click “Yes” on confirmation message			
2. Archive_ Invalid	2. Admin left a field unsatisfied	1. Hover on Maintenance 2. Click “Archive” 3. Enter Academic Year 4. Click “Archive Semester” 5. Click “Yes” on confirmation message			

Manage Users

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Verify User	1. Admin verified a registration request	1. Hover on Maintenance 2. Click “Users” 3. Click on a pending reservation request 4. Select “User”			

My Requests

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. My Requests	1. User wants to check his/her reservation requests	1. Login 2. Click on “My Requests”			

Approve Reservation Requests

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Approve Request	1. Admin approves a reservation request	1. Click on “Check Requests” 2. Select pending reservation request 3. Click “Approve”			

Disapprove Reservation Requests

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Disapprove Request	1. Admin disapproves a reservation request	1. Click on "Check Requests" 2. Select pending reservation request 3. Click "Disapprove"			

Cancel Reservation Request

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Cancel_ Valid	1. User wants to cancel his/her reservation request	1. Click on "Your requests" 2. Select pending reservation request 3. Click "Cancel" 4. Enter reason for cancellation 5. Click "Submit"			
2. Cancel_ invalid	2. User leaves reason for cancellation blank	1. Click on "Your requests" 2. Select pending reservation request 3. Click "Cancel" 4. Click "Submit"			

Scheduling

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Schedule_ Valid	1. Dept. Head satisfies all required fields	1. Select timeframe 2. Enter Course 3. Enter Room 4. Enter Faculty 5. Enter Section 6. Click "Apply"			
2. Schedule_ Invalid	2. Dept. Head leaves a field unsatisfied	1. Select timeframe 2. Enter Course 3. Enter Room 4. Enter Faculty 5. Click "Apply"			
3. Schedule_ Overlap	3. Dept. Head tries to add schedule on occupied timeframe	1. Select occupied timeframe			

Generate Report

Test Case	Test Case Scenario	Steps	Expected Results	Actual Results	Remarks
1. Report_ Success	1. Dept. Head successfully creates a report.	1. Click on "Scheduling" 2. Select view 3. Select schedule 4. Click "Generate Report" 5. Click "Yes"			

Implementation Plan

Scheduling and Reservation System for IICS

TASK NAME	START DATE	END DATE	DURATION (WORK DAYS)	TEAM MEMBER	PERCENT COMPLETE
Chapter 1 (Including revision					
Scope and Limitations	September 4, 2018	October2,2018	28	Quitor	100%
Project Context	September 4, 2018	October2,2018	28	Silvestre	100%
Purpose and Description	September 4, 2018	October2,2018	28	Valdes	100%
General Objectives	September 4, 2018	October2,2018	28	Plata	100%
Specific Objectives	September 4, 2018	October2,2018	28	Quitor	100%
Chapter 2					
Related Litirature	October 3, 2018	October 9, 2018	6	Quitor	100%
			6	Silvestre	100%
			6	Valdes	100%
			6	Plata	100%
Chapter 3					
Functional Requirements	October 16, 2018	October 19, 2018		Valdes	100%
Non-Functional Requirements	October 16, 2018	October 19, 2018		Valdes	100%
Methodology	October 19, 2018	October 20, 2018		Valdes	100%
System Design	October 19, 2018	October 24, 2018		Valdes	100%
Master Plan	October 29, 2018	November 02, 2018		Valdes	100%
Acceptance Plan	October 29, 2018	November 02, 2018		Valdes	100%
Hardware Requirements	November 08,	November 10, 2018		Silvestre	
Software Requirements	November 08,	November 10, 2018		Silvestre	
Developtment Plan	November 10, 2018			Plata	
Test plan	November 10, 2018			Valdes	10%
Gantt Chart	November 10, 2018	November 10, 2018		Quitor	100%
Coding					
Register	October 29, 2018	November 1, 2018		Plata/Quitor	
Login	October 29, 2018	November 1, 2018		Plata/Quitor	
Calendar Feature	October 29, 2018	November 1, 2018		Silvestre/Quit	
Maintinance Feature	October 29, 2018	November 1, 2018		Plata/Quitor	
User Verification	November 7, 2018	November 7, 2018		Plata/Quitor	
Encryption	November 7, 2018	November 7, 2018		Silvestre/Quit	

Figure 3.8.1: Gantt Chart Part 1 of 4

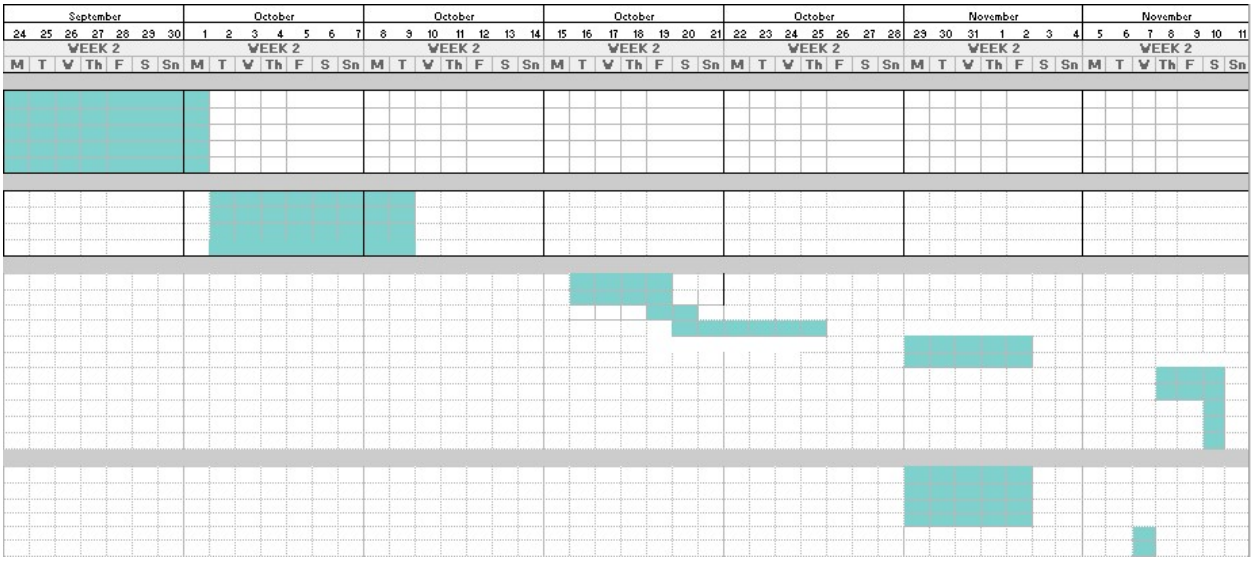


Figure 3.8.2: Gantt Chart Part 2 of 4

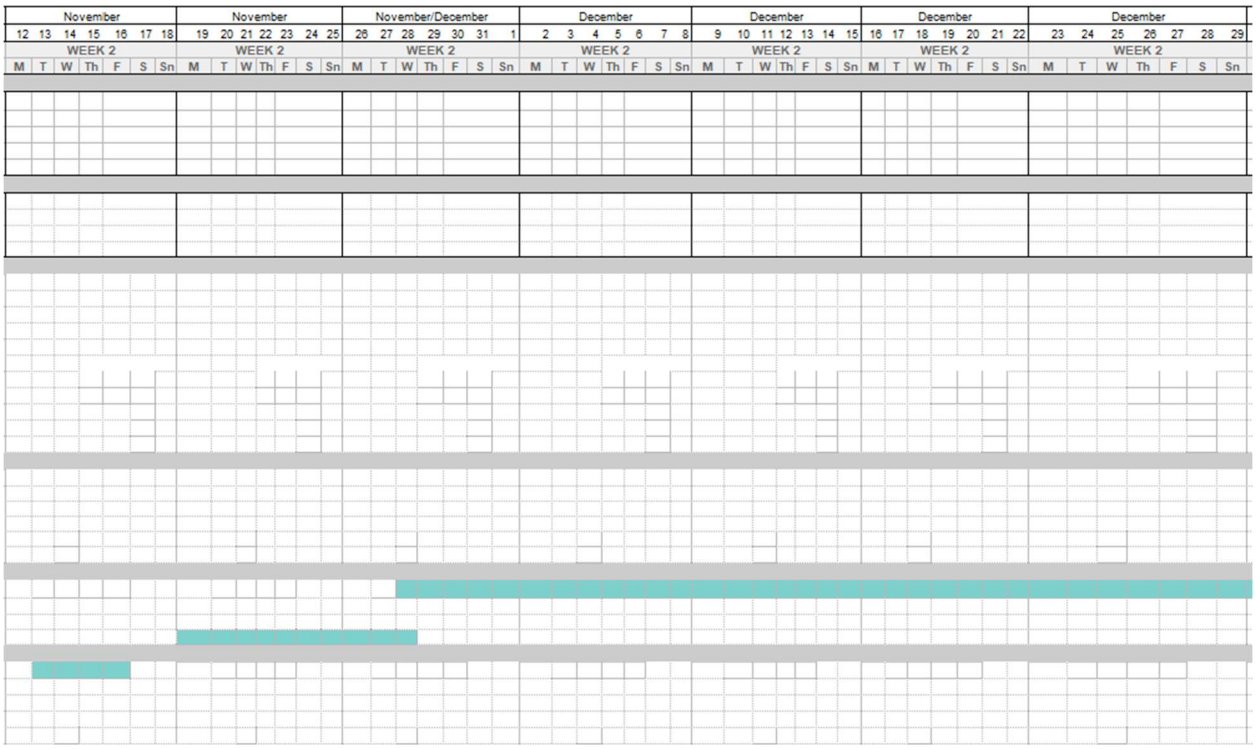


Figure 3.8.3: Gantt Chart Part 3 of 4

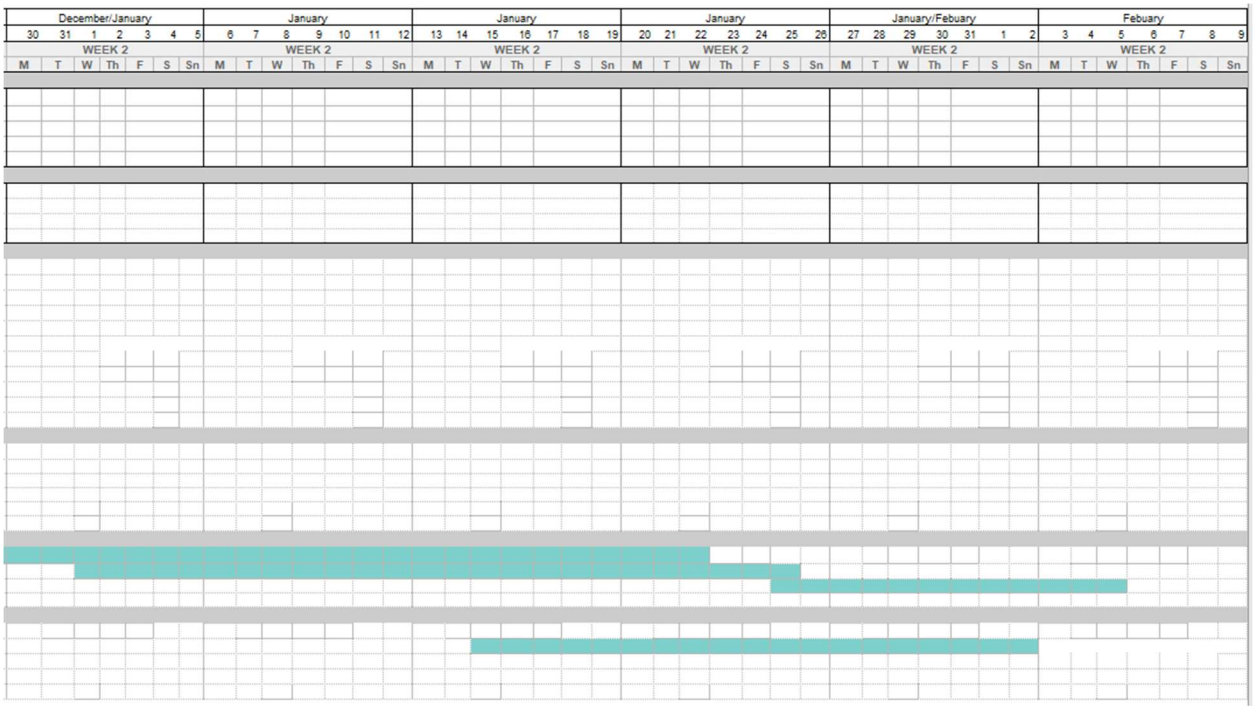


Figure 3.8.4: Gantt Chart Part 4 of 4

Chapter 4: Implementation, Results and Discussion

Requirements Documentation

Detailed Use Case Diagrams

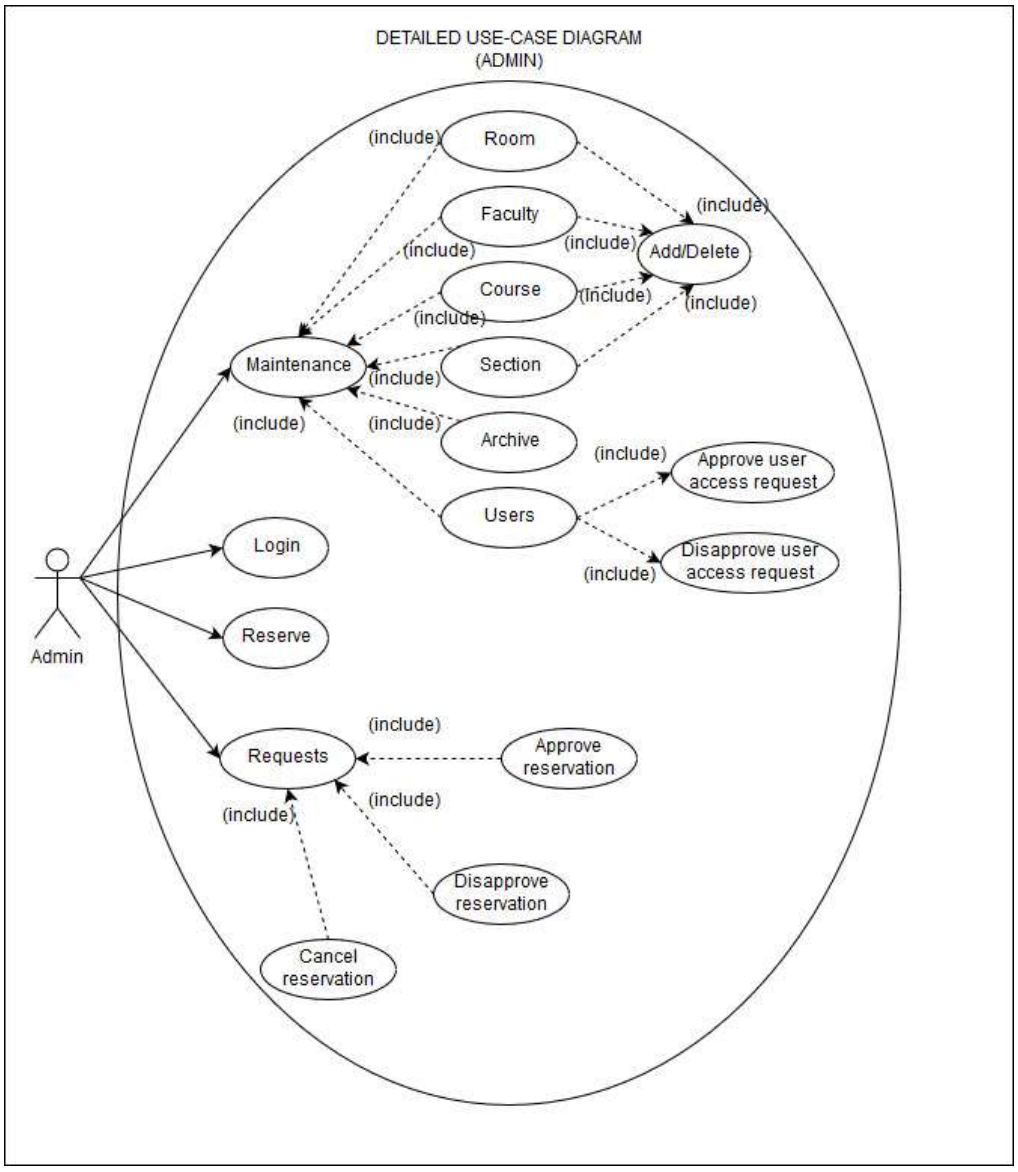


Figure 4.1.1: Detailed Use Case for ADMIN

The administrators of the system can login into the system. They can also make reservations, which means that they can also cancel reservation requests. Since they are admins, they can immediately approve of their own requests in the Maintenance function. Admins also have access to all of the Maintenance function, which are the Room, Faculty, Course, and Section Management, wherein they can add or delete data. They also have access to the User, and Requests Management function, wherein they can approve or deny verification and reservation requests, respectively. Admins are the only ones who can use the Archive function, which will delete but keep a record of the current requests in the schedule.

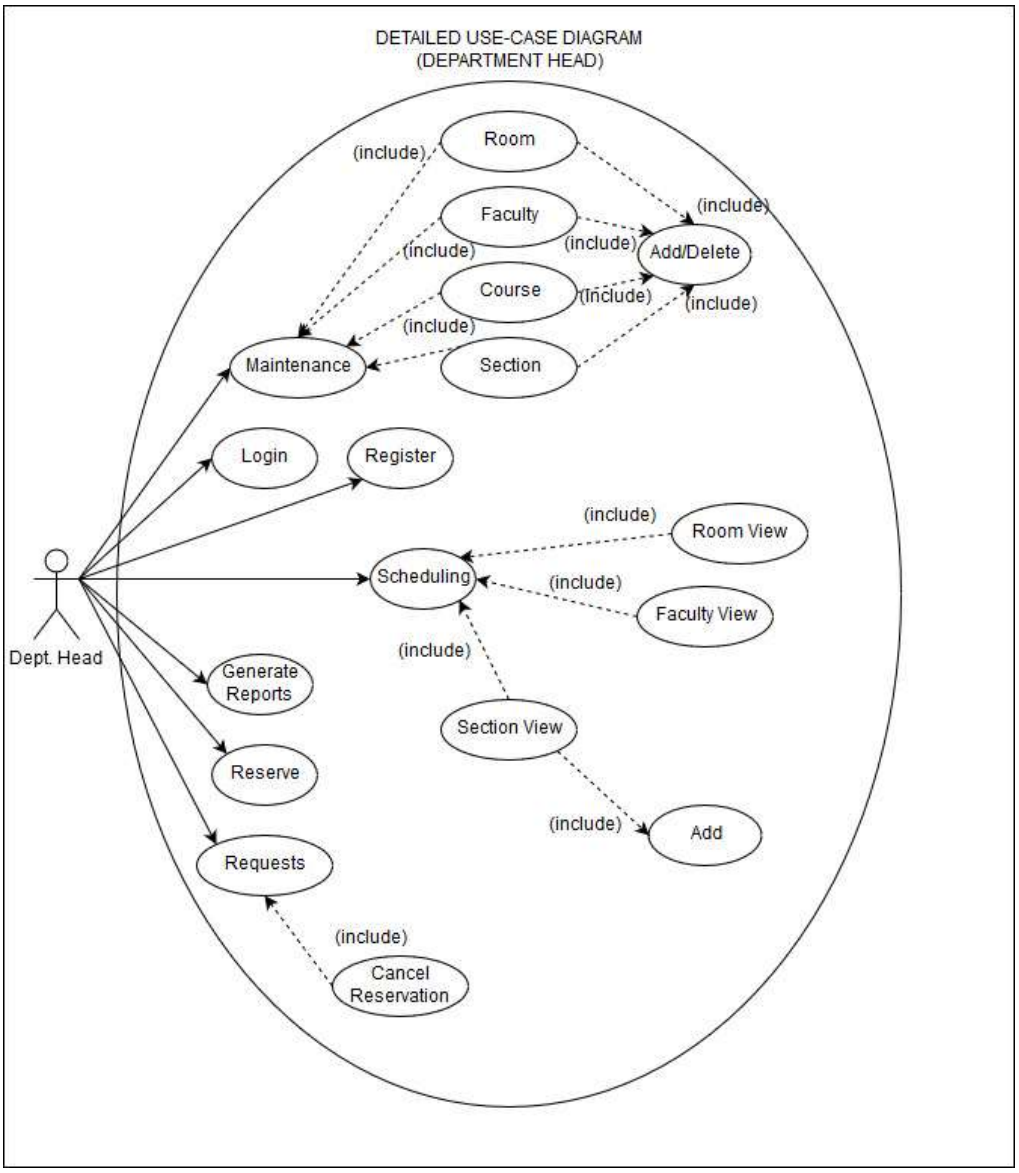


Figure 4.1.2: Detailed Use Case for DEPARTMENT HEAD

Department Heads, before logging in, must register their account first. A checkbox for Department Head identification is available in the register window. Once logged in, Dept. Heads can make reservations, and cancel requests if needed. They also have access to the Maintenance windows, and they can add or delete data. However, they cannot access the User Management, Requests Management, and the Archive functions. Department Heads have access to the Scheduling function, which allows them to view schedules for a

particular room, faculty, or section. They can also add schedules for rooms, faculty, and sections. Once finished, department heads can generate a report of the selected schedule in the form of an excel file.

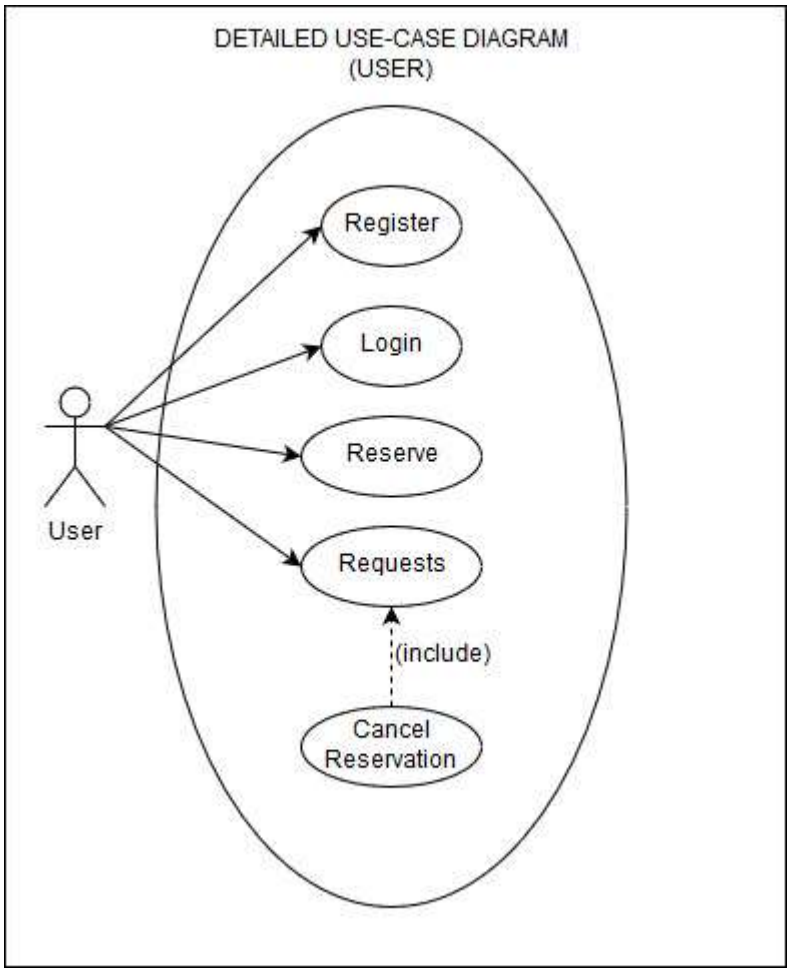


Figure 4.1.3: Detailed Use Case for USER

New users must first register an account. Once verified by the admin, they will be granted access into the system, albeit with limited privileges—accounts with "user" as their access level has access only to the reservation function of the system. They can also cancel their reservation requests.

Actual System Requirements

Software Requirements	Google Chrome Version 72.0.3626.121 (Official Build) (64-bit)
	Mozilla Fire Fox Version 65.0.2 (Official Build) (64-bit)

Minimum Hardware Requirements	RAM: 4GB
	OS: Windows 7 32-bit

Recommended Hardware Requirements	RAM: 16GB
	OS: Windows 10 64-bit

Design of Software, Systems, Product, and/or Processes

System Sequence Diagrams

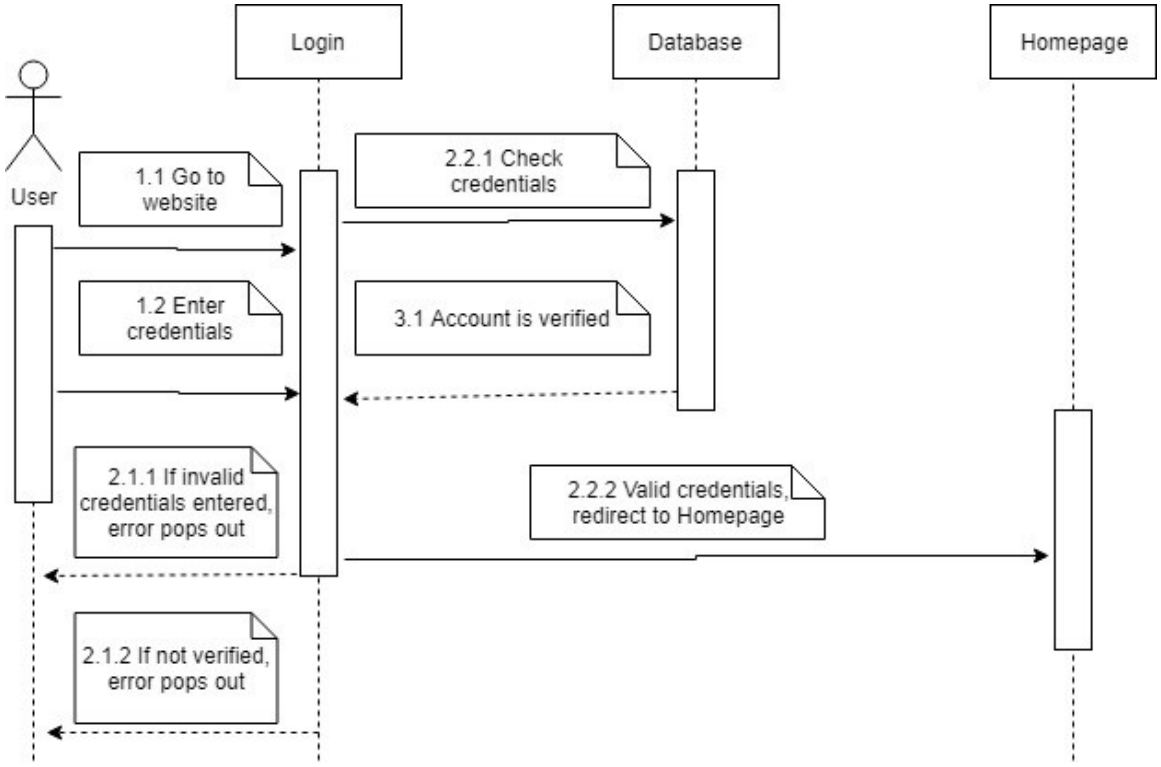


Figure 4.2.1: System Sequence Diagram for Login

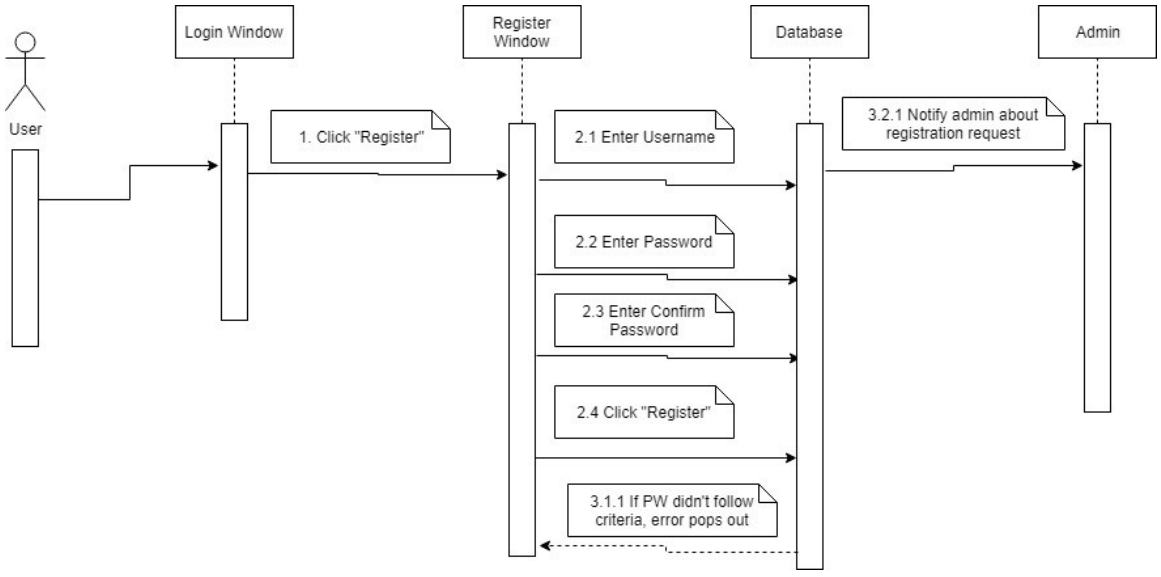


Figure 4.2.2: System Sequence Diagram for Register Part 1 of 2

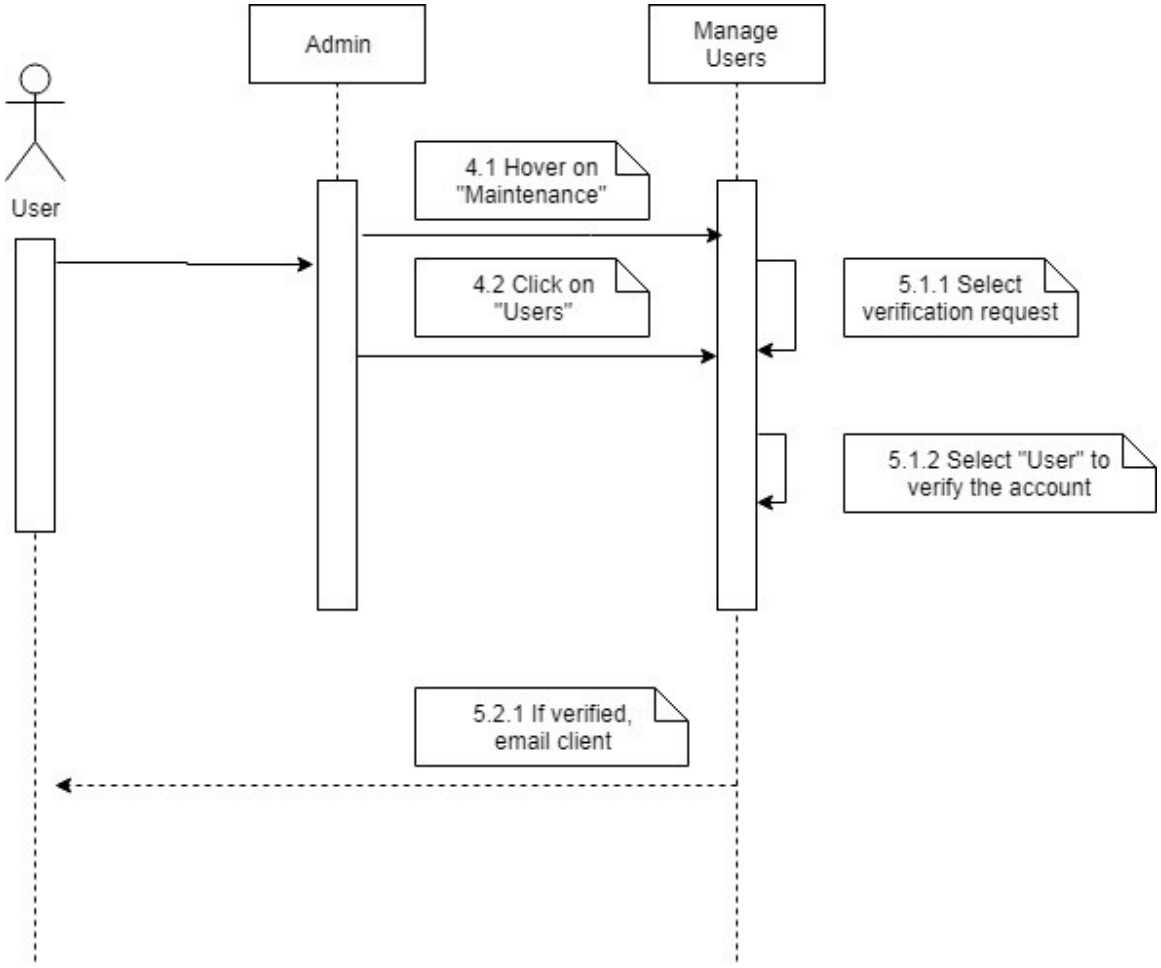


Figure 4.2.3: System Sequence Diagram for Register Part 2 of 2

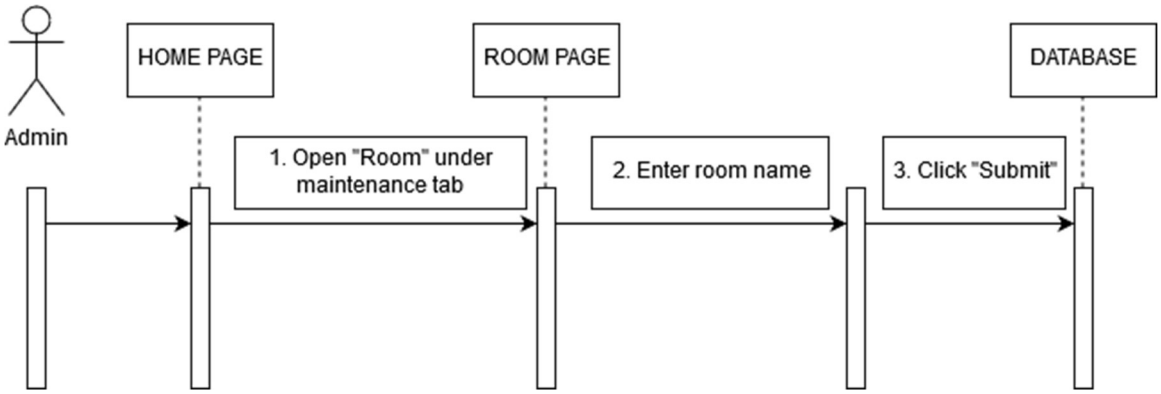


Figure 4.2.4: System Sequence Diagram for Room Add

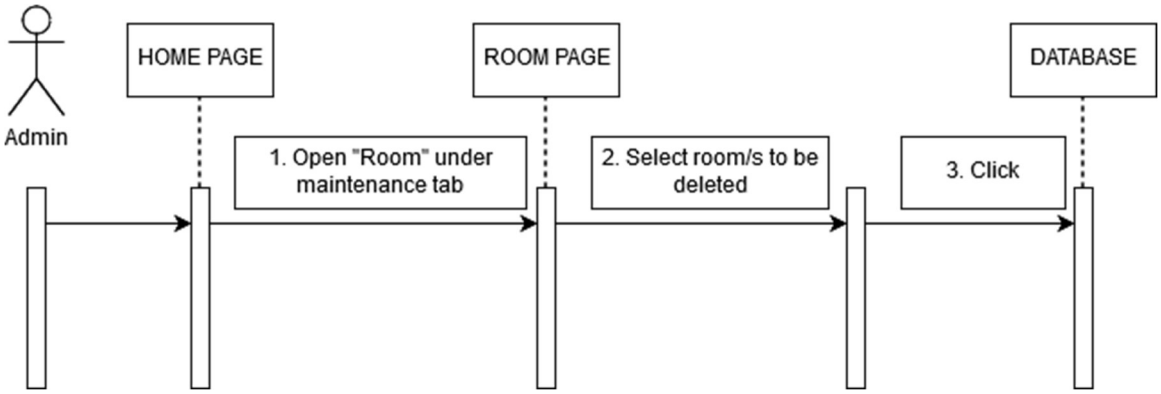


Figure 4.2.5: System Sequence Diagram for Room Delete

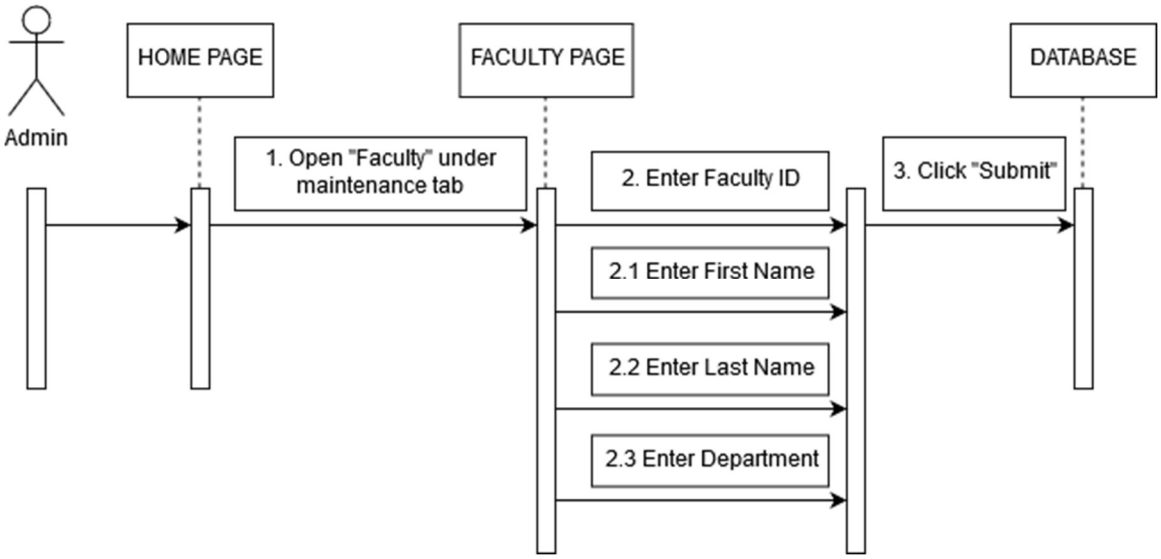


Figure 4.2.6: System Sequence Diagram for Faculty Add

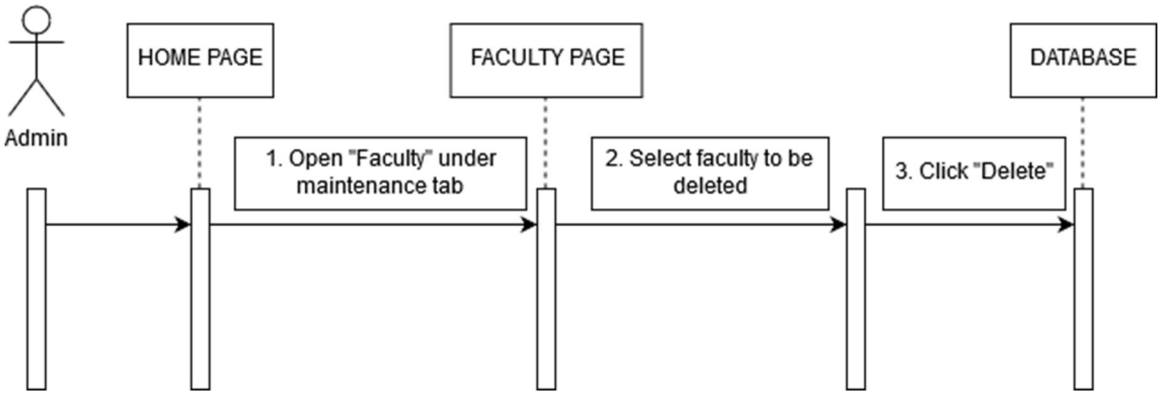


Figure 4.2.7: System Sequence Diagram for Faculty Delete

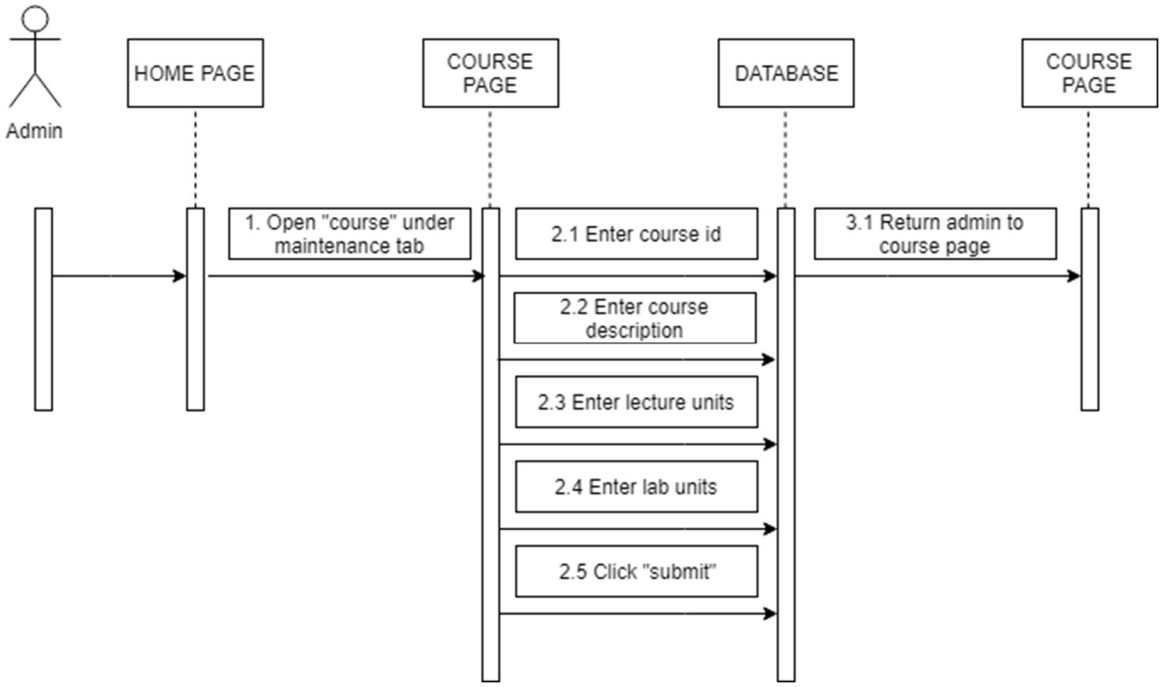


Figure 4.2.8: System Sequence Diagram for Course Add

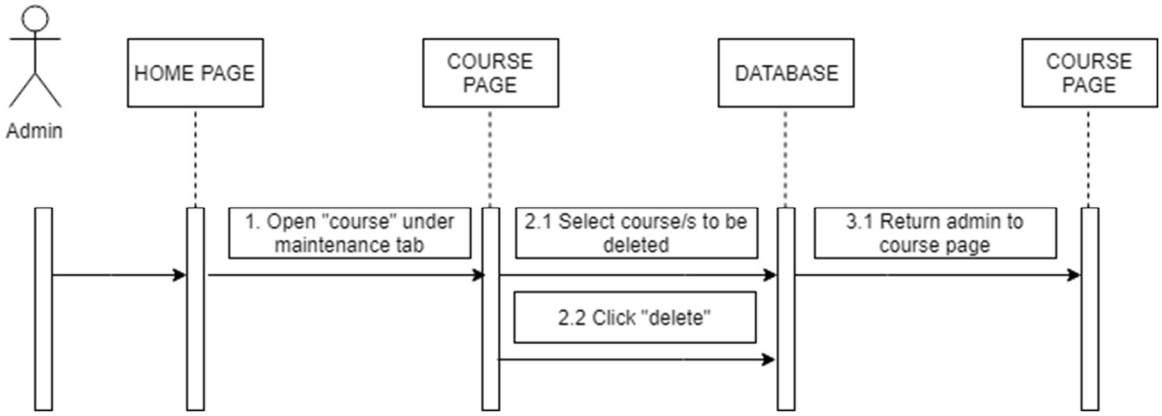


Figure 4.2.9: System Sequence Diagram for Course Delete

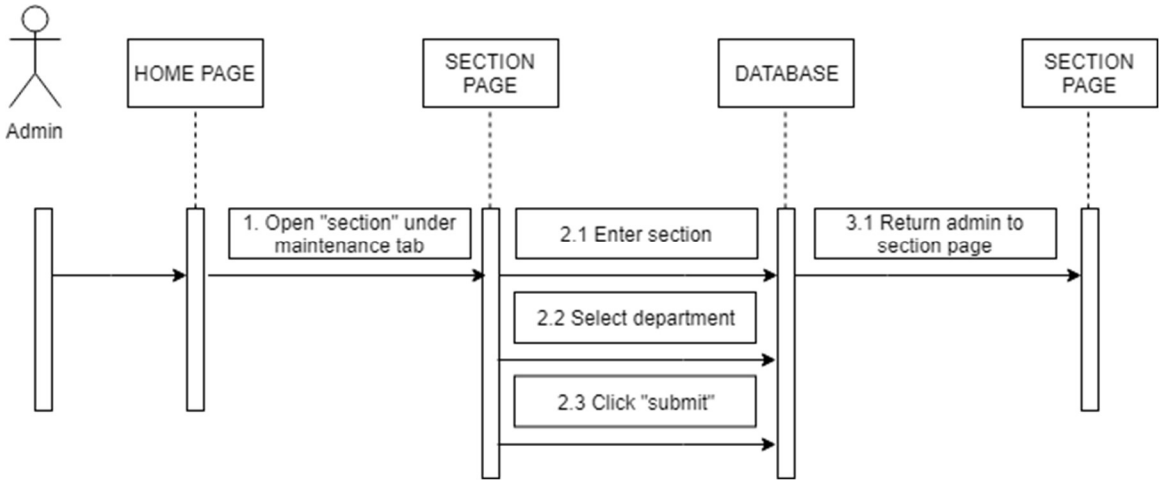


Figure 4.2.10: System Sequence Diagram for Section Add

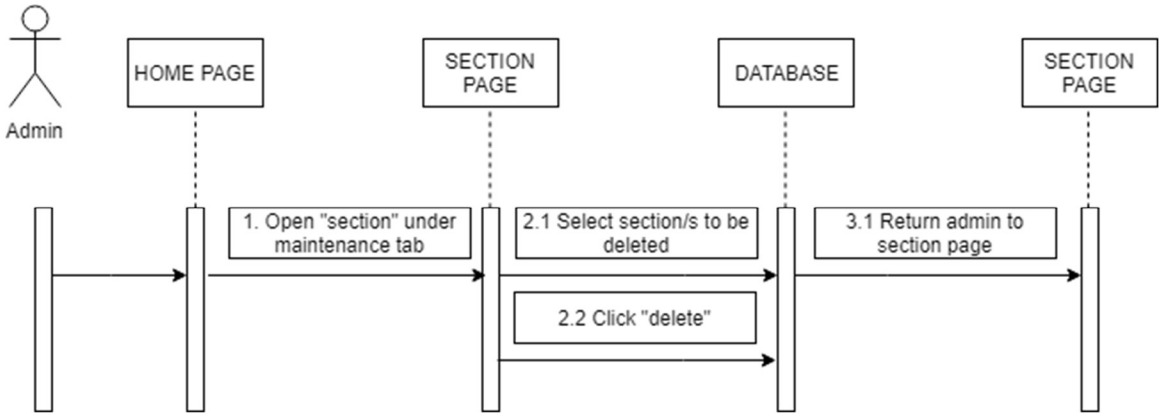


Figure 4.2.11: System Sequence Diagram for Section Delete

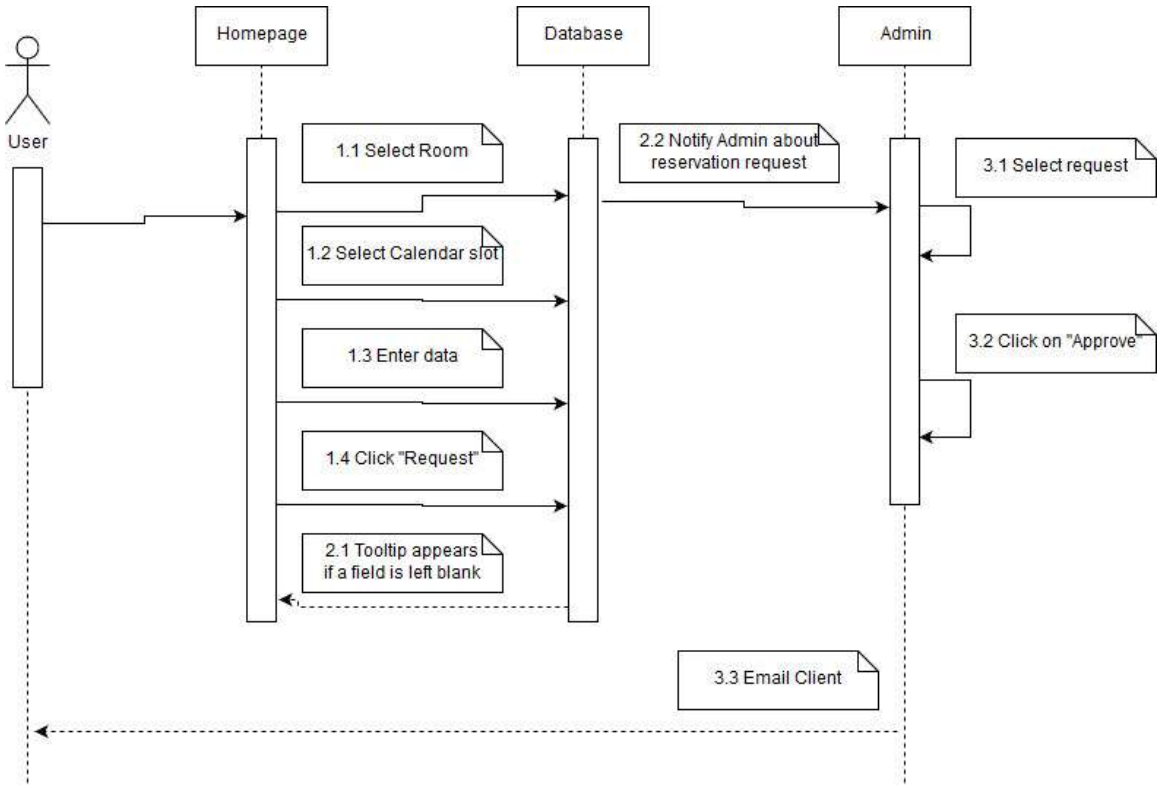


Figure 4.2.12: System Sequence Diagram for Reservation

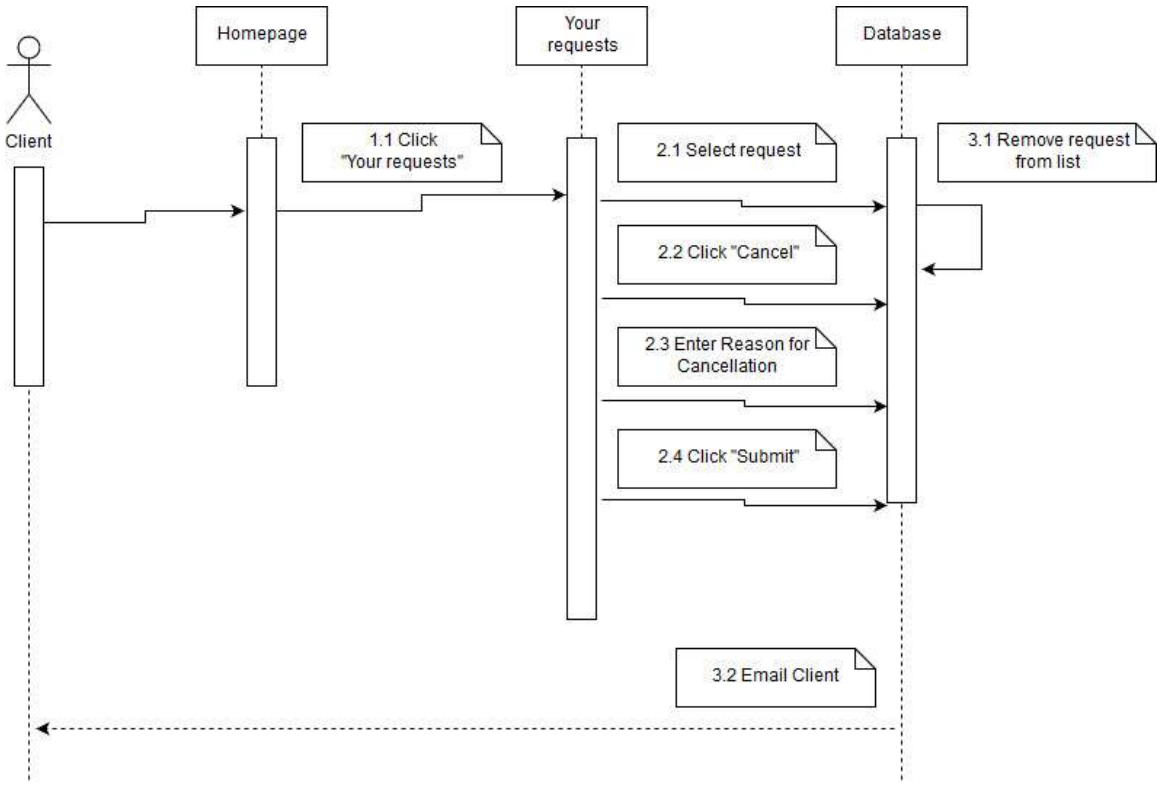


Figure 4.2.13: System Sequence Diagram for Cancel Reservation

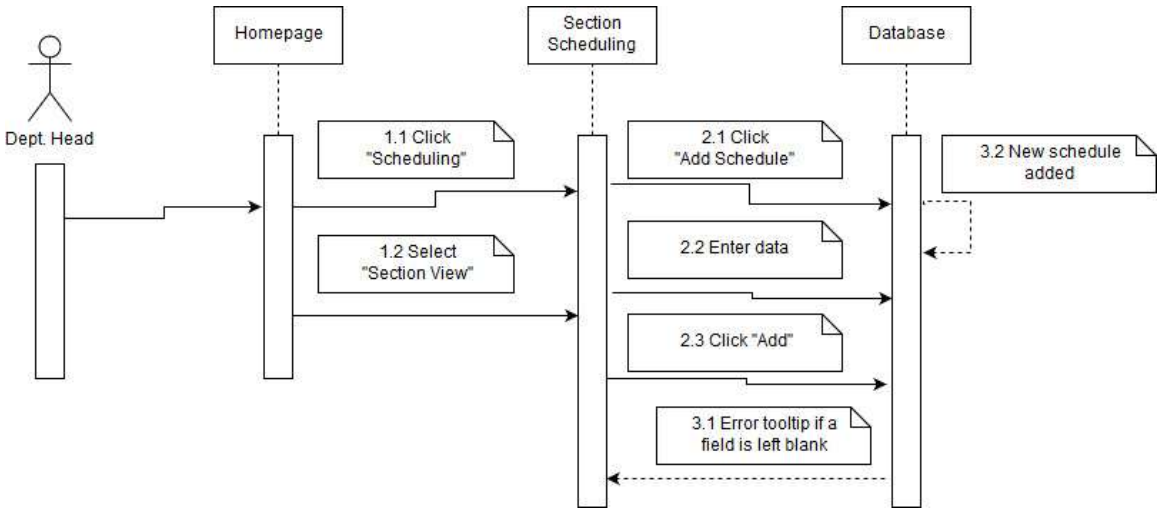


Figure 4.2.14: System Sequence Diagram for Add Schedule

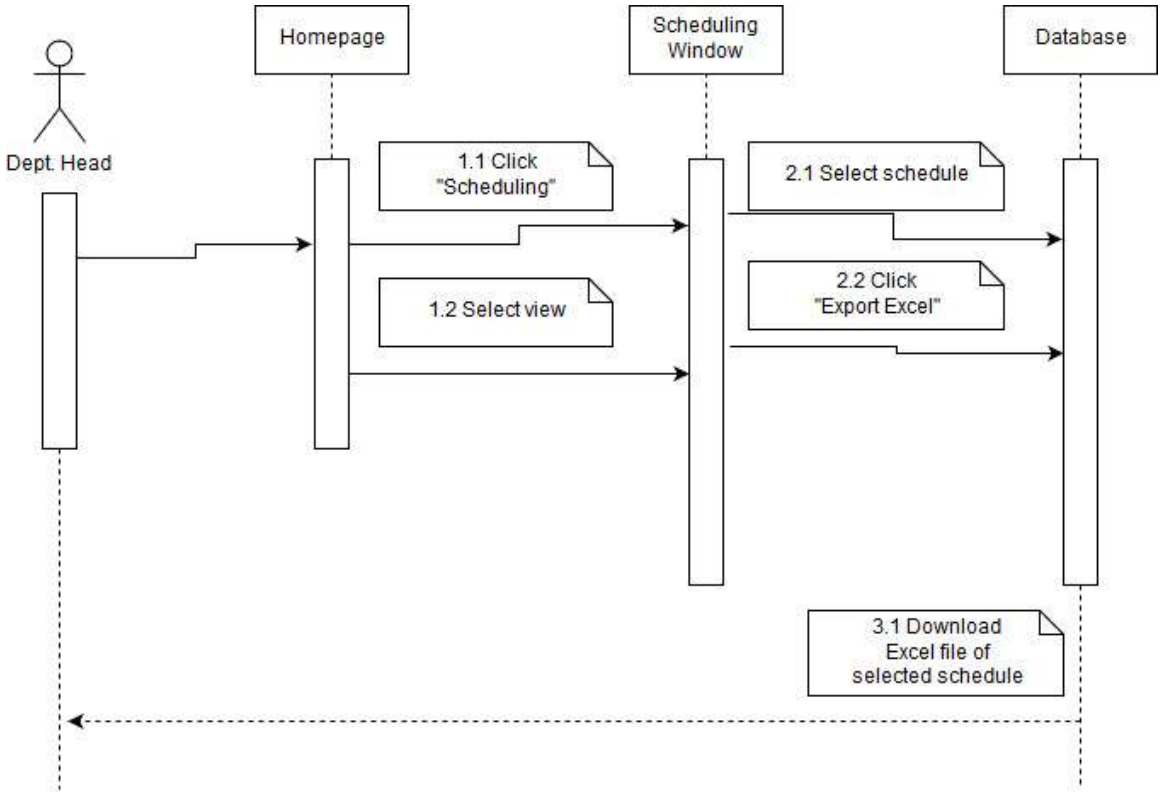


Figure 4.2.15: System Sequence Diagram for Generate Report

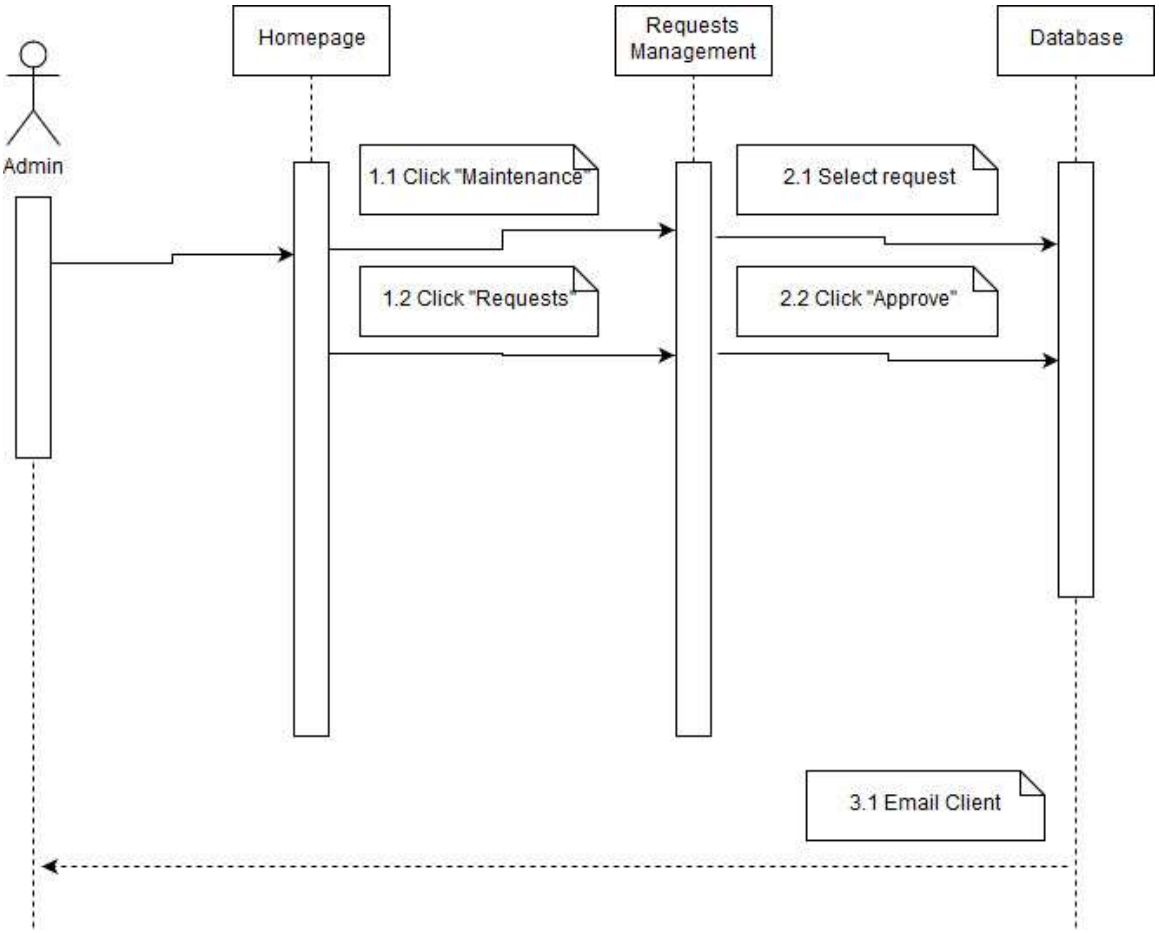


Figure 4.2.16: System Sequence Diagram for Requests

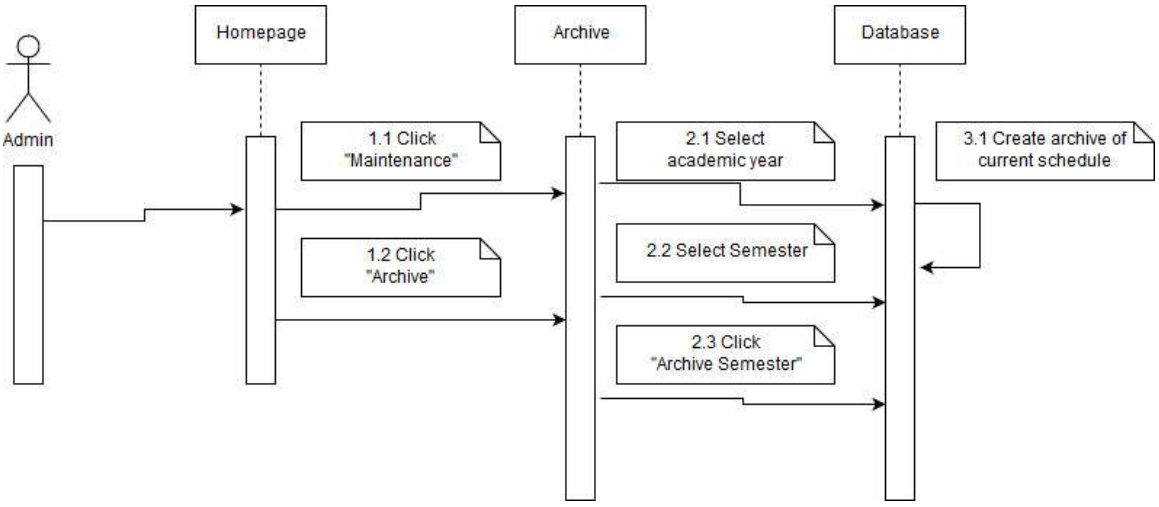


Figure 4.2.17: System Sequence Diagram for Archive

Database Design

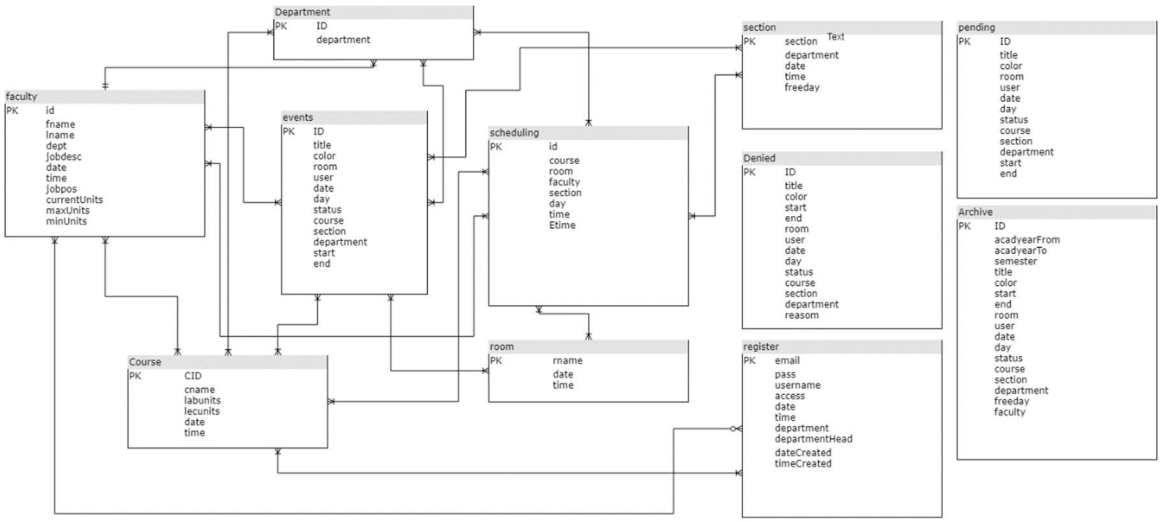


Figure 4.3.1: Database Design

Class Diagram

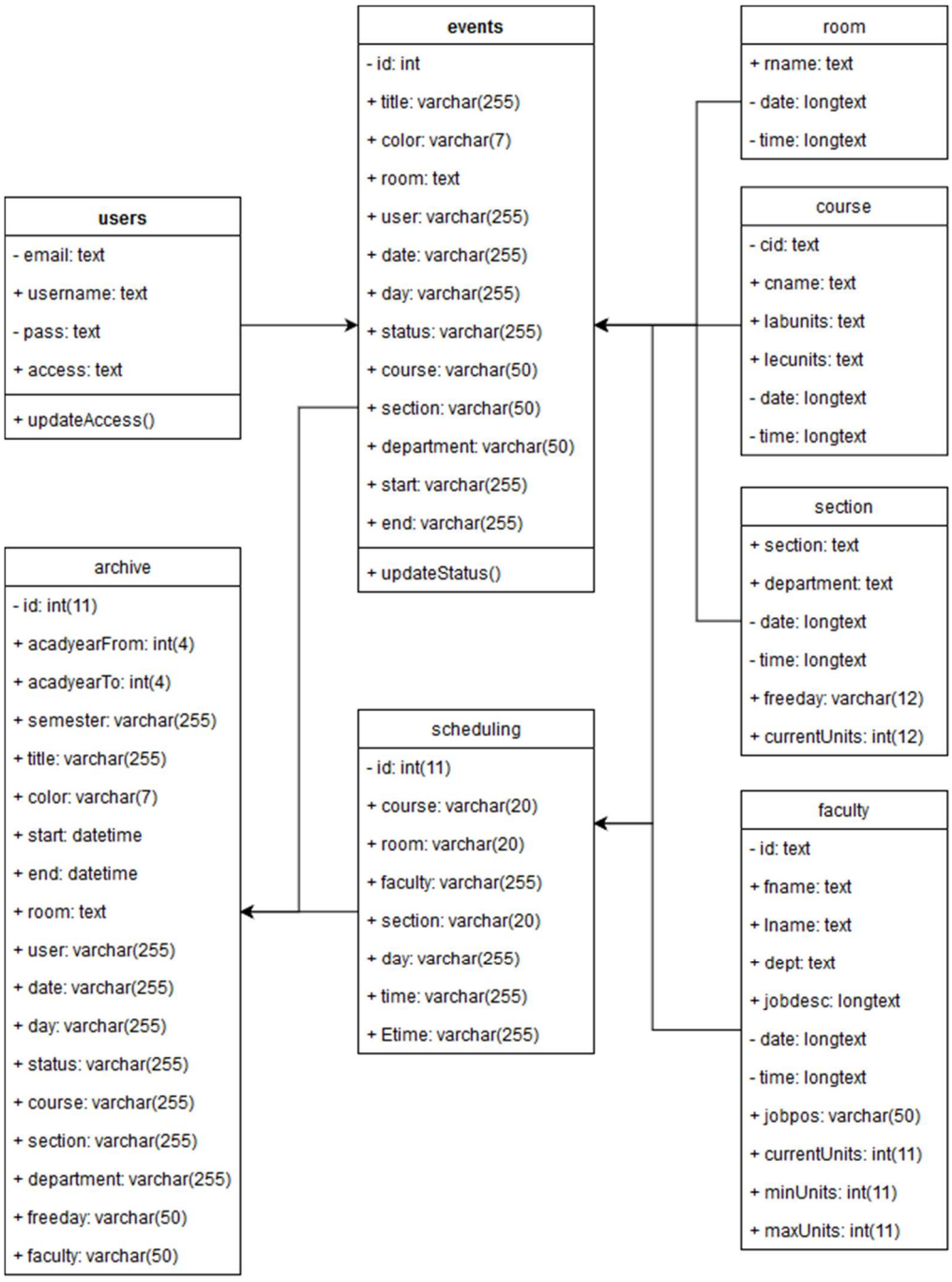


Figure 4.4.1: Class Diagram

Testing and Results

Summary of Test Cases

Module Name: Login

Pre-Conditions: User has a validated account

Actors: User/Admin/Department Head

Test Case Scenario:

1. Enter username
2. Enter password
3. Click “Login”

Post-Conditions: The user is directed into the homepage.

Module Name: Register

Pre-Conditions: User has visited the website, and clicked “Register” in the Login Window

Actors: User/Admin/Department Head

Test Case Scenario:

1. Enter email
2. Enter username
3. Enter password
4. Enter confirm password
5. Select department
6. Specify if department head or not
7. Click “Submit”

Post-Conditions: The system will generate a message stating that the request has been successfully sent to the admin for verification.

Module Name: Reserve

Pre-Conditions: User has logged in

Actors: User/Admin/Department Head

Test Case Scenario:

1. Click on a calendar slot
2. Enter Title
3. Select Department
4. Select Section
5. Select Course
6. Select Purpose
7. Select Room
8. Enter Start date
9. Enter End date
10. Click “Request”

Post-Conditions: The system will display a message that the request has been sent to the admin for approval.

Module Name: Maintenance_Rooms

Pre-Conditions: User has accessed the Room Maintenance window

Actors: Admin/Department Head

Test Case Scenario:

1. Click “Current rooms in the database”

Post-Conditions: The system will display the current rooms in the system.

Module Name: Maintenance_Rooms_Add

Pre-Conditions: User has accessed the Room Maintenance window

Actors: Admin/Department Head

Test Case Scenario:

1. Click “Add new rooms”
2. Enter Room Name
3. Click “Submit”

Post-Conditions: The system will generate the new room, and will be displayed in the current rooms.

Module Name: Maintenance_Rooms_BatchAdd

Pre-Conditions: User has accessed the “Batch encoding for rooms” list

Actors: Admin/Department Head

Test Case Scenario:

1. Click “Batch encoding for rooms”
2. Enter number of rooms
3. Click “Submit”
4. Enter Room Names
5. Click “Submit”

Post-Conditions: The system must generate the rooms entered, and must be displayed in the current rooms.

Module Name: Maintenance_Rooms_Delete

Pre-Conditions: User has accessed the Room Maintenance window

Actors: Admin/Department Head

Test Case Scenario:

1. Click “Delete room”
2. Select room/s to be deleted.
3. Click “Delete”

Post-Conditions: The selected room/s will be deleted from the system, and will not appear in the “Current rooms in the database”.

Module Name: Maintenance_Faculty

Pre-Conditions: User has accessed the Faculty Maintenance window

Actors: Admin/Department Head

Test Case Scenario:

1. Click “Current faculty members in the database”

Post-Conditions: The system will display the current faculty data in the system.

Module Name: Maintenance_Faculty_Add

Pre-Conditions: User has accessed the Faculty Maintenance window

Actors: Admin/Department Head

Test Case Scenario:

1. Click “Add new faculty member”
2. Enter Faculty ID
3. Enter First Name
4. Enter Last Name
5. Select Department
6. Select Job Description
7. Select Position
8. Click “Submit”

Post-Conditions: The system will generate the new faculty member, and will be displayed in the current faculty members in the database.

Module Name: Maintenance_Faculty_BatchAdd

Pre-Conditions: User has accessed the "Batch encoding for faculty" list

Actors: Admin/Department Head

Test Case Scenario:

1. Click "Batch encoding for faculty"
2. Enter number of faculty
3. Click "Submit"
4. Enter faculty information
5. Click "Submit"

Post-Conditions: The system must generate the faculty entered, and must be displayed in the current faculty.

Module Name: Maintenance_Faculty_Delete

Pre-Conditions: User has accessed the Faculty Maintenance window

Actors: Admin/Department Head

Test Case Scenario:

1. Click "Delete faculty"
2. Select faculty to be deleted.
3. Click "Delete"

Post-Conditions: The selected faculty will be deleted from the system, and will not appear in the "Current faculty in the database".

Module Name: Maintenance_Courses

Pre-Conditions: User has accessed the Course Maintenance window

Actors: Admin/Department Head

Test Case Scenario:

1. Click "Current courses in the database"

Post-Conditions: The system will display the current courses in the system.

Module Name: Maintenance_Course_Add

Pre-Conditions: User has accessed the Course Maintenance window

Actors: Admin/Department Head

Test Case Scenario:

1. Click “Add new course”
2. Enter Course ID
3. Enter Course Description
4. Enter Lecture Units
5. Enter Lab Units
6. Click “Submit”

Post-Conditions: The system will generate the new course, and will be displayed in the current courses in the database.

Module Name: Maintenance_Course_BatchAdd

Pre-Conditions: User has accessed the “Batch encoding for course” list

Actors: Admin/Department Head

Test Case Scenario:

1. Click “Batch encoding for courses”
2. Enter number of courses
3. Click “Submit”
4. Enter course information
5. Click “Submit”

Post-Conditions: The system must generate the coures entered, and must be displayed in the current courses.

Module Name: Maintenance_Course_Delete

Pre-Conditions: User has accessed the Course Maintenance window

Actors: Admin/Department Head

Test Case Scenario:

1. Click “Delete course”
2. Select course/s to be deleted.
3. Click “Delete”

Post-Conditions: The selected course will be deleted from the system, and will not appear in the “Current courses in the database”.

Module Name: Maintenance_Section

Pre-Conditions: User has accessed the Section Maintenance window

Actors: Admin/Department Head

Test Case Scenario:

1. Click “Current sections in the database”

Post-Conditions: The system will display the current sections in the system.

Module Name: Maintenance_Section_Add

Pre-Conditions: User has accessed the Section Maintenance window

Actors: Admin/Department Head

Test Case Scenario:

1. Click “Add new section”
2. Enter Section
3. Select Department
4. Select Free Day
5. Click “Submit”

Post-Conditions: The system will generate the new section, and will be displayed in the current sections in the database.

Module Name: Maintenance_Section_BatchAdd

Pre-Conditions: User has accessed the "Batch encoding for sections" list

Actors: Admin/Department Head

Test Case Scenario:

1. Click "Batch encoding for section"
2. Enter number of sections
3. Click "Submit"
4. Enter section information
5. Click "Submit"

Post-Conditions: The system must generate the sections entered, and must be displayed in the current sections.

Module Name: Maintenance_Section_Delete

Pre-Conditions: User has accessed the Section Maintenance window

Actors: Admin/Department Head

Test Case Scenario:

1. Click "Delete section"
2. Select section/s to be deleted.
3. Click "Delete"

Post-Conditions: The selected section will be deleted from the system, and will not appear in the "Current sections in the database".

Module Name: Maintenance_User

Pre-Conditions: User has accessed the User Maintenance window

Actors: Admin

Test Case Scenario:

1. Click "Current users in the database"

Post-Conditions: The system will display the list of current users in the database.

Module Name: Maintenance_User_Pending

Pre-Conditions: User has accessed the User Maintenance window

Actors: Admin

Test Case Scenario:

1. Click “Pending users in the database”
2. Select pending user
3. Click “Update”

Post-Conditions: The system will generate a popup message stating that the user has been successfully updated or verified, and will be reflected in the current users in the database as “user”.

Module Name: Maintenance_User_Edit

Pre-Conditions: User has accessed the User Maintenance window

Actors: Admin

Test Case Scenario:

1. Click “Edit user access”
2. Select user
3. Click “Update”

Post-Conditions: The system will generate a popup message stating that the user has been updated.

Module Name: Maintenance_User_Delete

Pre-Conditions: User has accessed the User Maintenance window

Actors: Admin

Test Case Scenario:

1. Click “Delete user accounts”
2. Select account
3. Click “Delete”

Post-Conditions: The selected account/s must be deleted from the database, and must no longer be reflected in the list of user accounts.

Module Name: Maintenance_Requests

Pre-Conditions: User has accessed the Maintenance-Requests window

Actors: Admin

Test Case Scenario:

1. Click “Maintenance”
2. Click “Requests”

Post-Conditions: The admin must be able to see the list of reservation requests along with their respective information.

Module Name: Maintenance_Requests_Approve

Pre-Conditions: User has accessed the Maintenance-Requests window

Actors: Admin

Test Case Scenario:

1. Click “Maintenance”
2. Click “Requests”
3. Approve particular request

Post-Conditions: The system must generate a popup message stating that the request has been approved.

Module Name: Maintenance_Requests_Deny

Pre-Conditions: User has accessed the Maintenance-Requests window

Actors: Admin

Test Case Scenario:

1. Click “Maintenance”
2. Click “Requests”
3. Deny particular request

Post-Conditions: The system will generate a popup message stating that the reservation request has been denied.

Module Name: Maintenance_Archive

Pre-Conditions: User has accessed the Archive window

Actors: Admin

Test Case Scenario:

1. Select academic year
2. Select semester
3. Click “Archive Semester”
4. Click “OK” in popup message

Post-Conditions: The current schedule must be archived, and the calendar must return to its clean state.

Module Name: Scheduling_Room_Export

Pre-Conditions: User wants a softcopy of a room schedule

Actors: Department Head

Test Case Scenario:

1. Click “Scheduling”
2. Click “Room View”
3. Click “Export Excel”

Post-Conditions: The system must be able to download a softcopy of the room schedules.

Module Name: Scheduling_Faculty_Export

Pre-Conditions: User wants a softcopy of a faculty schedule

Actors: Department Head

Test Case Scenario:

1. Click “Scheduling
2. Click “Faculty View:
3. Click “Export Excel”

Post-Conditions: The system must be able to download a softcopy of the faculty schedules.

Module Name: Scheduling_Section_Export

Pre-Conditions: User wants a softcopy of a particular schedule

Actors: Department Head

Test Case Scenario:

1. Click “Schedule”
2. Click “Section View”
3. Click “Export Excel”

Post-Conditions: The system must be able to download a softcopy of the section schedules.

Module Name: Scheduling_Section_Add

Pre-Conditions: Department Head has accessed the Section View of the Scheduling module

Actors: Department Head

Test Case Scenario:

1. Click “Add Schedule”
2. Select Course
3. Select Room
4. Select Faculty
5. Select Section
6. Select Day
7. Select Start Time
8. Select End Time
9. Click “Add”

Post-Conditions: The system must be able to add the schedule in the Excel file.

Module Name: Requests

Pre-Conditions: User wants to view his/her requests

Actors: User/Admin/Department Head

Test Case Scenario:

1. Click “Your requests”

Post-Conditions: The system must be able to generate the list of pending, approved, denied, and cancelled reservation requests.

Module Name: Requests_Cancel

Pre-Conditions: User wants to cancel a reservation request

Actors: User/Admin/Department Head

Test Case Scenario:

1. Click “Your requests”
2. Click “cancel” on particular reservation request
3. Enter Reason for Cancellation
4. Click “Submit”

Post-Conditions: The system asks for the reason for cancellation of request. Once the user submits the form, a popup message will appear stating that the reservation request has been cancelled.

Chapter 5: Conclusion and Recommendations

Bibliography

1.3.1 What is MySQL? (n.d.). From MySQL: https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html

Andrade, D. (2015, March 18). *The Benefits of Educational Technology*. From Tech Learning: https://www.techlearning.com/tl-advisor-blog/9059

Babich, N. (2017, October 4). *Minimalistic Design With Large Impact: Functional Minimalism For Web Design*. From Smashing Magazine: https://www.smashingmagazine.com/2017/10/functional-minimal-web-design/

Calendar. (n.d.). From ORO Valley Bicycle: http://orovalleybicycle.com/calendar/

Calhoon, K. B. (2014, July 1). *Google Calendar and Gmail-- Class Reminders*. From Google Books: https://books.google.com.ph/books?id=IxRdBAAAQBAJ&printsec=frontcover&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false

Components of a Website. (n.d.). From StrateComm: http://www.stratecomm.com/faqs/components/

Darling, S. (2016, August 19). *UI Toolkit: The Benefits of User Interface Design*. From Graphic Mint: https://graphicmint.com/blog/user-interface-design/

Davidson, E. (2018). *What Is a Computerized Vs. Manual Scheduling System?* From Chron: https://smallbusiness.chron.com/computerized-vs-manual-scheduling-system-11956.html#

Demonstration of a Meeting Room Reservation System. (n.d.). From SuperSaaS: https://www.supersaas.com/schedule/demo/Meeting_Rooms

Dunn, N. (2016, May 5). *Accessing your Local Web Server from a Mobile Device using XAMPP*. From Webucator: https://www.webucator.com/blog/2016/05/accessing-your-local-web-server-from-a-mobile-device-using-xampp/

Gravelle, R. (n.d.). *Making the Switch from PhpExcel to PhpSpreadsheet*. From HTML Goodies: https://www.htmlgoodies.com/beyond/making-the-switch-from-phpexcel-to-phpspreadsheet.html

Jacobs, M. (2016, January 13). *Real time calculation with PHP & array*. From Stack Overflow: https://stackoverflow.com/questions/34770788/real-time-calculation-with-php-array

Kelly-Barton, C. (2018, March 15). *10 Essential Components of A Successful Website*. From HostGator: https://www.hostgator.com/blog/essential-components-successful-website/

Marah, B. J. (2018, Febuary). *Mastering Bootstrap 4 Second Edition*. From Google Books: https://books.google.com.ph/books?id=NpZNDwAAQBAJ&printsec=frontcover&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false

Pandey, R. (2015, December 3). *Data Verification is an Important Factor for Retaining Customers*. From Iinkedin: https://www.linkedin.com/pulse/data-verification-important-factor-retaining-customers-rajeev-pandey

PowerKiKi. (2019, January 2). *PHPExcel*. From GitHub: https://github.com/PHPOffice/PHPExcel

Rutherford, Z. (2018, February 27). *Fixed Navigation Bars: Pros and Cons*. From awwwards: https://www.awwwards.com/fixed-navigation-bars-pros-and-cons.html

slicedtoad. (2015, April 1). *Simple Repeating Events*. From Stack Overflow: https://stackoverflow.com/questions/15161654/recurring-events-in-fullcalendar

Sloan, K. (2018, Febuary 2). *4 Benefits of Using a Scheduling System in Your Business*. From Calendar: https://www.calendar.com/blog/scheduling-system-business/

Stapler, R. (2016, April 13). *PHPExcel Tutorial - How to install PHPExcel*. From Youtube: https://www.youtube.com/watch?v=r6tg2sySwto

The benefits of web-based applications. (n.d.). From Magic Web Solutions: https://www.magicwebsolutions.co.uk/blog/the-benefits-of-web-based-applications.htm

Tom. (2011, May 3). *jQuery KeyUp and Click*. From Stack Overflow: https://stackoverflow.com/questions/5871206/jquery-keyup-and-click

Tooltips. (n.d.). From Bootstrap: https://getbootstrap.com/docs/4.0/components/tooltips/

user3263266. (2014, February 5). *Want to show/hide div based on dropdown box selection*. From Stack Overflow: https://stackoverflow.com/questions/21584752/want-to-show-hide-div-based-on-dropdown-box-selection

Why do we use PHP? (n.d.). From Blue Line Media: https://www.bluelinemedia.co.uk/blog/entry/web-design/blog/why-php

Appendices

Relevant Source Code

Register

```
<?php
session_start();
$servername = "localhost";
$username = "Admin";  //your user name for php my admin
$password = "Admin123";  //password
$databasename = "schedulereservation"; //db name
//$uName=$_SESSION["uName"];
$conn = new mysqli($servername, $username, $password, $databasename) or
die(mysqli_error()); //Connect to server

$trim2a = str_replace('<', '', $_POST['eMail']);
$trim1a = str_replace('>', '', $trim2a);
$email = str_replace(' ', '', $trim1a);

$trim2b = str_replace('<', '', $_POST['uName']);
$trim1b = str_replace('>', '', $trim2b);
$uname = str_replace(' ', '', $trim1b);

$trim2c = str_replace('<', '', $_POST['pWord']);
$trim1c = str_replace('>', '', $trim2c);
$pword = str_replace(' ', '', $trim1c);

$trim2d = str_replace('<', '', $_POST['cpWord']);
$trim1d = str_replace('>', '', $trim2d);
$cpword = str_replace(' ', '', $trim1d);

$dept = ($_POST['department']);
$Dhead = ($_POST['Dhead']);
$No = "No";

require 'PHPMailerAutoload.php';

date_default_timezone_set("Asia/Manila");
$date=date("M/d/Y");
$time=date("H:i:s");

$query= mysqli_query($conn, "SELECT * FROM register WHERE email = '$email'");
$exist = mysqli_num_rows($query);

$query= mysqli_query($conn, "SELECT * FROM register WHERE username = '$uname'");
$usercheck = mysqli_num_rows($query);

if($exist <= 0)//check if email exists
{
```

```
 if($usercheck <= 0)//check if username exists
 {
  if($pword == $cpword) //if password matches confirm password
  {
   if(!preg_match("#[0-9]+#",$pword))
   { //checks password if it has numbers
      Print '<script>alert("Your Password Must Contain At Least 1 Number!");</script>';
//Prompts the user
      Print '<script>window.location.assign("index.php");</script>'; // redirects to login.php
   }
   elseif(!preg_match("#[A-Z]+#",$pword))
   { //checks password if it has uppercase letter
      Print '<script>alert("Your Password Must Contain At Least 1 Capital Letter!");</script>';
//Prompts the user
      Print '<script>window.location.assign("index.php");</script>'; // redirects to login.php
   }
   elseif(!preg_match("#[a-z]+#",$pword))
   { //checks password if it has lowercase letter
      Print '<script>alert("Your Password Must Contain At Least 1 Lowercase
Letter!");</script>'; //Prompts the user
      Print '<script>window.location.assign("index.php");</script>'; // redirects to login.php
   }
   elseif(strlen($pword)<8)
   { //checks password if it is at least 8 characters long
      Print '<script>alert("Your Password Must Be at Least 8 Characters!");</script>'; //Prompts
the user
      Print '<script>window.location.assign("index.php");</script>'; // redirects to login.php
   }
   elseif($Dhead == "Yes")
   {
    $pass_encrypt = base64_encode($pword);
    mysqli_select_db($conn,"schedulereservation") or die("Cannot connect to database");
//Connect to database
    mysqli_query($conn, "INSERT INTO `register`(`email`, `pass`, `username`, `access`,
`department`, `departmentHead`, `dateCreated`, `timeCreated`) VALUES
('$email','$pass_encrypt','$uname','newuser','$dept', 'Yes', '$date', '$time')"); //insert record into
database
    Print '<script>alert("Your request has been successfully sent to the admin for
verification.");</script>';
    Print '<script>window.location.assign("index.php");</script>'; // redirects to login

    $mail = new PHPMailer;

    $mail->SMTPDebug = 0;                          // Enable verbose debug output

    $mail->isSMTP();                               // Set mailer to use SMTP
    $mail->Host = 'smtp.gmail.com';  // Specify main and backup SMTP servers
    $mail->SMTPAuth = true;                        // Enable SMTP authentication
    $mail->Username = 'IICS.SUPREMACY@gmail.com';          // SMTP username
    $mail->Password = 'bonak_kid';                 // SMTP password
    $mail->SMTPSecure = 'tls';                     // Enable TLS encryption, `ssl` also accepted
```

```
        $mail->Port = 587;                                    // TCP port to connect to

        $mail->setFrom('IICS.SUPREMACY@gmail.com', 'IICS');  //sender
        $mail->addAddress($email, 'User!');     // reciever
        //$mail->addAddress('ellen@example.com');               // Name is optional
        //$mail->addReplyTo('christian.academics.ust@gmail.com');
        //$mail->addCC('cc@example.com');
        //$mail->addBCC('bcc@example.com');

        //$mail->addAttachment('/var/tmp/file.tar.gz');         // Add attachments
        //$mail->addAttachment('/tmp/image.jpg', 'new.jpg');    // Optional name
        $mail->isHTML(true);                                  // Set email format to HTML

        $mail->Subject = 'User account creation';
        $mail->Body    = 'Hello, '.$uname.'<br><br>
        This is to inform you that your user account request is on the way. Form contents are as
follows: <br><br>
        EMAIL: '. $email .' <br>
        USERNAME: '. $uname .' <br>
        DATE REQUESTED: '. $date .' <br>
        TIME REQUESTED: '. $time .'<br>
        DEPARTMENT: '. $dept .'<br><br>
        Please wait for the admin to approve your account';
        //$mail->AltBody = 'This is the body in plain text for non-HTML mail clients';
        if(!$mail->send())
        {
           echo 'Message could not be sent.';
           echo 'Mailer Error: ' . $mail->ErrorInfo;
        } else
        {
           echo 'Message has been sent';
        }
      }
      elseif($Dhead == "No")
      {
        $pass_encrypt = base64_encode($pword);
        mysqli_select_db($conn,"schedulereservation") or die("Cannot connect to database");
//Connect to database
        mysqli_query($conn, "INSERT INTO `register`(`email`, `pass`, `username`, `access`,
`department`, `departmentHead`, `dateCreated`, `timeCreated`) VALUES
('$email','$pass_encrypt','$uname','newuser','$dept', 'No', '$date', '$time')"); //insert record into
database
        Print '<script>alert("Your request has been successfully sent to the admin for
verification.");</script>';
        Print '<script>window.location.assign("index.php");</script>'; // redirects to login

        $mail = new PHPMailer;

        $mail->SMTPDebug = 0;                                 // Enable verbose debug output

        $mail->isSMTP();                                      // Set mailer to use SMTP
```

```
        $mail->Host = 'smtp.gmail.com';  // Specify main and backup SMTP servers
        $mail->SMTPAuth = true;                               // Enable SMTP authentication
        $mail->Username = 'IICS.SUPREMACY@gmail.com';                 // SMTP username
        $mail->Password = 'bonak_kid';                           // SMTP password
        $mail->SMTPSecure = 'tls';                            // Enable TLS encryption, `ssl` also accepted
        $mail->Port = 587;                                    // TCP port to connect to

        $mail->setFrom('IICS.SUPREMACY@gmail.com', 'IICS');  //sender
        $mail->addAddress($email, 'User');     // reciever
        //$mail->addAddress('ellen@example.com');               // Name is optional
        //$mail->addReplyTo('christian.academics.ust@gmail.com');
        //$mail->addCC('cc@example.com');
        //$mail->addBCC('bcc@example.com');

        //$mail->addAttachment('/var/tmp/file.tar.gz');         // Add attachments
        //$mail->addAttachment('/tmp/image.jpg', 'new.jpg');    // Optional name
        $mail->isHTML(true);                                  // Set email format to HTML

        $mail->Subject = 'User account creation';
        $mail->Body    = 'Hello, '.$uname.'<br><br>
        This is to inform you that your user account request is on the way. Form contents are as
follows:<br><br>
        EMAIL: '. $email .' <br>
        USERNAME: '. $uname .' <br>
        DATE REQUESTED: '. $date .' <br>
        TIME REQUESTED: '. $time .'<br>
        DEPARTMENT: '. $dept .'<br><br>
        Please wait for the admin to approve your account';
        //$mail->AltBody = 'This is the body in plain text for non-HTML mail clients';
        if(!$mail->send())
        {
           echo 'Message could not be sent.';
           echo 'Mailer Error: ' . $mail->ErrorInfo;
        }
        else
        {
           echo 'Message has been sent';
        }
      }
    }
     else
     {
       Print '<script>alert("Please make sure the credentials match. Try again.");</script>'; //Prompts
the user
       Print '<script>window.location.assign("index.php");</script>'; // redirects to login.php
     }
   }
   else{
     Print '<script>alert("User Already Exists. Try again.");</script>'; //Prompts the user
     Print '<script>window.location.assign("index.php");</script>'; // redirects to login.php
   }
```

```
}
else
{
  Print '<script>alert("User Already Exists. Try again.");</script>'; //Prompts the user
  Print '<script>window.location.assign("index.php");</script>'; // redirects to login.php
}
 ?>
```

Reservation

```
<?php
session_start();
$servername = "localhost";
$username = "Admin";  //your user name for php my admin
$password = "Admin123";  //password
$databasename = "schedulereservation"; //db name
$uName=$_SESSION["uName"];
$mail = $_SESSION["emailadd"];
$conn = new mysqli($servername, $username, $password, $databasename) or
die(mysqli_error()); //Connect to server
if (empty($_SESSION['uName'])) {
    header('Location: index.php');
    exit;
}
try
{
  $bdd = new PDO('mysql:host=localhost;dbname=schedulereservation;charset=utf8', 'root', '');
}
catch(Exception $e)
{
        die('Error : '.$e->getMessage());
}
//$sessionVar = $_SESSION['room_name'];
$test = $_SESSION["roomName"];
$scheduling = "SELECT * FROM scheduling WHERE room = '$test'";
$sql = "SELECT * FROM events";

if (empty($_SESSION["roomName"])) {
  $sql2 = "SELECT * FROM events";
  $scheduling2 = "SELECT * FROM scheduling";
  $test = "All Rooms";
}
elseif ($test == "All Rooms") {
  $sql2 = "SELECT * FROM events";
  $scheduling2 = "SELECT * FROM scheduling";
}
else {
  $sql2 = "SELECT * FROM events WHERE room = '$test'";
  $scheduling2 = "SELECT * FROM scheduling WHERE room = '$test'";
}
$schedreq = $bdd->prepare($scheduling);
$schedreq->execute();
```

```
$scheds = $schedreq->fetchAll();
$schedreq2 = $bdd->prepare($scheduling2);
$schedreq2->execute();
$scheds2 = $schedreq2->fetchAll();
$req = $bdd->prepare($sql);
$req->execute();
$req2 = $bdd->prepare($sql2);
$req2->execute();
$events = $req->fetchAll();
$events2 = $req2->fetchAll();
$status = "SELECT * FROM events WHERE status ='pending'";
$variable = "<script>event.id</script>";
//$pending = "INSERT INTO events (status) VALUES ('pending') WHERE SELECT * FROM
events WHERE id = $variable";
?>

<!DOCTYPE html>
<html lang="en">
<head>
  <title>Home</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
  <link rel="stylesheet" href="styles/styles.css">
  <link rel="stylesheet" href="fullcalendar/fullcalendar.css">
  <link href="https://fonts.googleapis.com/css?family=Raleway|Ubuntu" rel="stylesheet">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
  <!-- Google Calendar FrontEnd START-->
  <script src="fullcalendar/lib/moment.min.js"></script>
  <script src="fullcalendar/gcal.js"></script>
  <link href='fullcalendar/fullcalendar.css' rel='stylesheet' />
  <link href='fullcalendar-scheduler/scheduler.css' rel='stylesheet' />
  <script src='https://momentjs.com/downloads/moment.min.js'></script>
  <script src='fullcalendar/fullcalendar.js'></script>
  <script src='fullcalendar-scheduler/scheduler.js'></script>

<!-- googleCalendarApiKey: 'AIzaSyCSv839KzO6s20ar3ayut5CW24y8Cczz_Q', -->
<!-- schedulerLicenseKey: 'CC-Attribution-NonCommercial-NoDerivatives', -->
  <script>
  $(document).ready(function() {
    var date = new Date()
    var d    = date.getDate(),
        m    = date.getMonth(),
        y    = date.getFullYear()
        $('#calendar').fullCalendar({
          schedulerLicenseKey: 'CC-Attribution-NonCommercial-NoDerivatives',
          header: {
            left: 'prev,next today',
            center: 'title',
```

```
          right: 'month,agendaWeek,agendaDay'
        },
        allDaySlot: false,
        defaultDate: new Date(y, m, d),
        editable: false,
        eventLimit: 6, // allow "more" link when too many events
        selectable: true,
        selectHelper: true,
        timeFormat: 'hh:mm t',
        eventOverlap: false,
        //selectOverlap: false,
        resourceEventOverlap:false,
        themeSystem: 'bootstrap3',
        minTime: '07:00',
        maxTime: '22:00',
        displayEventEnd: true,

        eventLimitClick: 'agendaDay',
        eventLimitText: 'more events',
        views: {
          agendaDay: { eventLimit: true }
        },
        eventRender: function(event, $el) {
          $el.popover({
            title: event.title,
            content: event.room + "<br>" + event.course + "<br>" + event.section + "<br>" +
event.department,
            trigger: 'hover',
            html: 'true',
            placement: 'top',
            container: 'body',
          });
        },
          select: function(start, end, jsEvent, view) {
             if(end.isBefore(moment().add(1,'hour').format())) {
             $('#calendar').fullCalendar('unselect');
             return false;             }

          else {
            $('#ModalAdd #start').val(moment(start).format('YYYY-MM-DD'));
            $('#ModalAdd #end').val(moment(end).format('YYYY-MM-DD'));
            $('#ModalAdd').modal('show');
          }
        },
//mark
        function(event, dayDelta, minuteDelta, allDay, revertFunc, jsEvent, ui, view) {
            /// deny overlap of event
            var start = new Date(event.start);
            var end = new Date(event.end);

            var overlap = $('#calendar').fullCalendar('clientEvents', function(ev) {
```

```
            if( ev == event) {
               return false;
            }
            var estart = new Date(ev.start);
            var eend = new Date(ev.end);

            return (
               ( Math.round(start) > Math.round(estart) && Math.round(start) < Math.round(eend)
)
               ||
               ( Math.round(end) > Math.round(estart) && Math.round(end) < Math.round(eend) )
               ||
               ( Math.round(start) < Math.round(estart) && Math.round(end) > Math.round(eend)
)
            );
         });
         if (overlap.length){
            revertFunc();
            return false;
         }
      },
//end mark

      events: [
        <?php foreach ($scheds as $sched):
          $q1 = $sched['course'];
          $q2 = $sched['time'];
          $q3 = $sched['Etime'];
          $q4 = $sched['day'];
          $q5 = $sched['section'];
          $q6 = $sched['course'];
          $q7 = $sched['room'];

             if ($q4 == "Monday")    { $day = "[1]"; }
          else if ($q4 == "Tuesday")   { $day = "[2]"; }
          else if ($q4 == "Wednesday") { $day = "[3]"; }
          else if ($q4 == "Thursday")  { $day = "[4]"; }
          else if ($q4 == "Friday")    { $day = "[5]"; }
          else if ($q4 == "Saturday")  { $day = "[6]"; }
        ?>
        {
        title: "<?php echo $q1; ?>" + " - " + "<?php echo $q5; ?>",
        section: '<?php echo $q5; ?>',
        course: '<?php echo $q6; ?>',
        room: '<?php echo $q7; ?>',
        department: '<?php echo ""; ?>',
        start: '<?php echo $q2; ?>',
        end: '<?php echo $q3; ?>',
        color: '<?php echo '#26ab79'; ?>',
        dow: <?php echo $day; ?>
        },
```

```
          <?php endforeach; ?>

          <?php foreach($events2 as $event): //from reservation
            $start = explode(" ", $event['start']);
            $end = explode(" ", $event['end']);

            if($start[1] == '00:00:00') {
              $start = $start[0];
            }
            else {
              $start = $event['start'];
            }
            if($end[1] == '00:00:00') {
              $end = $end[0];
            }
            else {
              $end = $event['end'];
            }
          ?>
          {
            id: '<?php echo $event['id']; ?>',
            title: '<?php echo $event['title']; ?>',
            course: '<?php echo $event['course']; ?>',
            section: '<?php echo $event['section']; ?>',
            department: '<?php echo $event['department']; ?>',
            start: '<?php echo $start; ?>',
            end: '<?php echo $end; ?>',
            color: '<?php echo '#F00' ?>',
            room: '<?php echo $event['room']; ?>',
          },
          <?php endforeach; ?>
        ]
      });
});
</script>
</head>

<!-- Modal -->
<div class="modal fade" id="ModalAdd" tabindex="-1" role="dialog" aria-
labelledby="myModalLabel">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <form class="form-horizontal" method="POST" action="addEvent.php" onsubmit="return
confirm('Your request will be sent for approval. Continue?');">
        <div class="modal-header">
          <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span aria-
hidden="true">&times;</span></button>
          <h4 class="modal-title" id="myModalLabel">New Reservation Request</h4>
        </div>

        <div class="modal-body">
```

```
        <div class="form-group">
          <label for="request" class="col-sm-2 control-label">Title</label>
          <div class="col-sm-10">
            <input type="text" name="title" class="form-control" id="title" placeholder="Title of
the Request" required="required" autocomplete="off">
          </div>
        </div>

        <div class="form-group">
          <label for="department" class="col-sm-2 control-label">Department</label>
          <div class="col-sm-10">
            <?php
              $query2 = $conn->prepare("SELECT * FROM department"); // displays all requests
with same project as user
              $query2->execute(); // actually perform the query
              $result2 = $query2->get_result(); // retrieve the result so it can be used inside PHP
              $r2 = $result2->fetch_array(MYSQLI_ASSOC); // bind the data from the first result
row to $r
                if ($result2 -> num_rows > 0)
                {
                  echo '<select name="department" class="form-control" id="department"
required="required">
                  <option disabled value="" selected hidden>Select one</option>';//column heads
                  do
                  {
                    echo "<option>".$r2['department']."</option>";//DISPLAY TITLE
                  }
                  while ($r2 = $result2 -> fetch_assoc());
                  echo '</select>';
                }
                else
                {
                  echo 'EMPTY';
                }
              ?>
          </div>
        </div>

        <div class="form-group">
          <label for="section" class="col-sm-2 control-label">Section</label>
          <div class="col-sm-10">
            <?php
              $query2 = $conn->prepare("SELECT * FROM section"); // displays all requests with
same project as user
              $query2->execute(); // actually perform the query
              $result2 = $query2->get_result(); // retrieve the result so it can be used inside PHP
              $r2 = $result2->fetch_array(MYSQLI_ASSOC); // bind the data from the first result
row to $r
                if ($result2 -> num_rows > 0)
                {
```

```
                    echo '<select name="section" class="form-control" id="section"
required="required">
                    <option disabled value="" selected hidden>Select one</option>';//column heads
                    do
                    {
                        echo "<option>".$r2['section']."</option>";//DISPLAY TITLE
                    }
                    while ($r2 = $result2 -> fetch_assoc());
                    echo '</select>';
                }
                else
                {
                    echo 'EMPTY';
                }
              ?>
          </div>
        </div>

        <div class="form-group">
          <label for="course" class="col-sm-2 control-label">Course</label>
          <div class="col-sm-10">
            <?php
              $query2 = $conn->prepare("SELECT * FROM course"); // displays all requests with
same project as user
              $query2->execute(); // actually perform the query
              $result2 = $query2->get_result(); // retrieve the result so it can be used inside PHP
              $r2 = $result2->fetch_array(MYSQLI_ASSOC); // bind the data from the first result
row to $r
                if ($result2 -> num_rows > 0)
                {
                    echo '<select name="course" class="form-control" id="course"
required="required">
                    <option disabled value="" selected hidden>Select one</option>';//column heads
                    do
                    {
                        echo "<option>".$r2['cid']."</option>";//DISPLAY TITLE
                    }
                    while ($r2 = $result2 -> fetch_assoc());
                    echo '</select>';
                }
                else
                {
                    echo 'EMPTY';
                }
              ?>
          </div>
        </div>

        <div class="form-group">
          <label for="color" class="col-sm-2 control-label">Purpose</label>
          <div class="col-sm-10">
```

```
            <select name="color" class="form-control" id="color" required="required">
              <option disabled value="" selected hidden>Select one</option>
              <option style="color:#40E0D0;" value="Make-up-Classes">&#9724; Make-up
Classes</option>
              <option style="color:#008000;" value="Org Groups">&#9724; Org Groups</option>
              <option style="color:#FFD700;" value="Meetings">&#9724; Meetings</option>
              <option style="color:#FF8C00;" value="Others">&#9724; Others</option>
              <option style="color:#FF0000;" value="Emergency">&#9724; Emergency</option>
            </select>
          </div>
        </div>

        <div class="form-group">
          <label for="department" class="col-sm-2 control-label">Room</label>
          <div class="col-sm-10">
            <?php
              $query2 = $conn->prepare("SELECT * FROM room"); // displays all requests with
same project as user
              $query2->execute(); // actually perform the query
              $result2 = $query2->get_result(); // retrieve the result so it can be used inside PHP
              $r2 = $result2->fetch_array(MYSQLI_ASSOC); // bind the data from the first result
row to $r
                if ($result2 -> num_rows > 0)
                {
                  echo '<select name="room" class="form-control" id="room" required="required">
                  <option disabled value="" selected hidden>Select one</option>';//column heads
                  do
                  {
                    echo "<option>".$r2['rname']."</option>";//DISPLAY TITLE
                  }
                  while ($r2 = $result2 -> fetch_assoc());
                  echo '</select>';
                }
                else
                {
                  echo 'EMPTY';
                }
              ?>
            </select>
          </div>
        </div>

        <div class="form-group">
        <label for="startTime" class="col-sm-2 control-label">Start Time</label>
        <div class="col-sm-3">
          <select name="startTime" class="form-control" id="startTime" onchange="run()">
            <option disabled value="" selected>Select one</option>
            <option value="07l00">7:00 AM</option>
            <option value="07l30">7:30 AM</option>
            <option value="08l00">8:00 AM</option>
            <option value="08l30">8:30 AM</option>
```

```
            <option value="09l00">9:00 AM</option>
            <option value="09l30">9:30 AM</option>
            <option value="10l00">10:00 AM</option>
            <option value="10l30">10:30 AM</option>
            <option value="11l00">11:00 AM</option>
            <option value="11l30">11:30 AM</option>
            <option value="12l00">12:00 PM</option>
            <option value="12l30">12:30 PM</option>
            <option value="13l00">1:00 PM</option>
            <option value="13l30">1:30 PM</option>
            <option value="14l00">2:00 PM</option>
            <option value="14l30">2:30 PM</option>
            <option value="15l00">3:00 PM</option>
            <option value="15l30">3:30 PM</option>
            <option value="16l00">4:00 PM</option>
            <option value="16l30">4:30 PM</option>
            <option value="17l00">5:00 PM</option>
            <option value="17l30">5:30 PM</option>
            <option value="18l00">6:00 PM</option>
            <option value="18l30">6:30 PM</option>
            <option value="19l00">7:00 PM</option>
            <option value="19l30">7:30 PM</option>
            <option value="20l00">8:00 PM</option>
            <option value="20l30">8:30 PM</option>
          </select>
        </div>
      </div>

      <div class="form-group">
        <label for="endTime" class="col-sm-2 control-label">End Time</label>
        <div class="col-sm-3">
          <select name="endTime" class="form-control" id="endTime" onchange="run()">
            <option disabled value="" selected hidden>Select one</option>
          </select>
        </div>
      </div>

      <div class="form-group">
        <label for="start" class="col-sm-2 control-label">Start date</label>
        <div class="col-sm-10">
          <input type="date" id="start" name="start" value="2000-05-05" readonly>
        </div>
      </div>

      <script>
        function run() {
          if (document.getElementById("startTime").value ==
document.getElementById("endTime").value) {
          $('#request').attr('disabled','disabled');
        }
```

```
        else if (document.getElementById("startTime").value >=
document.getElementById("endTime").value) {
        $('#request').attr('disabled','disabled');
       }
        else if (document.getElementById("startTime").value == 0 ||
document.getElementById("endTime").value == 0) {
        $('#request').attr('disabled','disabled');
       }
        else if (document.getElementById("startTime").value < "07l00" &&
document.getElementById("endTime").value > "10l00") {
        $('#request').attr('disabled','disabled');
       }
        else {
        $('#request').removeAttr('disabled');
       }
       $('#startTime').on('change', function(){
       $('#endTime').html('');
       if($('#startTime').val()=='07l00'){
          $('#endTime').append('<option value="07l30">7:30 AM</option>');
          $('#endTime').append('<option value="08l00">8:00 AM</option>');
          $('#endTime').append('<option value="08l30">8:30 AM</option>');
          $('#endTime').append('<option value="09l00">9:00 AM</option>');
          $('#endTime').append('<option value="09l30">9:30 AM</option>');
          $('#endTime').append('<option value="10l00">10:00 AM</option>');
       }
       else if($('#startTime').val()=='07l30'){
        $('#endTime').append('<option value="08l00">8:00 AM</option>');
        $('#endTime').append('<option value="08l30">8:30 AM</option>');
        $('#endTime').append('<option value="09l00">9:00 AM</option>');
        $('#endTime').append('<option value="09l30">9:30 AM</option>');
        $('#endTime').append('<option value="10l00">10:00 AM</option>');
        $('#endTime').append('<option value="10l30">10:30 AM</option>');
       }
       else if($('#startTime').val()=='08l00'){
        $('#endTime').append('<option value="08l30">8:30 AM</option>');
        $('#endTime').append('<option value="09l00">9:00 AM</option>');
        $('#endTime').append('<option value="09l30">9:30 AM</option>');
        $('#endTime').append('<option value="10l00">10:00 AM</option>');
        $('#endTime').append('<option value="10l30">10:30 AM</option>');
        $('#endTime').append('<option value="11l00">11:00 AM</option>');
       }
       else if($('#startTime').val()=='08l30'){
        $('#endTime').append('<option value="09l00">9:00 AM</option>');
        $('#endTime').append('<option value="09l30">9:30 AM</option>');
        $('#endTime').append('<option value="10l00">10:00 AM</option>');
        $('#endTime').append('<option value="10l30">10:30 AM</option>');
        $('#endTime').append('<option value="11l00">11:00 AM</option>');
        $('#endTime').append('<option value="11l30">11:30 AM</option>');
       }
       else if($('#startTime').val()=='09l00'){
        $('#endTime').append('<option value="09l30">9:30 AM</option>');
```

```
  $('#endTime').append('<option value="10l00">10:00 AM</option>');
  $('#endTime').append('<option value="10l30">10:30 AM</option>');
  $('#endTime').append('<option value="11l00">11:00 AM</option>');
  $('#endTime').append('<option value="11l30">11:30 AM</option>');
  $('#endTime').append('<option value="12l00">12:00 PM</option>');
}
else if($('#startTime').val()=='09l30'){
  $('#endTime').append('<option value="10l00">10:00 AM</option>');
  $('#endTime').append('<option value="10l30">10:30 AM</option>');
  $('#endTime').append('<option value="11l00">11:00 AM</option>');
  $('#endTime').append('<option value="11l30">11:30 AM</option>');
  $('#endTime').append('<option value="12l00">12:00 PM</option>');
  $('#endTime').append('<option value="12l30">12:30 PM</option>');
}
else if($('#startTime').val()=='10l00'){
  $('#endTime').append('<option value="10l30">10:30 AM</option>');
  $('#endTime').append('<option value="11l00">11:00 AM</option>');
  $('#endTime').append('<option value="11l30">11:30 AM</option>');
  $('#endTime').append('<option value="12l00">12:00 PM</option>');
  $('#endTime').append('<option value="12l30">12:30 PM</option>');
  $('#endTime').append('<option value="13l00">1:00 PM</option>');
}
else if($('#startTime').val()=='10l30'){
  $('#endTime').append('<option value="11l00">11:00 AM</option>');
  $('#endTime').append('<option value="11l30">11:30 AM</option>');
  $('#endTime').append('<option value="12l00">12:00 PM</option>');
  $('#endTime').append('<option value="12l30">12:30 PM</option>');
  $('#endTime').append('<option value="13l00">1:00 PM</option>');
  $('#endTime').append('<option value="13l30">1:30 PM</option>');
}
else if($('#startTime').val()=='11l00'){
  $('#endTime').append('<option value="11l30">11:30 AM</option>');
  $('#endTime').append('<option value="12l00">12:00 PM</option>');
  $('#endTime').append('<option value="12l30">12:30 PM</option>');
  $('#endTime').append('<option value="13l00">1:00 PM</option>');
  $('#endTime').append('<option value="13l30">1:30 PM</option>');
  $('#endTime').append('<option value="14l00">2:00 PM</option>');
}
else if($('#startTime').val()=='11l00'){
  $('#endTime').append('<option value="11l30">11:30 AM</option>');
  $('#endTime').append('<option value="12l00">12:00 PM</option>');
  $('#endTime').append('<option value="12l30">12:30 PM</option>');
  $('#endTime').append('<option value="13l00">1:00 PM</option>');
  $('#endTime').append('<option value="13l30">1:30 PM</option>');
  $('#endTime').append('<option value="14l00">2:00 PM</option>');
}
else if($('#startTime').val()=='11l30'){
  $('#endTime').append('<option value="12l00">12:00 PM</option>');
  $('#endTime').append('<option value="12l30">12:30 PM</option>');
  $('#endTime').append('<option value="13l00">1:00 PM</option>');
  $('#endTime').append('<option value="13l30">1:30 PM</option>');
```

```
  $('#endTime').append('<option value="14l00">2:00 PM</option>');
  $('#endTime').append('<option value="14l30">2:30 PM</option>');
}
else if($('#startTime').val()=='12l00'){
  $('#endTime').append('<option value="13l00">1:00 PM</option>');
  $('#endTime').append('<option value="13l30">1:30 PM</option>');
  $('#endTime').append('<option value="14l00">2:00 PM</option>');
  $('#endTime').append('<option value="14l30">2:30 PM</option>');
  $('#endTime').append('<option value="15l00">3:00 PM</option>');
}
else if($('#startTime').val()=='12l30'){
  $('#endTime').append('<option value="13l30">1:30 PM</option>');
  $('#endTime').append('<option value="14l00">2:00 PM</option>');
  $('#endTime').append('<option value="14l30">2:30 PM</option>');
  $('#endTime').append('<option value="15l00">3:00 PM</option>');
  $('#endTime').append('<option value="15l30">3:30 PM</option>');
}
else if($('#startTime').val()=='13l00'){
  $('#endTime').append('<option value="14l00">2:00 PM</option>');
  $('#endTime').append('<option value="14l30">2:30 PM</option>');
  $('#endTime').append('<option value="15l00">3:00 PM</option>');
  $('#endTime').append('<option value="15l30">3:30 PM</option>');
  $('#endTime').append('<option value="16l00">4:00 PM</option>');
}
else if($('#startTime').val()=='13l30'){
  $('#endTime').append('<option value="14l30">2:30 PM</option>');
  $('#endTime').append('<option value="15l00">3:00 PM</option>');
  $('#endTime').append('<option value="15l30">3:30 PM</option>');
  $('#endTime').append('<option value="16l00">4:00 PM</option>');
  $('#endTime').append('<option value="16l30">4:30 PM</option>');
}
else if($('#startTime').val()=='14l00'){
  $('#endTime').append('<option value="15l00">3:00 PM</option>');
  $('#endTime').append('<option value="15l30">3:30 PM</option>');
  $('#endTime').append('<option value="16l00">4:00 PM</option>');
  $('#endTime').append('<option value="16l30">4:30 PM</option>');
  $('#endTime').append('<option value="17l00">5:00 PM</option>');
}
else if($('#startTime').val()=='14l30'){
  $('#endTime').append('<option value="15l30">3:30 PM</option>');
  $('#endTime').append('<option value="16l00">4:00 PM</option>');
  $('#endTime').append('<option value="16l30">4:30 PM</option>');
  $('#endTime').append('<option value="17l00">5:00 PM</option>');
  $('#endTime').append('<option value="17l30">5:30 PM</option>');
}
else if($('#startTime').val()=='15l00'){
  $('#endTime').append('<option value="16l00">4:00 PM</option>');
  $('#endTime').append('<option value="16l30">4:30 PM</option>');
  $('#endTime').append('<option value="17l00">5:00 PM</option>');
  $('#endTime').append('<option value="17l30">5:30 PM</option>');
  $('#endTime').append('<option value="18l00">6:00 PM</option>');
```

```
}
else if($('#startTime').val()=='15l30'){
  $('#endTime').append('<option value="16l30">4:30 PM</option>');
  $('#endTime').append('<option value="17l00">5:00 PM</option>');
  $('#endTime').append('<option value="17l30">5:30 PM</option>');
  $('#endTime').append('<option value="18l00">6:00 PM</option>');
  $('#endTime').append('<option value="18l30">6:30 PM</option>');
}
else if($('#startTime').val()=='16l00'){
  $('#endTime').append('<option value="17l00">5:00 PM</option>');
  $('#endTime').append('<option value="17l30">5:30 PM</option>');
  $('#endTime').append('<option value="18l00">6:00 PM</option>');
  $('#endTime').append('<option value="18l30">6:30 PM</option>');
  $('#endTime').append('<option value="19l00">7:00 PM</option>');
}
else if($('#startTime').val()=='16l30'){
  $('#endTime').append('<option value="17l30">5:30 PM</option>');
  $('#endTime').append('<option value="18l00">6:00 PM</option>');
  $('#endTime').append('<option value="18l30">6:30 PM</option>');
  $('#endTime').append('<option value="19l00">7:00 PM</option>');
  $('#endTime').append('<option value="19l30">7:30 PM</option>');
}
else if($('#startTime').val()=='17l00'){
  $('#endTime').append('<option value="18l00">6:00 PM</option>');
  $('#endTime').append('<option value="18l30">6:30 PM</option>');
  $('#endTime').append('<option value="19l00">7:00 PM</option>');
  $('#endTime').append('<option value="19l30">7:30 PM</option>');
  $('#endTime').append('<option value="20l00">8:00 PM</option>');
}
else if($('#startTime').val()=='17l30'){
  $('#endTime').append('<option value="18l30">6:30 PM</option>');
  $('#endTime').append('<option value="19l00">7:00 PM</option>');
  $('#endTime').append('<option value="19l30">7:30 PM</option>');
  $('#endTime').append('<option value="20l00">8:00 PM</option>');
  $('#endTime').append('<option value="20l30">8:30 PM</option>');
}
else if($('#startTime').val()=='18l00'){
  $('#endTime').append('<option value="19l00">7:00 PM</option>');
  $('#endTime').append('<option value="19l30">7:30 PM</option>');
  $('#endTime').append('<option value="20l00">8:00 PM</option>');
  $('#endTime').append('<option value="20l30">8:30 PM</option>');
  $('#endTime').append('<option value="21l00">9:00 PM</option>');
}
else if($('#startTime').val()=='18l30'){
  $('#endTime').append('<option value="19l30">7:30 PM</option>');
  $('#endTime').append('<option value="20l00">8:00 PM</option>');
  $('#endTime').append('<option value="20l30">8:30 PM</option>');
  $('#endTime').append('<option value="21l00">9:00 PM</option>');
}
else if($('#startTime').val()=='19l00'){
  $('#endTime').append('<option value="20l00">8:00 PM</option>');
```

```
          $('#endTime').append('<option value="20l30">8:30 PM</option>');
          $('#endTime').append('<option value="21l00">9:00 PM</option>');
        }

        else if($('#startTime').val()=='19l30'){
          $('#endTime').append('<option value="20l30">8:30 PM</option>');
          $('#endTime').append('<option value="21l00">9:00 PM</option>');
        }
        else if($('#startTime').val()=='20l00'){
          $('#endTime').append('<option value="21l00">9:00 PM</option>');
        }
        else if($('#startTime').val()=='20l30'){
          $('#endTime').append('<option value="21l00">9:00 PM</option>');
        }
       });
       }
       </script>
       </div>
       <div class="modal-footer">
         <!--<button type="button" class="buttonz" data-dismiss="modal">Close</button>-->
         <button type="submit" class="btn btn-primary buttonz" id="request"
name="request">Submit</button>
       </div>
     </form>
   </div>
  </div>
</div>

<body>
<div class="navbar navbar-inverse navbar-fixed-top major" role="navigation">
   <div class="container-fluid">
      <div class="navbar-header">
         <button type="button" class="navbar-toggle" data-toggle="collapse" data-
target=".navbar-collapse">
            <span class="sr-only">Toggle navigation</span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
         </button>
         <a class="navbar-brand" href="home.php">UST-ICS</a>
      </div>
      <div class="collapse navbar-collapse">
         <ul class="nav navbar-nav navbar-right">
            <li><a href="#">Hello,
              <?php
              echo $_SESSION['uName'];
              echo ' !';
              ?>
              </a></li>
            <li><a href="index.php"><span class="glyphicon glyphicon-log-in"></span> Log-
out</a></li>
```

```
        </ul>
        <ul class="nav navbar-nav">
          <li class="active"><a href="home.php">Home</a></li>
          <li><a href="requests.php">Your requests</a></li>
          <?php
            $query = $conn->prepare("SELECT * FROM register WHERE
username='$uName'"); // prepate a query
            $query->execute(); // actually perform the query
            $result = $query->get_result(); // retrieve the result so it can be used inside PHP
            $r = $result->fetch_array(MYSQLI_ASSOC); // bind the data from the first result row
to $r
            if ($r['access']=="DeptHead")
            {
          ?>
          <li>
            <a href="#" class="dropdown-toggle" data-toggle="dropdown">Scheduling <b
class="caret"></b></a>
            <ul class="dropdown-menu multi-level">
              <li><a href="roomView.php">Room view</a></li>
              <li><a href="facultyView.php">Faculty view</a></li>
              <li><a href="sectionView.php">Section view</a></li>
              <!--
              <li class="divider"></li>
              <li><a href="#">Separated link Scheduling</a></li>
              <li class="divider"></li>
              <li><a href="#">One more separated link Scheduling</a></li>
              -->
            </ul>
          </li>

          <li>
            <a href="#" class="dropdown-toggle" data-toggle="dropdown">Maintenance <b
class="caret"></b></a>
            <ul class="dropdown-menu">
              <li><a href="room.php">Room</a></li>
              <li><a href="faculty.php">Faculty</a></li>
              <li><a href="course.php">Course</a></li>
              <li><a href="section.php">Section</a></li>
              <!--
              <li class="divider"></li>
              <li><a href="#">Separated link Scheduling</a></li>
              <li class="divider"></li>
              <li><a href="#">One more separated link Scheduling</a></li>
              -->
            </ul>
          </li>
            <?php
            }
            ?>
          <?php
```

```
            $query = $conn->prepare("SELECT * FROM register WHERE
username='$uName'"); // prepate a query
            $query->execute(); // actually perform the query
            $result = $query->get_result(); // retrieve the result so it can be used inside PHP
            $r = $result->fetch_array(MYSQLI_ASSOC); // bind the data from the first result row
to $r
            if ($r['access']=="admin")
            {
          ?>
          <li>
            <a href="#" class="dropdown-toggle" data-toggle="dropdown">Maintenance <b
class="caret"></b></a>
            <ul class="dropdown-menu">
              <li><a href="room.php">Room</a></li>
              <li><a href="faculty.php">Faculty</a></li>
              <li><a href="course.php">Course</a></li>
              <li><a href="section.php">Section</a></li>
              <li><a href="archive.php">Archive</a></li>
              <li><a href="users.php">Users</a></li>
              <li><a href="reqMain.php">Requests</a></li>
              <!--
              <li class="divider"></li>
              <li><a href="#">Separated link Scheduling</a></li>
              <li class="divider"></li>
              <li><a href="#">One more separated link Scheduling</a></li>
              -->
            </ul>
          </li>
          <?php
          }
          ?>
        </ul>
      </div><!--/.nav-collapse -->
    </div>
  </div>

<div class="container-fluid text-center minor">
  <div class="row content">
    <div class="col-sm-1 sidenav">
    </div>

    <div class="col-sm-10 text-left" style="background-color: white;">
      <h1><?php echo $test; ?> schedules and events</h1>
      <hr>
      <!-- Room Dropdown START-->
      <?php
        $query2 = $conn->prepare("SELECT * FROM room"); // displays all requests with same
project as user
        $query2->execute(); // actually perform the query
        $result2 = $query2->get_result(); // retrieve the result so it can be used inside PHP
```

```
      $r2 = $result2->fetch_array(MYSQLI_ASSOC); // bind the data from the first result row to
$r
          if ($result2 -> num_rows > 0)
          {
           $main = $r2['rname'];
             echo '
             <form action="roomSelect.php" method="post">
             <div class="dropdown">
                   <button class="btn btn-default dropdown-toggle" type="button" data-
toggle="dropdown">'.$test.'
                    <span class="caret"></span></button>
                    <ul class="dropdown-menu dropdown-menu-left">
                      <li><a href="home.php" style="text-align:center;">All Rooms</a></li>
                      <li class="divider"></li>';//column heads
             do
             {
                echo '<li><button type="submit" style="
                   background:none!important;
                   color:inherit;
                   border:none;
                   text-align:center;
                   padding:0!important;
                   font: inherit;
                   cursor: pointer;
                   width:100%;" name="rname" id="rname"
                value="'.$r2['rname'].'"> '.$r2['rname'].' </button></li>
                <li class="divider"></li>';//DISPLAY TITLE
             }
             while ($r2 = $result2 -> fetch_assoc());
             echo '          </ul>
                   </div>
                   </form>';
          }
          else
          {
             echo 'EMPTY';
          }
        ?>
    <!-- Room Dropdown END -->
    <!-- GOOGLE CALENDAR EMBED CODE -->
    <br>
           <div class="responsiveCal">
      <div class="responsiveCal major" style="margin-top: 5%;">
       <div id="calendar"></div>
      </div>
     </div>
     <hr>
     <div>
     </div>
    </div>
    <div class="col-sm-1 sidenav">
```

```
    </div>
  </div>
</div>
<footer class="container-fluid text-center fixed-bottom">
  <p>Copyright © University of Santo Tomas - Institute of Information and Computing Sciences.
All rights reserved 2018</p>
</footer>
</body>
</html>
```

Scheduling

```
$sheetNames = $phpExcel->getSheetNames();
if(in_array($section, $sheetNames))
{
  $phpExcel->getSheetByName($section);
  $phpExcel->setActiveSheetIndexByName($section);

  $phpExcel->getActiveSheet()->mergeCells("A1:M1");
  $phpExcel->getActiveSheet()->getStyle('A1:M1')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
  $phpExcel->getActiveSheet()->SetCellValue("A1", "University of Santo Tomas");
  $phpExcel->getActiveSheet()->getStyle('A1:M1')->getFont()->setBold( true )->setSize(12);

  $phpExcel->getActiveSheet()->mergeCells("A2:M2");
  $phpExcel->getActiveSheet()->getStyle('A2:M2')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
  $phpExcel->getActiveSheet()->SetCellValue("A2", "Institute of Information and Computing
Sciences");
  $phpExcel->getActiveSheet()->getStyle('A2:M2')->getFont()->setBold( true )->setItalic( true )-
>setSize(12);

  $phpExcel->getActiveSheet()->mergeCells("A3:M3");
  $phpExcel->getActiveSheet()->getStyle('A3:M3')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
  $phpExcel->getActiveSheet()->SetCellValue("A3", "C L A S S   S C H E D U L E");
  $phpExcel->getActiveSheet()->getStyle('A3:M3')->getFont()->setBold( true )->setSize(12);

  $phpExcel->getActiveSheet()->mergeCells("A4:M4");
  $phpExcel->getActiveSheet()->getStyle('A4:M4')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
  $phpExcel->getActiveSheet()->SetCellValue("A4", "2nd Term A.Y. 2018-2019");
  $phpExcel->getActiveSheet()->getStyle('A4:M4')->getFont()->setBold( true )->setSize(12)-
>setUnderline(true);

  $phpExcel->getActiveSheet()->mergeCells("A6:M6");
  $phpExcel->getActiveSheet()->getStyle('A6:M6')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
  $phpExcel->getActiveSheet()->SetCellValue("A6", "CLASS SCHEDULE ".$section."");
  $phpExcel->getActiveSheet()->getStyle('A6:M6')->getFont()->setBold( true )->setSize(12)-
>setUnderline(true);
```

```
  $phpExcel->getActiveSheet()->getStyle('A7')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
  $phpExcel->getActiveSheet()->SetCellValue("A7", "TIME");
  $phpExcel->getActiveSheet()->getStyle('A7')->getFont()->setBold( true )->setSize(12);

  $phpExcel->getActiveSheet()->mergeCells("B7:C7");
  $phpExcel->getActiveSheet()->getStyle('B7:C7')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
  $phpExcel->getActiveSheet()->SetCellValue("B7", "MONDAY");
  $phpExcel->getActiveSheet()->getStyle('B7:C7')->getFont()->setBold( true )->setSize(12);

  $phpExcel->getActiveSheet()->mergeCells("D7:E7");
  $phpExcel->getActiveSheet()->getStyle('D7:E7')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
  $phpExcel->getActiveSheet()->SetCellValue("D7", "TUESDAY");
  $phpExcel->getActiveSheet()->getStyle('D7:E7')->getFont()->setBold( true )->setSize(12);

  $phpExcel->getActiveSheet()->mergeCells("F7:G7");
  $phpExcel->getActiveSheet()->getStyle('F7:G7')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
  $phpExcel->getActiveSheet()->SetCellValue("F7", "WEDNESDAY");
  $phpExcel->getActiveSheet()->getStyle('F7:G7')->getFont()->setBold( true )->setSize(12);

  $phpExcel->getActiveSheet()->mergeCells("H7:I7");
  $phpExcel->getActiveSheet()->getStyle('H7:I7')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
  $phpExcel->getActiveSheet()->SetCellValue("H7", "THURSDAY");
  $phpExcel->getActiveSheet()->getStyle('H7:I7')->getFont()->setBold( true )->setSize(12);

  $phpExcel->getActiveSheet()->mergeCells("J7:K7");
  $phpExcel->getActiveSheet()->getStyle('J7:K7')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
  $phpExcel->getActiveSheet()->SetCellValue("J7", "FRIDAY");
  $phpExcel->getActiveSheet()->getStyle('J7:K7')->getFont()->setBold( true )->setSize(12);

  $phpExcel->getActiveSheet()->mergeCells("L7:M7");
  $phpExcel->getActiveSheet()->getStyle('L7:M7')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
  $phpExcel->getActiveSheet()->SetCellValue("L7", "SATURDAY");
  $phpExcel->getActiveSheet()->getStyle('L7:M7')->getFont()->setBold( true )->setSize(12);

  $phpExcel->getActiveSheet()->SetCellValue("A8", "7:00 AM");
  $phpExcel->getActiveSheet()->getStyle('A8')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A9", "to");
  $phpExcel->getActiveSheet()->getStyle('A9')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A10", "7:30 AM");
  $phpExcel->getActiveSheet()->getStyle('A10')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A8:A10')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
```

```
$phpExcel->getActiveSheet()->mergeCells("B9:C9");
$phpExcel->getActiveSheet()->mergeCells("B10:C10");

$phpExcel->getActiveSheet()->mergeCells("D9:E9");
$phpExcel->getActiveSheet()->mergeCells("D10:E10");

$phpExcel->getActiveSheet()->mergeCells("F9:G9");
$phpExcel->getActiveSheet()->mergeCells("F10:G10");

$phpExcel->getActiveSheet()->mergeCells("H9:I9");
$phpExcel->getActiveSheet()->mergeCells("H10:I10");

$phpExcel->getActiveSheet()->mergeCells("J9:K9");
$phpExcel->getActiveSheet()->mergeCells("J10:K10");

$phpExcel->getActiveSheet()->mergeCells("L9:M9");
$phpExcel->getActiveSheet()->mergeCells("L10:M10");


$phpExcel->getActiveSheet()->SetCellValue("A11", "7:30 AM");
$phpExcel->getActiveSheet()->getStyle('A11')->getFont()->setBold( true )->setSize(12);
$phpExcel->getActiveSheet()->SetCellValue("A12", "to");
$phpExcel->getActiveSheet()->getStyle('A12')->getFont()->setBold( true )->setSize(12);
$phpExcel->getActiveSheet()->SetCellValue("A13", "8:00 AM");
$phpExcel->getActiveSheet()->getStyle('A13')->getFont()->setBold( true )->setSize(12);
$phpExcel->getActiveSheet()->getStyle('A11:A13')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

$phpExcel->getActiveSheet()->mergeCells("B12:C12");
$phpExcel->getActiveSheet()->mergeCells("B13:C13");

$phpExcel->getActiveSheet()->mergeCells("D12:E12");
$phpExcel->getActiveSheet()->mergeCells("D13:E13");

$phpExcel->getActiveSheet()->mergeCells("F12:G12");
$phpExcel->getActiveSheet()->mergeCells("F13:G13");

$phpExcel->getActiveSheet()->mergeCells("H12:I12");
$phpExcel->getActiveSheet()->mergeCells("H13:I13");

$phpExcel->getActiveSheet()->mergeCells("J12:K12");
$phpExcel->getActiveSheet()->mergeCells("J13:K13");

$phpExcel->getActiveSheet()->mergeCells("L12:M12");
$phpExcel->getActiveSheet()->mergeCells("L13:M13");

$phpExcel->getActiveSheet()->SetCellValue("A14", "8:00 AM");
$phpExcel->getActiveSheet()->getStyle('A14')->getFont()->setBold( true )->setSize(12);
$phpExcel->getActiveSheet()->SetCellValue("A15", "to");
$phpExcel->getActiveSheet()->getStyle('A15')->getFont()->setBold( true )->setSize(12);
$phpExcel->getActiveSheet()->SetCellValue("A16", "8:30 AM");
```

```
  $phpExcel->getActiveSheet()->getStyle('A16')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A14:A16')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B15:C15");
  $phpExcel->getActiveSheet()->mergeCells("B16:C16");

  $phpExcel->getActiveSheet()->mergeCells("D15:E15");
  $phpExcel->getActiveSheet()->mergeCells("D16:E16");

  $phpExcel->getActiveSheet()->mergeCells("F15:G15");
  $phpExcel->getActiveSheet()->mergeCells("F16:G16");

  $phpExcel->getActiveSheet()->mergeCells("H15:I15");
  $phpExcel->getActiveSheet()->mergeCells("H16:I16");

  $phpExcel->getActiveSheet()->mergeCells("J15:K15");
  $phpExcel->getActiveSheet()->mergeCells("J16:K16");

  $phpExcel->getActiveSheet()->mergeCells("L15:M15");
  $phpExcel->getActiveSheet()->mergeCells("L16:M16");

  $phpExcel->getActiveSheet()->SetCellValue("A17", "8:30 AM");
  $phpExcel->getActiveSheet()->getStyle('A17')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A18", "to");
  $phpExcel->getActiveSheet()->getStyle('A18')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A19", "9:00 AM");
  $phpExcel->getActiveSheet()->getStyle('A19')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A17:A19')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B18:C18");
  $phpExcel->getActiveSheet()->mergeCells("B19:C19");

  $phpExcel->getActiveSheet()->mergeCells("D18:E18");
  $phpExcel->getActiveSheet()->mergeCells("D19:E19");

  $phpExcel->getActiveSheet()->mergeCells("F18:G18");
  $phpExcel->getActiveSheet()->mergeCells("F19:G19");

  $phpExcel->getActiveSheet()->mergeCells("H18:I18");
  $phpExcel->getActiveSheet()->mergeCells("H19:I19");

  $phpExcel->getActiveSheet()->mergeCells("J18:K18");
  $phpExcel->getActiveSheet()->mergeCells("J19:K19");

  $phpExcel->getActiveSheet()->mergeCells("L18:M18");
  $phpExcel->getActiveSheet()->mergeCells("L19:M19");

  $phpExcel->getActiveSheet()->SetCellValue("A20", "9:00 AM");
  $phpExcel->getActiveSheet()->getStyle('A20')->getFont()->setBold( true )->setSize(12);
```

```
  $phpExcel->getActiveSheet()->SetCellValue("A21", "to");
  $phpExcel->getActiveSheet()->getStyle('A21')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A22", "9:30 AM");
  $phpExcel->getActiveSheet()->getStyle('A22')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A20:A22')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B21:C21");
  $phpExcel->getActiveSheet()->mergeCells("B22:C22");

  $phpExcel->getActiveSheet()->mergeCells("D21:E21");
  $phpExcel->getActiveSheet()->mergeCells("D22:E22");

  $phpExcel->getActiveSheet()->mergeCells("F21:G21");
  $phpExcel->getActiveSheet()->mergeCells("F22:G22");

  $phpExcel->getActiveSheet()->mergeCells("H21:I21");
  $phpExcel->getActiveSheet()->mergeCells("H22:I22");

  $phpExcel->getActiveSheet()->mergeCells("J21:K21");
  $phpExcel->getActiveSheet()->mergeCells("J22:K22");

  $phpExcel->getActiveSheet()->mergeCells("L21:M21");
  $phpExcel->getActiveSheet()->mergeCells("L22:M22");

  $phpExcel->getActiveSheet()->SetCellValue("A23", "9:30 AM");
  $phpExcel->getActiveSheet()->getStyle('A23')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A24", "to");
  $phpExcel->getActiveSheet()->getStyle('A24')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A25", "10:00 AM");
  $phpExcel->getActiveSheet()->getStyle('A25')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A23:A25')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B24:C24");
  $phpExcel->getActiveSheet()->mergeCells("B25:C25");

  $phpExcel->getActiveSheet()->mergeCells("D24:E24");
  $phpExcel->getActiveSheet()->mergeCells("D25:E25");

  $phpExcel->getActiveSheet()->mergeCells("F24:G24");
  $phpExcel->getActiveSheet()->mergeCells("F25:G25");

  $phpExcel->getActiveSheet()->mergeCells("H24:I24");
  $phpExcel->getActiveSheet()->mergeCells("H25:I25");

  $phpExcel->getActiveSheet()->mergeCells("J24:K24");
  $phpExcel->getActiveSheet()->mergeCells("J25:K25");

  $phpExcel->getActiveSheet()->mergeCells("L24:M24");
  $phpExcel->getActiveSheet()->mergeCells("L25:M25");
```

```
  $phpExcel->getActiveSheet()->SetCellValue("A26", "10:00 AM");
  $phpExcel->getActiveSheet()->getStyle('A26')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A27", "to");
  $phpExcel->getActiveSheet()->getStyle('A27')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A28", "10:30 AM");
  $phpExcel->getActiveSheet()->getStyle('A28')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A26:A28')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B27:C27");
  $phpExcel->getActiveSheet()->mergeCells("B28:C28");

  $phpExcel->getActiveSheet()->mergeCells("D27:E27");
  $phpExcel->getActiveSheet()->mergeCells("D28:E28");

  $phpExcel->getActiveSheet()->mergeCells("F27:G27");
  $phpExcel->getActiveSheet()->mergeCells("F28:G28");

  $phpExcel->getActiveSheet()->mergeCells("H27:I27");
  $phpExcel->getActiveSheet()->mergeCells("H28:I28");

  $phpExcel->getActiveSheet()->mergeCells("J27:K27");
  $phpExcel->getActiveSheet()->mergeCells("J28:K28");

  $phpExcel->getActiveSheet()->mergeCells("L27:M27");
  $phpExcel->getActiveSheet()->mergeCells("L28:M28");


  $phpExcel->getActiveSheet()->SetCellValue("A29", "10:30 AM");
  $phpExcel->getActiveSheet()->getStyle('A29')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A30", "to");
  $phpExcel->getActiveSheet()->getStyle('A30')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A31", "11:00 AM");
  $phpExcel->getActiveSheet()->getStyle('A31')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A29:A31')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B30:C30");
  $phpExcel->getActiveSheet()->mergeCells("B31:C31");

  $phpExcel->getActiveSheet()->mergeCells("D30:E30");
  $phpExcel->getActiveSheet()->mergeCells("D31:E31");

  $phpExcel->getActiveSheet()->mergeCells("F30:G30");
  $phpExcel->getActiveSheet()->mergeCells("F31:G31");

  $phpExcel->getActiveSheet()->mergeCells("H30:I30");
  $phpExcel->getActiveSheet()->mergeCells("H31:I31");

  $phpExcel->getActiveSheet()->mergeCells("J30:K30");
```

```
  $phpExcel->getActiveSheet()->mergeCells("J31:K31");

  $phpExcel->getActiveSheet()->mergeCells("L30:M30");
  $phpExcel->getActiveSheet()->mergeCells("L31:M31");

  $phpExcel->getActiveSheet()->SetCellValue("A32", "11:00 AM");
  $phpExcel->getActiveSheet()->getStyle('A32')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A33", "to");
  $phpExcel->getActiveSheet()->getStyle('A33')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A34", "11:30 AM");
  $phpExcel->getActiveSheet()->getStyle('A34')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A32:A34')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B33:C33");
  $phpExcel->getActiveSheet()->mergeCells("B34:C34");

  $phpExcel->getActiveSheet()->mergeCells("D33:E33");
  $phpExcel->getActiveSheet()->mergeCells("D34:E34");

  $phpExcel->getActiveSheet()->mergeCells("F33:G33");
  $phpExcel->getActiveSheet()->mergeCells("F34:G34");

  $phpExcel->getActiveSheet()->mergeCells("H33:I33");
  $phpExcel->getActiveSheet()->mergeCells("H34:I34");

  $phpExcel->getActiveSheet()->mergeCells("J33:K33");
  $phpExcel->getActiveSheet()->mergeCells("J34:K34");

  $phpExcel->getActiveSheet()->mergeCells("L33:M33");
  $phpExcel->getActiveSheet()->mergeCells("L34:M34");

  $phpExcel->getActiveSheet()->SetCellValue("A35", "11:30 AM");
  $phpExcel->getActiveSheet()->getStyle('A35')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A36", "to");
  $phpExcel->getActiveSheet()->getStyle('A36')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A37", "12:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A37')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A35:A37')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B36:C36");
  $phpExcel->getActiveSheet()->mergeCells("B37:C37");

  $phpExcel->getActiveSheet()->mergeCells("D36:E36");
  $phpExcel->getActiveSheet()->mergeCells("D37:E37");

  $phpExcel->getActiveSheet()->mergeCells("F36:G36");
  $phpExcel->getActiveSheet()->mergeCells("F37:G37");

  $phpExcel->getActiveSheet()->mergeCells("H36:I36");
```

```
  $phpExcel->getActiveSheet()->mergeCells("H37:I37");

  $phpExcel->getActiveSheet()->mergeCells("J36:K36");
  $phpExcel->getActiveSheet()->mergeCells("J37:K37");

  $phpExcel->getActiveSheet()->mergeCells("L36:M36");
  $phpExcel->getActiveSheet()->mergeCells("L37:M37");

  $phpExcel->getActiveSheet()->SetCellValue("A38", "12:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A38')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A39", "to");
  $phpExcel->getActiveSheet()->getStyle('A39')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A40", "12:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A40')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A38:A40')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B39:C39");
  $phpExcel->getActiveSheet()->mergeCells("B40:C40");

  $phpExcel->getActiveSheet()->mergeCells("D39:E39");
  $phpExcel->getActiveSheet()->mergeCells("D40:E40");

  $phpExcel->getActiveSheet()->mergeCells("F39:G39");
  $phpExcel->getActiveSheet()->mergeCells("F40:G40");

  $phpExcel->getActiveSheet()->mergeCells("H39:I39");
  $phpExcel->getActiveSheet()->mergeCells("H40:I40");

  $phpExcel->getActiveSheet()->mergeCells("J39:K39");
  $phpExcel->getActiveSheet()->mergeCells("J40:K40");

  $phpExcel->getActiveSheet()->mergeCells("L39:M39");
  $phpExcel->getActiveSheet()->mergeCells("L40:M40");

  $phpExcel->getActiveSheet()->SetCellValue("A41", "12:30 AM");
  $phpExcel->getActiveSheet()->getStyle('A41')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A42", "to");
  $phpExcel->getActiveSheet()->getStyle('A42')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A43", "1:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A43')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A41:A43')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B42:C42");
  $phpExcel->getActiveSheet()->mergeCells("B43:C43");

  $phpExcel->getActiveSheet()->mergeCells("D42:E42");
  $phpExcel->getActiveSheet()->mergeCells("D43:E43");

  $phpExcel->getActiveSheet()->mergeCells("F42:G42");
```

```
  $phpExcel->getActiveSheet()->mergeCells("F43:G43");

  $phpExcel->getActiveSheet()->mergeCells("H42:I42");
  $phpExcel->getActiveSheet()->mergeCells("H43:I43");

  $phpExcel->getActiveSheet()->mergeCells("J42:K42");
  $phpExcel->getActiveSheet()->mergeCells("J43:K43");

  $phpExcel->getActiveSheet()->mergeCells("L42:M42");
  $phpExcel->getActiveSheet()->mergeCells("L43:M43");

  $phpExcel->getActiveSheet()->SetCellValue("A44", "1:00 AM");
  $phpExcel->getActiveSheet()->getStyle('A44')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A45", "to");
  $phpExcel->getActiveSheet()->getStyle('A45')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A46", "1:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A46')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A44:A46')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B45:C45");
  $phpExcel->getActiveSheet()->mergeCells("B46:C46");

  $phpExcel->getActiveSheet()->mergeCells("D45:E45");
  $phpExcel->getActiveSheet()->mergeCells("D46:E46");

  $phpExcel->getActiveSheet()->mergeCells("F45:G45");
  $phpExcel->getActiveSheet()->mergeCells("F46:G46");

  $phpExcel->getActiveSheet()->mergeCells("H45:I45");
  $phpExcel->getActiveSheet()->mergeCells("H46:I46");

  $phpExcel->getActiveSheet()->mergeCells("J45:K45");
  $phpExcel->getActiveSheet()->mergeCells("J46:K46");

  $phpExcel->getActiveSheet()->mergeCells("L45:M45");
  $phpExcel->getActiveSheet()->mergeCells("L46:M46");

  $phpExcel->getActiveSheet()->SetCellValue("A47", "1:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A47')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A48", "to");
  $phpExcel->getActiveSheet()->getStyle('A48')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A49", "2:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A49')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A47:A49')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B48:C48");
  $phpExcel->getActiveSheet()->mergeCells("B49:C49");

  $phpExcel->getActiveSheet()->mergeCells("D48:E48");
```

```
  $phpExcel->getActiveSheet()->mergeCells("D49:E49");

  $phpExcel->getActiveSheet()->mergeCells("F48:G48");
  $phpExcel->getActiveSheet()->mergeCells("F49:G49");

  $phpExcel->getActiveSheet()->mergeCells("H48:I48");
  $phpExcel->getActiveSheet()->mergeCells("H49:I49");

  $phpExcel->getActiveSheet()->mergeCells("J48:K48");
  $phpExcel->getActiveSheet()->mergeCells("J49:K49");

  $phpExcel->getActiveSheet()->mergeCells("L48:M48");
  $phpExcel->getActiveSheet()->mergeCells("L49:M49");

  $phpExcel->getActiveSheet()->SetCellValue("A50", "2:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A50')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A51", "to");
  $phpExcel->getActiveSheet()->getStyle('A51')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A52", "2:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A52')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A50:A52')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B51:C51");
  $phpExcel->getActiveSheet()->mergeCells("B52:C52");

  $phpExcel->getActiveSheet()->mergeCells("D51:E51");
  $phpExcel->getActiveSheet()->mergeCells("D52:E52");

  $phpExcel->getActiveSheet()->mergeCells("F51:G51");
  $phpExcel->getActiveSheet()->mergeCells("F52:G52");

  $phpExcel->getActiveSheet()->mergeCells("H51:I51");
  $phpExcel->getActiveSheet()->mergeCells("H52:I52");

  $phpExcel->getActiveSheet()->mergeCells("J51:K51");
  $phpExcel->getActiveSheet()->mergeCells("J52:K52");

  $phpExcel->getActiveSheet()->mergeCells("L51:M51");
  $phpExcel->getActiveSheet()->mergeCells("L52:M52");

  $phpExcel->getActiveSheet()->SetCellValue("A53", "2:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A53')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A54", "to");
  $phpExcel->getActiveSheet()->getStyle('A54')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A55", "3:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A55')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A53:A55')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B54:C54");
```

```
$phpExcel->getActiveSheet()->mergeCells("B55:C55");

$phpExcel->getActiveSheet()->mergeCells("D54:E54");
$phpExcel->getActiveSheet()->mergeCells("D55:E55");

$phpExcel->getActiveSheet()->mergeCells("F54:G54");
$phpExcel->getActiveSheet()->mergeCells("F55:G55");

$phpExcel->getActiveSheet()->mergeCells("H54:I54");
$phpExcel->getActiveSheet()->mergeCells("H55:I55");

$phpExcel->getActiveSheet()->mergeCells("J54:K54");
$phpExcel->getActiveSheet()->mergeCells("J55:K55");

$phpExcel->getActiveSheet()->mergeCells("L54:M54");
$phpExcel->getActiveSheet()->mergeCells("L55:M55");

$phpExcel->getActiveSheet()->SetCellValue("A56", "3:00 PM");
$phpExcel->getActiveSheet()->getStyle('A56')->getFont()->setBold( true )->setSize(12);
$phpExcel->getActiveSheet()->SetCellValue("A57", "to");
$phpExcel->getActiveSheet()->getStyle('A57')->getFont()->setBold( true )->setSize(12);
$phpExcel->getActiveSheet()->SetCellValue("A58", "3:30 PM");
$phpExcel->getActiveSheet()->getStyle('A58')->getFont()->setBold( true )->setSize(12);
$phpExcel->getActiveSheet()->getStyle('A56:A58')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

$phpExcel->getActiveSheet()->mergeCells("B57:C57");
$phpExcel->getActiveSheet()->mergeCells("B58:C58");

$phpExcel->getActiveSheet()->mergeCells("D57:E57");
$phpExcel->getActiveSheet()->mergeCells("D58:E58");

$phpExcel->getActiveSheet()->mergeCells("F57:G57");
$phpExcel->getActiveSheet()->mergeCells("F58:G58");

$phpExcel->getActiveSheet()->mergeCells("H57:I57");
$phpExcel->getActiveSheet()->mergeCells("H58:I58");

$phpExcel->getActiveSheet()->mergeCells("J57:K57");
$phpExcel->getActiveSheet()->mergeCells("J58:K58");

$phpExcel->getActiveSheet()->mergeCells("L57:M57");
$phpExcel->getActiveSheet()->mergeCells("L58:M58");

$phpExcel->getActiveSheet()->SetCellValue("A59", "3:30 PM");
$phpExcel->getActiveSheet()->getStyle('A59')->getFont()->setBold( true )->setSize(12);
$phpExcel->getActiveSheet()->SetCellValue("A60", "to");
$phpExcel->getActiveSheet()->getStyle('A60')->getFont()->setBold( true )->setSize(12);
$phpExcel->getActiveSheet()->SetCellValue("A61", "4:00 PM");
$phpExcel->getActiveSheet()->getStyle('A61')->getFont()->setBold( true )->setSize(12);
```

```
  $phpExcel->getActiveSheet()->getStyle('A59:A61')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B60:C60");
  $phpExcel->getActiveSheet()->mergeCells("B61:C61");

  $phpExcel->getActiveSheet()->mergeCells("D60:E60");
  $phpExcel->getActiveSheet()->mergeCells("D61:E61");

  $phpExcel->getActiveSheet()->mergeCells("F60:G60");
  $phpExcel->getActiveSheet()->mergeCells("F61:G61");

  $phpExcel->getActiveSheet()->mergeCells("H60:I60");
  $phpExcel->getActiveSheet()->mergeCells("H61:I61");

  $phpExcel->getActiveSheet()->mergeCells("J60:K60");
  $phpExcel->getActiveSheet()->mergeCells("J61:K61");

  $phpExcel->getActiveSheet()->mergeCells("L60:M60");
  $phpExcel->getActiveSheet()->mergeCells("L61:M61");

  $phpExcel->getActiveSheet()->SetCellValue("A62", "4:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A62')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A63", "to");
  $phpExcel->getActiveSheet()->getStyle('A63')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A64", "4:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A64')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A62:A64')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B63:C63");
  $phpExcel->getActiveSheet()->mergeCells("B64:C64");

  $phpExcel->getActiveSheet()->mergeCells("D63:E63");
  $phpExcel->getActiveSheet()->mergeCells("D64:E64");

  $phpExcel->getActiveSheet()->mergeCells("F63:G63");
  $phpExcel->getActiveSheet()->mergeCells("F64:G64");

  $phpExcel->getActiveSheet()->mergeCells("H63:I63");
  $phpExcel->getActiveSheet()->mergeCells("H64:I64");

  $phpExcel->getActiveSheet()->mergeCells("J63:K63");
  $phpExcel->getActiveSheet()->mergeCells("J64:K64");

  $phpExcel->getActiveSheet()->mergeCells("L63:M63");
  $phpExcel->getActiveSheet()->mergeCells("L64:M64");

  $phpExcel->getActiveSheet()->SetCellValue("A65", "4:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A65')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A66", "to");
```

```
  $phpExcel->getActiveSheet()->getStyle('A66')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A67", "5:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A67')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A65:A67')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B66:C66");
  $phpExcel->getActiveSheet()->mergeCells("B67:C67");

  $phpExcel->getActiveSheet()->mergeCells("D66:E66");
  $phpExcel->getActiveSheet()->mergeCells("D67:E67");

  $phpExcel->getActiveSheet()->mergeCells("F66:G66");
  $phpExcel->getActiveSheet()->mergeCells("F67:G67");

  $phpExcel->getActiveSheet()->mergeCells("H66:I66");
  $phpExcel->getActiveSheet()->mergeCells("H67:I67");

  $phpExcel->getActiveSheet()->mergeCells("J66:K66");
  $phpExcel->getActiveSheet()->mergeCells("J67:K67");

  $phpExcel->getActiveSheet()->mergeCells("L66:M66");
  $phpExcel->getActiveSheet()->mergeCells("L67:M67");

  $phpExcel->getActiveSheet()->SetCellValue("A68", "5:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A68')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A69", "to");
  $phpExcel->getActiveSheet()->getStyle('A69')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A70", "5:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A70')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A68:A70')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B69:C69");
  $phpExcel->getActiveSheet()->mergeCells("B70:C70");

  $phpExcel->getActiveSheet()->mergeCells("D69:E69");
  $phpExcel->getActiveSheet()->mergeCells("D70:E70");

  $phpExcel->getActiveSheet()->mergeCells("F69:G69");
  $phpExcel->getActiveSheet()->mergeCells("F70:G70");

  $phpExcel->getActiveSheet()->mergeCells("H69:I69");
  $phpExcel->getActiveSheet()->mergeCells("H70:I70");

  $phpExcel->getActiveSheet()->mergeCells("J69:K69");
  $phpExcel->getActiveSheet()->mergeCells("J70:K70");

  $phpExcel->getActiveSheet()->mergeCells("L69:M69");
  $phpExcel->getActiveSheet()->mergeCells("L70:M70");
```

```
  $phpExcel->getActiveSheet()->SetCellValue("A71", "5:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A71')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A72", "to");
  $phpExcel->getActiveSheet()->getStyle('A72')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A73", "6:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A73')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A71:A73')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B72:C72");
  $phpExcel->getActiveSheet()->mergeCells("B73:C73");

  $phpExcel->getActiveSheet()->mergeCells("D72:E72");
  $phpExcel->getActiveSheet()->mergeCells("D73:E73");

  $phpExcel->getActiveSheet()->mergeCells("F72:G72");
  $phpExcel->getActiveSheet()->mergeCells("F73:G73");

  $phpExcel->getActiveSheet()->mergeCells("H72:I72");
  $phpExcel->getActiveSheet()->mergeCells("H73:I73");

  $phpExcel->getActiveSheet()->mergeCells("J72:K72");
  $phpExcel->getActiveSheet()->mergeCells("J73:K73");

  $phpExcel->getActiveSheet()->mergeCells("L72:M72");
  $phpExcel->getActiveSheet()->mergeCells("L73:M73");

  $phpExcel->getActiveSheet()->SetCellValue("A74", "6:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A74')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A75", "to");
  $phpExcel->getActiveSheet()->getStyle('A75')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A76", "6:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A76')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A74:A76')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B75:C75");
  $phpExcel->getActiveSheet()->mergeCells("B76:C76");

  $phpExcel->getActiveSheet()->mergeCells("D75:E75");
  $phpExcel->getActiveSheet()->mergeCells("D76:E76");

  $phpExcel->getActiveSheet()->mergeCells("F75:G75");
  $phpExcel->getActiveSheet()->mergeCells("F76:G76");

  $phpExcel->getActiveSheet()->mergeCells("H75:I75");
  $phpExcel->getActiveSheet()->mergeCells("H76:I76");

  $phpExcel->getActiveSheet()->mergeCells("J75:K75");
  $phpExcel->getActiveSheet()->mergeCells("J76:K76");
```

```
  $phpExcel->getActiveSheet()->mergeCells("L75:M75");
  $phpExcel->getActiveSheet()->mergeCells("L76:M76");

  $phpExcel->getActiveSheet()->SetCellValue("A77", "6:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A77')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A78", "to");
  $phpExcel->getActiveSheet()->getStyle('A78')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A79", "7:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A79')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A77:A79')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B78:C78");
  $phpExcel->getActiveSheet()->mergeCells("B79:C79");

  $phpExcel->getActiveSheet()->mergeCells("D78:E78");
  $phpExcel->getActiveSheet()->mergeCells("D79:E79");

  $phpExcel->getActiveSheet()->mergeCells("F78:G78");
  $phpExcel->getActiveSheet()->mergeCells("F79:G79");

  $phpExcel->getActiveSheet()->mergeCells("H78:I78");
  $phpExcel->getActiveSheet()->mergeCells("H79:I79");

  $phpExcel->getActiveSheet()->mergeCells("J78:K78");
  $phpExcel->getActiveSheet()->mergeCells("J79:K79");

  $phpExcel->getActiveSheet()->mergeCells("L78:M78");
  $phpExcel->getActiveSheet()->mergeCells("L79:M79");

  $phpExcel->getActiveSheet()->SetCellValue("A80", "7:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A80')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A81", "to");
  $phpExcel->getActiveSheet()->getStyle('A81')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A82", "7:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A82')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A80:A82')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B81:C81");
  $phpExcel->getActiveSheet()->mergeCells("B82:C82");

  $phpExcel->getActiveSheet()->mergeCells("D81:E81");
  $phpExcel->getActiveSheet()->mergeCells("D82:E82");

  $phpExcel->getActiveSheet()->mergeCells("F81:G81");
  $phpExcel->getActiveSheet()->mergeCells("F82:G82");

  $phpExcel->getActiveSheet()->mergeCells("H81:I81");
  $phpExcel->getActiveSheet()->mergeCells("H82:I82");
```

```
  $phpExcel->getActiveSheet()->mergeCells("J81:K81");
  $phpExcel->getActiveSheet()->mergeCells("J82:K82");

  $phpExcel->getActiveSheet()->mergeCells("L81:M81");
  $phpExcel->getActiveSheet()->mergeCells("L82:M82");

  $phpExcel->getActiveSheet()->SetCellValue("A83", "7:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A83')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A84", "to");
  $phpExcel->getActiveSheet()->getStyle('A84')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A85", "8:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A85')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A83:A85')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B84:C84");
  $phpExcel->getActiveSheet()->mergeCells("B85:C85");

  $phpExcel->getActiveSheet()->mergeCells("D84:E84");
  $phpExcel->getActiveSheet()->mergeCells("D85:E85");

  $phpExcel->getActiveSheet()->mergeCells("F84:G84");
  $phpExcel->getActiveSheet()->mergeCells("F85:G85");

  $phpExcel->getActiveSheet()->mergeCells("H84:I84");
  $phpExcel->getActiveSheet()->mergeCells("H85:I85");

  $phpExcel->getActiveSheet()->mergeCells("J84:K84");
  $phpExcel->getActiveSheet()->mergeCells("J85:K85");

  $phpExcel->getActiveSheet()->mergeCells("L84:M84");
  $phpExcel->getActiveSheet()->mergeCells("L85:M85");

  $phpExcel->getActiveSheet()->SetCellValue("A86", "8:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A86')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A87", "to");
  $phpExcel->getActiveSheet()->getStyle('A87')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A88", "8:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A88')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A86:A88')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B87:C87");
  $phpExcel->getActiveSheet()->mergeCells("B88:C88");

  $phpExcel->getActiveSheet()->mergeCells("D87:E87");
  $phpExcel->getActiveSheet()->mergeCells("D88:E88");

  $phpExcel->getActiveSheet()->mergeCells("F87:G87");
  $phpExcel->getActiveSheet()->mergeCells("F88:G88");
```

```
  $phpExcel->getActiveSheet()->mergeCells("H87:I87");
  $phpExcel->getActiveSheet()->mergeCells("H88:I88");

  $phpExcel->getActiveSheet()->mergeCells("J87:K87");
  $phpExcel->getActiveSheet()->mergeCells("J88:K88");

  $phpExcel->getActiveSheet()->mergeCells("L87:M87");
  $phpExcel->getActiveSheet()->mergeCells("L88:M88");

  $phpExcel->getActiveSheet()->SetCellValue("A89", "8:30 PM");
  $phpExcel->getActiveSheet()->getStyle('A89')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A90", "to");
  $phpExcel->getActiveSheet()->getStyle('A90')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->SetCellValue("A91", "9:00 PM");
  $phpExcel->getActiveSheet()->getStyle('A91')->getFont()->setBold( true )->setSize(12);
  $phpExcel->getActiveSheet()->getStyle('A89:A91')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);

  $phpExcel->getActiveSheet()->mergeCells("B90:C90");
  $phpExcel->getActiveSheet()->mergeCells("B91:C91");

  $phpExcel->getActiveSheet()->mergeCells("D90:E90");
  $phpExcel->getActiveSheet()->mergeCells("D91:E91");

  $phpExcel->getActiveSheet()->mergeCells("F90:G90");
  $phpExcel->getActiveSheet()->mergeCells("F91:G91");

  $phpExcel->getActiveSheet()->mergeCells("H90:I90");
  $phpExcel->getActiveSheet()->mergeCells("H91:I91");

  $phpExcel->getActiveSheet()->mergeCells("J90:K90");
  $phpExcel->getActiveSheet()->mergeCells("J91:K91");

  $phpExcel->getActiveSheet()->mergeCells("L90:M90");
  $phpExcel->getActiveSheet()->mergeCells("L91:M91");

  $THICC = array('borders' => array('allborders' => array('style' =>
PHPExcel_Style_Border::BORDER_THICK)));
  $THINN = array('borders' => array('allborders' => array('style' =>
PHPExcel_Style_Border::BORDER_THIN)));
  $outTHICC = array('borders' => array('outline' => array('style' =>
PHPExcel_Style_Border::BORDER_THICK)));
  $outTHINN = array('borders' => array('outline' => array('style' =>
PHPExcel_Style_Border::BORDER_THIN)));


  $phpExcel->getActiveSheet()->getStyle('B8:M8')->applyFromArray($THINN);
  $phpExcel->getActiveSheet()->getStyle('B9:M10')->applyFromArray($THINN);

  $phpExcel->getActiveSheet()->getStyle('B11:M11')->applyFromArray($THINN);
  $phpExcel->getActiveSheet()->getStyle('B12:M13')->applyFromArray($THINN);
```

```
$phpExcel->getActiveSheet()->getStyle('B14:M14')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B15:M16')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B17:M17')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B18:M19')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B20:M20')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B21:M22')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B23:M23')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B24:M25')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B26:M26')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B27:M28')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B29:M29')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B30:M31')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B32:M32')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B33:M34')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B35:M35')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B36:M37')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B38:M38')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B39:M40')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B41:M41')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B42:M43')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B44:M44')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B45:M46')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B47:M47')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B48:M49')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B50:M50')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B51:M52')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B53:M53')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B54:M55')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B56:M56')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B57:M58')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B59:M59')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B60:M61')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B62:M62')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B63:M64')->applyFromArray($THINN);
```

```
$phpExcel->getActiveSheet()->getStyle('B65:M65')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B66:M67')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B68:M68')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B69:M70')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B71:M71')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B72:M73')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B74:M74')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B75:M76')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B74:M74')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B75:M76')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B77:M77')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B78:M79')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B80:M80')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B81:M82')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B83:M83')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B84:M85')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B86:M86')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B87:M88')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B89:M89')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('B90:M91')->applyFromArray($THINN);

$phpExcel->getActiveSheet()->getStyle('B8:A10')->applyFromArray($THINN);
$phpExcel->getActiveSheet()->getStyle('A7:M7')->applyFromArray($THICC);

$phpExcel->getActiveSheet()->getStyle('B8:C91')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('D8:E91')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('F8:G91')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('H8:I91')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('J8:K91')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('L8:M91')->applyFromArray($outTHICC);

$phpExcel->getActiveSheet()->getColumnDimension('A')->setWidth(15);

$phpExcel->getActiveSheet()->getStyle('A8:M10')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A11:M13')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A14:M16')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A17:M19')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A20:M22')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A23:M25')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A26:M28')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A29:M31')->applyFromArray($outTHICC);
```

```
$phpExcel->getActiveSheet()->getStyle('A32:M34')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A35:M37')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A38:M40')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A41:M43')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A44:M46')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A47:M49')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A50:M52')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A53:M55')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A56:M58')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A59:M61')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A62:M64')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A65:M67')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A68:M70')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A71:M73')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A74:M76')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A77:M79')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A80:M82')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A83:M85')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A86:M88')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A89:M91')->applyFromArray($outTHICC);

$phpExcel->getActiveSheet()->getStyle('A8:A10')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A11:A13')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A14:A16')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A17:A19')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A20:A22')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A23:A25')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A26:A28')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A29:A31')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A32:A34')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A35:A37')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A38:A40')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A41:A43')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A44:A46')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A47:A49')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A50:A52')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A53:A55')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A56:A58')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A59:A61')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A62:A64')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A65:A67')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A68:A70')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A71:A73')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A74:A76')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A77:A79')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A80:A82')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A83:A85')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A86:A88')->applyFromArray($outTHICC);
$phpExcel->getActiveSheet()->getStyle('A89:A91')->applyFromArray($outTHICC);
```

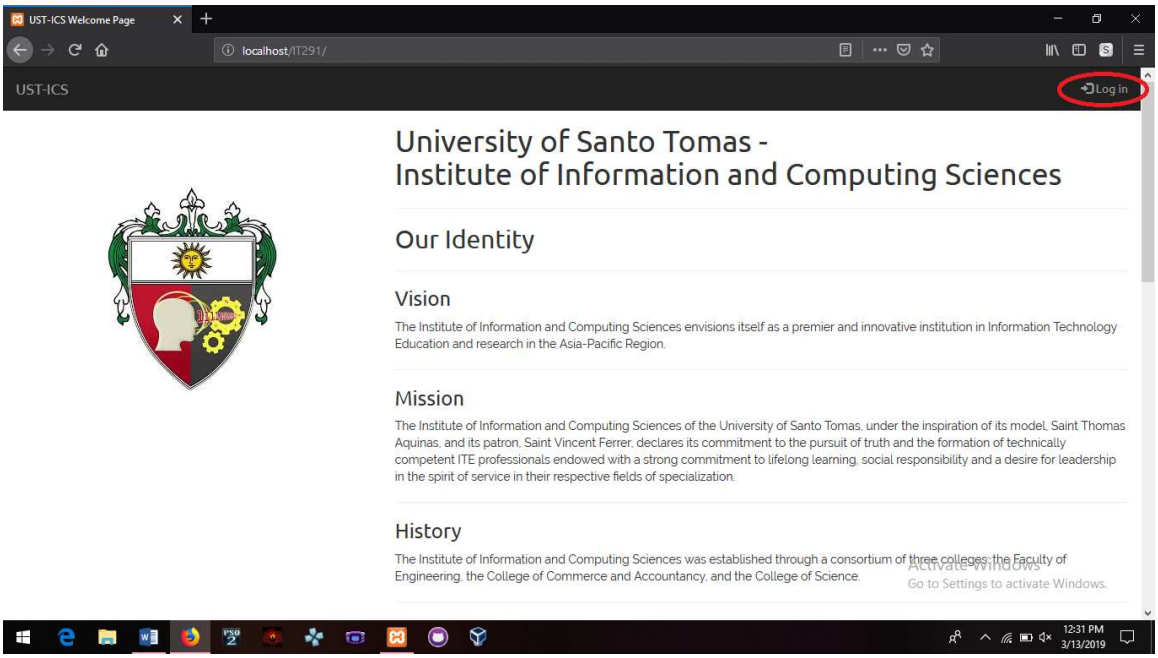
```
  $phpExcel->getActiveSheet()->getStyle('A8:A10')->applyFromArray($THINN); // End of
Scheduling layout
}


if($day == 'Monday' && $time == '07:00' && $timeOut == '07:30')     //START OF MON
          {
            $ValueSect = $phpExcel->getActiveSheet()->getCellByColumnAndRow(1,9)-
>getValue();
            $ValueRoom = $phpExcel2->getActiveSheet()->getCellByColumnAndRow(1,9)-
>getValue();
            $ValueFact = $phpExcel3->getActiveSheet()->getCellByColumnAndRow(1,9)-
>getValue();
            if(empty($ValueSect) && empty($ValueRoom) && empty($ValueFact)) //check time
constraint
            {
              $firstUNIT = $cour['labunits'] + $cour['lecunits'];
              $finalUNIT = $fact['currentUnits'] + $firstUNIT;
              if($fact['maxUnits'] < $finalUNIT) //check time constraint
              {
                Print '<script>alert("There is conflict in either Professor Units or Schedule, Please try
again.");</script>'; //Prompts the user
          Print '<script>window.location.assign("sectionView.php");</script>'; // redirects to
login.php
              }
              else {
            mysqli_query($conn, "UPDATE `faculty` SET `currenUnits` = '$finalUNIT' WHERE
`id` = '$id'");
$sql = "INSERT INTO scheduling (course, room, faculty, section, day, time, Etime) values
('$course', '$room', '$faculty', '$section', '$day', '$time', '$timeOut')";

if($days != "This" && $times != "This" && $timeOuts !== "This")
{
  $sql2 = "INSERT INTO scheduling (course, room, faculty, section, day, time, Etime) values
('$course', '$room', '$faculty', '$section', '$days', '$times', '$timeOuts')";
  $query = $bdd->prepare( $sql2 );
  if ($query == false) {
   print_r($bdd->errorInfo());
   die ('Prepare Error');
  }
  $sth = $query->execute();
  if ($sth == false) {
   print_r($query->errorInfo());
  }
}
echo $sql;
$query = $bdd->prepare( $sql );
if ($query == false) {
 print_r($bdd->errorInfo());
 die ('Prepare Error');
}
```

```
$sth = $query->execute();
if ($sth == false) {
 print_r($query->errorInfo());
 die ('Execute Error');
}
            $phpExcel->getActiveSheet()->SetCellValue("B8", $room);
            $phpExcel->getActiveSheet()->SetCellValue("C8", $section);
            $phpExcel->getActiveSheet()->SetCellValue("B9", $course);
            $phpExcel->getActiveSheet()->getStyle('B9:B10')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
            $phpExcel->getActiveSheet()->getStyle('B9')->getFont()->setBold( true )->setSize(12);
            $phpExcel->getActiveSheet()->SetCellValue("B10", $faculty);

            $phpExcel2->getActiveSheet()->SetCellValue("B8", $room);
            $phpExcel2->getActiveSheet()->SetCellValue("C8", $section);
            $phpExcel2->getActiveSheet()->SetCellValue("B9", $course);
            $phpExcel2->getActiveSheet()->getStyle('B9:B10')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
            $phpExcel2->getActiveSheet()->getStyle('B9')->getFont()->setBold( true )-
>setSize(12);
            $phpExcel2->getActiveSheet()->SetCellValue("B10", $faculty);

            $phpExcel3->getActiveSheet()->SetCellValue("B8", $room);
            $phpExcel3->getActiveSheet()->SetCellValue("C8", $section);
            $phpExcel3->getActiveSheet()->SetCellValue("B9", $course);
            $phpExcel3->getActiveSheet()->getStyle('B9:B10')->getAlignment()-
>setHorizontal(PHPExcel_Style_Alignment::HORIZONTAL_CENTER);
            $phpExcel3->getActiveSheet()->getStyle('B9')->getFont()->setBold( true )-
>setSize(12);
            $phpExcel3->getActiveSheet()->SetCellValue("B10", $faculty);
          }
          }
          else
          {
          Print '<script>alert("There is conflict in either Professor Units or Schedule, Please try
again.");</script>'; //Prompts the user
          Print '<script>window.location.assign("sectionView.php");</script>'; // redirects to
login.php
          }
          }
```

User's Guide, and Sample Outputs

1. Registering an account

    1.1 To register an account, click "Log in" at the upper-right corner of the website, then click "Register" on the popup window that appears.



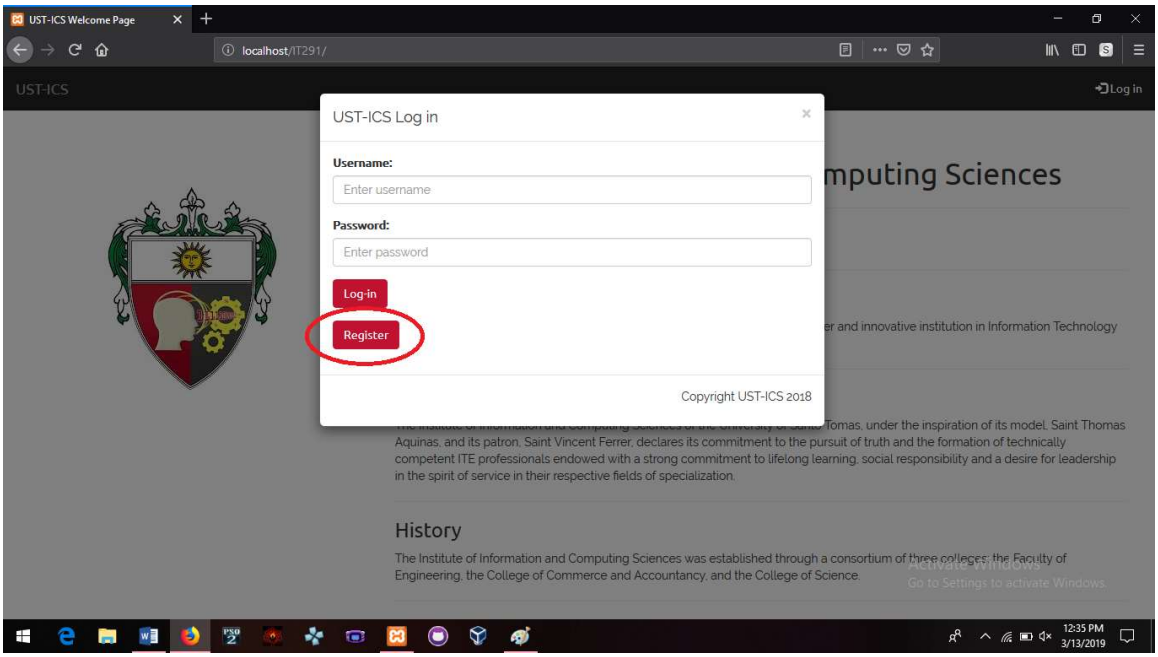


Figure 5.2.1: Login to Register page

1.2 In the Register window, enter your email, username, and password. Confirm your password, then select your department.

1.2.1 The password MUST have an uppercase, lowercase, AND a number. It also needs to be at least eight (8) characters long.

1.2.2 If a Department Head is the one registering, a checkbox for Dept. Head identification is also available.

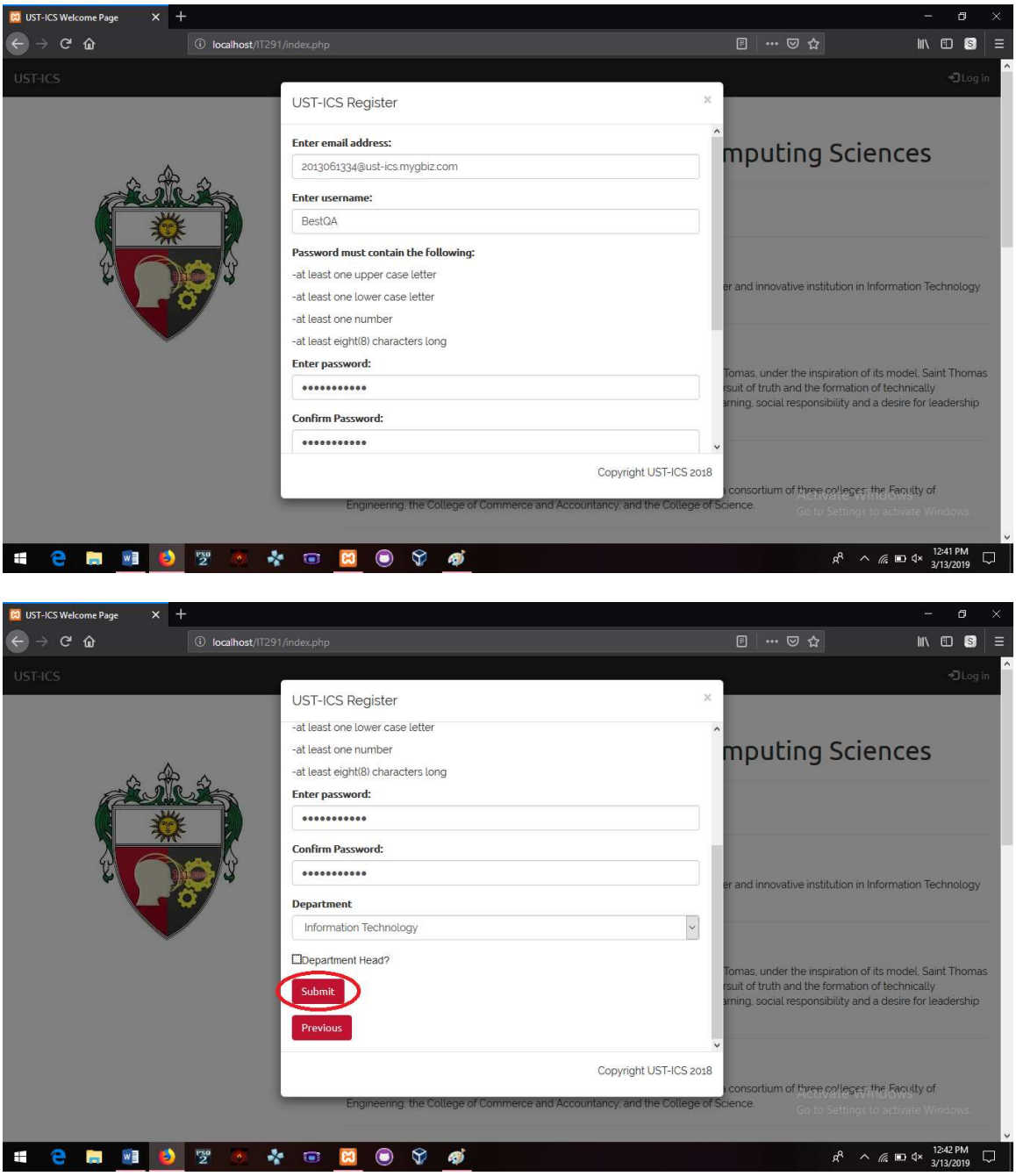


Figure 5.2.2: Registering an account

1.3 The admin will receive an email once the verification request has been submitted. Once the admin verifies your request, you may login into the system.

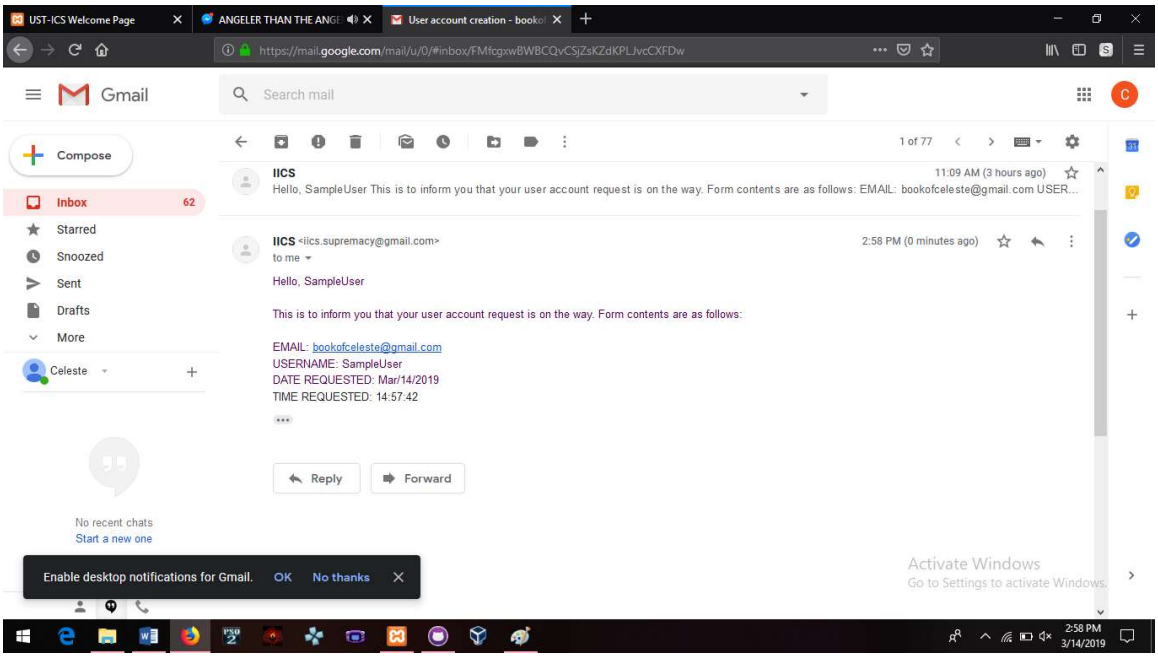


Figure 5.2.3: Email after account creation

1.3.1 You will receive an email once your account has been verified.

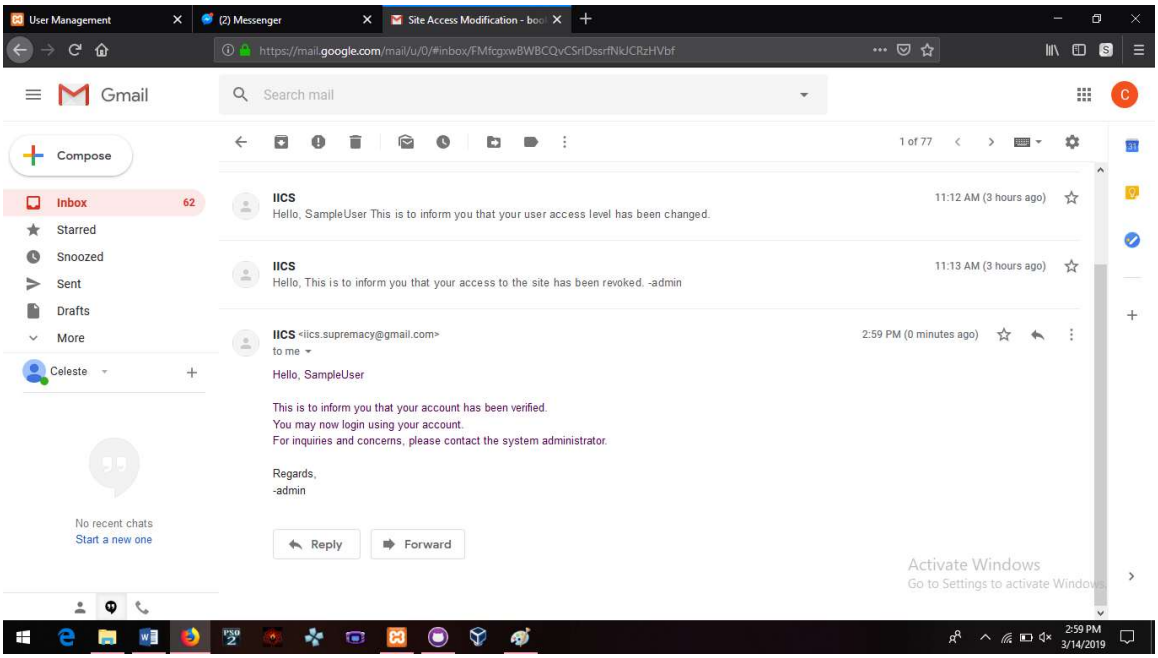


Figure 5.2.4: Email after account verification

2. Logging in

2.1 Once your account has been verified, return to the website and click “Log in”.

2.2 Enter your username and password, then click “Log-in”.

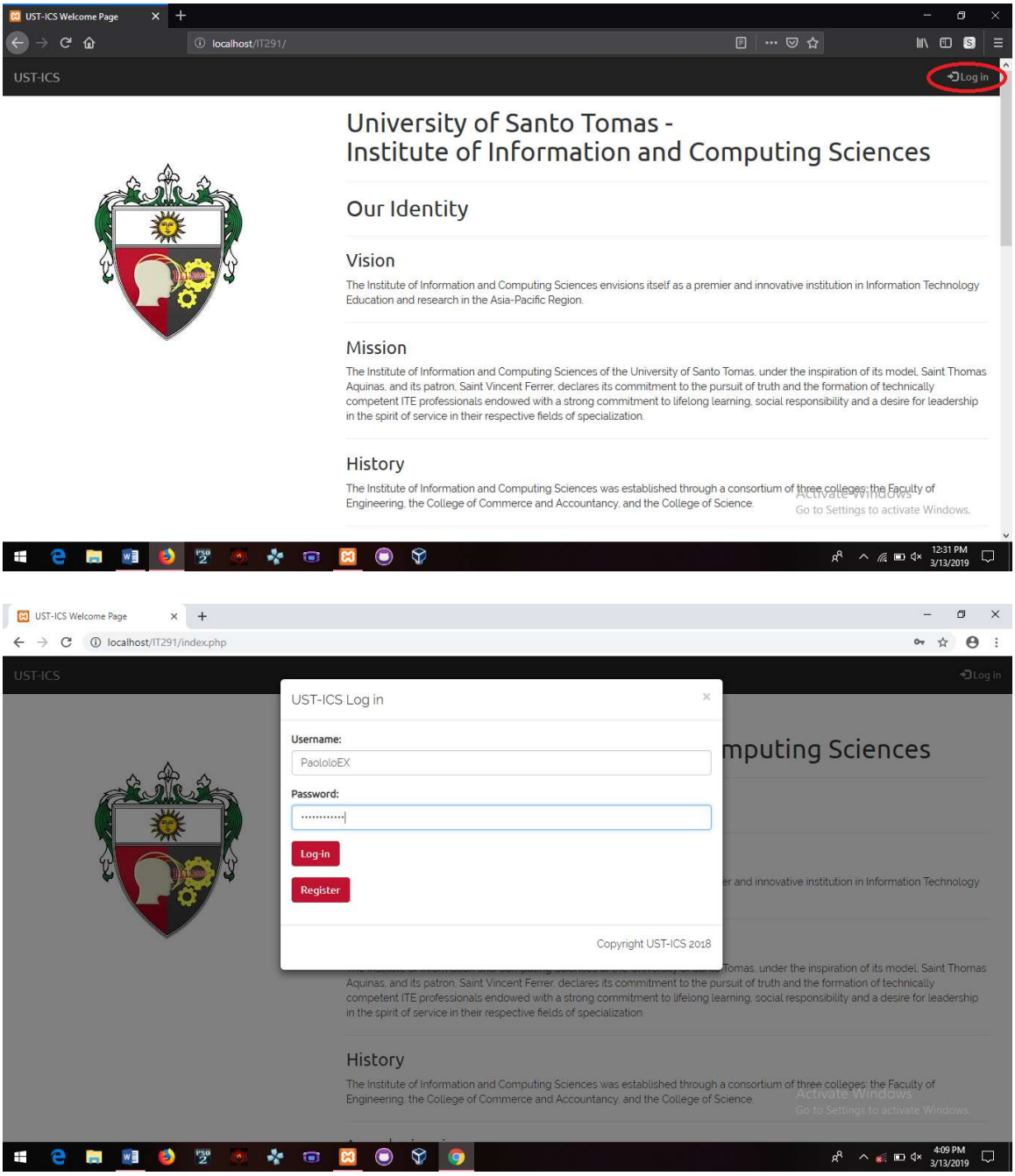


Figure 5.2.5: Logging in

3. Creating a reservation

3.1 Once logged in, click on the “All Rooms” dropdown to display the list of rooms.

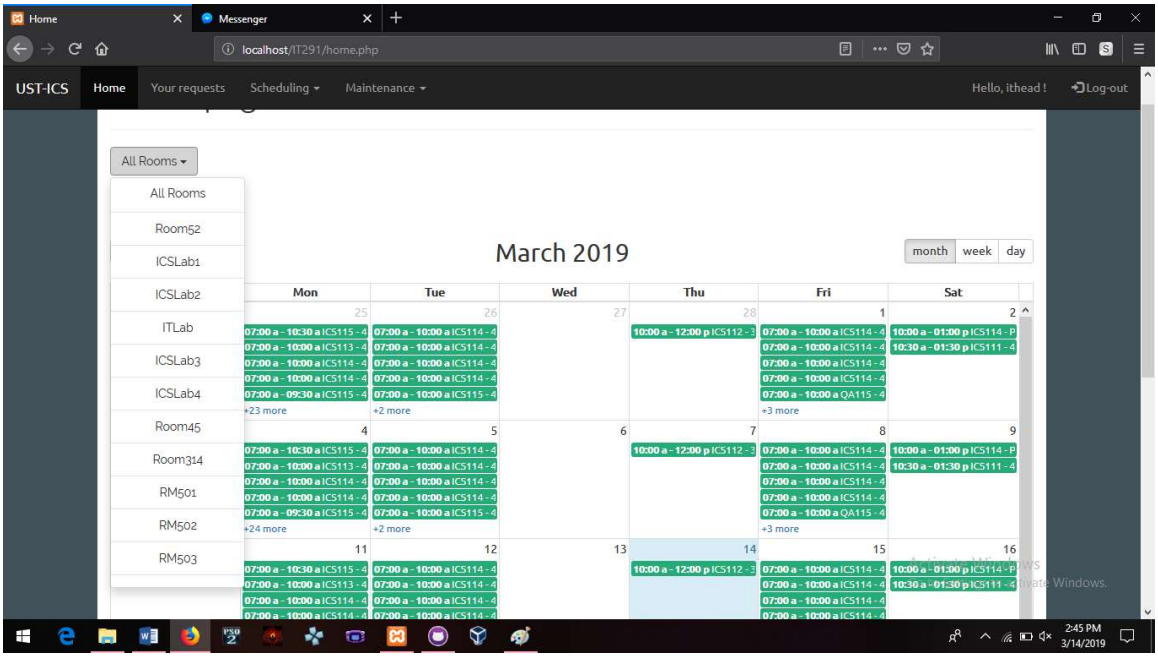


Figure 5.2.6: "All Rooms" dropdown

3.2 Select a room.

3.3 Once a room has been selected, click on a calendar slot.

3.3.1 You can only select the current or succeeding dates in the calendar.

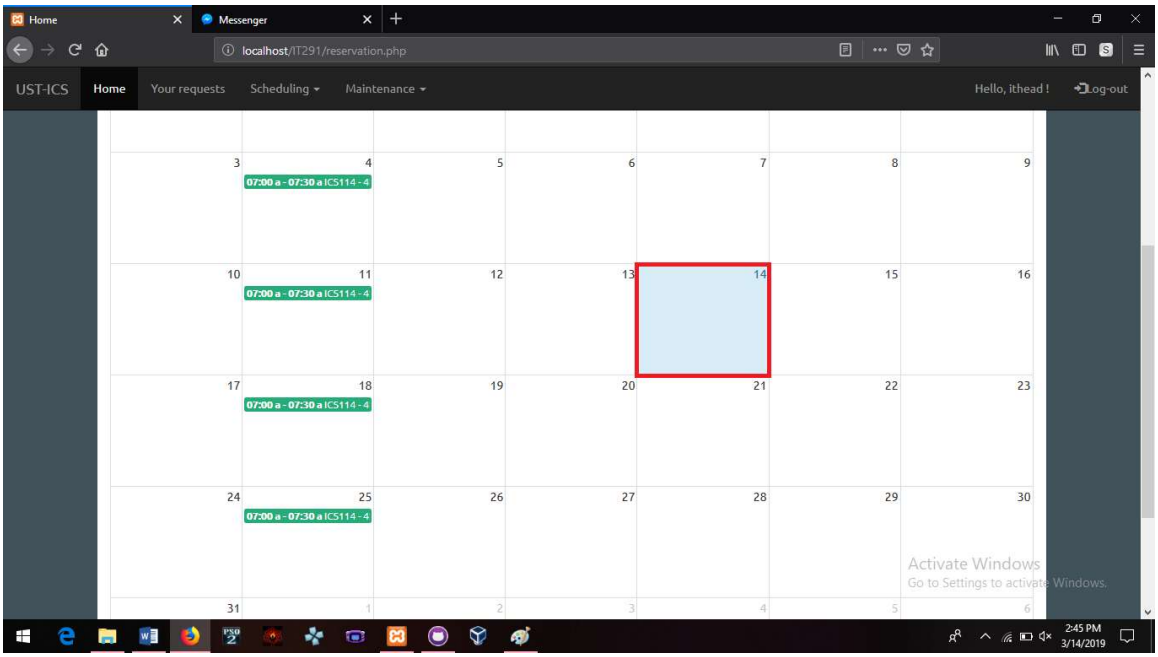


Figure 5.2.7: Click on a calendar slot

3.4 A popup window generates once a slot has been selected. In this window, enter the title of your reservation request. Then, select the department, section, course, purpose, room, start time, and end time. Finally, click "Request".

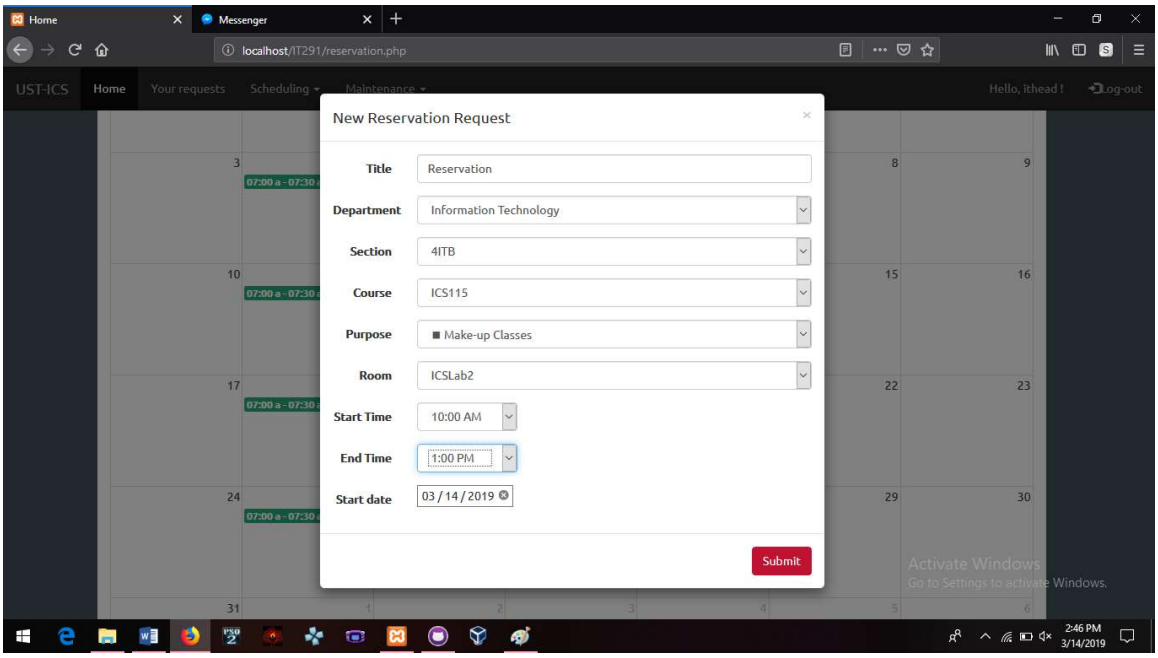


Figure 5.2.8: Reservation Form

3.5 Your request will be sent to the admin for approval. Once the admin approves your request, you will receive an email.

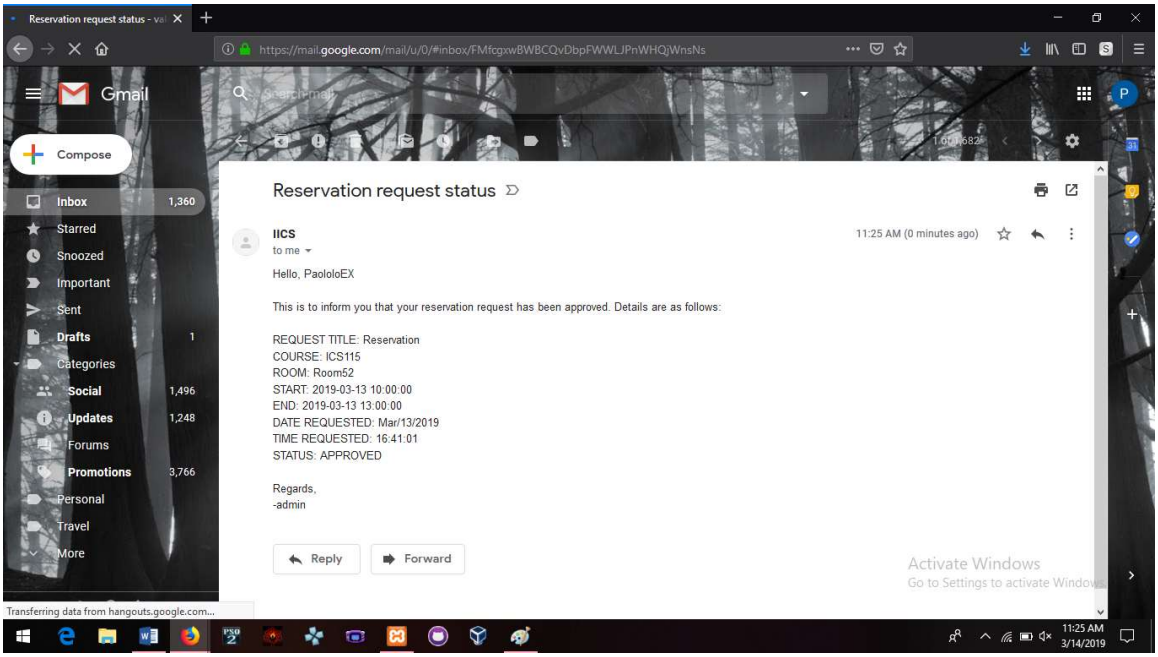


Figure 5.2.9: Approval of Reservation Request

3.5.1 Should the admin deny your request, you will also receive an email along with the reason of denial.

4. Viewing reservation requests

4.1 To view reservation requests, click on “Your requests” on the homepage.

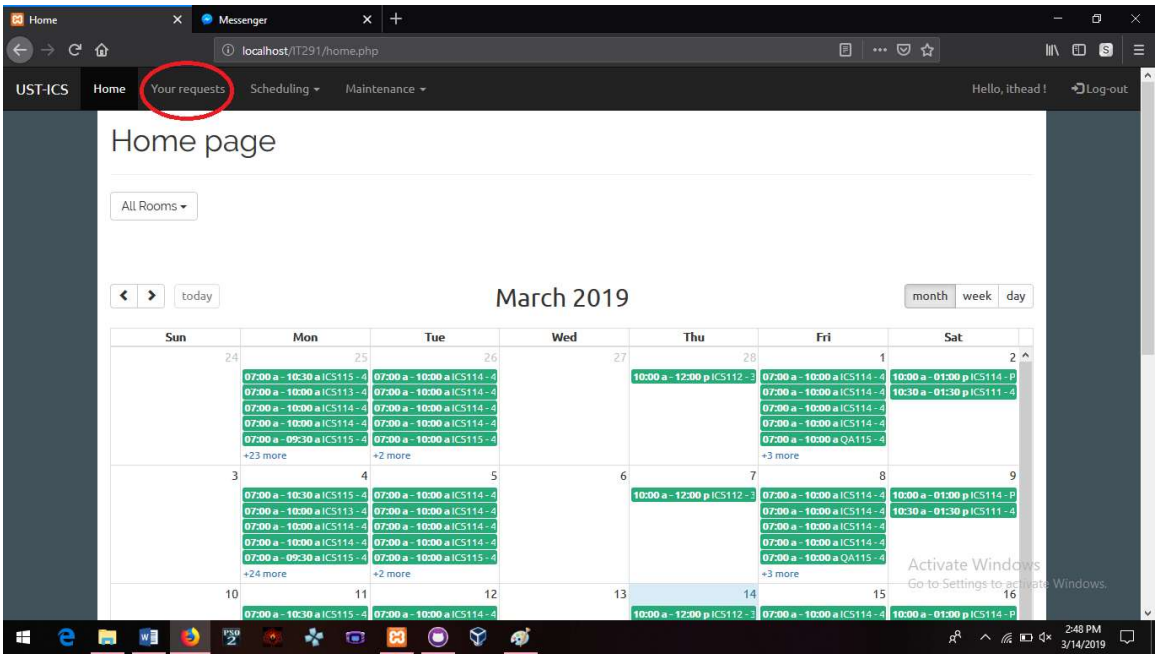


Figure 5.2.10: "Your requests" button

4.2 The list of pending, approved, and denied reservation requests will be displayed.

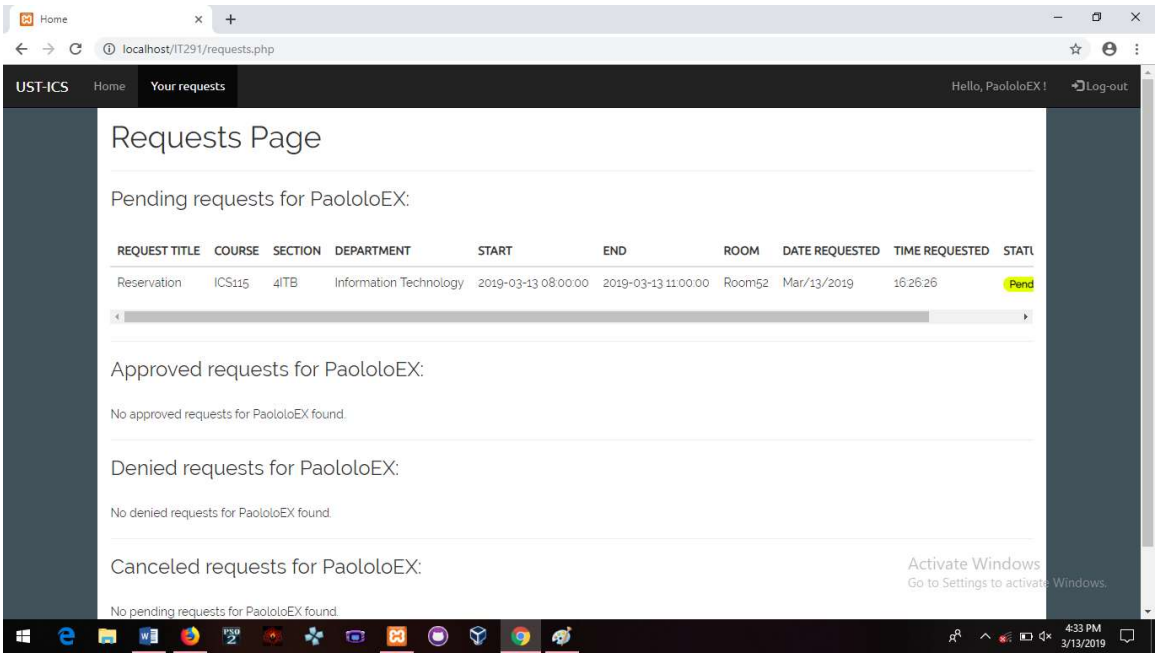


Figure 5.2.11: Requests Page

5. Cancelling a reservation request

5.1 To cancel a reservation request, click on “Your requests”.

5.2 In the Requests Page, click “Cancel” on a reservation request.

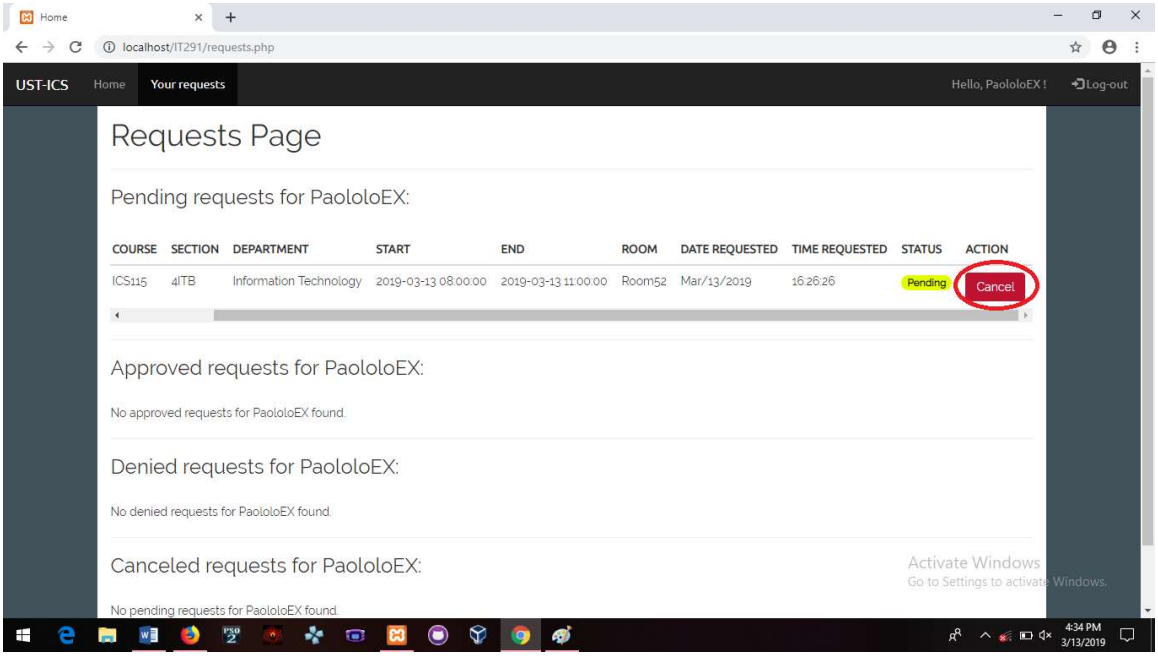


Figure 5.2.12: Requests "Cancel" button

5.3 Enter the Reason for Cancellation in the form that appears.

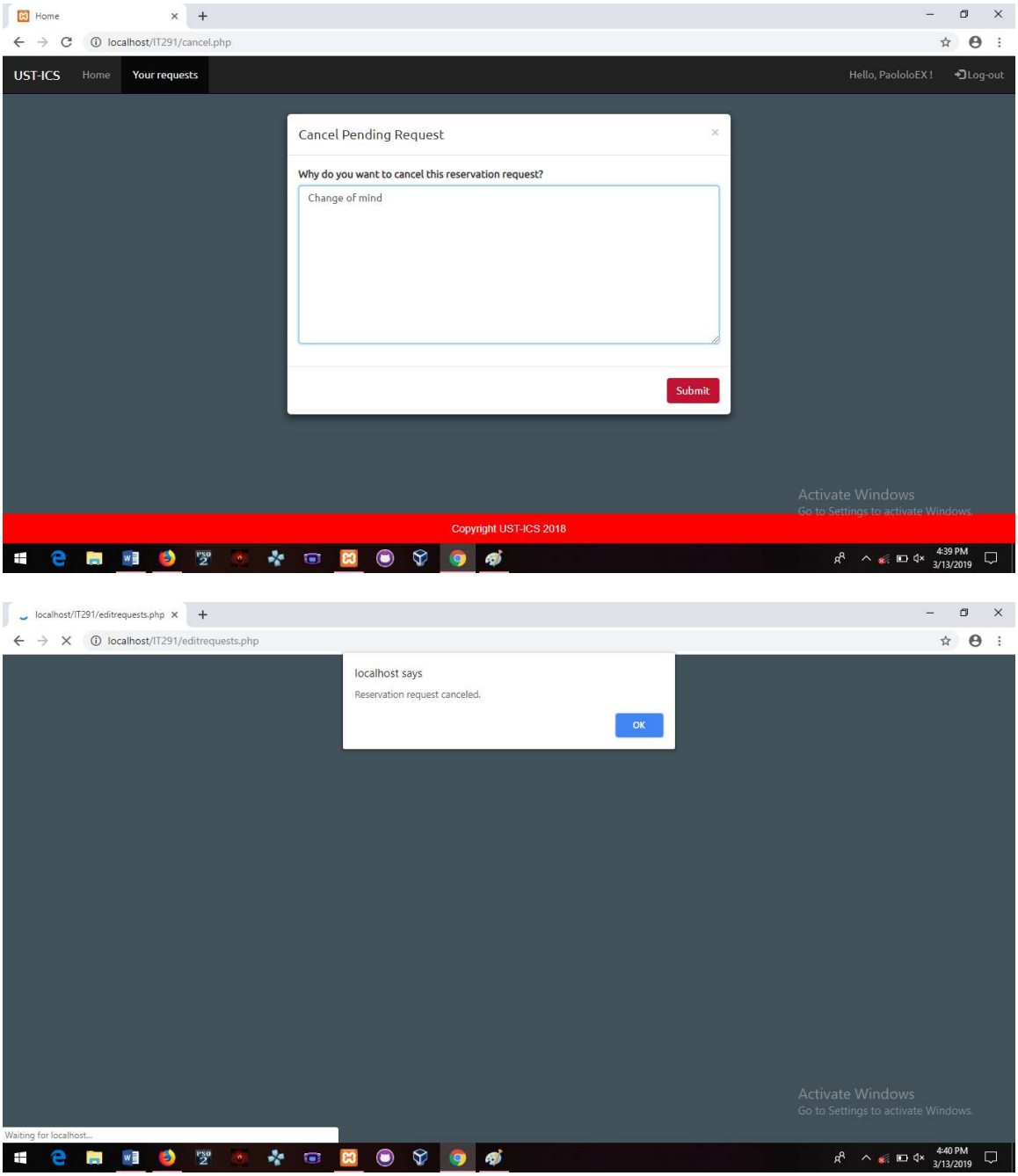


Figure 5.2.13: Reason for Cancellation

5.4 The request will be cancelled and removed from the pending list.

For both ADMIN and DEPARTMENT HEAD:

1. Using the Maintenance function

1.1 Checking the list of current data

1.1.1 To check the list of current data in the system, click on "Maintenance" in the homepage, then click on either Room, Faculty, Course, or Section. This will redirect you on the Management window.

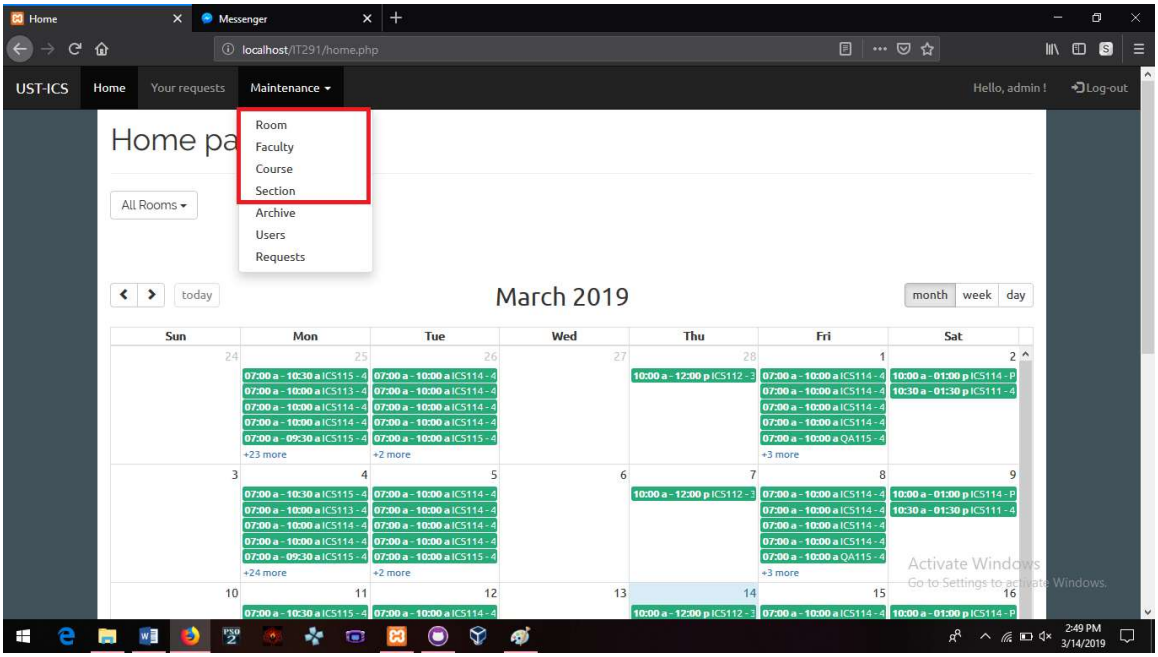


Figure 5.2.14: Maintenance for Room, Faculty, Course, and Section

1.1.2 Inside the Management window, the dropdown "Current rooms/faculty/courses/sections in the database" is available. Click on this dropdown to display the current data in the system.

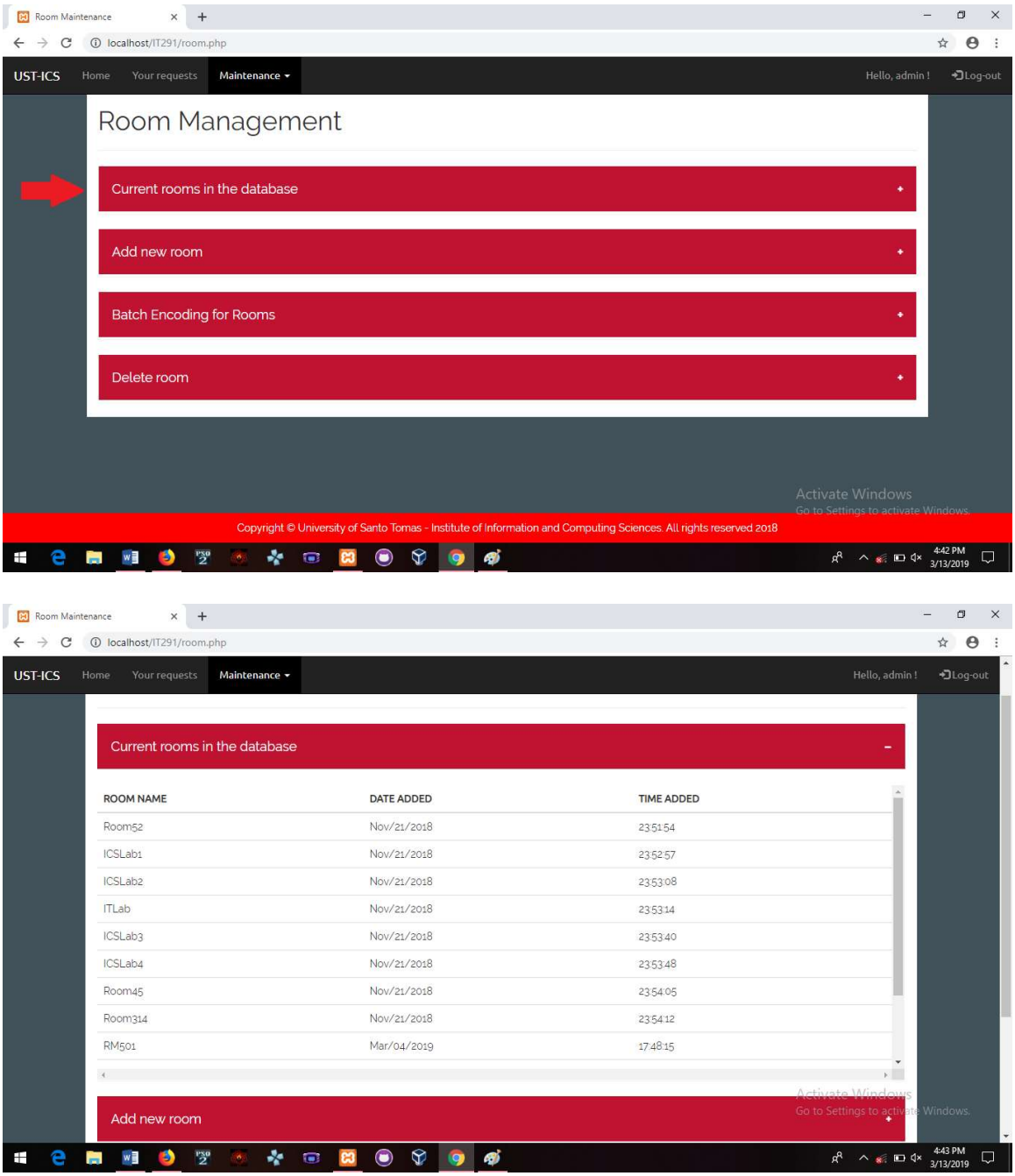


Figure 5.2.15: Current list

1.2 Adding data

1.2.1 To add data into the system, go to the Management window, then click on the “Add…” dropdown.

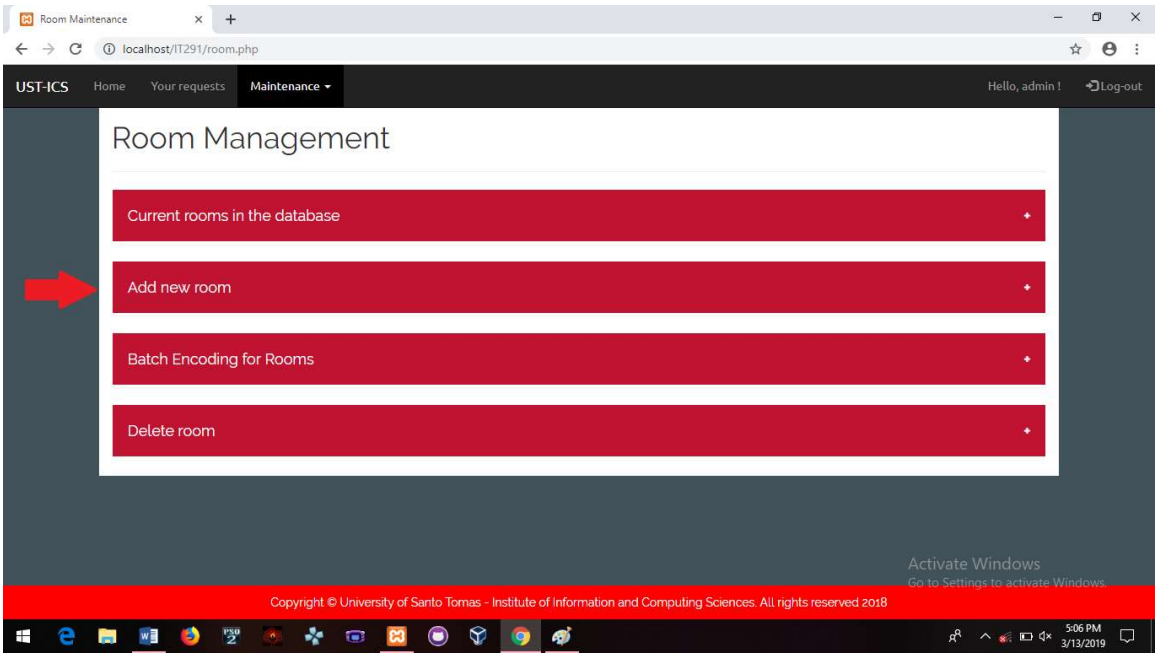
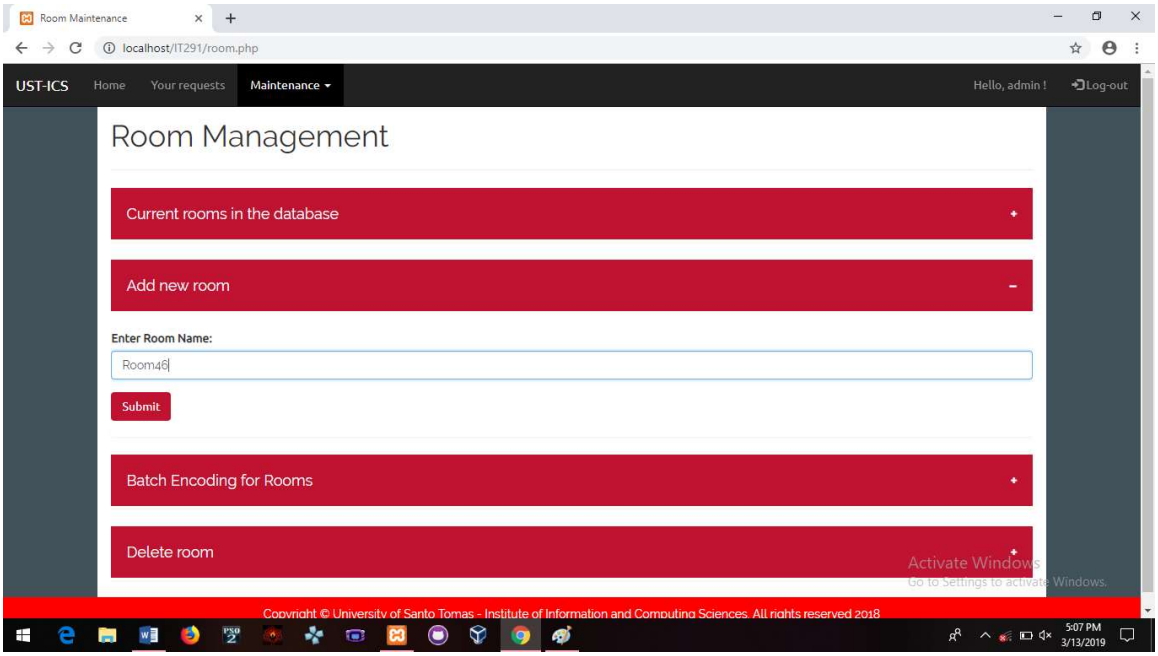


Figure 5.2.16: "Add new..." dropdown

1.2.2 Enter data into the fields displayed, then click “Submit”.



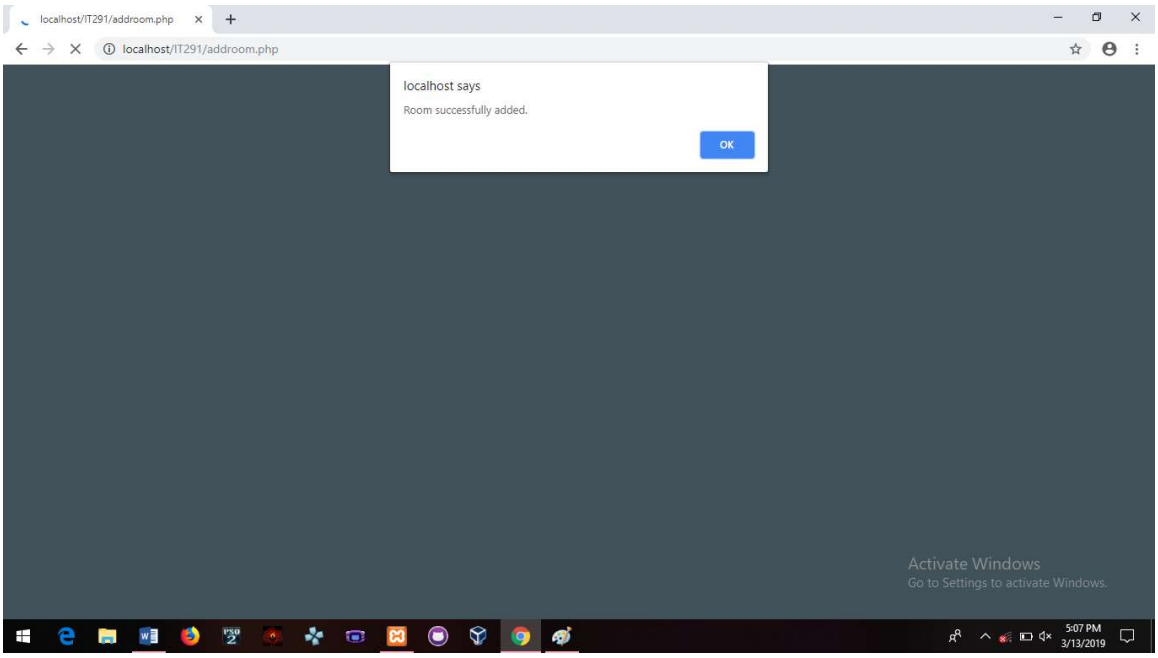


Figure 5.2.17: Adding new data

1.2.3 The newly added data should be displayed in the current list.

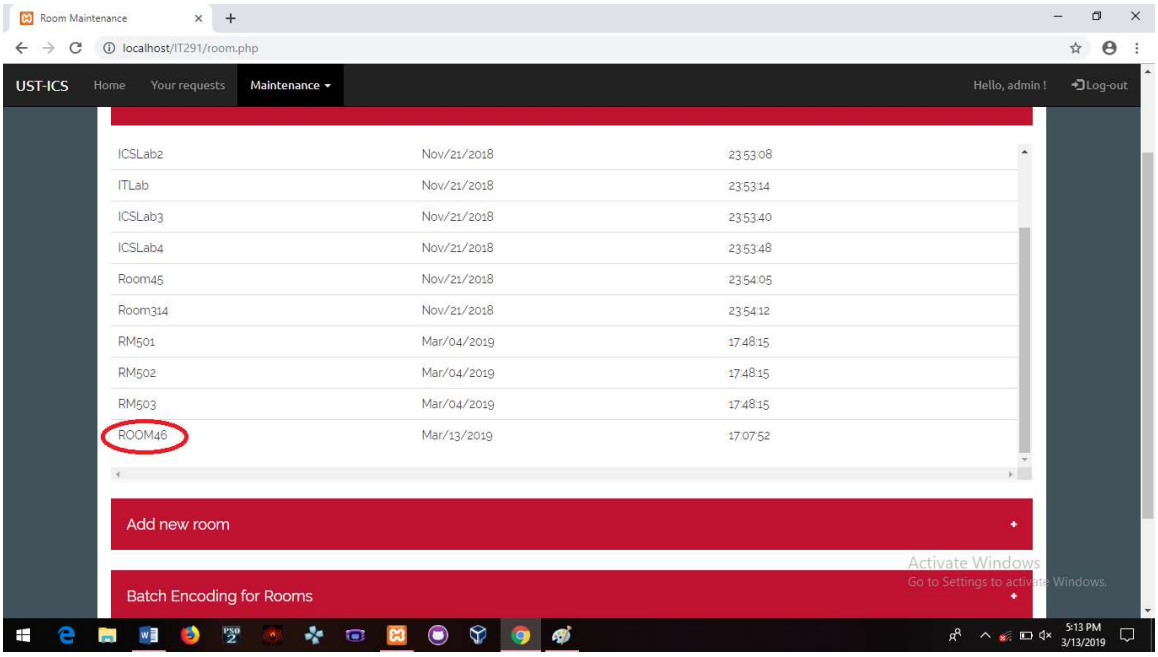


Figure 5.2.18: New data in current list

1.3 Batch encoding

1.3.1 To add data as a batch, go to the Management window, then click on the “Batch encoding…” dropdown.

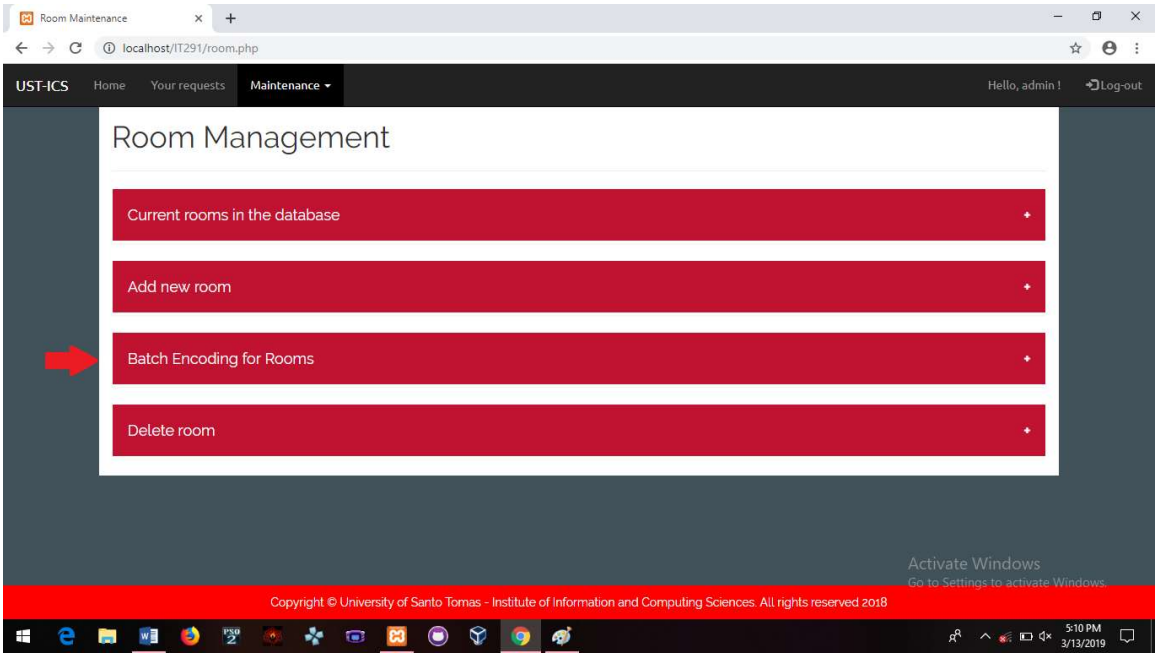


Figure 5.2.19: "Batch Encoding" dropdown

1.3.2 Enter the number of desired entries, then click “Submit”.

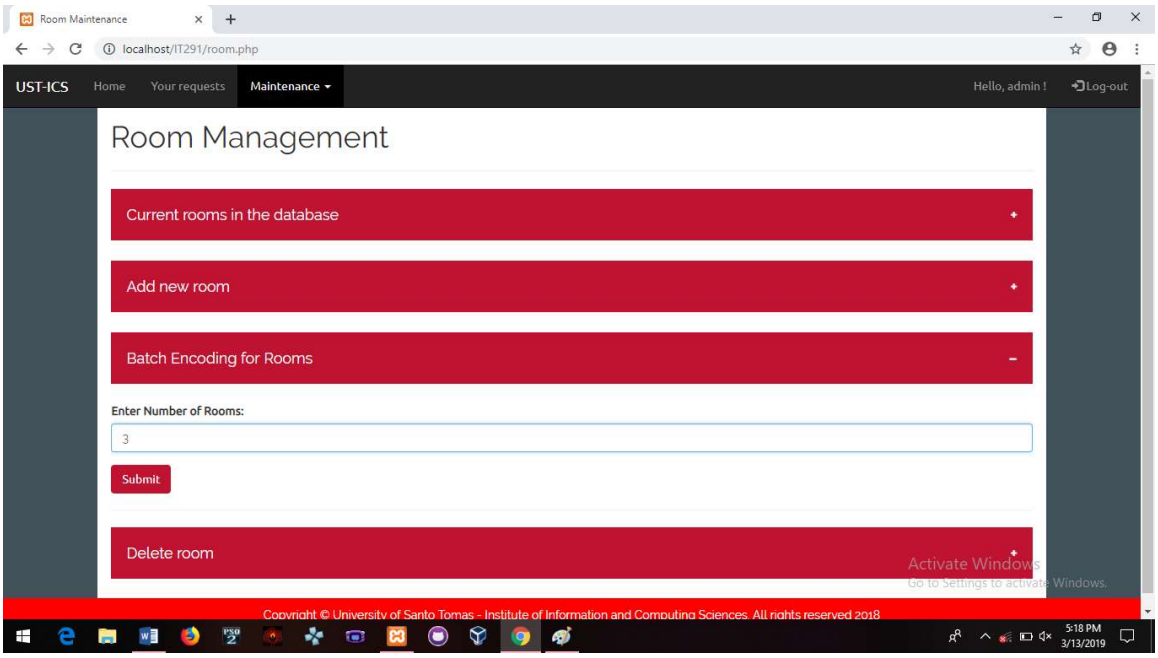


Figure 5.2.20: Batch Encoding number of entries

1.3.3 Enter data into the fields, then click “Submit”

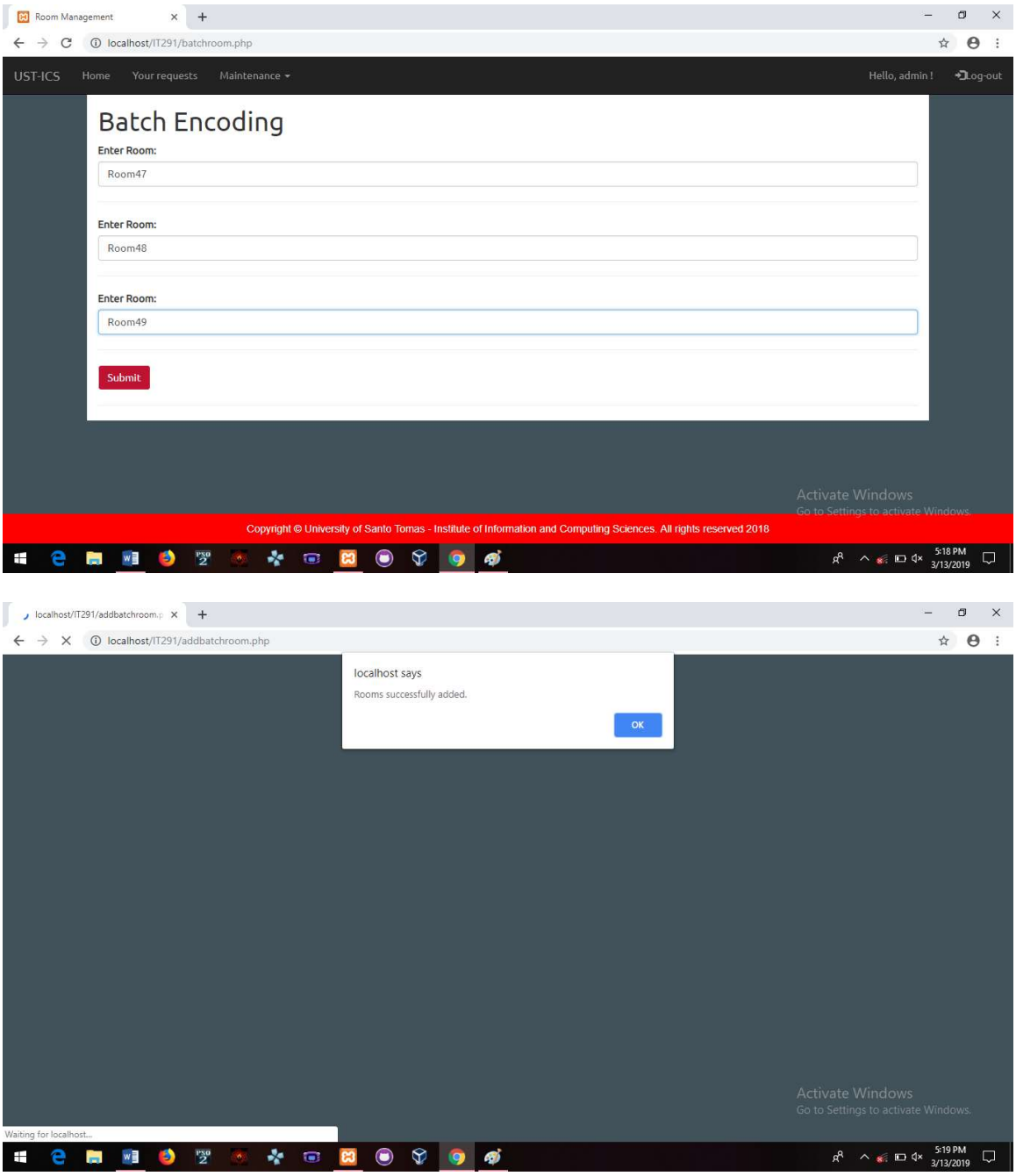


Figure 5.2.21: Batch Encoding data entries

1.3.4 The newly added data should be displayed in the current list.

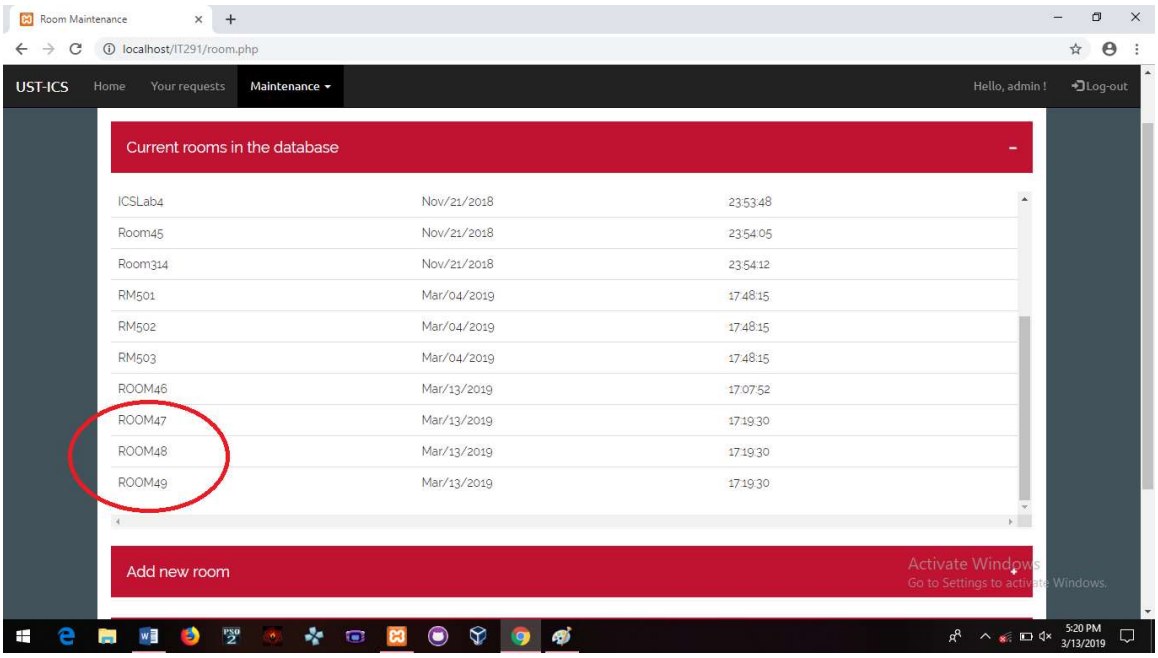


Figure 5.2.22: Batch Encoding result

1.4 Deleting data

1.4.1 To delete data from the system, go to the Management window, then click on the “Delete…” dropdown.

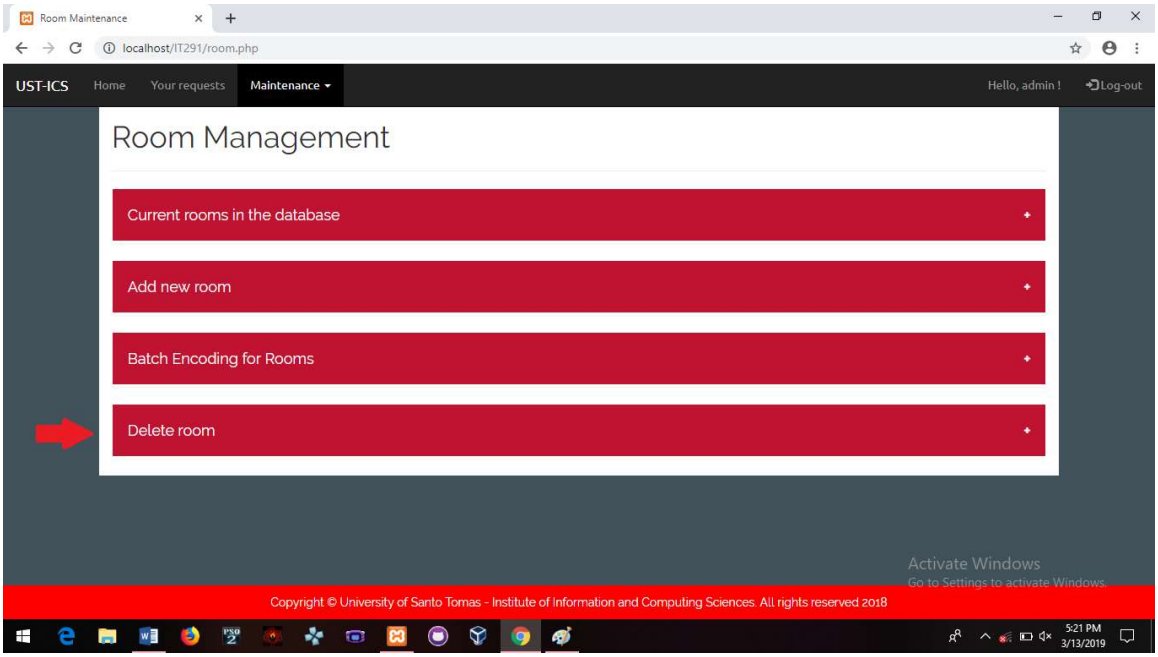


Figure 5.2.23: "Delete..." dropdown

1.4.2 Select data to be deleted, then click “Delete”.

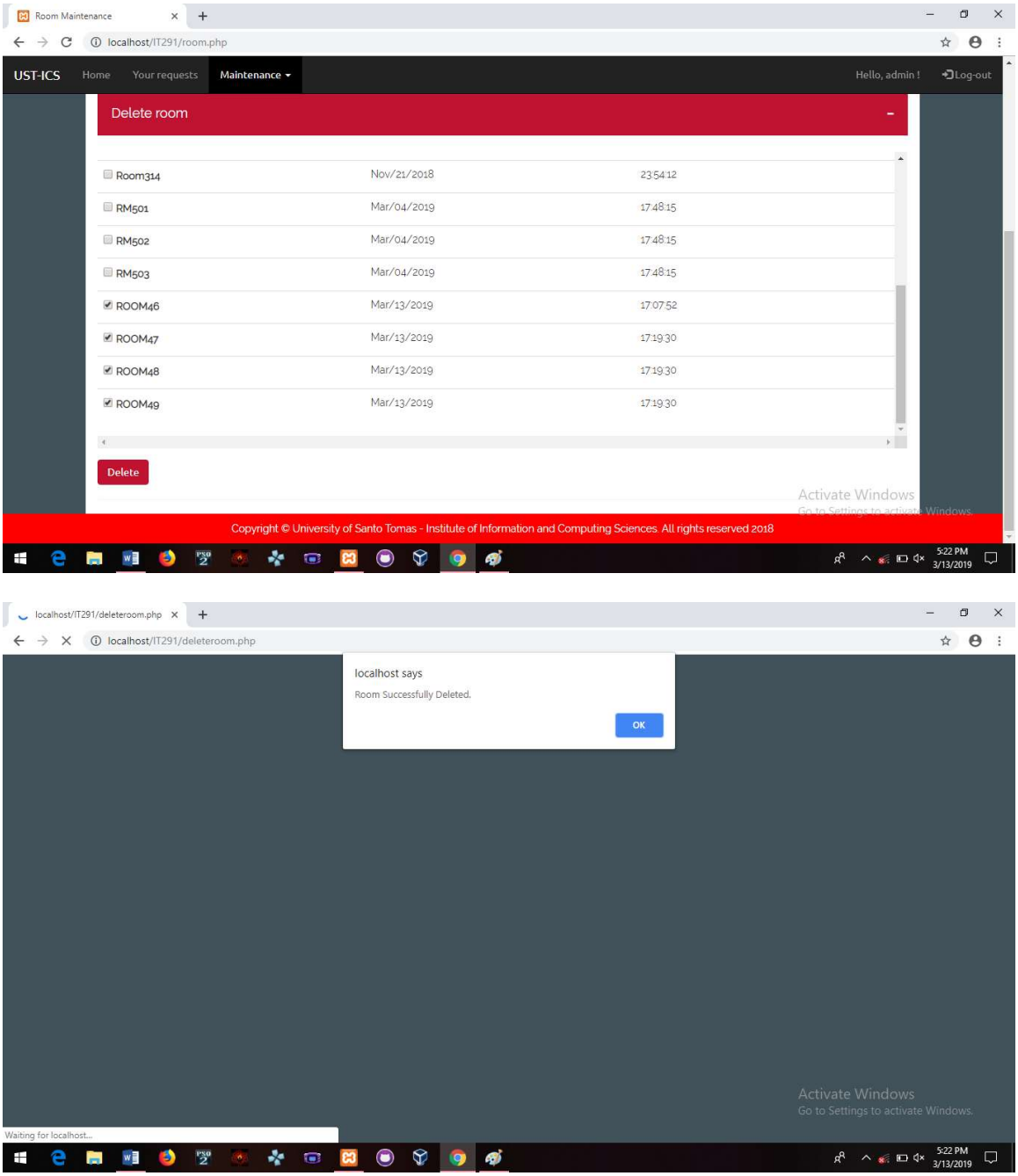


Figure 5.2.24: Data deletion

1.4.3 The selected data should be removed from the system.

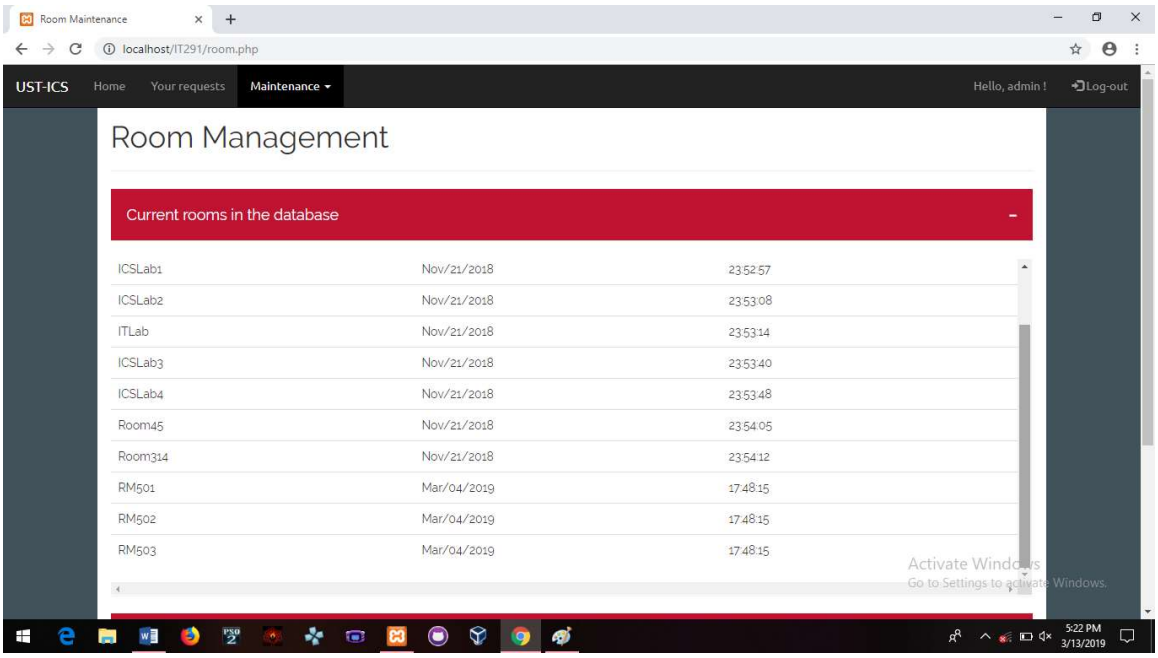


Figure 5.2.25: Delete data result

FOR ADMIN ONLY:

1. Checking the list of current users

    1.1 To check the list of current users in the system, click on “Maintenance”, then click “Users”.

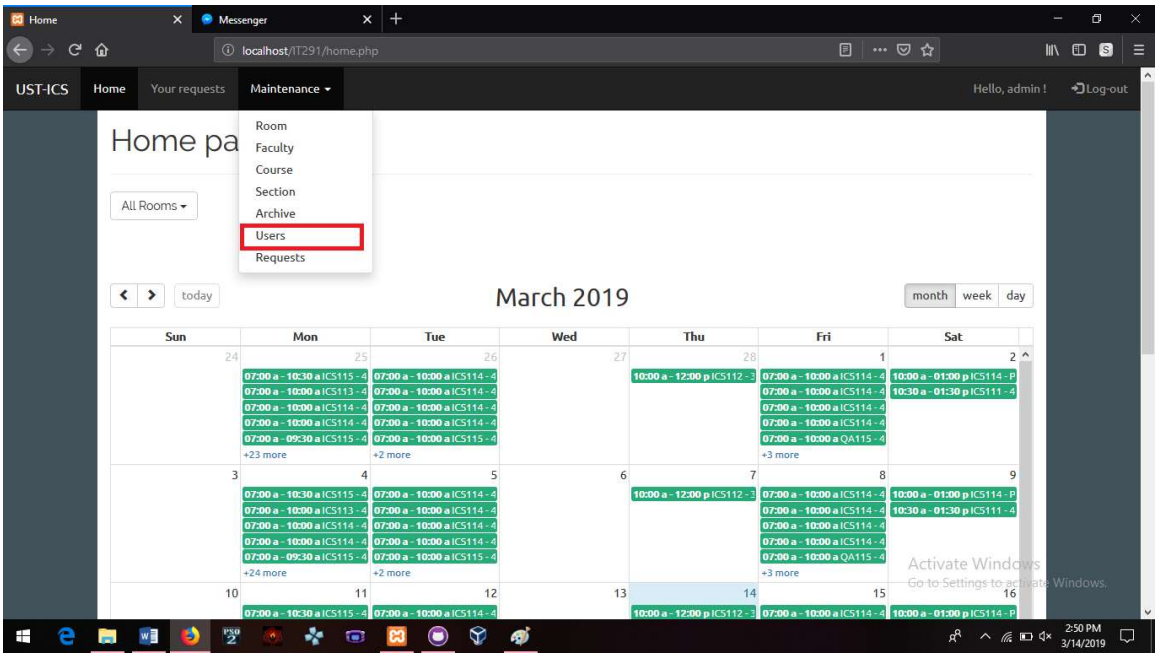
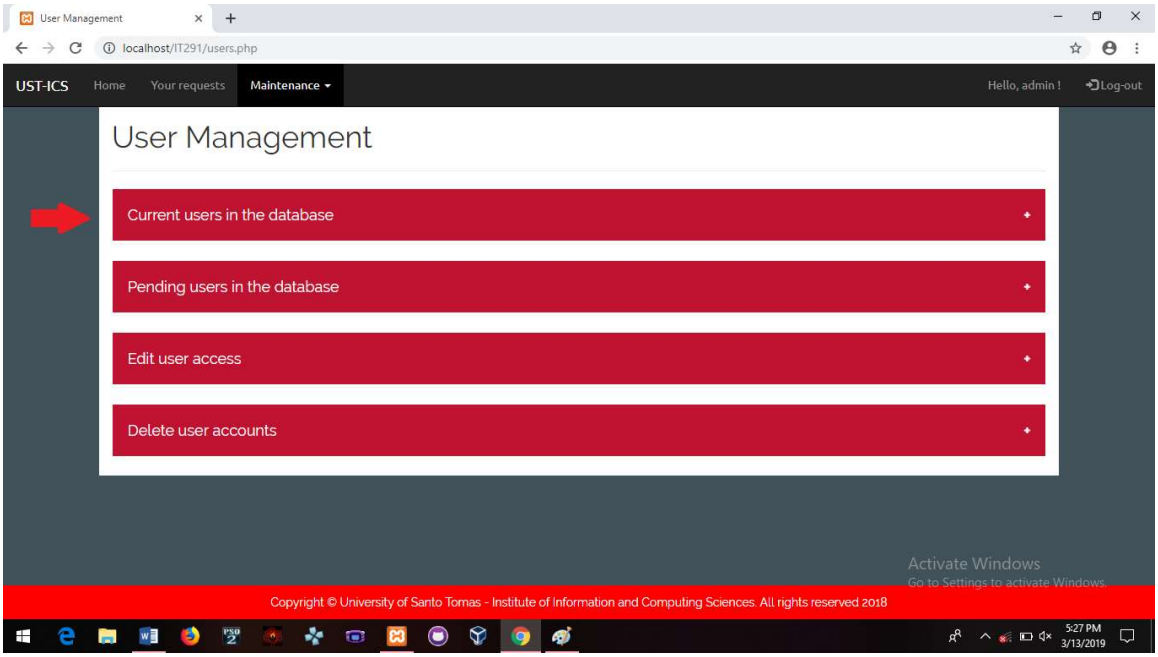


Figure 5.2.26: Maintenance for "Users"

1.2 Inside the User Management window, click on the “Current users in the database” dropdown to display the list of users in the system.



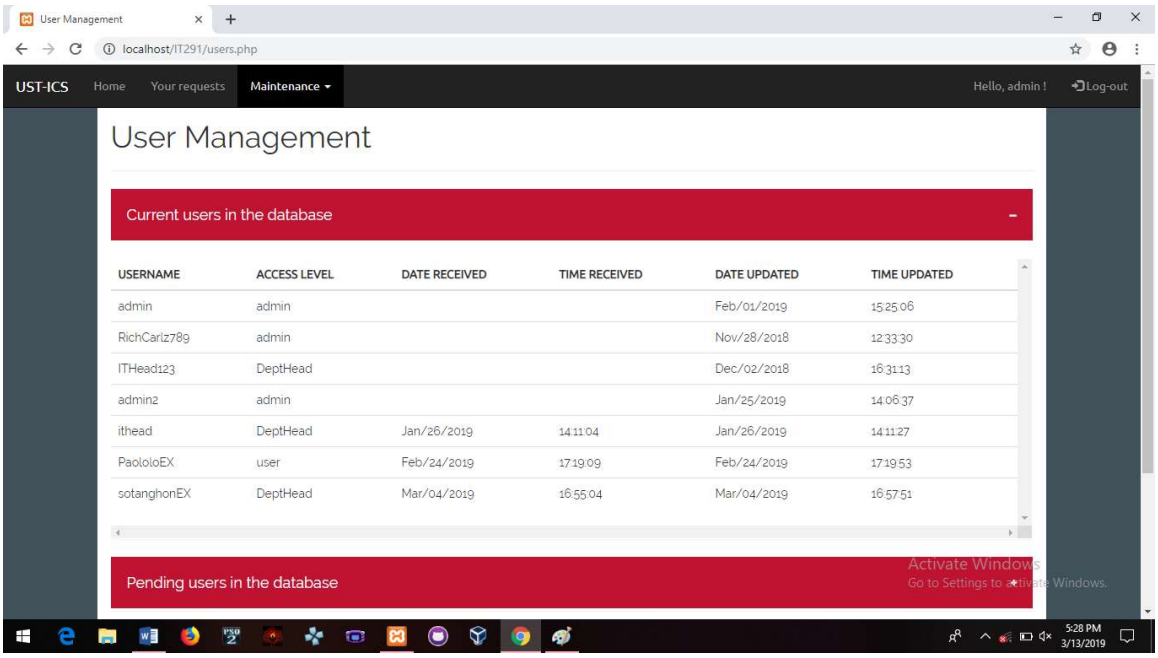
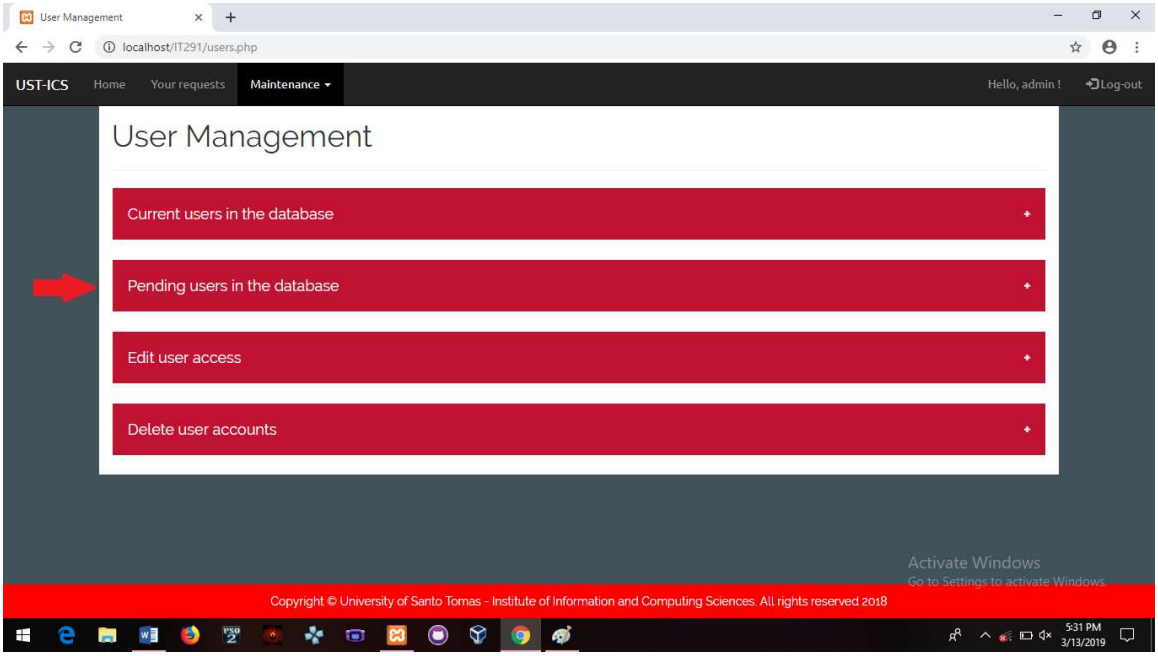


Figure 5.2.27: "Current users" dropdown

2 Verifying user accounts

2.1 To verify an account, go to the User Management window, then click on the “Pending users in the database” dropdown.



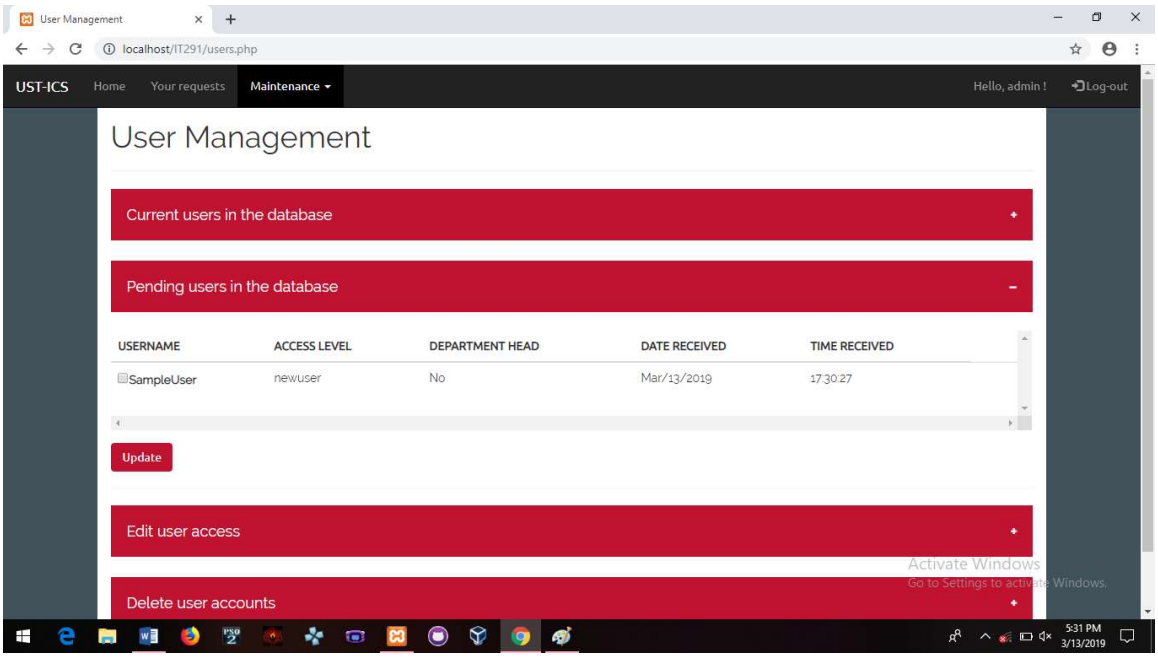
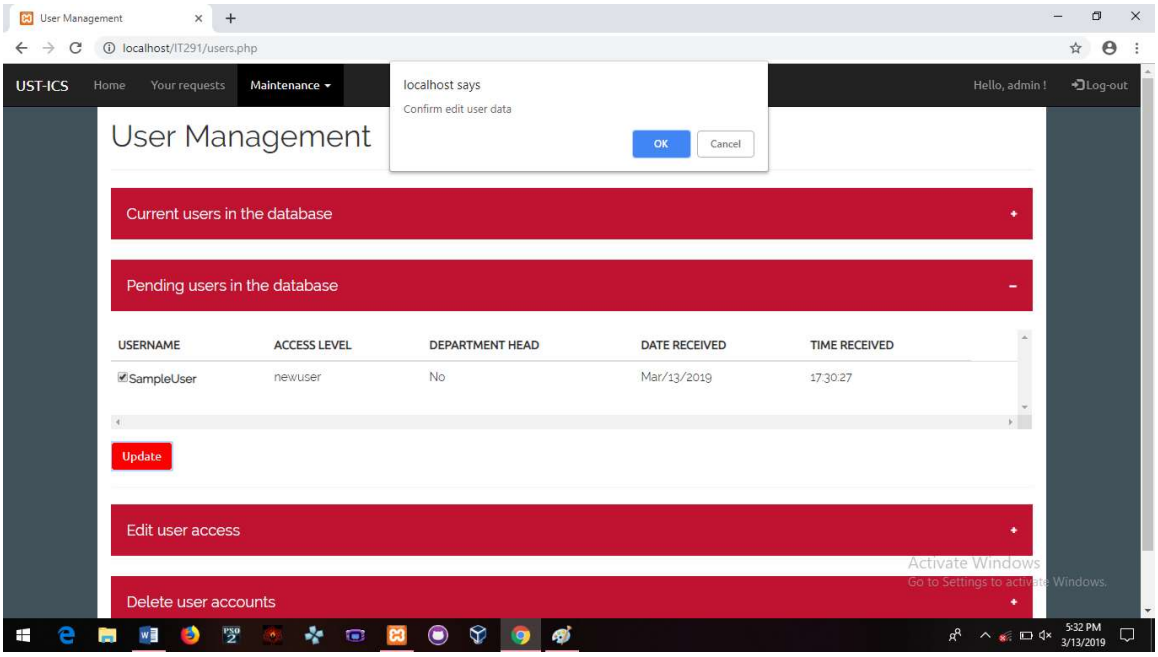


Figure 5.2.28: "Pending users" dropdown

2.2 Select the user/s you want to verify, then click on “Update”.



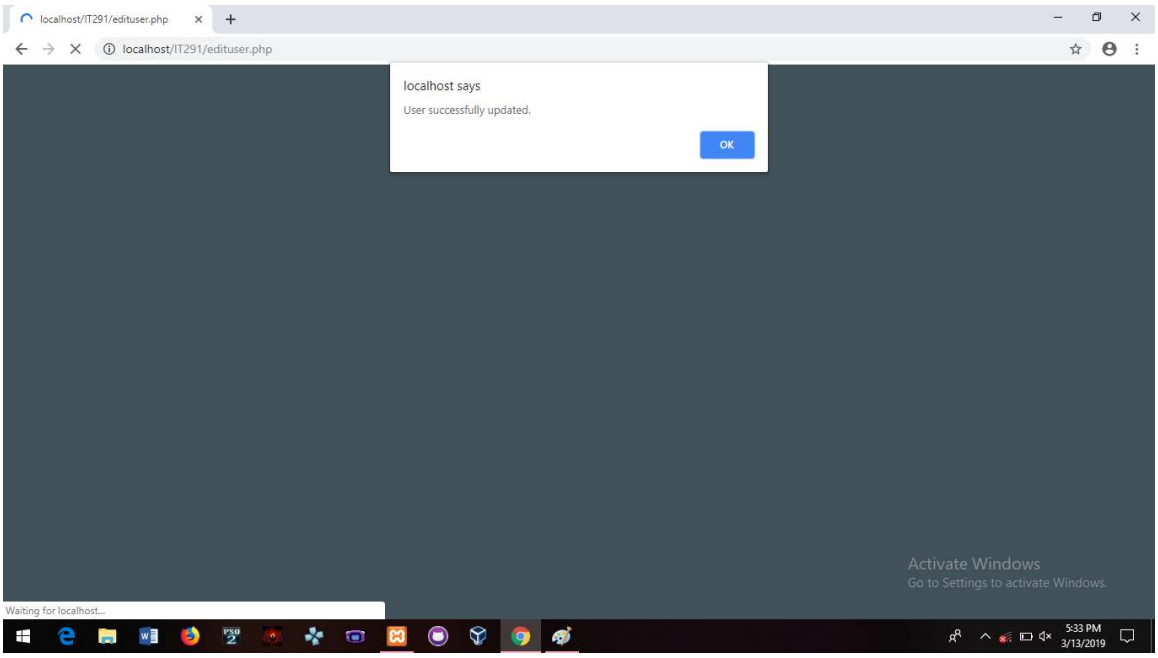
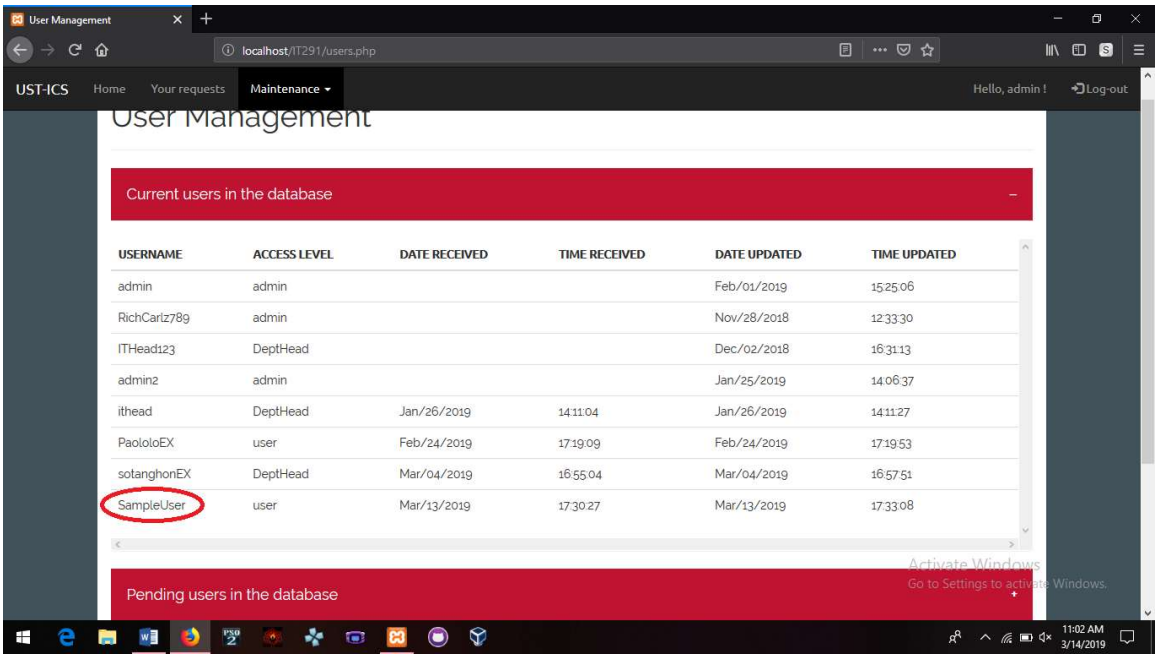


Figure 5.2.29: Account verification

2.3 The selected user's access level should become "user" once updated, and the particular user will receive an email regarding the update.



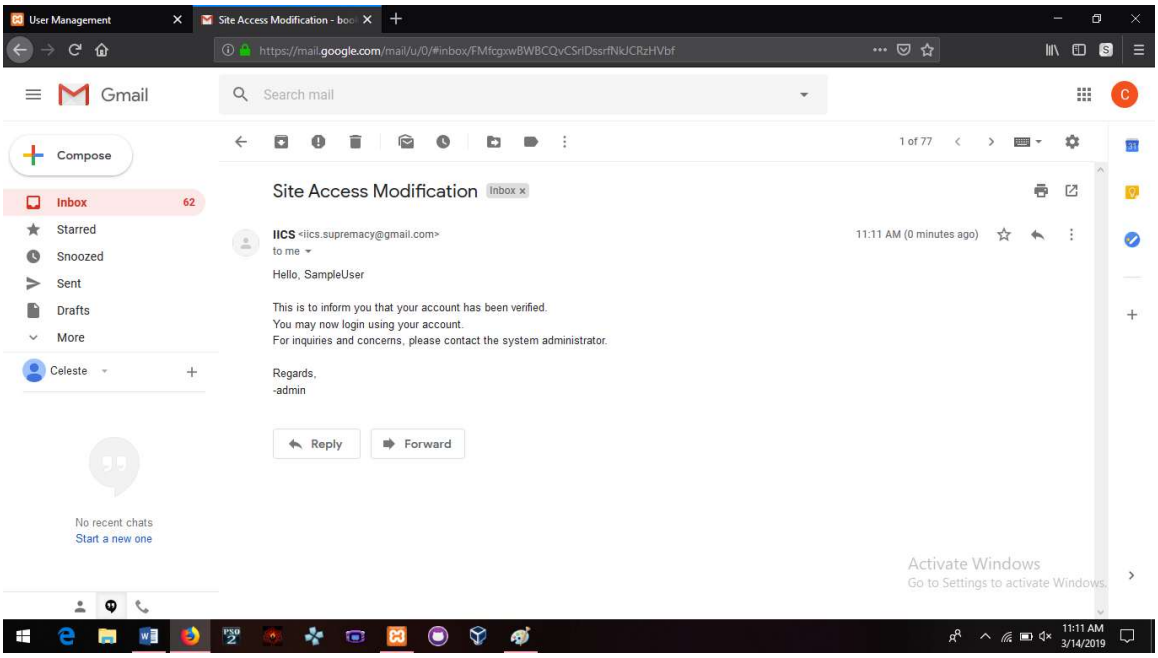


Figure 5.2.30: Email to user after verification

3 Updating user access level

3.1 To **make a user into an admin**, go to the User Management window, then click on the "Edit user access" dropdown.

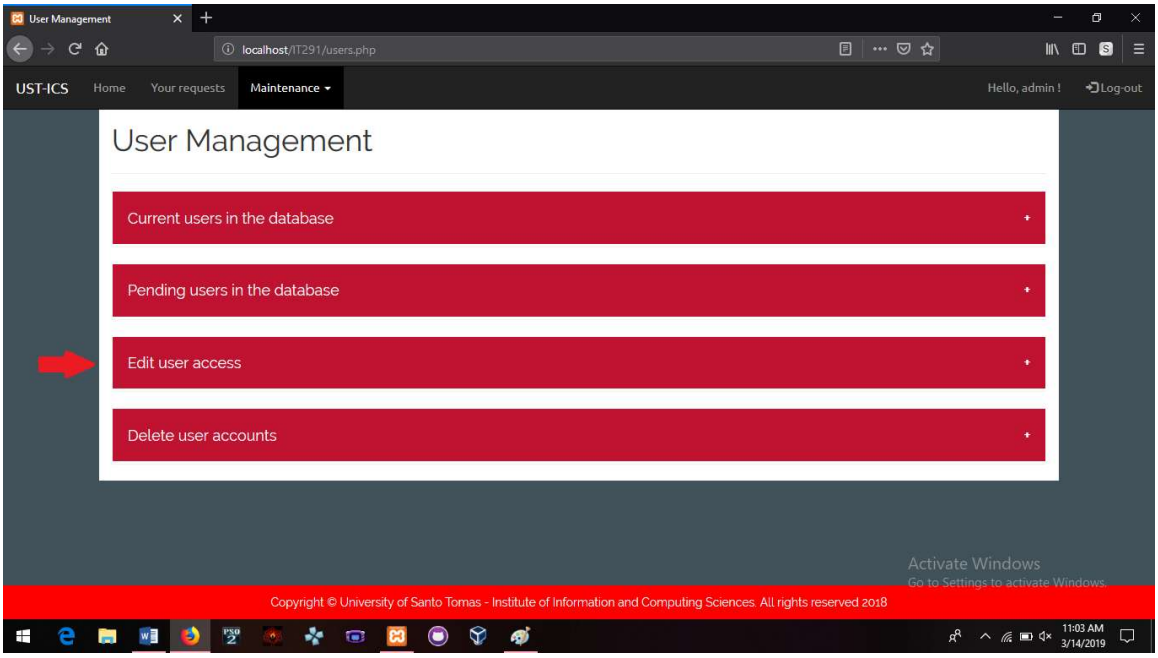


Figure 5.2.31: "Edit user access" dropdown

3.2 Select the user/s with “user” as their access level, then click on “Update”.

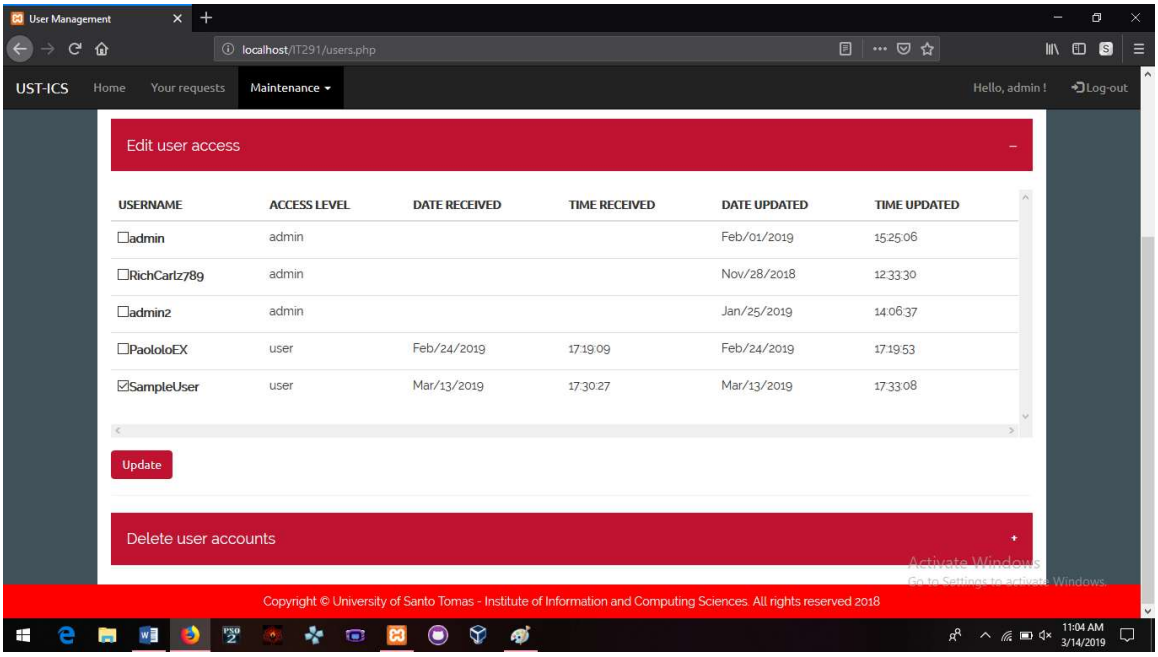
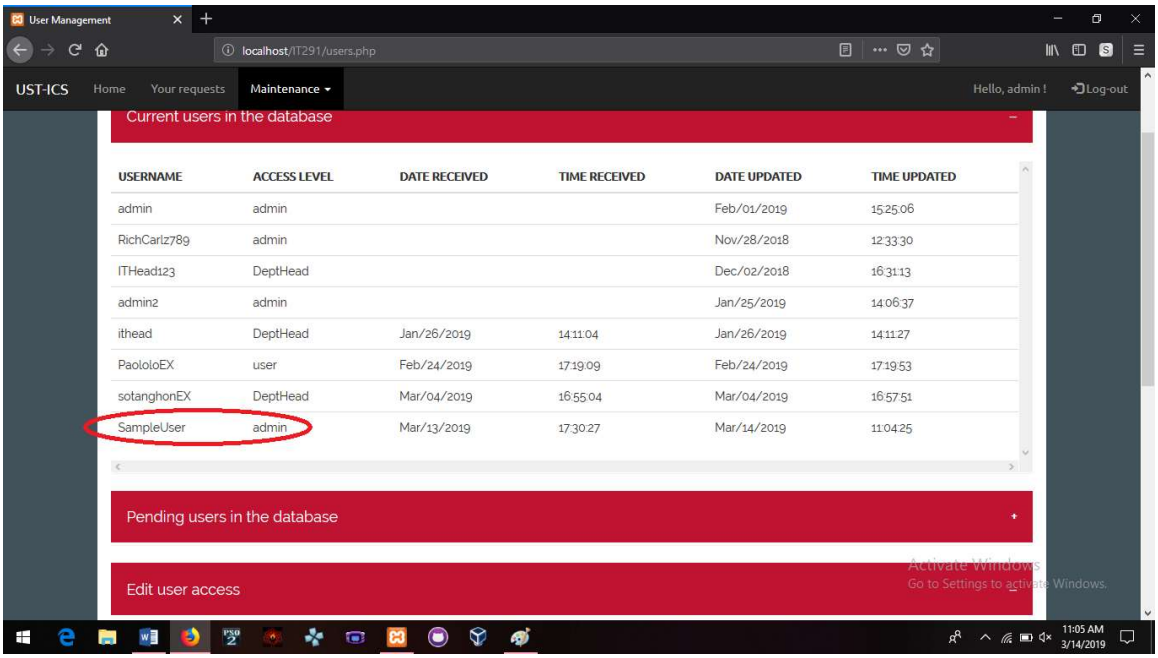


Figure 5.2.32: Selecting a user to update

3.3 The selected user’s access level should become “admin” once updated, and the particular user will receive an email regarding the update.



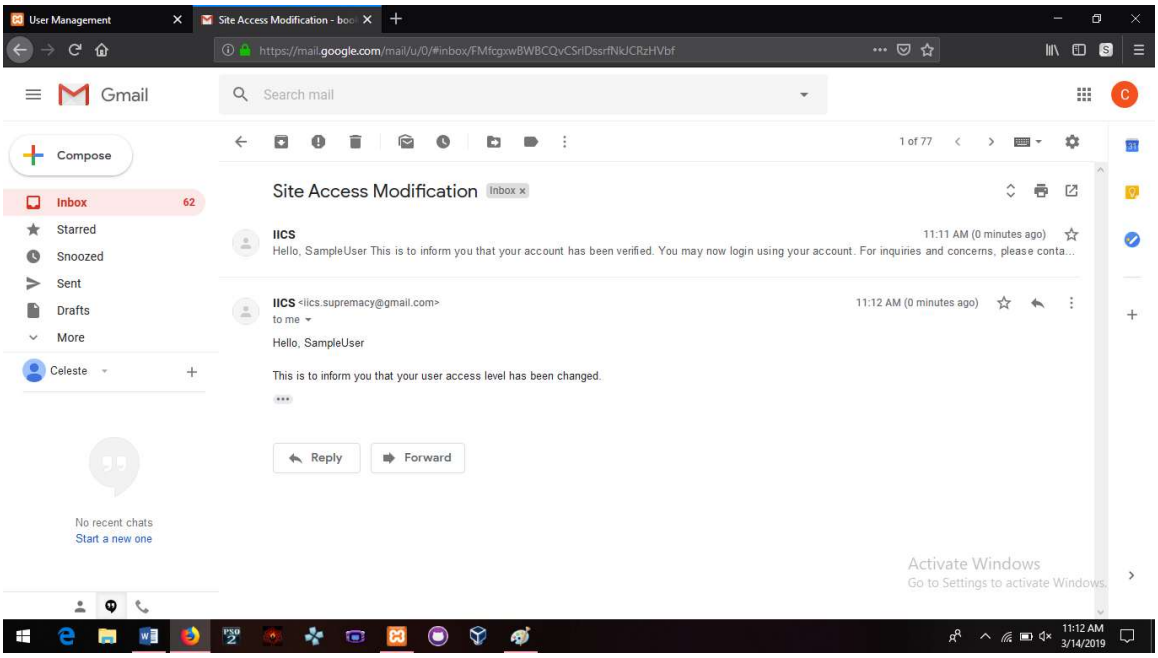


Figure 5.2.33: Email to user after account update

3.4 To **make an admin into a user**, repeat procedure 3.1.

3.5 Then, select the user/s with "admin" as their access level, then click on "Update".

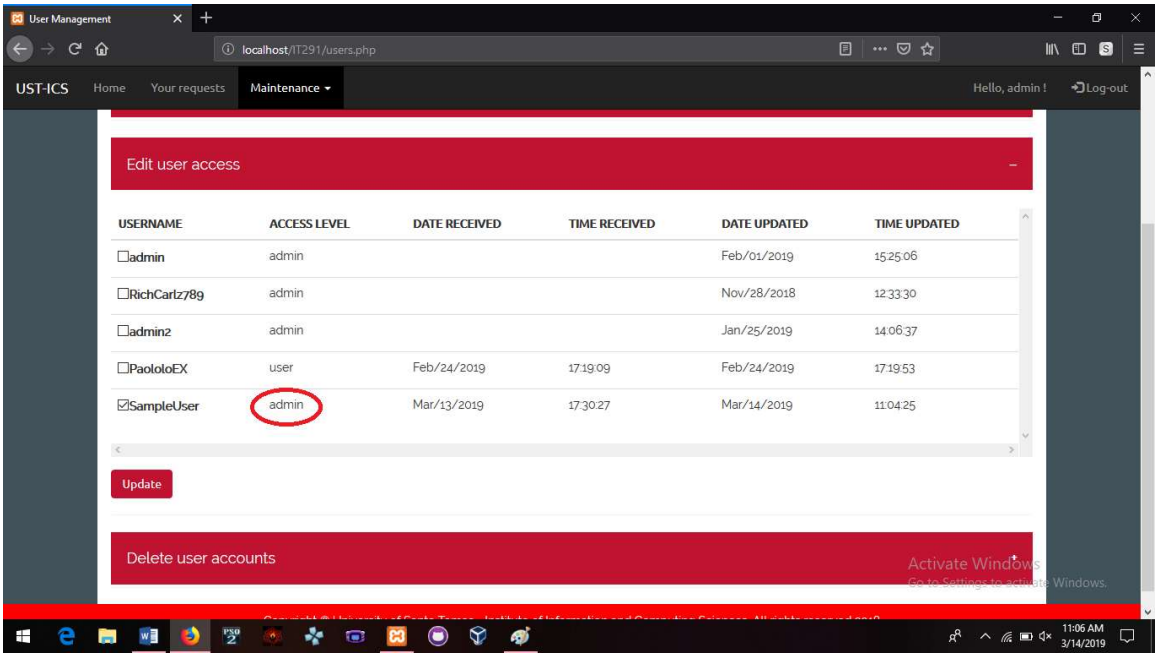
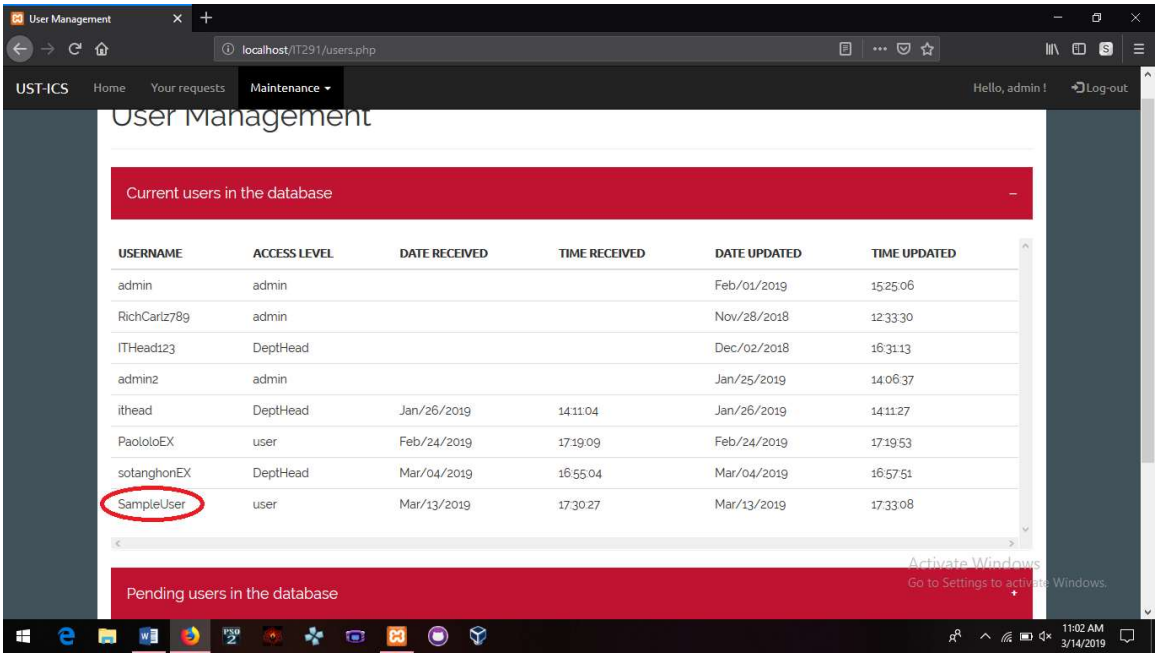


Figure 5.2.34: Updating an "admin"

3.6 The selected user's access level should become "user" once updated, and the particular user will receive an email regarding the update.



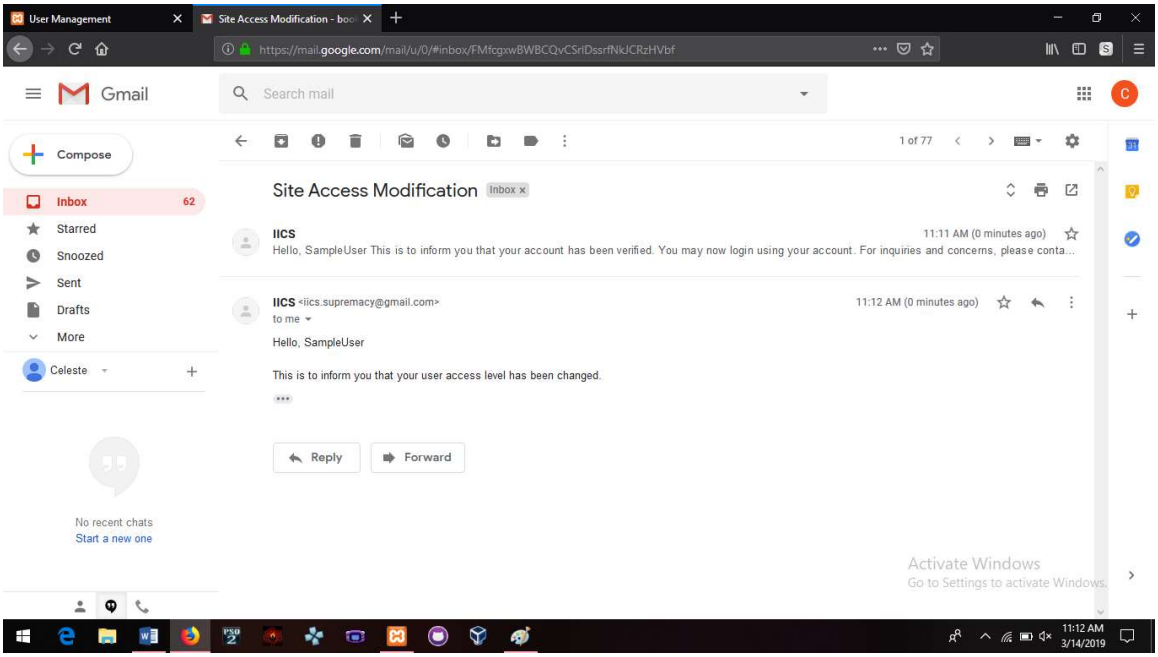


Figure 5.2.35: Email to admin after update

4 Deleting user accounts

4.1 To delete user accounts from the system, go to the User Management window, then click on the “Delete user accounts” dropdown.

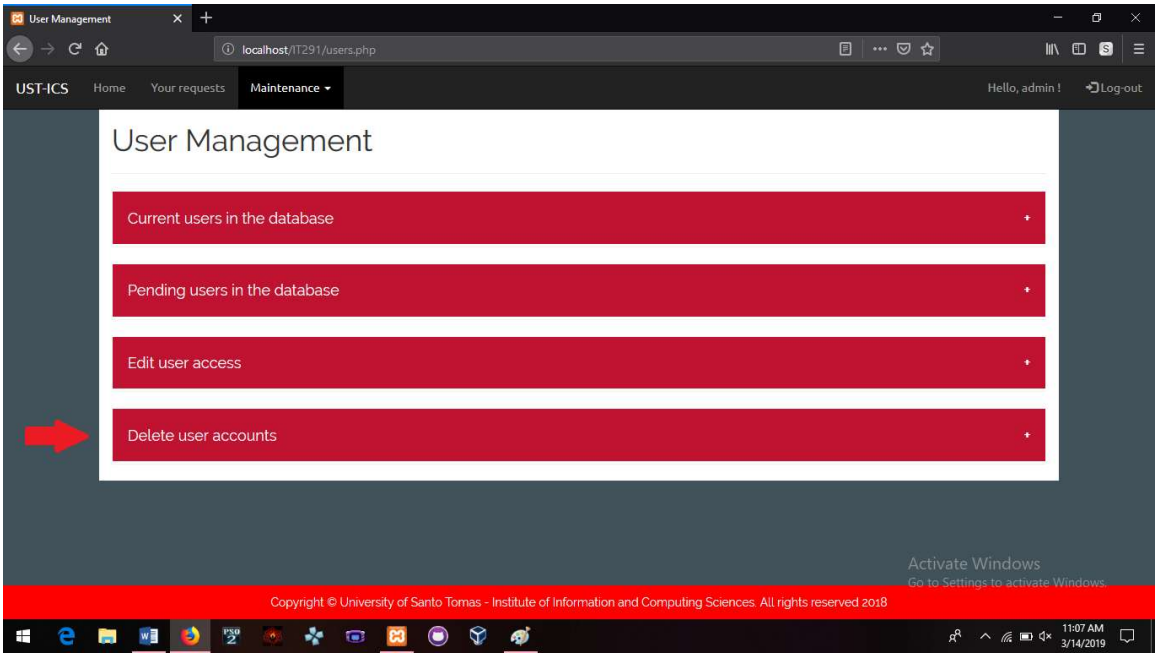
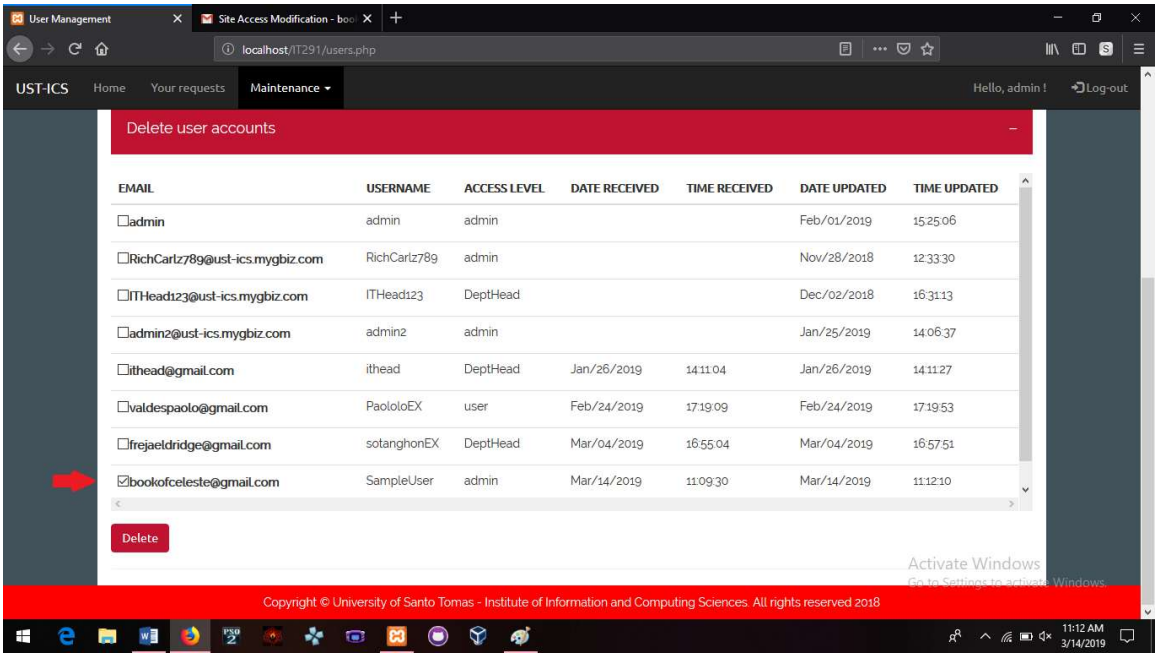


Figure 5.2.36: "Delete user accounts" dropdown

4.2 Select the account/s you want to remove, then click on “Delete”.



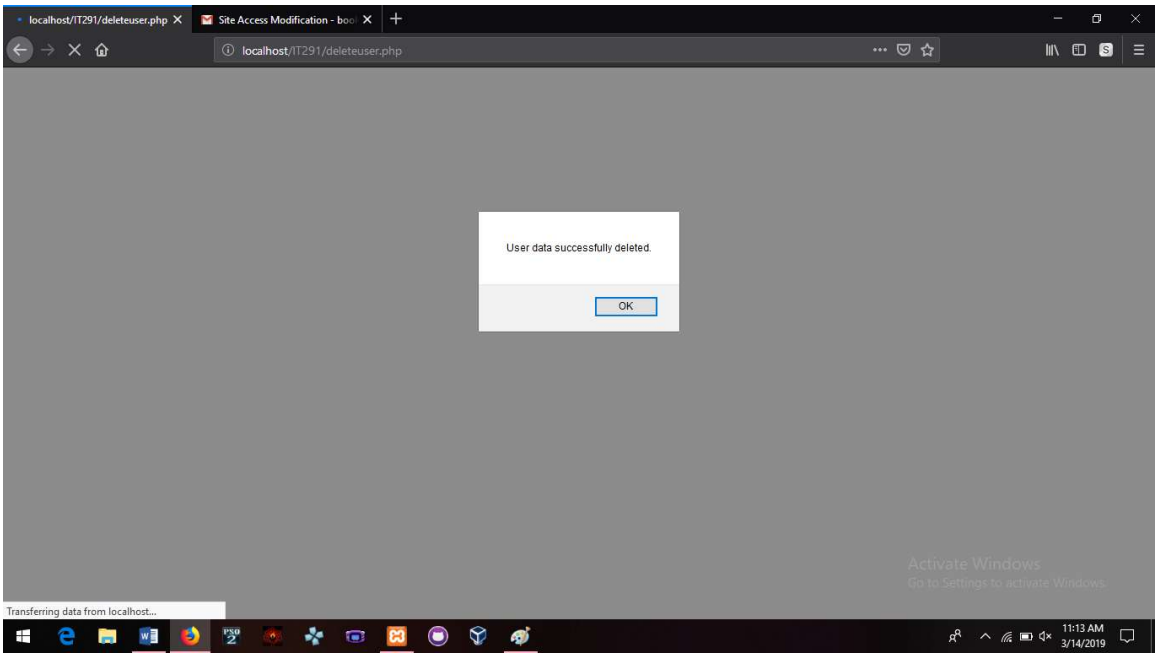
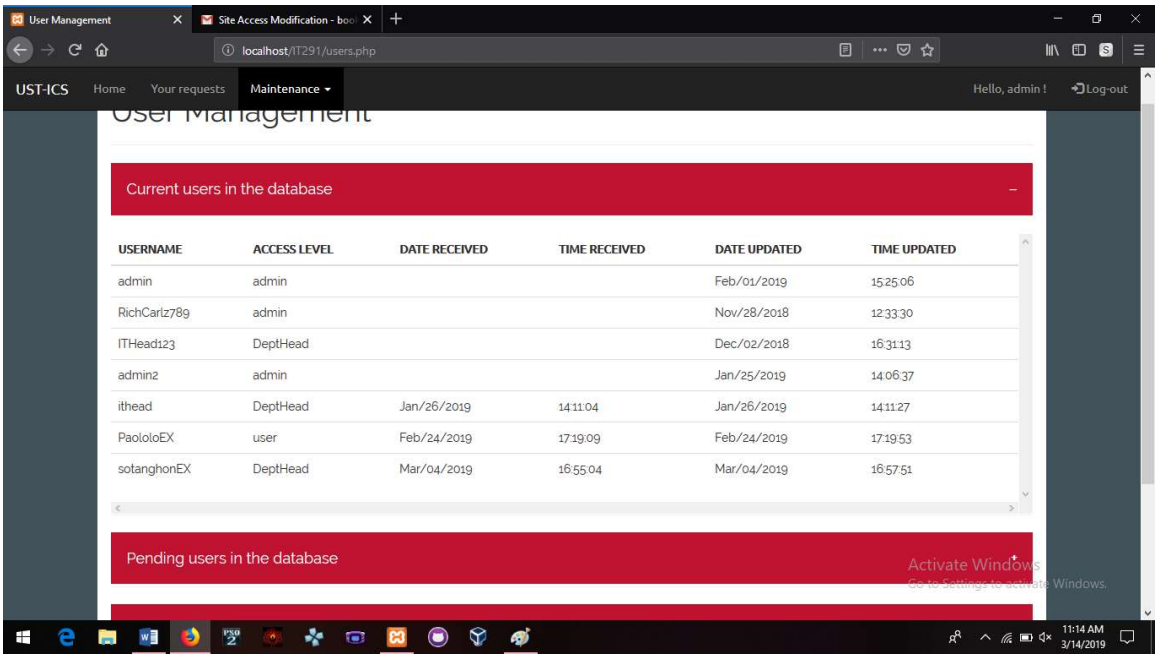


Figure 5.2.37: Account Deletion

4.3 The accounts should be removed from the system, and the user will be notified through email.



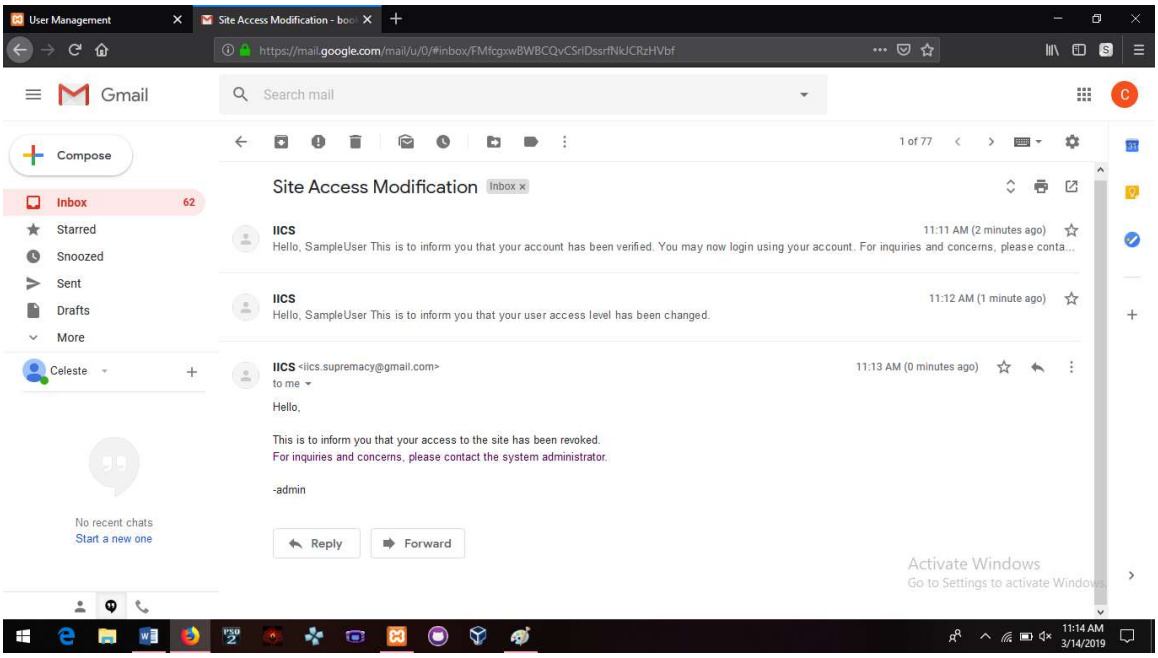


Figure 5.2.38: Email after account deletion

5 Managing reservation requests

5.1 To **approve requests**, click on “Maintenance”, then click on “Requests”.

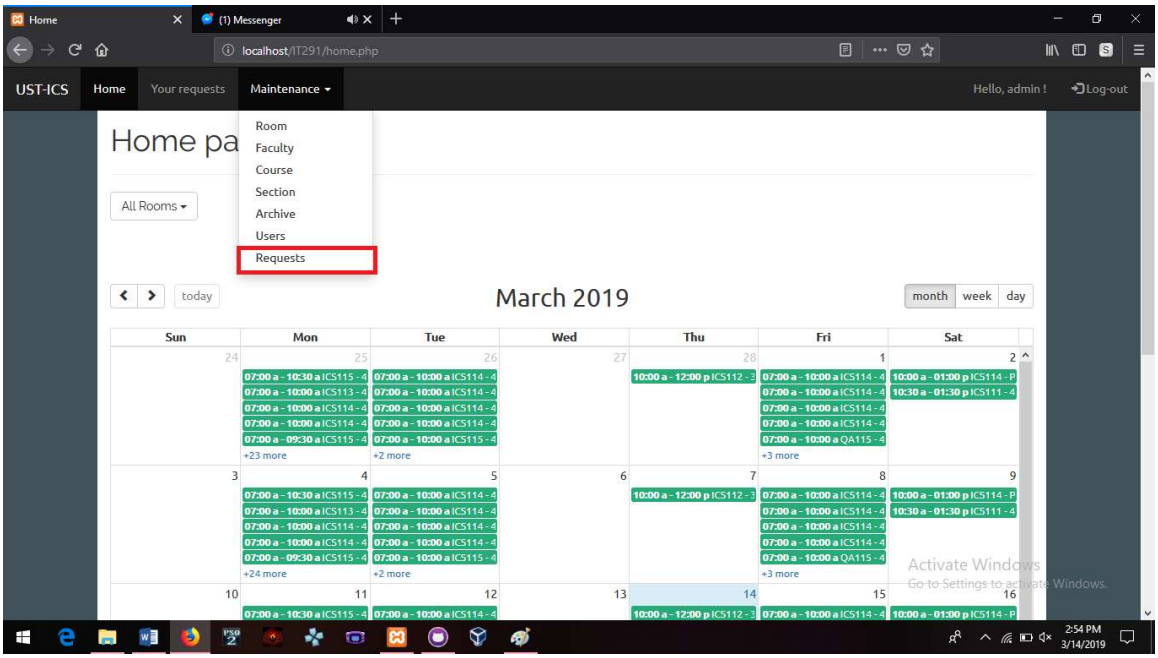


Figure 5.2.39: Maintenance for "Requests"

5.2 In the Requests Management window, pick a particular request, then click on “Approve”.

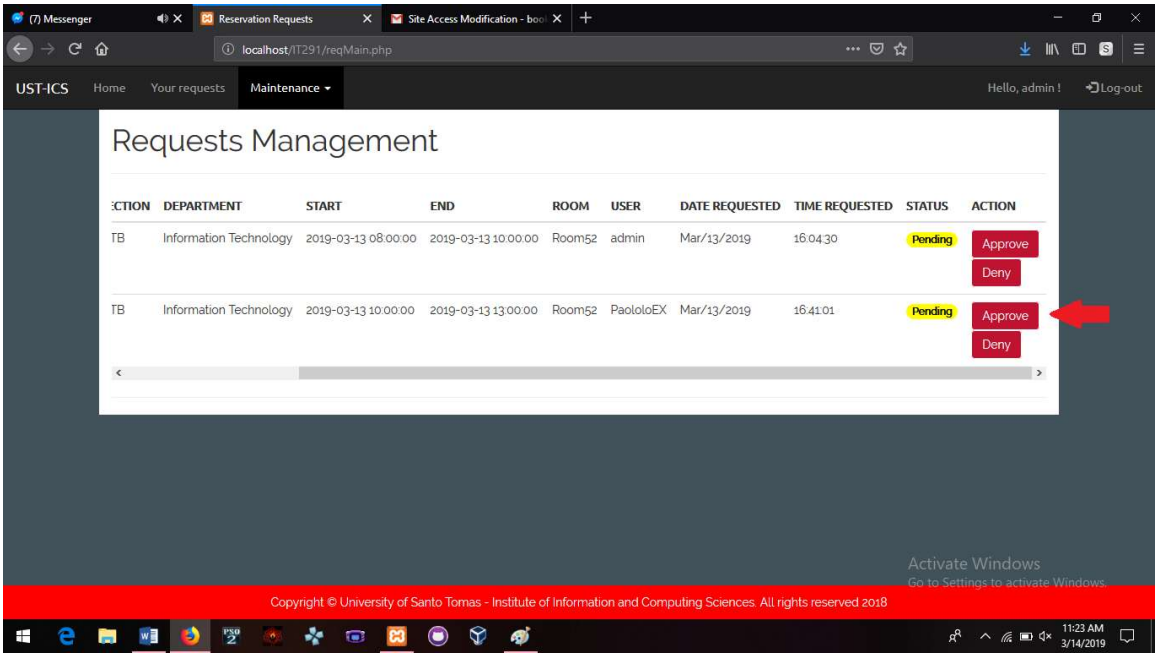


Figure 5.2. 40: "Approve" button in Requests Management

5.3 The user will receive an email after the approval.

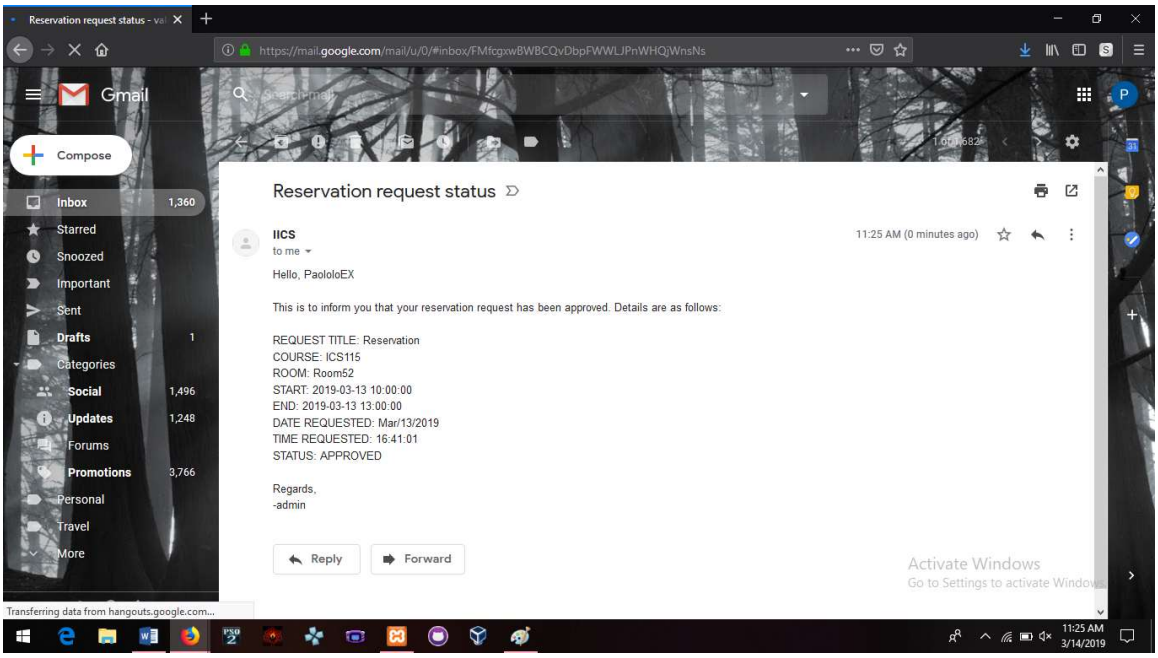


Figure 5.2.41: Approval of Request email

5.4 To **deny requests**, repeat procedure 5.1.

5.5 Then, pick a particular request, then click on “Deny”.

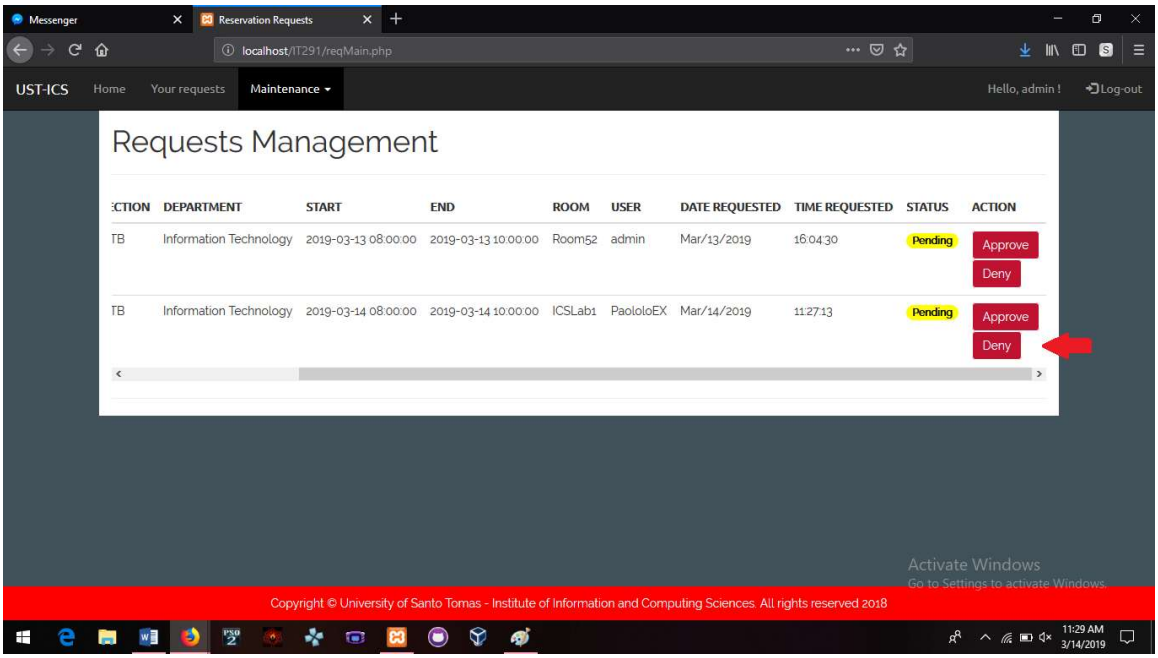


Figure 5.2.42: "Deny" button in Requests Management

5.6 Enter the reason for denial of request, then click “Submit”.

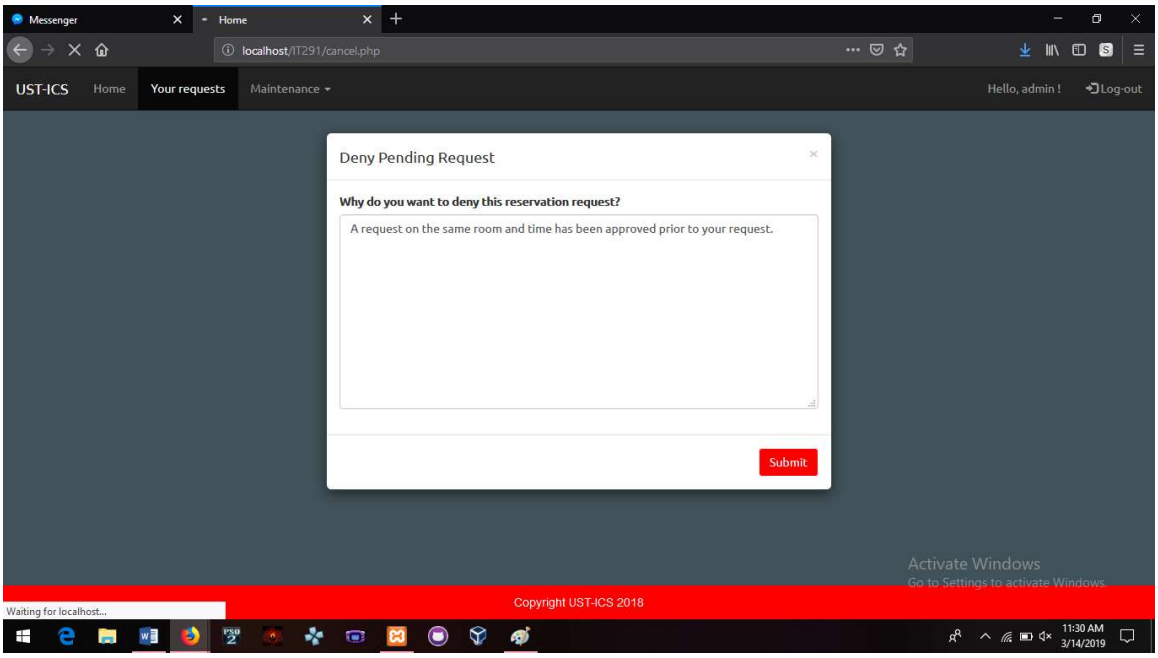


Figure 5.2.43: Deny pending request page

5.7 The user will receive an email after the denial, along with the reason entered.

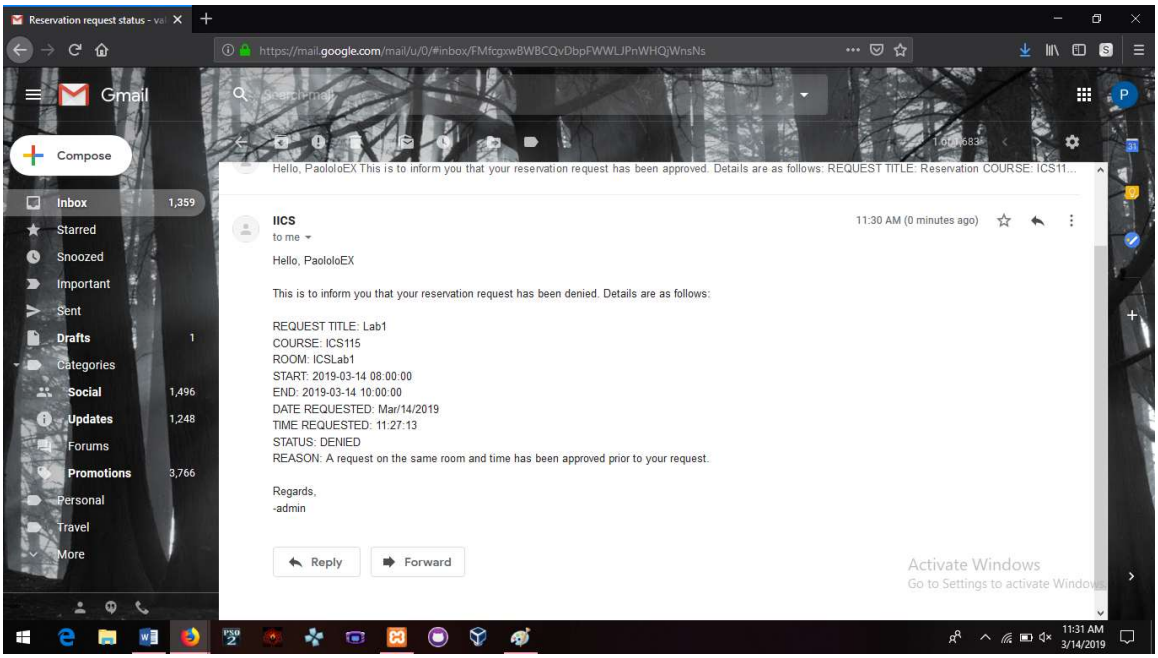


Figure 5.2.44: Email after denial of request

6 Creating an archive of requests

6.1 To create an archive of the requests, click on “Maintenance”, then click on “Archive”.

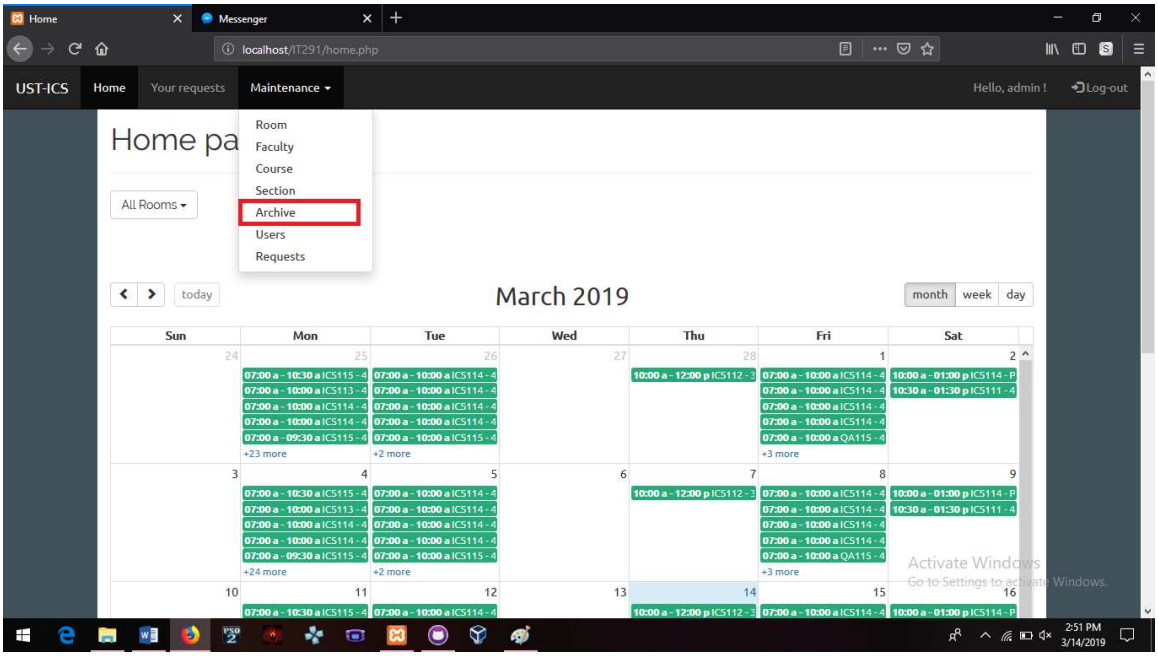


Figure 5.2.45: Maintenance for "Archive"

6.2 Select the Academic Year and Semester from the dropdowns available.

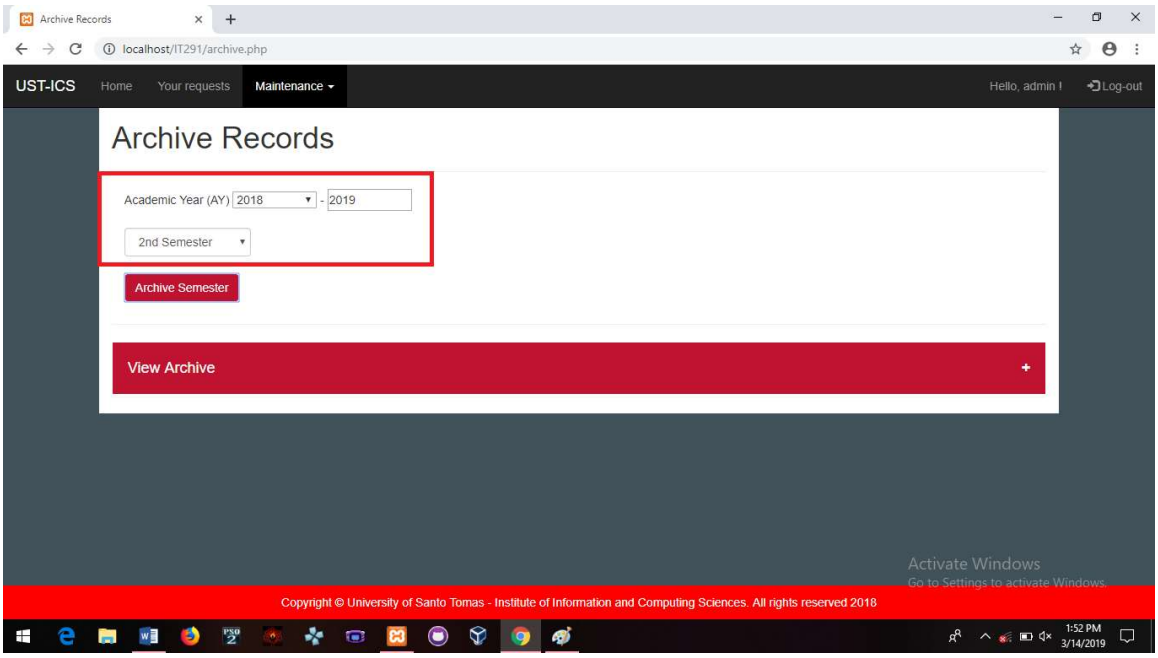


Figure 5.2.46: Academic Year and Semester dropdowns

6.3 Finally, click on Archive Semester, then click on "OK" on the popup window that appears.

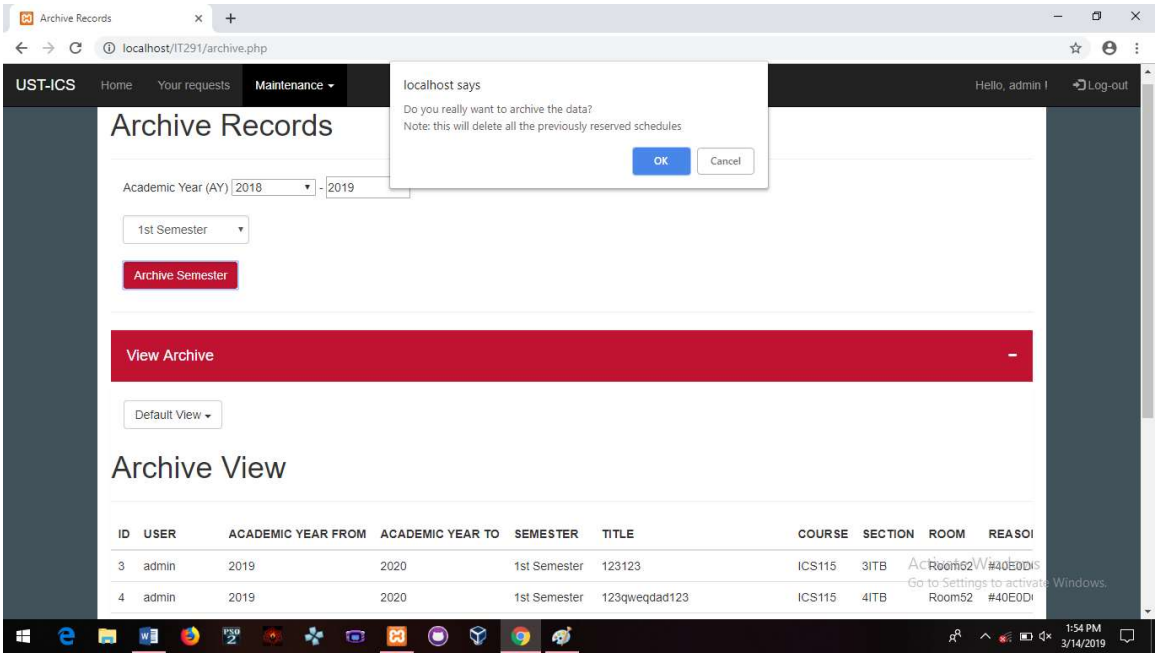


Figure 5.2.47: Archive confirmation message

6.4 To view the list of requests that have been archived, click on the “View Archive” dropdown.

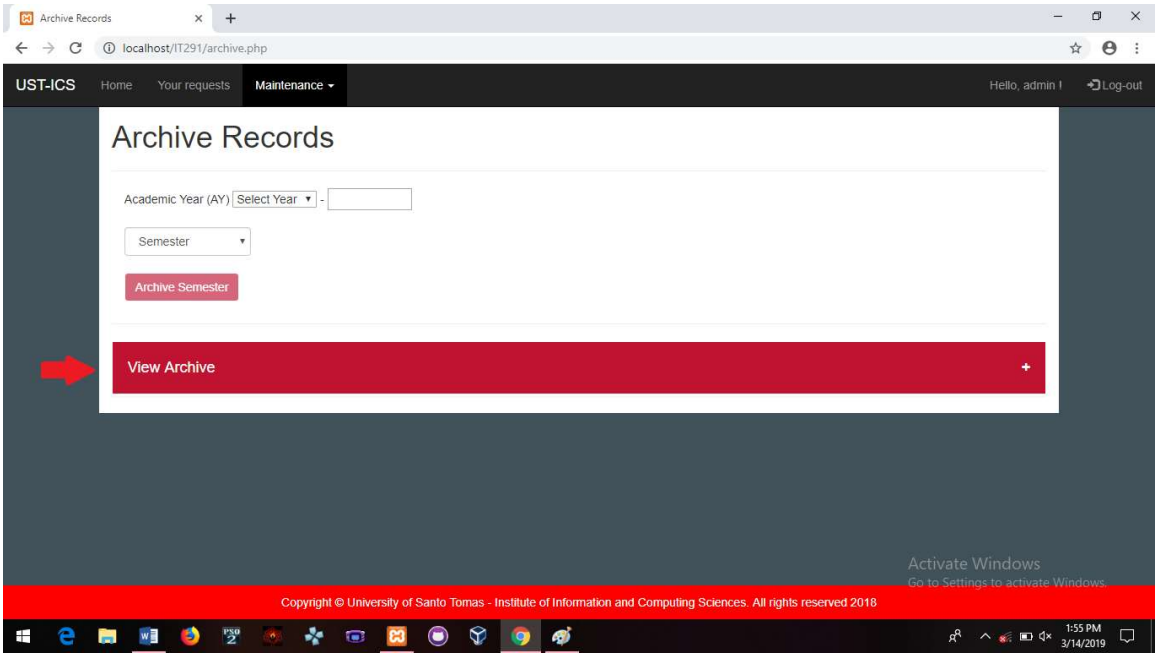


Figure 5.2.48: "View Archive" dropdown

6.5 The archive will be displayed once clicked.

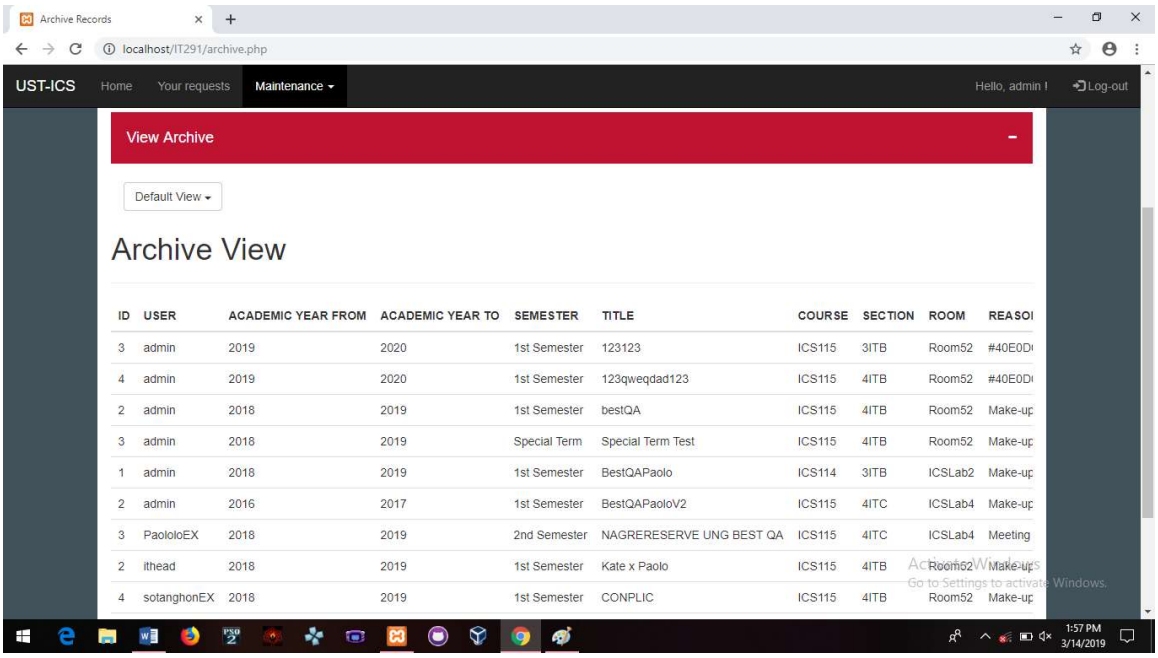


Figure 5.2.49: Archive list

6.6 To sort the list, click on the "Default View" dropdown and select your desired view.

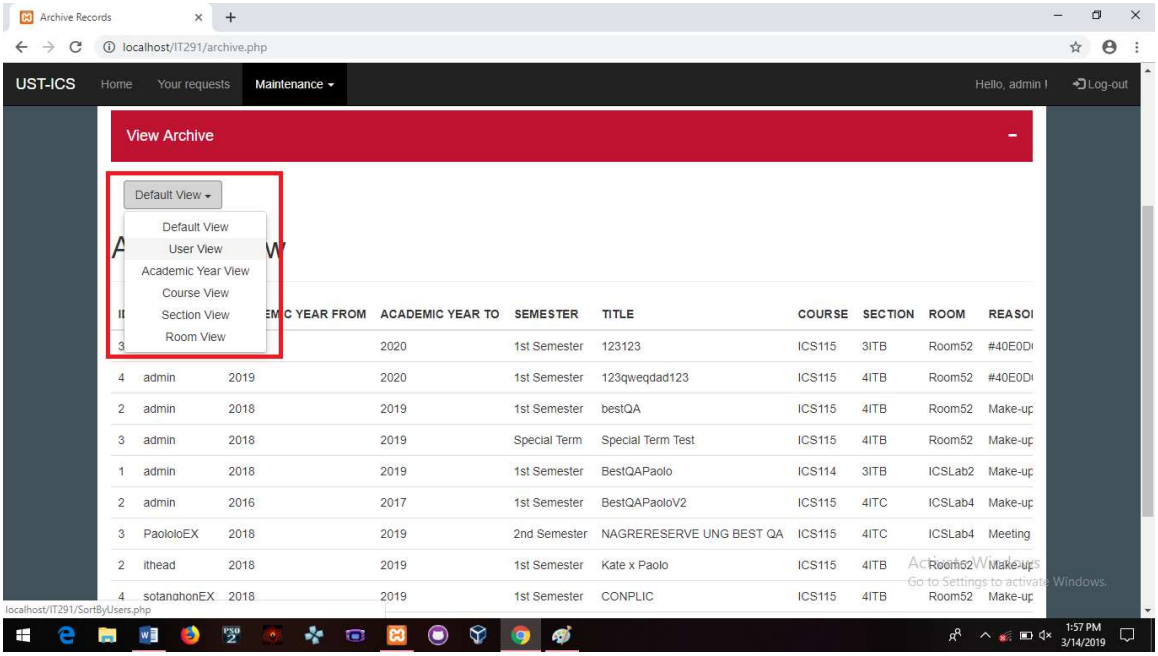


Figure 5.2.50: Archive view-sort dropdown

6.7 Then, select a particular User, Academic Year, Course, Section, or Room from the dropdown.

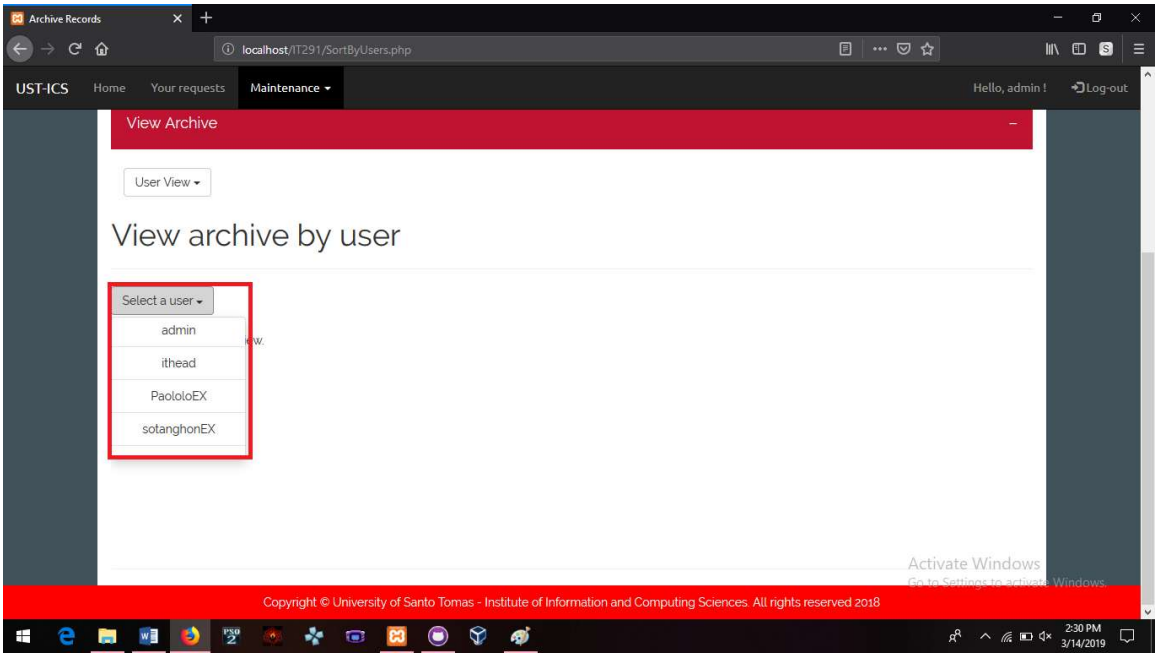


Figure 5.2.51: Archive user-sort dropdown

6.8 The list will be sorted based on the desired information selected.

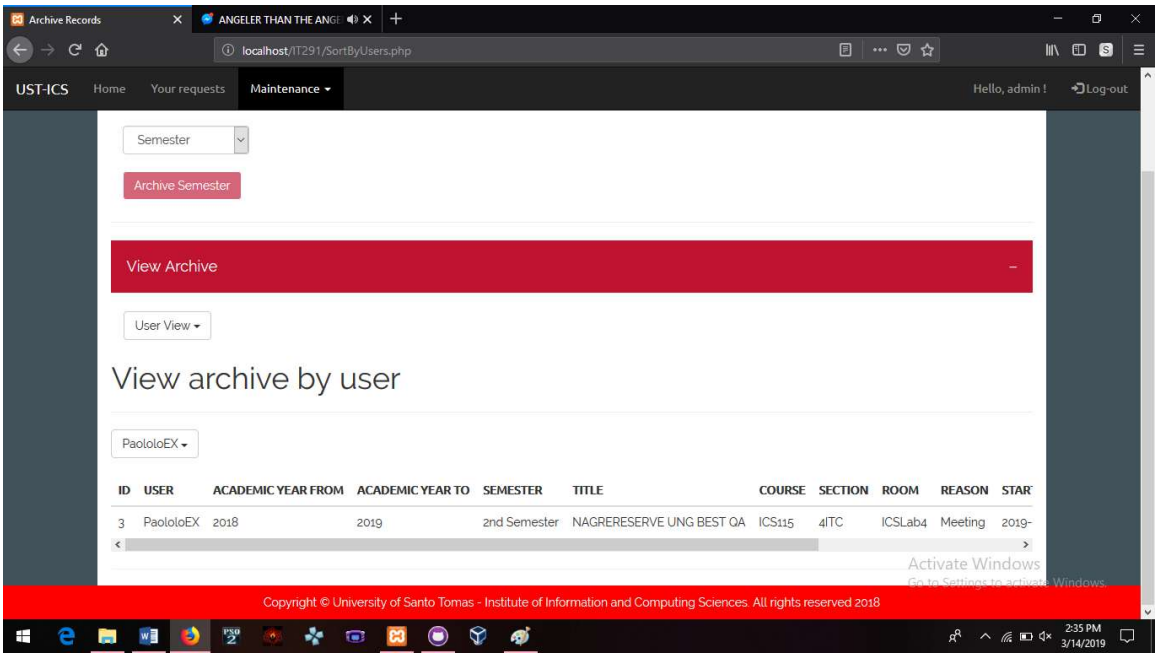


Figure 5.2.52: Sorting result

FOR DEPARTMENT HEAD ONLY:

1. Adding schedules

1.1 To add schedules, click on “Scheduling”, then click on Section View.

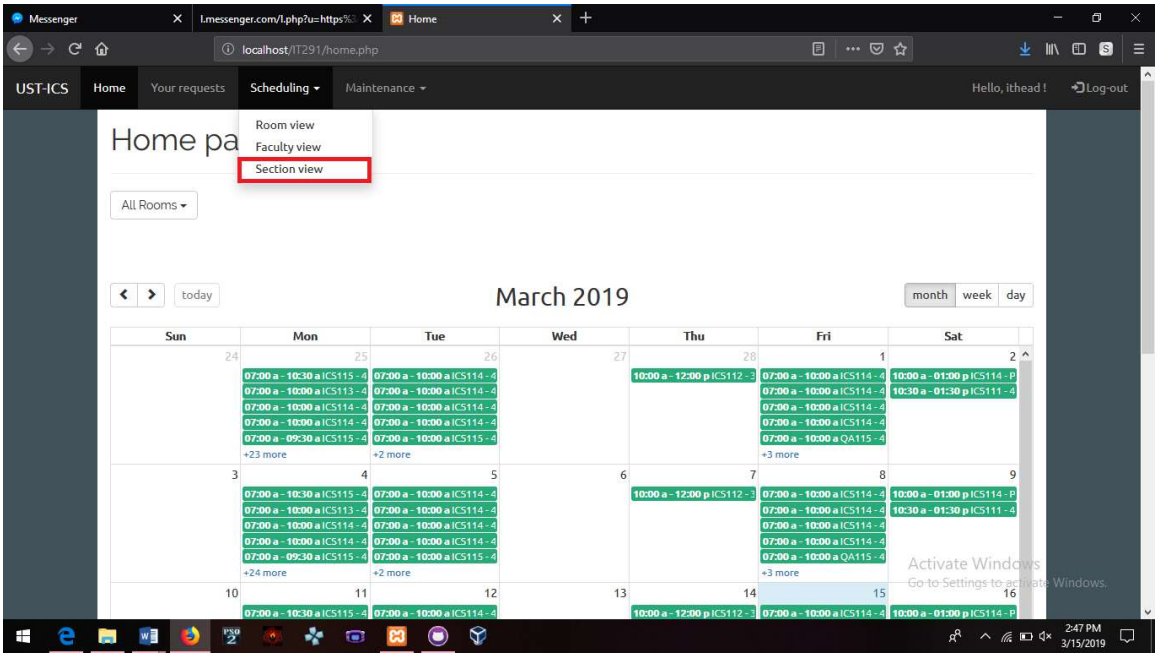


Figure 5.2.53: Scheduling button

1.2 In the Section View window, click on “Add Schedule”

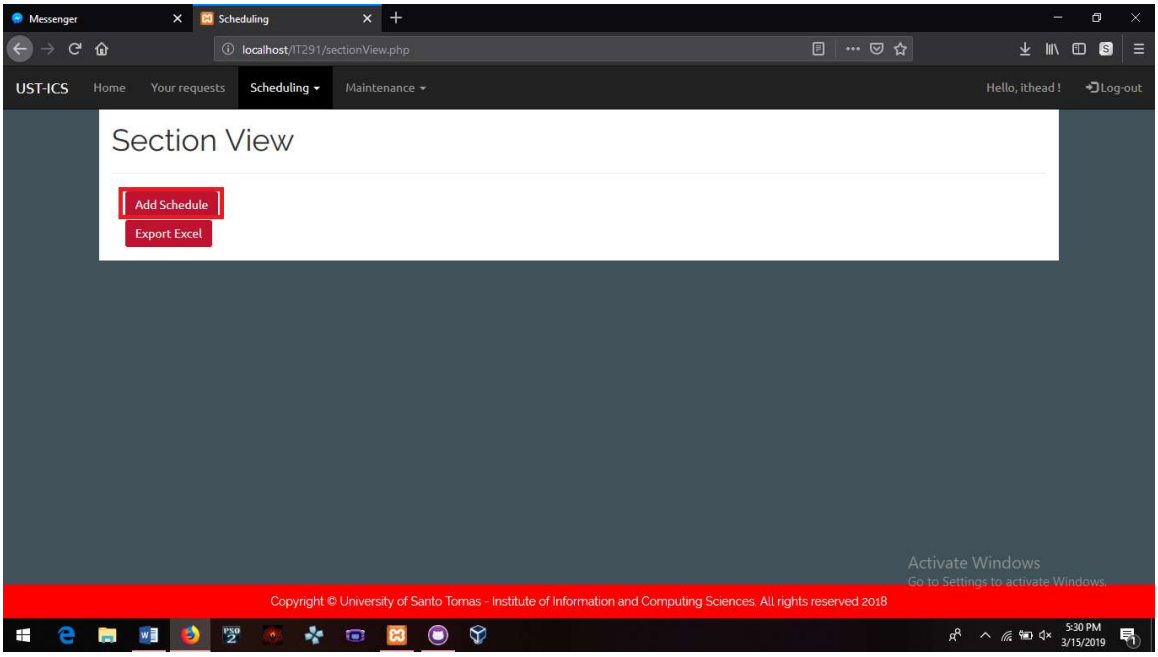


Figure 5.2.54: "Add Schedule" button

1.3 Fill up the form that appears, then click on “Add”.

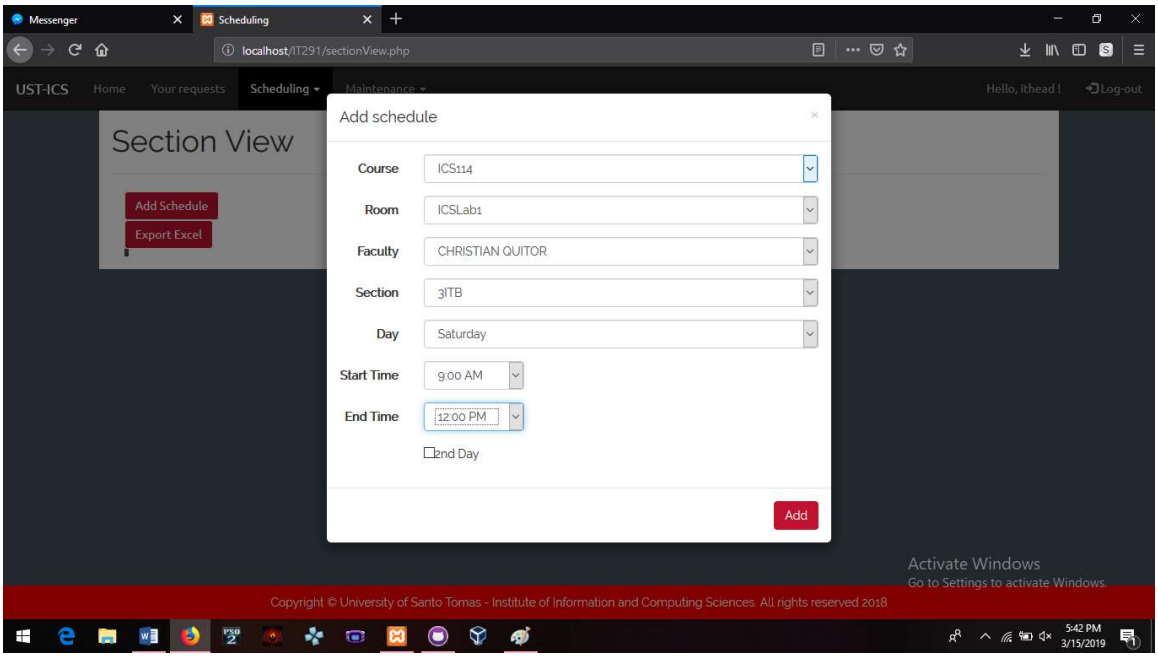
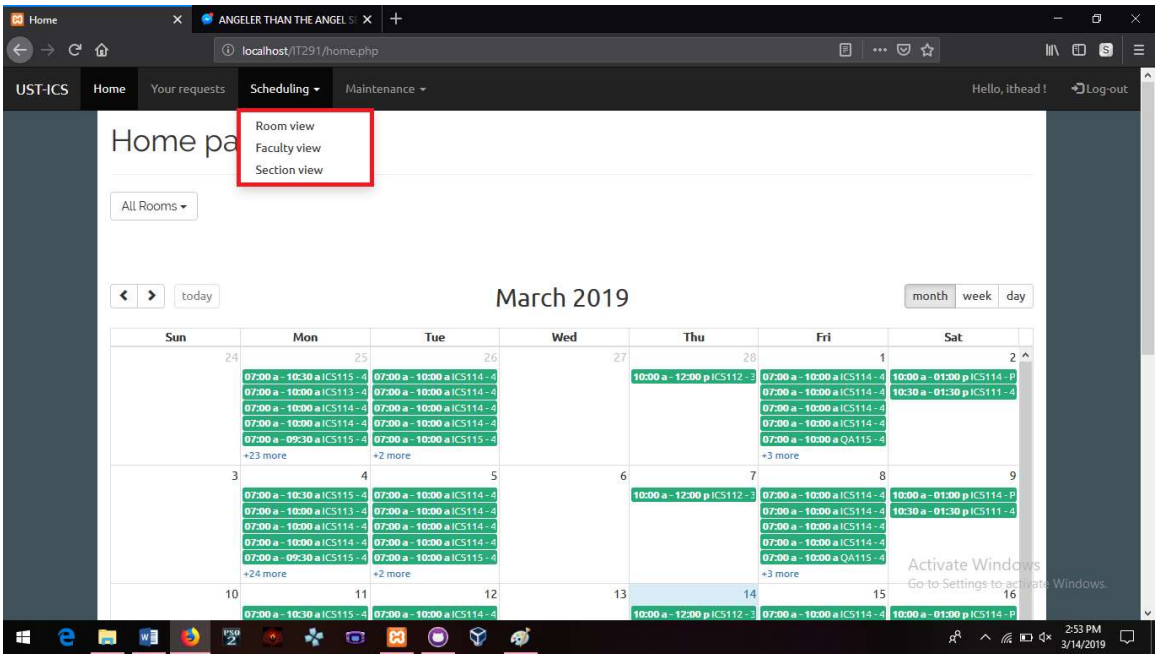


Figure 5.2.55: Schedule addition

1.4 The schedule will be reflected in the calendar at the home page, and at the Excel file that will be generated through “Export Excel”.

2 Generating reports

2.1 To generate an excel file of the schedule, click on either Room View, Faculty View, or Section View.



2.2 Inside the View window, click on “Export Excel”.

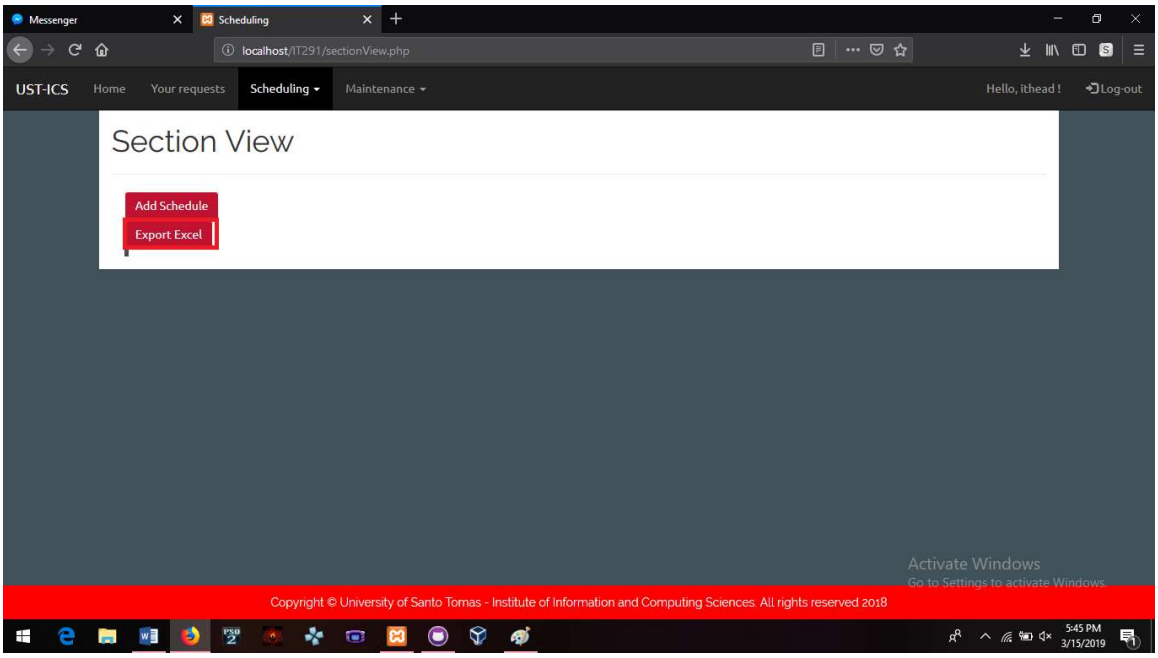
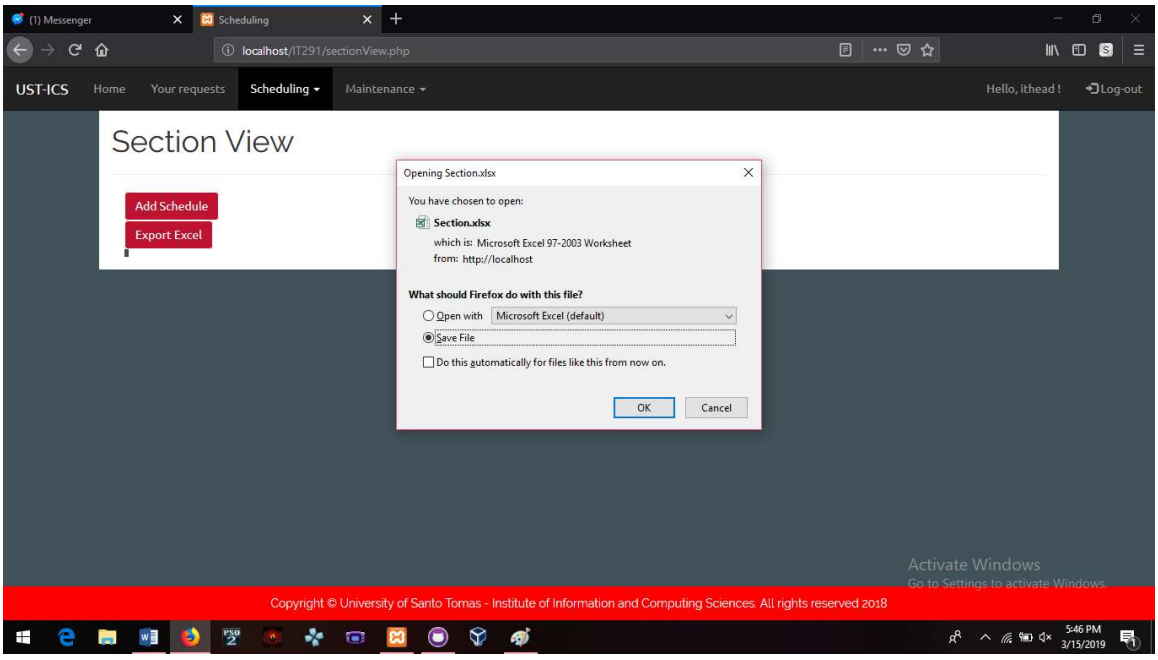


Figure 5.2.56: "Export Excel" button

2.3 The system will download the excel file of the chosen schedule.



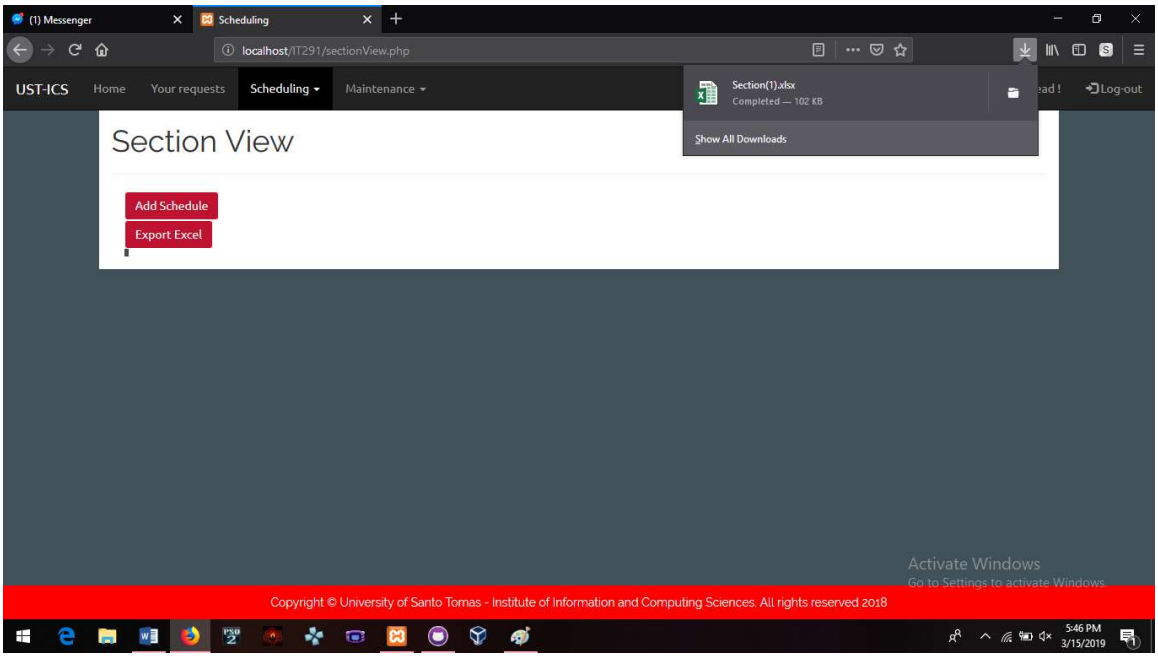


Figure 5.2.57: Excel download

Test Results

Test Case Name: login_valid

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Username and password matched	1. Enter “ithead” 2. Enter “Admin123” 3. Click “Login”	The user is granted access into the system.	The user has accessed the system.	PASSED

Test Case Name: login_invalid

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Username and password did not match	1. Enter “ithead” 2. Enter “admin123” 3. Click “Login”	The system will generate an error.	The system generated an error	PASSED

Test Case Name: login_unverifiedAcct

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User logs in with unverified account	1. Enter “Samp” 2. Enter “Sample123” 3. Click “Login”	The system will generate an error.	The system generated an error	PASSED

Test Case Name: login_blank_username

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Username is left blank	1. Blank username 2. Enter “Admin123” 3. Click “Login”	The system will generate an error tooltip.	The system generated an error tooltip	PASSED

Test Case Name: login_blank_password

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Password is left blank	1. Enter "ithead" 2. Blank password 3. Click "Login"	The system will generate an error tooltip.	The system generated an error tooltip	PASSED

Test Case Name: register_valid_nonHead

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User has made a successful registration	1. Enter "valdespaolo@gmail.com" 2. Enter "QAValdes" 3. Enter "QAisLife123" 4. Enter "QAisLife123" 5. Select "Information Technology" 6. Don't check Dept.Head box 7. Click "Submit"	The system must send the request to the admin for verification	The system sent the request to the admin for verification	PASSED

Test Case Name: register_valid_deptHead

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User has made a successful registration	1. Enter "valdespaolo@gmail.com" 2. Enter "QAValdes" 3. Enter "QAisLife123" 4. Enter "QAisLife123" 5. Select "Information Technology" 6. Check Dept.Head box 7. Click "Submit"	The system must send the request to the admin for verification	The system sent the request to the admin for verification	PASSED

Test Case Name: register_invalid

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Password and confirm password did not match	1. Enter "valdespaolo@gmail.com" 2. Enter "QAValdes" 3. Enter "QAisLife123" 4. Enter "QAisLife" 5. Select "Information Technology" 6. Don't check Dept.Head box 7. Click "Submit"	The system will generate an error	The system generated an error	PASSED

Test Case Name: register_takenEmail

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters a taken email.	1. Enter "valdespaolo@gmail.com" 2. Enter "QAValdes" 3. Enter "QAisLife123" 4. Enter "QAisLife" 5. Select "Information Technology" 6. Don't check Dept.Head box 7. Click "Submit"	The system will generate an error	The system generated an error	PASSED

Test Case Name: register_takenUsername

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters a taken username	1. Enter "valdespaolo@gmail.com" 2. Enter "QAValdes" 3. Enter "QAisLife123" 4. Enter "QAisLife" 5. Select "Information Technology" 6. Don't check Dept.Head box 7. Click "Submit"	The system will generate an error	The system generated an error	PASSED

Test Case Name: register_blank_email

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves email field blank	1. Blank email 2. Enter “QAValdes” 3. Enter “QAisLife123” 4. Enter “QAisLife123” 5. Select “Information Technology” 6. Don’t check Dept.Head box 7. Click “Submit”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: register_blank_username

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves username field blank	1. Enter email 2. Blank username 3. Enter “QAisLife123” 4. Enter “QAisLife123” 5. Select “Information Technology” 6. Don’t check Dept.Head box 7. Click “Submit”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: register_blank_password

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves password field blank	1. Enter email 2. Enter “QAValdes” 3. Blank password 4. Enter “QAisLife123” 5. Select “Information Technology” 6. Don’t check Dept.Head box 7. Click “Submit”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: register_blank_confirmPass

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves confirm password field blank	1. Enter email 2. Enter "QAValdes" 3. Enter "QAisLife123" 4. Blank confirm password 5. Select "Information Technology" 6. Don't check Dept.Head box 7. Click "Submit"	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: register_blank_department

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves department field blank	1. Enter email 2. Enter "QAValdes" 3. Enter "QAisLife123" 4. Enter "QAisLife123" 5. Blank department 6. Don't check Dept.Head box 7. Click "Submit"	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: register_password_noUpperCase

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Password complexity not satisfied	1. Enter email 2. Enter "QAValdes" 3. Enter "qaislife123" 4. Enter "qaislife123" 5. Select "Information Technology" 6. Don't check Dept.Head box 7. Click "Submit"	The system will generate an error.	The system generated an error	PASSED

Test Case Name: register_password_noLowerCase

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Password complexity not satisfied	1. Enter email 2. Enter “QAValdes” 3. Enter “QAISLIFE123” 4. Enter “QAISLIFE123” 5. Select “Information Technology” 6. Don’t check Dept.Head box 7. Click “Submit”	The system will generate an error.	The system generated an error	PASSED

Test Case Name: register_password_noNumber

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Password complexity not satisfied	1. Enter email 2. Enter “QAValdes” 3. Enter “QAisLife” 4. Enter “QAisLife” 5. Select “Information Technology” 6. Don’t check Dept.Head box 7. Click “Submit”	The system will generate an error.	The system generated an error	PASSED

Test Case Name: register_password_notEightChars

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Password complexity not satisfied	1. Enter email 2. Enter “QAValdes” 3. Enter “QALife” 4. Enter “QALife” 5. Blank department 6. Don’t check Dept.Head box 7. Click “Submit”	The system will generate an error.	The system generated an error	PASSED

Test Case Name: register_email

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Password complexity not satisfied	1. Enter email 2. Enter "QAValdes" 3. Enter "QAisLife123" 4. Enter "QAisLife123" 5. Select "Information Technology" 6. Don't check Dept.Head box 7. Click "Submit"	The specified email address must receive an email notification.	The specified email address did not receive anything.	FAILED

Test Case Name: register_email

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Password complexity not satisfied	1. Enter email 2. Enter "QAValdes" 3. Enter "QAisLife123" 4. Enter "QAisLife123" 5. Select "Information Technology" 6. Don't check Dept.Head box 7. Click "Submit"	The specified email address must receive an email notification.	The specified email address received the email notification.	PASSED

Test Case Name: reserve_valid

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User has made a successful reservation request	1. Click on a calendar slot. 2. Enter "Capstone" 3. Select "Information Technology" 4. Select "4ITH" 5. Select "IT 292" 6. Select "Make-Up Classes" 7. Select "Room 48" 8. Enter "10:00" 9. Enter "12:00" 10. Click "Request"	The reservation request will be sent to the admin for approval.	The reservation request was sent to the admin for approval.	PASSED

Test Case Name: reserve_time_overlap1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User entered an occupied timeframe	1. Click on a calendar slot. 2. Enter “Capstone” 3. Select “Information Technology” 4. Select “4ITH” 5. Select “IT 292” 6. Select “Make-Up Classes” 7. Select “Room 48” 8. Enter “10:00” 9. Enter “12:00” 10. Click “Request”	The system will generate an error	The reservation request was sent to the admin for approval	FAILED

Test Case Name: reserve_time_overlap2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User entered an occupied timeframe	1. Click on a calendar slot. 2. Enter “Capstone” 3. Select “Information Technology” 4. Select “4ITH” 5. Select “IT 292” 6. Select “Make-Up Classes” 7. Select “Room 48” 8. Enter “10:00” 9. Enter “12:00” 10. Click “Request”	The system will generate an error		

Test Case Name: reserve_blank_title

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves title field blank	1. Click on a calendar slot. 2. Blank title 3. Select “Information Technology” 4. Select “4ITH” 5. Select “IT 292” 6. Select “Make-Up Classes” 7. Select “Room 48” 8. Enter “10:00” 9. Enter “12:00” 10. Click “Request”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: reserve_blank_department

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves department field blank	1. Click on a calendar slot. 2. Enter “Capstone” 3. Blank department 4. Select “4ITH” 5. Select “IT 292” 6. Select “Make-Up Classes” 7. Select “Room 48” 8. Enter “10:00” 9. Enter “12:00” 10. Click “Request”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: reserve_blank_section

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves section field blank	1. Click on a calendar slot. 2. Enter “Capstone” 3. Select “Information Technology” 4. Blank section 5. Select “IT 292” 6. Select “Make-Up Classes” 7. Select “Room 48” 8. Enter “10:00” 9. Enter “12:00” 10. Click “Request”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: reserve_blank_course

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves course field blank	1. Click on a calendar slot. 2. Enter “Capstone” 3. Select “Information Technology” 4. Select “4ITH” 5. Blank course 6. Select “Make-Up Classes” 7. Select “Room 48” 8. Enter “10:00” 9. Enter “12:00” 10. Click “Request”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: reserve_blank_purpose

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves purpose field blank	1. Click on a calendar slot. 2. Enter “Capstone” 3. Select “Information Technology” 4. Select “4ITH” 5. Select “IT 292” 6. Blank purpose 7. Select “Room 48” 8. Enter “10:00” 9. Enter “12:00” 10. Click “Request”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: reserve_blank_room

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves room field blank	1. Click on a calendar slot. 2. Enter “Capstone” 3. Select “Information Technology” 4. Select “4ITH” 5. Select “IT 292” 6. Select “Make-Up Classes” 7. Blank room 8. Enter “10:00” 9. Enter “12:00” 10. Click “Request”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: reserve_email1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User makes a reservation	1. Click on a calendar slot. 2. Enter “Capstone” 3. Select “Information Technology” 4. Select “4ITH” 5. Select “IT 292” 6. Select “Make-Up Classes” 7. Select “Room 48” 8. Enter “10:00” 9. Enter “12:00” 10. Click “Request”	The specified email address must receive an email notification.	The specified email address did not receive the email notification.	FAILED

Test Case Name: reserve_email2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User makes a reservation	1. Click on a calendar slot. 2. Enter “Capstone” 3. Select “Information Technology” 4. Select “4ITH” 5. Select “IT 292” 6. Select “Make-Up Classes” 7. Select “Room 48” 8. Enter “10:00” 9. Enter “12:00” 10. Click “Request”	The specified email address must receive an email notification.	The specified email address received the email notification.	PASSED

Test Case Name: maint_rooms_current

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User wants to check current rooms	Content: - Room Name - Date Added - Time Added	The user will see the current rooms in the system.	The user saw the current rooms in the system.	PASSED

Test Case Name: maint_rooms_add_valid

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User wants to add rooms	1. Click "Add new rooms" 2. Enter Room52 3. Click Submit	The room will be successfully added.	The room is successfully added.	PASSED

Test Case Name: maint_rooms_blank

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves the add field blank	1. Click "Add new rooms" 2. Blank room name 3. Click Submit	An error tooltip will be generated	An error tooltip generated	PASSED

Test Case Name: maint_rooms_overlap1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters a duplicate of a room.	1. Click "Add new rooms" 2. Enter "Room52" 3. Click Submit 4. Enter "Room52" 5. Click Submit	An error will be generated.	An error has been generated.	PASSED

Test Case Name: maint_rooms_overlap2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters uppercase and lowercase names of the same room	1. Click "Add new rooms" 2. Enter "Room52" 3. Click Submit 4. Enter "room52" 5. Click Submit	An error will be generated.	The room has still been added.	FAILED

Test Case Name: maint_rooms_overlap3

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters uppercase and lowercase names of the same room	1. Click “Add new rooms” 2. Enter “Room52” 3. Click Submit 4. Enter “room52” 5. Click Submit	An error will be generated.	An error has been generated.	PASSED

Test Case Name: maint_batchrooms_zero1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters ‘0’	1. Click “Batch encoding for rooms” 2. Enter ‘0’ 3. Click Submit	User must encounter an error	The system still proceeded	FAILED

Test Case Name: maint_batchrooms_zero2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters ‘0’	1. Click “Batch encoding for rooms” 2. Enter ‘0’ 3. Click Submit	User must encounter an error	The user encountered an error	PASSED

Test Case Name: maint_rooms_delete

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User deletes a room	1. Click “Delete room” 2. Select “ROOM52” 3. Click “Delete”	The selected room must be deleted from the system.	The selected room has been deleted from the system.	PASSED

Test Case Name: maint_faculty_current

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User wants to check current faculty	Content: - ID - First Name - Last Name - Department - Job Description - Position - Date Added - Time Added	The user will see the current faculty data in the system.	The user saw the current faculty data in the system.	PASSED

Test Case Name: maint_faculty_add_valid

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User wants to add a new faculty member	1. Click “Add new faculty member” 2. Enter 2013061334 3. Enter “Paolo” 4. Enter “Valdes” 5. Select “Information Technology” 6. Select “Full-time” 7. Select “Prof.” 8. Click “Submit”	The faculty member will be successfully added.	The faculty member is successfully added.	PASSED

Test Case Name: maint_faculty_blankID

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves Faculty ID blank	1. Click “Add new faculty member” 2. Blank Faculty ID 3. Enter “Paolo” 4. Enter “Valdes” 5. Select “Information Technology” 6. Select “Full-time” 7. Select “Prof.”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

	8. Click “Submit”			

Test Case Name: maint_faculty_blankFirstName

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves First Name blank	1. Click “Add new faculty member” 2. Enter 2013061334 3. Blank first name 4. Enter “Valdes” 5. Select “Information Technology” 6. Select “Full-time” 7. Select “Prof.” 8. Click “Submit”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: maint_faculty_blankLastName

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves Last Name blank	1. Click “Add new faculty member” 2. Enter 2013061334 3. Enter “Paolo” 4. Blank last name 5. Select “Information Technology” 6. Select “Full-time” 7. Select “Prof.” 8. Click “Submit”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: maint_faculty_blankDepartment

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves Department blank	1. Click “Add new faculty member” 2. Enter 2013061334 3. Enter “Paolo” 4. Enter “Valdes”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

	5. Blank Department 6. Select “Full-time” 7. Select “Prof.” 8. Click “Submit”			

Test Case Name: maint_faculty_blankJobDescription

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves Job Description blank	1. Click “Add new faculty member” 2. Enter 2013061334 3. Enter “Paolo” 4. Enter “Valdes” 5. Select “Information Technology” 6. Blank Job Description 7. Select “Prof.” 8. Click “Submit”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: maint_faculty_blankPosition

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves Faculty ID blank	1. Click “Add new faculty member” 2. Blank Faculty ID 3. Enter “Paolo” 4. Enter “Valdes” 5. Select “Information Technology” 6. Select “Full-time” 7. Blank Position 8. Click “Submit”	The system will generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: maint_faculty_overlap

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters a duplicate	1. Click “Add new faculty member”	The system will generate an error.	The system generated an	PASSED

of a faculty member	2. Enter 2013061334 3. Enter “Paolo” 4. Enter “Valdes” 5. Select “Information Technology” 6. Select “Full-time” 7. Select “Prof.” 8. Click “Submit”		error.	

Test Case Name: maint_faculty_spChars

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User tries to enter special characters	1. Click “Add new faculty member” 2. Enter 2013061334 3. Enter “Paolo!@#” 4. Enter “Valdes” 5. Select “Information Technology” 6. Select “Full-time” 7. Select “Prof.” 8. Click “Submit”	No special characters must be generated	No special characters were generated	PASSED

Test Case Name: maint_batchfaculty_zero1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters zero	1. Click “Batch encoding for faculty” 2. Enter “0” 3. Click Submit	The system must generate an error	Zero was accepted	FAILED

Test Case Name: maint_batchfaculty_zero2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters zero	1. Click "Batch encoding for faculty" 2. Enter "0" 3. Click Submit	The system must generate an error	The system generated an error	PASSED

Test Case Name: maint_faculty_delete

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User deletes a faculty	1. Click "Delete room" 2. Select "Valdes Paolo" 3. Click "Delete"	The selected faculty must be deleted from the system.	The selected faculty has been deleted from the system.	PASSED

Test Case Name: maint_course_current

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User wants to check current courses	Content: - Course Code - Course Description - Lecture Units - Lab Units - Date Added - Time Added	The user will see the current courses in the system.	The user saw the current courses in the system.	PASSED

Test Case Name: maint_course_add_valid

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User wants to add a new course	1. Click “Add new course” 2. Enter “ICS111” 3. Enter “Computer Programming I” 4. Enter ‘3’ 5. Enter ‘1’ 6. Click “Submit”	The course will be successfully added	The course was successfully added	PASSED

Test Case Name: maint_course_blankID

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves the Course ID field blank	1. Click “Add new course” 2. Blank Course ID 3. Enter “Computer Programming I” 4. Enter ‘3’ 5. Enter ‘1’ 6. Click “Submit”	The system will generate an error tooltip.	The system generated an error tooltip.	PASSED

Test Case Name: maint_course_blankDescription

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves the Course Description field blank	1. Click “Add new course” 2. Enter “ICS111” 3. Blank Course Description 4. Enter ‘3’ 5. Enter ‘1’ 6. Click “Submit”	The system will generate an error tooltip.	The system generated an error tooltip.	PASSED

Test Case Name: maint_course_blankLecUnits

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves the Lecture Units field blank	1. Click "Add new course" 2. Enter "ICS111" 3. Enter "Computer Programming I" 4. Blank Lecture Units 5. Enter '1' 6. Click "Submit"	The system will generate an error tooltip.	The system generated an error tooltip.	PASSED

Test Case Name: maint_course_blankLabUnits

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves the Lab Units field blank	1. Click "Add new course" 2. Enter "ICS111" 3. Enter "Computer Programming I" 4. Enter '3' 5. Blank Lab Units 6. Click "Submit"	The system will generate an error tooltip.	The system generated an error tooltip.	PASSED

Test Case Name: maint_course_zeroUnits1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters 0 units for both lecture and lab	1. Click "Add new course" 2. Enter "ICS111" 3. Enter "Computer Programming I" 4. Enter '0' 5. Enter '0' 6. Click "Submit"	The system will generate an error.	The course was still added.	FAILED

Test Case Name: maint_course_zeroUnits2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters 0 units for both lecture and lab	1. Click "Add new course" 2. Enter "ICS111" 3. Enter "Computer Programming I" 4. Enter '0' 5. Enter '0' 6. Click "Submit"	The system will generate an error.	The system generated an error.	PASSED

Test Case Name: maint_course_overlap

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters a duplicate of an existing course	1. Click "Add new course" 2. Enter "ICS111" 3. Enter "Computer Programming I" 4. Enter '3' 5. Enter '1' 6. Click "Submit"	The system will generate an error.	The system generated an error	PASSED

Test Case Name: maint_batchcourse_zero1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User tries to enter '0'	1. Click "Batch encoding for courses" 2. Enter '0' 3. Click "Submit"	The system will generate an error.	The system did not generate an error	FAILED

Test Case Name: maint_batchcourse_zero2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User tries to enter '0'	1. Click "Batch encoding for courses" 2. Enter '0' 3. Click "Submit"	The system will generate an error.	The system generated an error.	PASSED

Test Case Name: maint_course_delete

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User deletes a course	1. Click "Delete course" 2. Select "ICS115" 3. Click "Delete"	The selected course must be deleted from the system.	The selected course has been deleted from the system.	PASSED

Test Case Name: maint_section_current

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User wants to check current sections	Content: - Section - Department - Free Day - Date Added - Time Added	The user will see the current sections in the system.	The user saw the current sections in the system.	PASSED

Test Case Name: maint_section_add_valid

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User wants to add a new section	1. Click "Add new section" 2. Enter "4ITA" 3. Select "Information Technology" 4. Select "Monday" 5. Click "Submit"	The section will be successfully added	The section was successfully added	PASSED

Test Case Name: maint_section_blankSec

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves the section field blank	1. Click "Add new section" 2. Blank Section 3. Select "Information Technology" 4. Select "Monday" 5. Click "Submit"	The system will generate an error tooltip.	The system generated an error tooltip.	PASSED

Test Case Name: maint_section_blankDept

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves the Department field blank	1. Click "Add new section" 2. Enter "4ITA" 3. Blank Department 4. Select "Monday" 5. Click "Submit"	The system will generate an error tooltip.	The system generated an error tooltip.	PASSED

Test Case Name: maint_section_blankDay

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User leaves the Free Day field blank	1. Click "Add new section" 2. Enter "4ITA" 3. Select "Information Technology" 4. Blank Free Day 5. Click "Submit"	The system will generate an error tooltip.	The system generated an error tooltip.	PASSED

Test Case Name: maint_section_sameSection1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters an existing section.	1. Click "Add Section" 2. Enter "4ITG" 3. Click Submit	The system will generate an error.	The section still registered.	FAILED

Test Case Name: maint_section_sameSection2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters an existing section	1. Click "Add Section" 2. Enter "4ITG" 3. Click Submit	The system will generate an error.	The system generated an error	PASSED

Test Case Name: maint_section_diffFreeDay1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters an existing section, but different free day	1. Click "Add Section" 2. Enter "4ITG" 3. Select "Tuesday" 4. Click Submit	The system will generate an error.	The section has been added.	FAILED

Test Case Name: maint_section_diffFreeDay2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User enters an existing section, but different free day	1. Click "Add Section" 2. Enter "4ITG" 3. Select "Tuesday" 4. Click Submit	The system will generate an error.	The system generated an error.	PASSED

Test Case Name: maint_batchsection_zero1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User tries to enter '0'	1. Click "Batch encoding for Sections" 2. Enter '0' 3. Click "Submit"	The system will generate an error.	The system did not generate an error	FAILED

Test Case Name: maint_batchsection_zero2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User tries to enter '0'	1. Click "Batch encoding for Sections" 2. Enter '0' 3. Click "Submit"	The system will generate an error.	The system generated an error	PASSED

Test Case Name: maint_section_delete

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User deletes a section	1. Click "Delete section" 2. Select "3ITB" 3. Click "Delete"	The selected section must be deleted from the system.	The selected section has been deleted from the system.	PASSED

Test Case Name: user_current

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin wants to check current users	Content: - Username - Access Level - Date Received - Time Received - Date Updated - Time Updated	The current users in the database must be seen.	The current users in the database were seen.	PASSED

Test Case Name: user_pending

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin wants to check pending users	Content: - Username - Access Level - Date Received - Time Received - Date Updated - Time Updated	The pending users in the database must be seen.	The pending users in the database were seen.	PASSED

Test Case Name: user_pending_update

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin updates a new user to be verified	1. Click “Pending users in the database” 2. Select “Pao.Valdes” 3. Click “Update”	The pending user must be verified.	The pending user was verified.	PASSED

Test Case Name: user_pending_updateDeptHead

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin updates a new user to be verified	1. Click “Pending users in the database” 2. Select “Dept.Head” 3. Click “Update”	The pending user must be verified.	The pending user was verified.	PASSED

Test Case Name: user_pending_updateEmail1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin updates a new user to be verified	1. Click “Pending users in the database” 2. Select “Pao.Valdes” 3. Click “Update” 4. Check email	The user must receive an email.	The user did not receive an email	FAILED

Test Case Name: user_pending_updateEmail2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin updates a new user to be verified	1. Click “Pending users in the database” 2. Select “Pao.Valdes” 3. Click “Update” 4. Check email	The user must receive an email.	The user received an email.	PASSED

Test Case Name: user_edit

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin checks users that can be updated.	1. Click “Edit user access”	The admin must see the non-Dept.Head users.	The admin saw the non-Dept.Head users.	PASSED

Test Case Name: user_edit_user

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin updates a “user”	1. Click “Edit user access” 2. Select user with “user” as access level 3. Click “Update”	The “user” must become an “admin”	The “user” became an “admin”	PASSED

Test Case Name: user_edit_admin

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin updates an "admin"	1. Click "Edit user access" 2. Select user with "admin" as access level 3. Click "Update"	The "admin" must become a "user"	The "admin" became a "user"	PASSED

Test Case Name: user_edit_userEmail1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User checks for email	1. Click "Edit user access" 2. Select user with "user" as access level 3. Click "Update" 4. Check email	The user must receive an email	The email was received by a different user.	FAILED

Test Case Name: user_edit_userEmail2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User checks for email	1. Click "Edit user access" 2. Select user with "user" as access level 3. Click "Update" 4. Check email	The user must receive an email	The user received an email.	PASSED

Test Case Name: user_edit_adminEmail1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin checks for email	1. Click "Edit user access" 2. Select user with "admin" as access level 3. Click "Update" 4. Check email	The admin must receive an email	The email was received by a different user.	FAILED

Test Case Name: user_edit_adminEmail2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin checks for email	1. Click “Edit user access” 2. Select user with “admin” as access level 3. Click “Update” 4. Check email	The admin must receive an email	The admin received an email.	PASSED

Test Case Name: user_delete

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin wants to delete an account.	1. Click “Delete user accounts” 2. Select “Pao.Valdes” 3. Click “Delete”	The selected account/s must be deleted.	The selected account/s was deleted.	PASSED

Test Case Name: user_deleteEmail1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin wants to delete an account.	1. Click “Delete user accounts” 2. Select “Pao.Valdes” 3. Click “Delete” 4. Check email	The recipient must receive an email.	The email was received by another recipient.	FAILED

Test Case Name: user_deleteEmail2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin wants to delete an account.	1. Click “Delete user accounts” 2. Select “Pao.Valdes” 3. Click “Delete” 4. Check email	The recipient must receive an email.	The email was received by another recipient.	PASSED

Test Case Name: maint_requests

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin wants to check pending reservation requests.	Content: - Request Title - Course - Section - Department - Start - End - Room - User - Date Requested - Time Requested - Status - Action	The admin must be able to see the list of reservation requests.	The admin was able to see the list of reservation requests.	PASSED

Test Case Name: maint_requests_approve

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin approves particular reservation request.	1. Approve particular request	The request must be approved.	The request has been approved.	PASSED

Test Case Name: maint_requests_approveEmail1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin approves particular reservation request.	1. Approve particular request 2. Check email	The recipient must receive an email	The recipient did not receive an email.	FAILED

Test Case Name: maint_requests_approveEmail2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin approves particular reservation request.	1. Approve particular request 2. Check email	The recipient must receive an email	The recipient received an email.	PASSED

Test Case Name: maint_requests_deny

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin denies particular reservation request.	1. Deny particular request 2. Enter Reason for Denial 3. Click “Submit”	The request must be denied.	The request has been denied.	PASSED

Test Case Name: maint_requests_denyEmail1

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin denies particular reservation request.	1. Deny particular request 2. Check email	The recipient must receive an email	The email was received by a different recipient	FAILED

Test Case Name: maint_requests_denyEmail2

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin denies particular reservation request.	1. Deny particular request 2. Check email	The recipient must receive an email	The recipient received the email.	PASSED

Test Case Name: maint_archive

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin archives schedule	1. Select "2019" 2. Select "2nd Semester" 3. Click "Archive Semester" 4. Click "OK" in popup message	The current schedule must be archived.	The schedule has been archived.	PASSED

Test Case Name: maint_archive_blankYear

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin leaves academic year blank	1. Blank academic year 2. Select "2nd Semester" 3. Click "Archive Semester" 4. Click "OK" in popup message	The system must generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: maint_archive_blankSem

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Admin leaves semester blank	1. Select "2018" 2. Blank semester 3. Click "Archive Semester" 4. Click "OK" in popup message	The system must generate an error tooltip	The system generated an error tooltip	PASSED

Test Case Name: sched_roomExport

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Department Head wants a softcopy of room schedules	1. Click "Scheduling" 2. Click "Room View" 3. Click "Export Excel"	The system must download a softcopy of the schedule	The system downloaded a softcopy of the schedule	PASSED

Test Case Name: sched_facultyExport

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Department Head wants a softcopy of a faculty schedule	1. Click “Scheduling” 2. Click “Faculty View” 3. Click “Export Excel”	The system must download a softcopy of the schedule	The system downloaded a softcopy of the schedule	PASSED

Test Case Name: sched_sectionExport

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Department Head wants a softcopy of a section schedule	1. Click “Scheduling” 2. Click “Section View” 3. Click “Export Excel”	The system must download a softcopy of the schedule	The system downloaded a softcopy of the schedule	PASSED

Test Case Name: sched_addValid

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Department Head wants to add a new schedule	1. Click “Add Schedule” 2. Select “ICS115” 3. Select “Room52” 4. Select “Alma Perol” 5. Select “4ITB” 6. Select “Friday” 7. Select “10:00AM” 8. Select “12:00PM” 9. Click “Add”	The system must generate a popup message.	The system generated a popup message.	PASSED

Test Case Name: sched_blankCourse

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Department Head left Course	1. Click “Add Schedule” 2. Blank Course 3. Select “Room52”	The system must generate an	The system generated an error tooltip.	PASSED

field blank.	4. Select “Alma Perol” 5. Select “4ITB” 6. Select “Friday” 7. Select “10:00AM” 8. Select “12:00PM” 9. Click “Add”	error tooltip.		

Test Case Name: sched_blankRoom

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Department Head left Room field blank.	1. Click “Add Schedule” 2. Select “ICS115” 3. Blank Room 4. Select “Alma Perol” 5. Select “4ITB” 6. Select “Friday” 7. Select “10:00AM” 8. Select “12:00PM” 9. Click “Add”	The system must generate an error tooltip.	The system generated an error tooltip.	PASSED

Test Case Name: sched_blankFaculty

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Department Head left Faculty field blank.	1. Click “Add Schedule” 2. Select “ICS115” 3. Select “Room52” 4. Blank Faculty 5. Select “4ITB” 6. Select “Friday” 7. Select “10:00AM” 8. Select “12:00PM” 9. Click “Add”	The system must generate an error tooltip.	The system generated an error tooltip.	PASSED

Test Case Name: sched_blankSection

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Department Head left Section field blank.	1. Click “Add Schedule” 2. Select “ICS115” 3. Select “Room52” 4. Select “Alma Perol” 5. Blank Section 6. Select “Friday” 7. Select “10:00AM” 8. Select “12:00PM” 9. Click “Add”	The system must generate an error tooltip.	The system generated an error tooltip.	PASSED

Test Case Name: sched_blankDay

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Department Head left Day field blank.	1. Click “Add Schedule” 2. Select “ICS115” 3. Select “Room52” 4. Select “Alma Perol” 5. Select “4ITB” 6. Blank Day 7. Select “10:00AM” 8. Select “12:00PM” 9. Click “Add”	The system must generate an error tooltip.	The system generated an error tooltip.	PASSED

Test Case Name: sched_blankStartTime

Test Case	Steps	Expected Results	Actual Results	Remarks
1. Department Head left Start Time field blank.	1. Click “Add Schedule” 2. Select “ICS115” 3. Select “Room52” 4. Select “Alma Perol” 5. Select “4ITB” 6. Select “Friday” 7. Blank Start Time 8. Click “Add”	The system must generate an error tooltip.	The system generated an error tooltip.	PASSED

Test Case Name: requests_view

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User wants to view a reservation request	Content: - Pending requests - Approved requests - Denied requests - Cancelled requests	The system must display the list of reservation requests.	The system displayed the list of reservation requests.	PASSED

Test Case Name: requests_cancel

Test Case	Steps	Expected Results	Actual Results	Remarks
1. User wants to cancel a reservation request	1. Click “Your requests” 2. Click “Cancel” 3. Enter Reason for Cancellation 4. Click “Submit”	The particular request must be removed from the list of requests.	The particular request has been removed	PASSED

Curriculum Vitae

Quitor, Christian Jeus D.
231 E. F . Fernandez St.,
San Juan, Metro Manila
Mobile no.: 0916-649-4037
E-mail Address: Christian.academics.ust@gmail.com

OBJECTIVE

To become associated in a company where I can enhance my knowledge and skills and to gain experience in the field that I have chosen.

EDUCATION

University of Santo Tomas, Espana, Manila, Philippines
Bachelor of Science in Information Technology
2014-2019

Don Bosco Sta. Mesa
June 2010- March 2014

AWARDS RECEIVED

IBM DB2 Associate, University of Santo Tomas, Manila, 2015

ACADEMIC PROJECTS

Project Manager in Kali Astacom Website, August, 2017
Quality Assurance in Oracle Laundry Transaction System, UST IICS, March 9, 2017
Project Manager and Developer in Room Reservation and Scheduling for IICS, August 2018

SKILLS & ABILITIES

- Computer: Microsoft Office, Adobe Premiere Pro
- Specialization: Web and Mobile Development
- Programming languages: C++, Java, PHP, Assembly Language, Javascript, Json, Jquery, C#, Javascript
- Oral and Written Communication in English and Filipino
- Other IT Related: Cisco Networking, MySQL and IBM DB2, Bootstrap, CSS, HTML, Ethical Hacking

Plata, Tristan Miguel C.

L21-G Blk 3, Centro de Buenviaje, Sto. Nino, Marikina City
tmcplata@gmail.com
+639355318451

Objective:

Seeking a summer internship where my knowledge in web design will be utilized while obtaining experience in the industry and allow me to explore career options in the IT sector.

Educational Background:

Tertiary:	University of Santo Tomas (2014-present) Bachelor of Science in Information Technology
Secondary:	San Lorenzo Ruiz de Manila School, Marikina City (2010-2014)
Primary:	St. Mary's Academy of Sto.Niño, Meycauayan, Bulacan (2004-2010)

Training/Seminars:

- Ethics and IT: Back to the beginning, Faculty of Arts and Letters, Prof. John Weckert, Benavides Auditorium, UST, February 2017
- Potatalk: Ruby on Rails Workshop, Engineering Conference Hall, UST, April 2017
- Developers Connect Philippines (DevCon), Medicine Auditorium, UST, April 2018

Awards/Certificates:

- Cyber Security Awareness, Department of Defense USA, April 26, 2017
- Identifying and Safeguarding Personally Identifiable Information, Department of Defense USA, April 26, 2017

Skills:

- Basic Knowledge in Struts2 framework
- Skill programming (Java, HTML, PHP, CSS)
- Basic Knowledge in MS Word, Excel, and PowerPoint
- Networking: Basic Router and Switching
- Able to communicate effectively within a team during collaborations
- Able to meet project deadlines
- Able to adapt to given situations

RICH CARL MICHAEL RIVERA SILVESTRE

32C APA Compound Philand Drive Tandang Sora Q.C
(+63)927 627 3067, (+63)925 507 7778
richcarl.silvestre@gmail.com

EDUCATION

Bachelor of Science in Information Technology
Institute of Information and Computing Sciences
University of Santo Tomas, España, Sampaloc, Manila
June 2019

CO-CURRICULAR ACTIVITES

UST Wushu Flaming Tigers , University of Santo Tomas
Membership, 2014

EXAMINATIONS TAKEN

PhilNITS: JITSE Exam Passer, April 2018

COMMUNITY ENGAGEMENT

- **National Service Training Program (NSTP),** University of Santo Tomas, Manila S.Y. 2015-2016

TRAININGS/SEMINARS ATTENDED

- **E-Waste-O Seminar, Institute of Information and Computing Sciences, Medicine Auditorium***, April 2018*
- **Data Privacy Act of the Philippines and Security Threats, Medicine Auditorium,** *UST, April 2018*
- **Security Threats, Medicine Auditorium,** *UST, April 2018*
- **Python Training***, Cryptors, Makati, November 2017*
- **Certificate of Completion in Python Training,** *Cryptors, Makati, November 2017*

- **CompTIA Linux+**, *Cybrary, August 2017*
- **Cryptography**, *Cybrary, July 2017*
- **Certificate of Completion in Cryptography,** *Cybrary, July 2017*
- **Certificate of Completion in Web Hacking: WAF Bypass and RCE Training,** *Cryptors, Makati,*
  June 2017
- **Web Hacking: WAF Bypass and RCE Training**, *Cryptors, Makati, June 2017*
- **Potatalk: Introduction to Git Seminar**, *Engineering AVR, Roque Roano Building, UST*
- **Potatalk: Ruby on Rails**, *Roque Roano Building, UST*

ACADEMIC PROJECTS

USTProject: Online Parish Services, Institute of Information and Computing Sciences, University of Santo Tomas, Present

SKILLS AND ABILITIES

- Basic programming skills in HTML, CSS, Java, C, C++, C#, Python, PHP, and MySQL.
- Other software: Familiar with Autodesk Maya, Adobe Animate, Adobe Dreamweaver, Macromedia Flash, and Unity Engine.
- Network: Networking Basics, Routing, Switching and Network Security.
- Knowledgeable in Linux and Unix Operating System.
- Oral and Written Communication in English, Filipino, Basic German, and Basic Spanish

PAOLO OTAZU VALDES

50 Pao Street, Santa Mesa Heights, Q.C.

Contact Nos.: 954-53-90/0917-751-2784

valdespaolo@gmail.com

Career Objective

To work in a company that will build my competence in business analytics and software testing.

Accomplishments

Philippine National I.T. Standards (PhilNITS): JITSE Exam Passer, Manila, April 2018

Educational Background

2013-2019 **University of Santo Tomas, España, Manila**

Currently pursuing a Bachelor of Science Degree in Information Technology, (2013 - 2018)

2009-2013 **Lourdes School of Quezon City, Kanlaon cor. Don Manuel St., Q.C.**

Completed high school, March 2013.

2003-2009	**Lourdes School of Quezon City, Kanlaon cor. Don Manuel St., Q.C.** Completed grade school, March 2009
2001-2003	**Lourdes School of Quezon City, Kanlaon cor. Don Manuel St., Q.C.** Completed Kindergarten-Prep.

Community Engagement

National Service Training Program (NSTP), University of Santo Tomas, Manila, 2014-2015

Skills and Abilities

- Good grammatical skills
- Analytic skills in system testing
- Document making
- Creative thinking
- Essay-writing skills
- Oral and written communication in English

Personal Background

Born in Quezon City, May 18, 1996. Literary Aspirant. Able to speak English and Tagalog