|  |  |  |
| --- | --- | --- |
| **Systems Design and Development** | | |
| Student ID: 18406552 | Date of Submission: 5th May | |
| Student name: Tenzin Dhundup Sherpa | Tutor: Mrs.Anita Gurung | Module id: CSY2030 |
| Video link: https://northampton.mediaspace.kaltura.com/media/Screen+Capture+-+2019+May+06+02A16A06/1\_l80a1mx8 | | |

# Introduction

This is the report of system design and development (CSY2030). This report consists of problem statement and solution for the system. The assessment was to create a University Human Resource System using object programming in JAVA. Our task was to create a graphical user interface through which Admin and Department Secretary can accessed the system to perform different task. The application would allow admin to add new lecturers, departments, view and delete lecturers and departments if necessary after logging into the system. Also, the department secretary could use the system to view the lecturer and add lecturer to certain departments. This program will help Admin and Department Secretary to tack the organization activity efficiently.

The report is divided into different segments. Each segment deal with different section of the task. The approach report segments are listed below

Introduction to report

Design of system (use case diagram, class diagram): On this part of the report the user shows the design of the system of the system with the help of visual representation of the system. The user has created use case diagram and class diagram of the operation of the system and explained in brief about the purpose of the design

System Description: This segment of the report shows the functionality of the system with the help of screenshot of the actual program that the developer created. Developer has also explained in brief of the functionality of the program with the help of various screenshots.

Testing: On this section, the developer shows two type of testing. Black box testing is included which help to understand what system supposed to do? In this section also, developer have included various output screenshots. Furthermore, the white box testing is also done to show how system works and included screenshots of related coding implementation.

Conclusion: This section of the report summarizes the task in a whole.

# Diagram

## Class Diagram

Figure 1 Use Case Diagram

Figure 1 Use Case Diagram

Figure 1 Use Case Diagram

*Figure 0 Class Diagram*

## Use Case Diagram

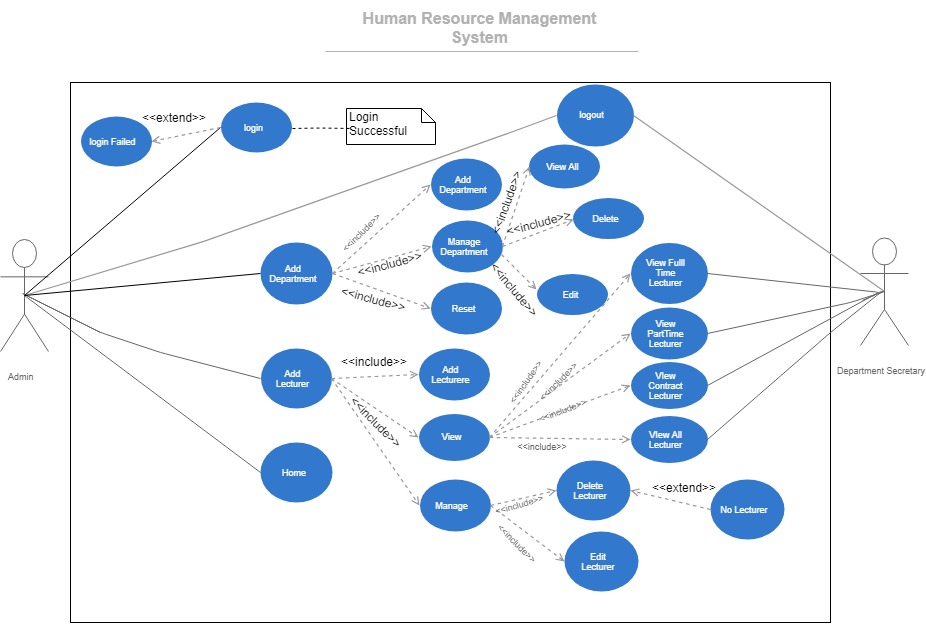


Figure 1 Use Case Diagram

**Documentation for Use Case Diagram**

1. **UC1 Login**

**Name:** UC1 Login

**Initiator**: Administrator, Department Secretary

**Goal:** Administrator log in to the system

Department Secretary log in to the system

**Pre-condition:** Administrator is not logged into the system.

Department Secretary is not logged into the system.

**Post-condition:** Administrator log into the system and redirects to respective homepage

The secretary log into the system and redirects to respective homepage

**Assumption:** Administrator and Department Secretary are both valid members.

**Main Success Scenario**

* Admin and Department Secretary log in to the system.
* Checks if Admin or Secretary is logged in.
* If valid username and password, then redirects to respective homepage.
* If Admin is logged in, Admin is redirects to admin homepage.
* If Secretary is logged in, Sectary homepage is accessed.

**Excludes:** Login failed.

1. **UC3 Add Department**

**Name**: UC2 Add Department

**Initiator:** Administrator

**Goal:** Adding new department

**Pre-condition:** Administrator should be logged in.

**Post-condition:** New Department added successfully by Administrator.

**Assumption:** Administrator can add new department.

**Main Success Scenario**

* Successfully adding new department

**Includes:** Add Department

View Department

Reset

1. **UC2 Add Lecturer**

**Name**: UC2 Add Lecturer

**Initiator:** Administrator

**Goal:** Adding new Lecturer.

**Pre-condition:** Administrator should be logged in.

**Post-condition:** New Lecturer added successfully by Administrator.

**Assumption:** Administrator can add new lecturers.

**Main Success Scenario**

* Successfully adding new lecturers.

**Includes:** Add Lecturer

View Lecturer

Manage Lecture.

1. **UC4 Home**

**Name:** UC4 Home

**Initiator**: Administrator, Department Secretary

**Goal:** Administrator log in to the system

Department Secretary log in to the system

**Pre-condition:** Administrator must be logged in.

Department Secretary must be logged in.

**Post-condition:** Administrator goes back to home page

The secretary goes back to homepage

**Assumption:** Administrator and Department Secretary are both valid members and logged in.

**Main Success Scenario**

* Admin redirects back to admin homepage
* Department Secretary redirect back to secretary homepage

1. **UC5 View Full Time Lecturer**

**Name**: UC5 View Full Time Lecturer

**Initiator:** Administrator and Department Secretary

**Goal:** Administrator and Department Secretary views the details of Full Time lecturer.

**Pre-condition:** Administrator is logged in to the system successfully.

Department Secretary is logged in to the system successfully.

**Post-condition:** Administrator views all the details of Full Time lecturer.

DepartmentSecretary views all the details of Full Time lecturer

**Assumption:**

* Administrationis logged in and is able to view full time lecturer information
* Department Secretaryis logged in and is able to view full time lecturer information

**Main Success Scenario**

* Administrator views all information of all Full Time lecturer.
* Department Secretary views all information of all Full Time lecturer.

1. **UC6 View Part Time Lecturer**

**Name**: UC6 View Part Time Lecturer

**Initiator:** Administrator and Department Secretary

**Goal:** Administrator and Department Secretary views the details of Part Time lecturer.

**Pre-condition:** Administrator is logged in to the system successfully.

Department Secretary is logged in to the system successfully.

**Post-condition:** Administrator views all the details of Part Time lecturers.

DepartmentSecretary views all the details of Part Time lecturers.

**Assumption:**

* Administrationis logged in and is able to view part time lecturer information
* Department Secretaryis logged in and is able to view part time lecturer information.

**Main Success Scenario**

* Administrator views all information of all Part Time lecturer.
* Department Secretary views all information of all Part Time lecturer.

1. **UC7 View Contract Lecturer**

**Name**: UC7 View Contract Lecturer

**Initiator:** Administrator and Department Secretary

**Goal:** Administrator and Department Secretary views the details of Contract Lecturer.

**Pre-condition:** Administrator is logged in to the system successfully.

Department Secretary is logged in to the system successfully.

**Post-condition:** Administrator views all the details of Contract lecturer.

DepartmentSecretary views all the details of Contract lecturer

**Assumption:**

* Administrationis logged in and is able to view contract lecturer information
* Department Secretaryis logged in and is able to view contract lecturer information.

**Main Success Scenario**

* Administrator views all information of all Contract lecturer.
* Department Secretary views all information of all Contract lecturer.

1. **UC8 View All Lecturers**

**Name**: UC8 View All Lecturer

**Initiator:** Administrator and Department Secretary

**Goal:** Administrator and Department Secretary views the details of All lecturers.

**Pre-condition:** Administrator is logged in to the system successfully.

Department Secretary is logged in to the system successfully.

**Post-condition:** Administrator views all the details of All lecturer.

DepartmentSecretary views all the details of All lecturer

**Assumption:**

* Administrationis logged in and is able to view all the lecturer’s information
* Department Secretaryis logged in and is able to view all the lecturer’s information

**Main Success Scenario**

* Administrator views all information of All lecturer.
* Department Secretary views all information of All lecturer.

1. **UC9 MANAGE Department**

**Name**: UC9 Manage Department

**Initiator:** Administrator

**Goal:** Redirecting the administrator to managing department page

**Pre-condition:** Administrator should be logged in.

**Post-condition:** Administrator can perform feature of manage department page.

**Assumption:** Administrator is logged in to the system and can manage department

**Main Success Scenario**

1. Successfully redirecting to manage department page

**Includes:** View ALL

Delete

Edit

1. **UC10 View ALL**

**Name**: UC10 View ALL(Department)

**Initiator:** Administrator

**Goal:** Administrator views all details of department that are in the system.

**Pre-condition:** Administrator should be logged in.

**Post-condition:** Administrator view the details of all the department that are in the system.

**Assumption:**

* Administrationis logged into the system and views the details information of department.

**Main Success Scenario**

* Administrator views all the details of department that are in the system.

1. **UC11 Delete Department**

**Name**: UC11 Delete Department

**Initiator:** Administrator

**Goal:** Administrator delete all details of department that are in the system.

**Pre-condition:** Administrator should be logged in.

**Post-condition:** Administrator delete the department by giving its id.

**Assumption:**

* Administrationis logged into the system and is authorized to delete the department from the system.

**Main Success Scenario**

* Administrator delete all the details of department that are in the system.

1. **UC12 Edit Department**

**Name**: UC12 Edit Department

**Initiator:** Administrator

**Goal:** Administrator changes details of department that are in the system.

**Pre-condition:** Administrator should be logged in.

**Post-condition:** Administrator changes the department details by giving id of that department.

**Assumption:**

* Administrationis logged into the system and is authorized to change the details of department that are in the system.

**Main Success Scenario**

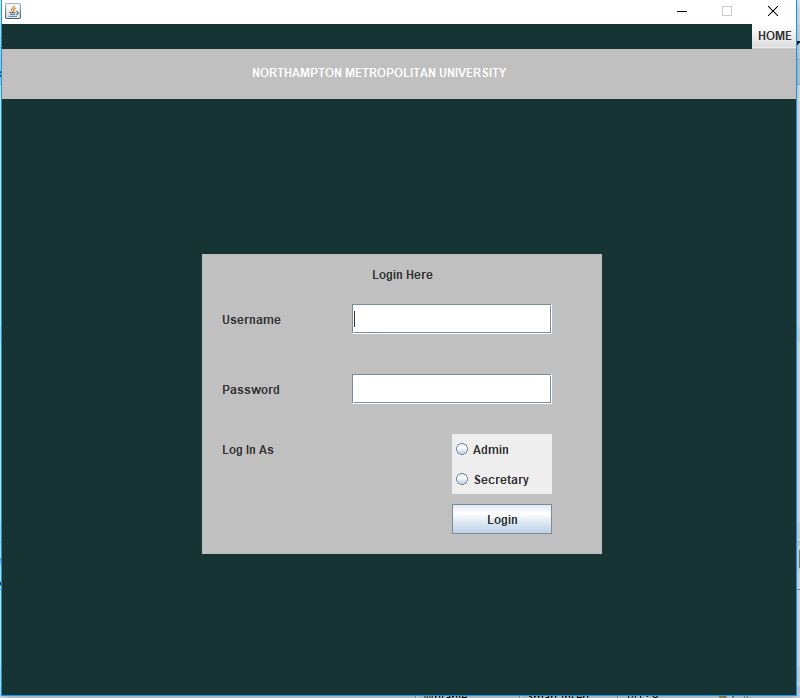
* Administrator changes the details of department that are in the system.

1. **UC13**

# System Description

This part includes the details about how the system looks and can be run by the user. It is like a guide book to user for using the newly developed human resource management system. It also discusses about the different component and function that are implemented in the system.

* **Login portal of the system**

Figure 2 Login portal for admin and secretary****

In the above figure, login portal to login to the system is shown. There are two fields where the username and password are to be given and if the admin is logging then, he/she should click on admin button which is just above the login button on the right-hand bottom corner. Similarly, if the secretary is logging in to the system, he/she should click on secretary before clicking on login button.

* **Logged In Message**

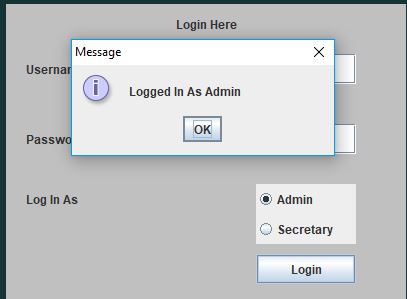


Figure 3 pop up message once logged in

So, after giving the username and password and clicked as admin, he/she is logged in to the system and message will be pop out as “log in as admin”.

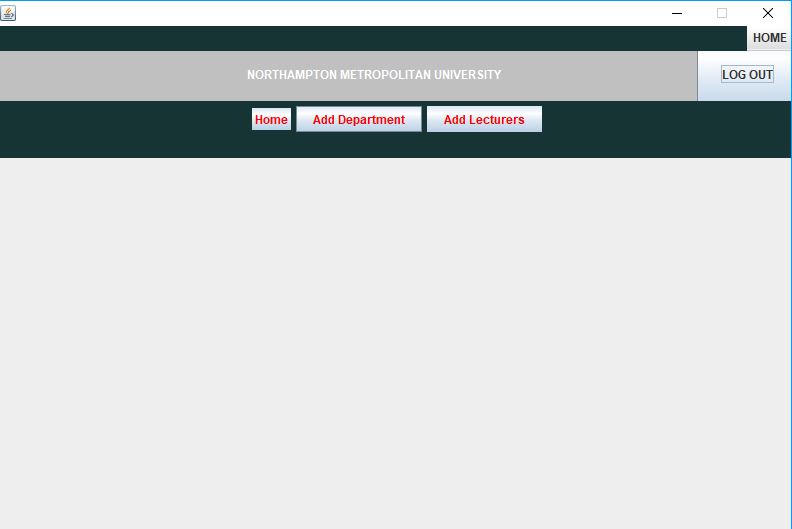
* **Admin Home page**

Figure 4 Admin Homepage

After the admin is successfully logged in, he/she is redirected to admin home page which is shown in figure 4. Here in the admin homepage, there are three functionalities that the admin can do. Admin can return back to current page by clicking on home button, Admin can add new department by clicking on add department button and likewise Admin can add new Lecturers by clicking on add lecturer button on top navigation bar.

* **Add Department**

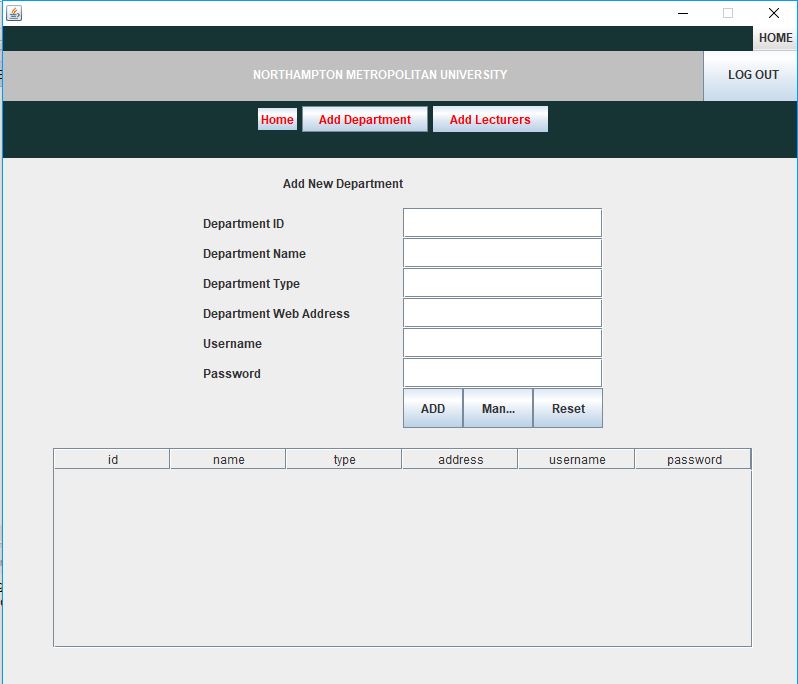
****

Figure 5 adding new department

Here in the above figure 5, the form to add new department is shown. Once the admin clicks on Add Department button, he/she is redirected to this page (shown above, fig 5). From this page, Admin can manage Admin Data and reset the data that are written in current forma and Admin add the new department by filing this form and after clicking ADD button, the department will be added.

* Manage Department

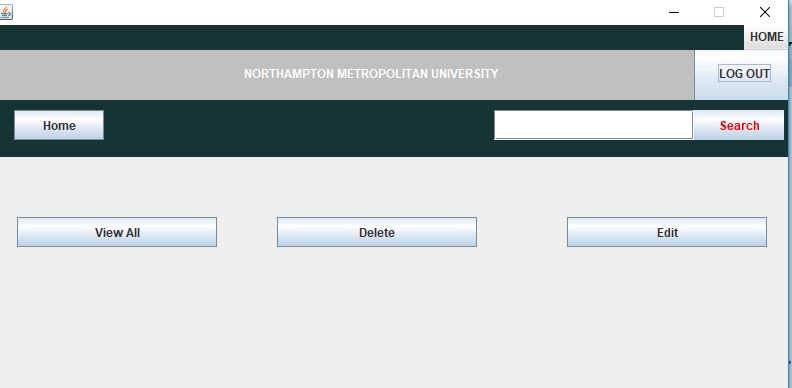


Figure 6 Manage Department page

In the figure 5, in add department page, there were three functionality that the Admin could perform. There were ADD, manage and reset functionality. So, if the Admin click on manage button then he/she is redirected to manage department page (figure 6). From here, Admin can view all the department data, delete and edit those department data.

* **View All Department**

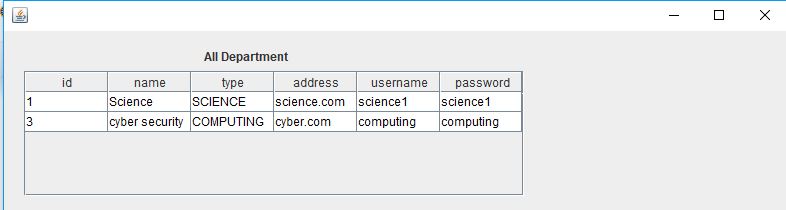
****

Figure 7 View all department data

In the figure 7, information of department that are in the system are shown in table. After the

Admin clicks on view all button, Admin can view this page.

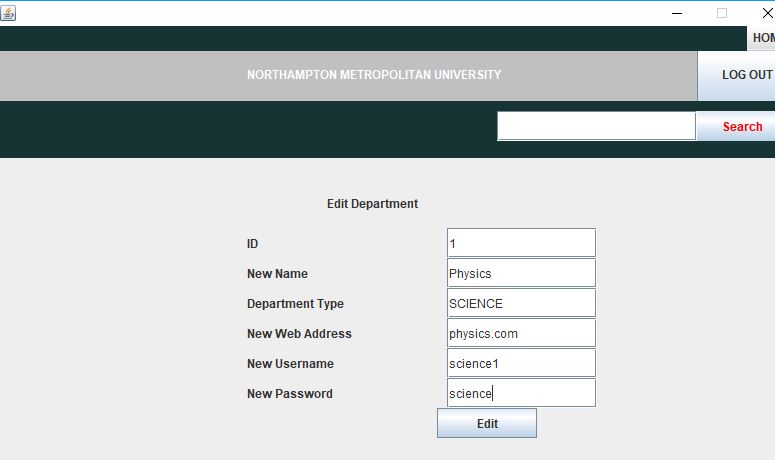
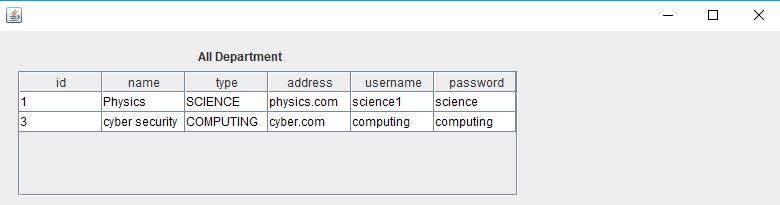
* While updating the department

Figure 8 Editing Department

In the figure 8, the form to edit the department details is shown. Here Admin is editing the data of department with ID 1. Previously we had department name as Science and web address as science.com and password as science1, but the new data in their place can be seen in above form. So, once the edit button is clicked, department details is updated.

* **Updated Department Details**

****Figure 9 Updated Department Details

So, in the figure 9, the new updated data are shown. Since the data was changed of department with ID 1, its data has been successfully updated.

* **Delete Department**

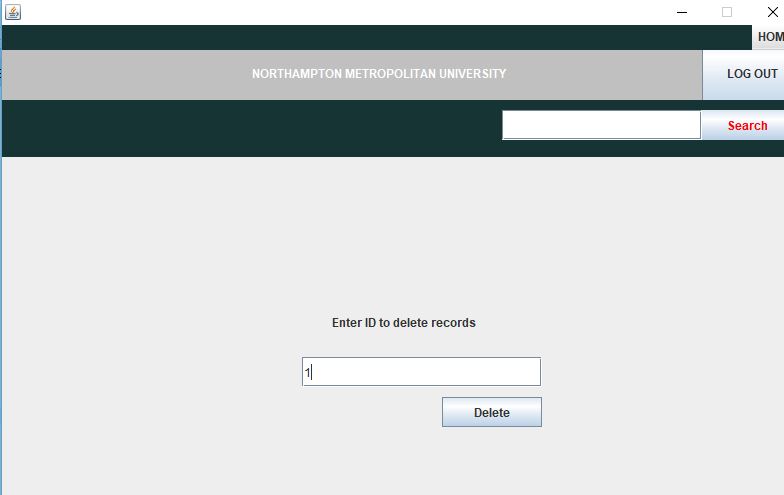
****

Figure 10 Deleting Department

In the above figure 10, deleting a department process is shown. Admin can delete any department by entering their ID and once the ID is entered, it can be delete from the system by clicking delete button.

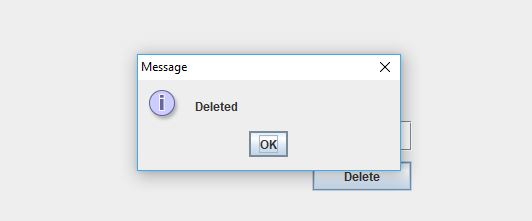


Figure 11 Deleted Message

Once the delete button is clicked, if the operation is successfully completed then system shows the on-screen message saying” Deleted”.

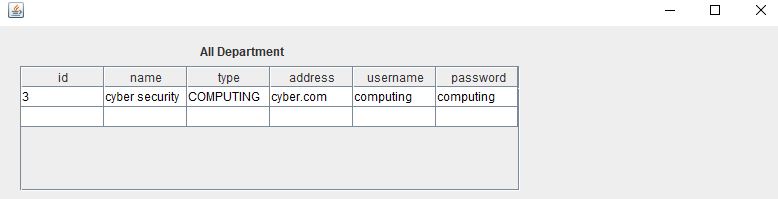


Figure 12 Department deleted from system

The figure 12 shows that the system has deleted department with ID 1 and it can be seen that there is only one row of data with ID (3). Since the Admin entered ID 1 while deleting process, its details is deleted from the system.

* **Add Lecturer**

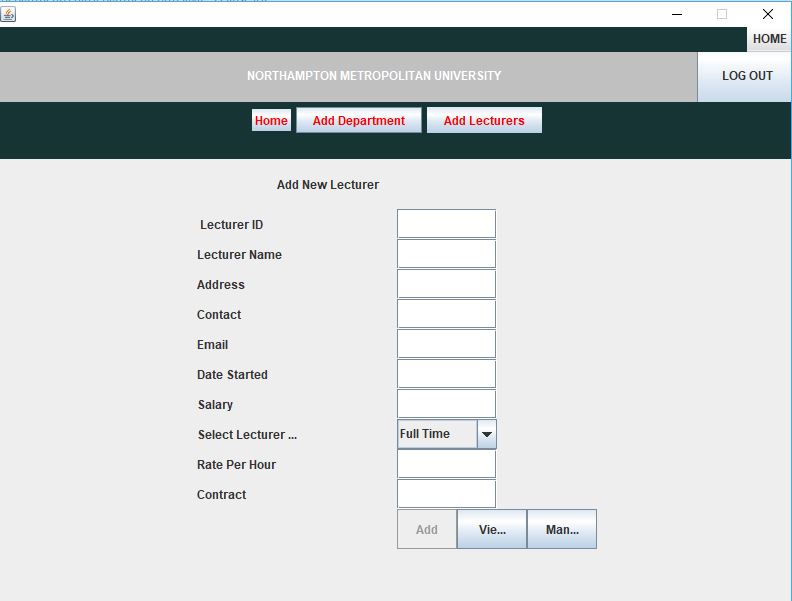
****

Figure 13 Add lecturer

In the figure 13, the process for adding a lecturer is shown. Once the Admin clicks on Add lecturer button from navigation bar, Admin is redirected to Adding new lecturer page (figure 13). Here, at first Add button is disabled here. But once the Admin fill the form to add new lecturer and selected the lecturer type, add button will be enabled and can add the new Lecturer by clicking on it. There are also View and Manage button beside Add button.

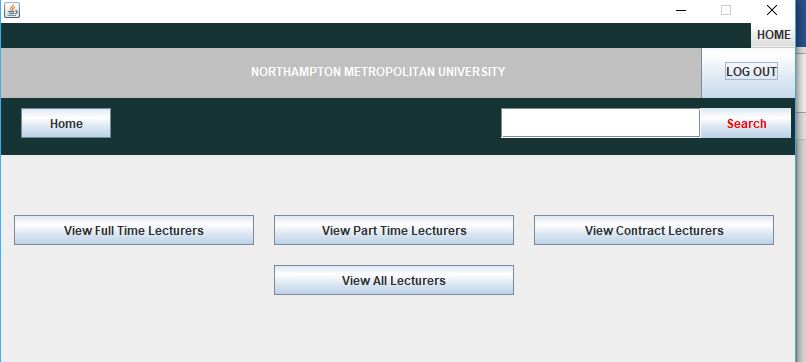


Figure 14 View Lecturer

So, if the View button is clicked, Admin is redirected into view lecturers page. Here, Admin can view different lecturers as per their type. When view full time lecturers is clicked, full time lecturer details are shown as per the figure below.

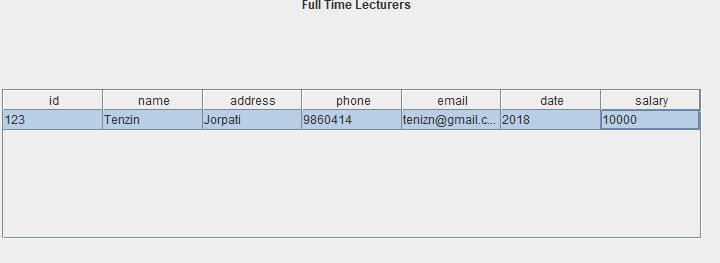


Figure 15 Full Time Lecturer Details

In this way, Admin can view the details of other lecturers also. Admin can also view all the lecturers that are in the system by clicking on View All Lecturers button.

* Manage Lecturers

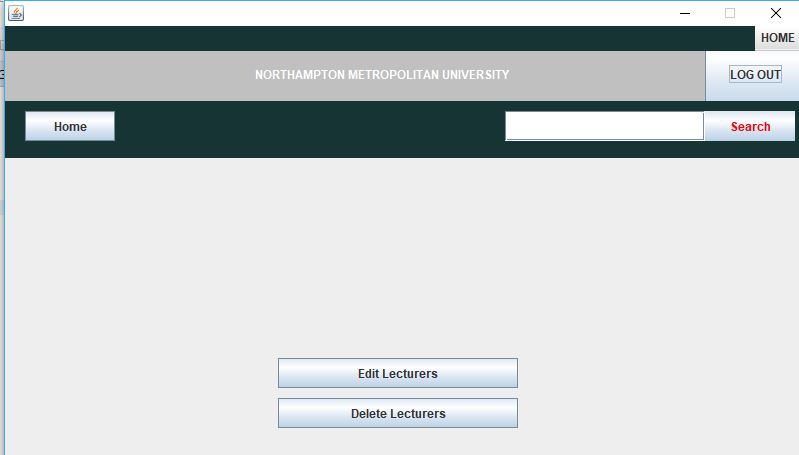
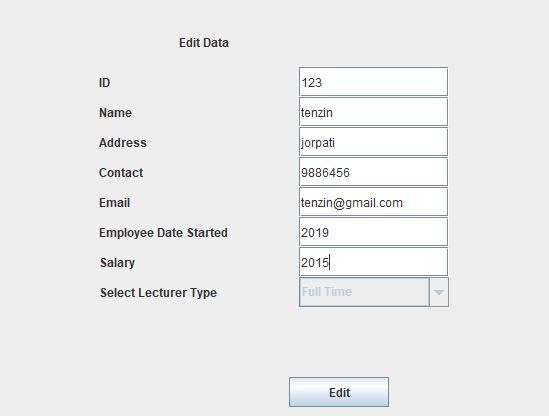


Figure 16 Manage Lecturers

In the figure 16 above, the page to edit and delete the lecturers is shown. When the Admin click on Manage button in Add lecturer page (see figure 13), Admin is directed to manage lecturers page. In this page, Admin can Edit and Delete the existing lecturer of system. If the Admin clicks on Edit lecturers’ button, a page to edit the current lecturers is opened. Below is the screenshot of how it looks like.

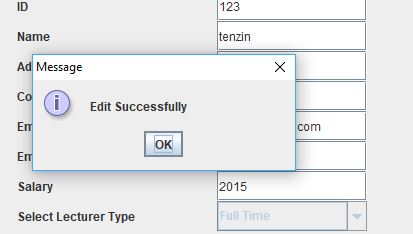
Figure 17 Editing lecturers

Figure 18 Edit successfully

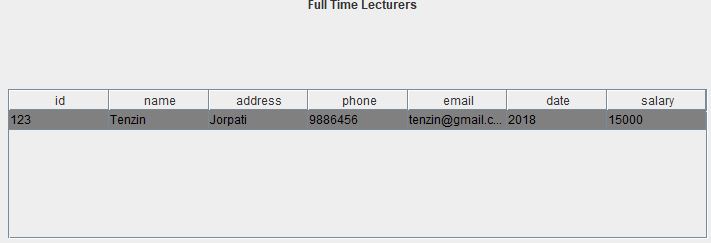


Figure 19 Updated Lecturers

So, if the edit process is successfully completed, system shows edit successfully message as in figure 18 and the updated data of lecturer can be seen on figure 19. Here, the admin has increased the salary of lecturer to 15000 from 10000 and it is successfully updated.

* **Secretary Login**

Since the Admin and secretary has different authority they cannot login to the system using same username and password. That’s why they are to be logged into the system separately.

Below figure shows the secretary login process

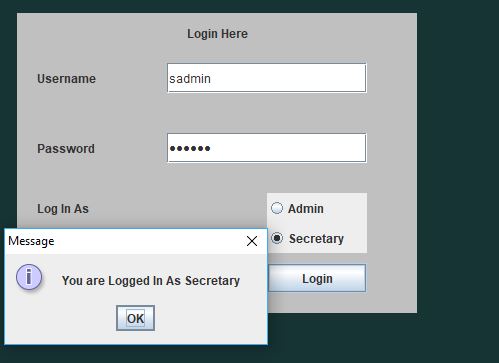


Figure 20 Login Secretary

If the user is secretary and he/she enters the username and password and login as secretary and if those username and password are valid then system displays message “You are logged in as secretary”.

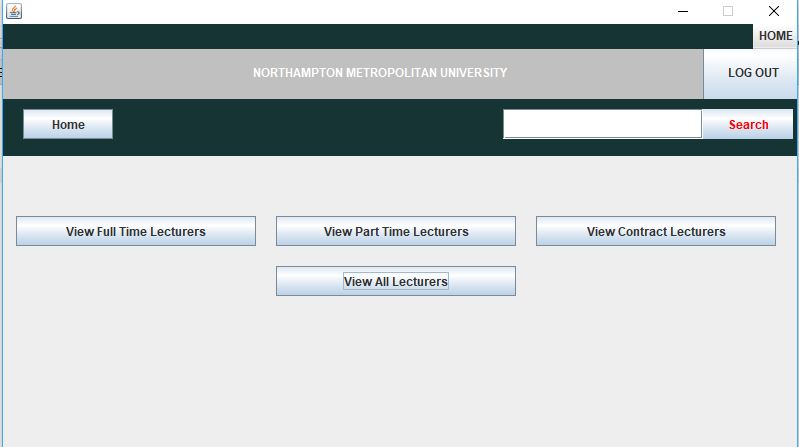
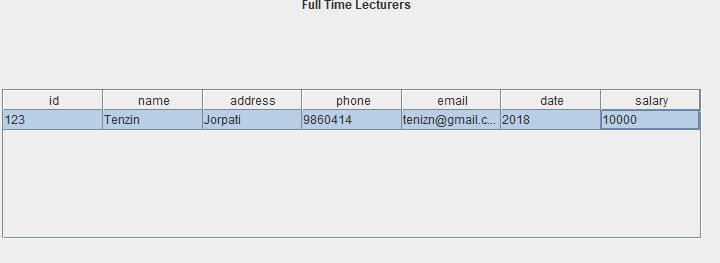


Figure 21 Secretary homepage

Once the secretary is successfully logged in, he/she is redirected to secretary homepage (figure 21). From here secretary can view lecturers of different type. If secretary clicks on view full time lecturers, details of all the full-time lecturers are shown and similarly if secretary clicks on view part time and contract lecturer, the details of each type of lecturers are shown. And by clicking on View All Lecturer, all the details of all the lecturers are shown in one page. (see figure 23)

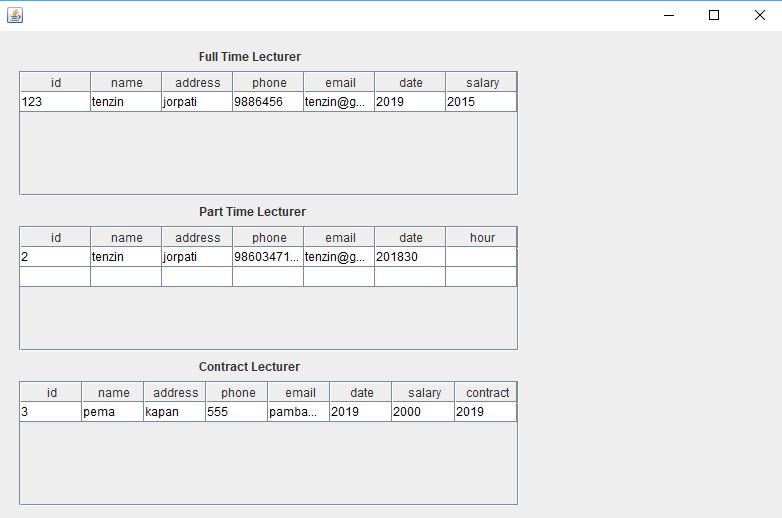
Figure 22 Full Time Lecturers

Figure 23 All Lecturers

# **Evidence of Testing**

Testing is one of the important features for developing a software. A testing is carried out in order to check whether the implemented function and component are working or not as per the requirement. Testing is done to know the possible bugs occurred or to find any error in the system. There is different approach to carried out the testing. In this software, the developer has specifically followed two method:

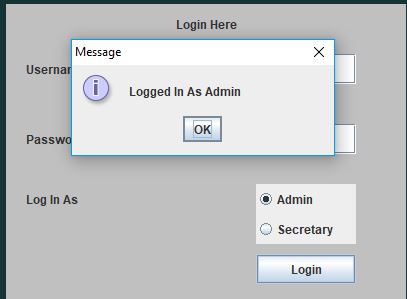
#### **Black Box Testing**

Black box testing is defined as a testing technique in which functionality of the Application Under Test (AUT) is tested without looking at the internal code structure, implementation details and knowledge of internal paths of the software. This type of testing is based entirely on software requirements and specifications. (Guru99, 2019)

* ***Log in as Admin***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.N. | Operations | Predicted Output | Actual Output | Results |
| 1 | Log in with correct username and password | Admin logged into the system successfully | Admin logged into the system successfully | Successful |
| 2 | Log in with incorrect username or password | Display error message saying incorrect username or password | Display error message saying incorrect username or password | Successful |

This above testing can be seen on below screenshot



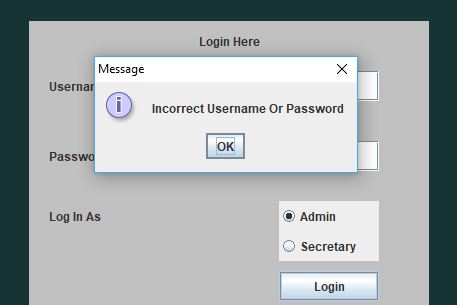
Figure 24 Logged in with correct username and password

Figure 25 Login with incorrect username or password

* **Log in as Secretary**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N.** | **Operations** | **Predicted Output** | **Actual Output** | **Results** |
| 1 | Log in with correct username and password | Secretary logged into the system successfully | Secretary logged into the system successfully | Successful |
| 2 | Log in with incorrect username or password | Display error message saying incorrect username or password | Display error message saying incorrect username or password | Successful |

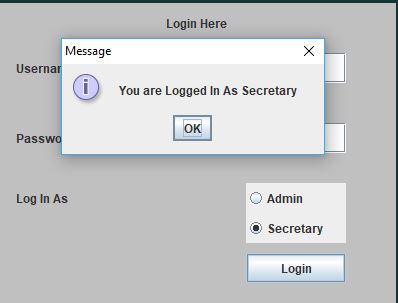


Figure 26 Login with correct username and password (Secretary)

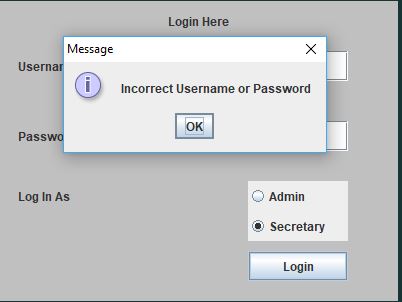
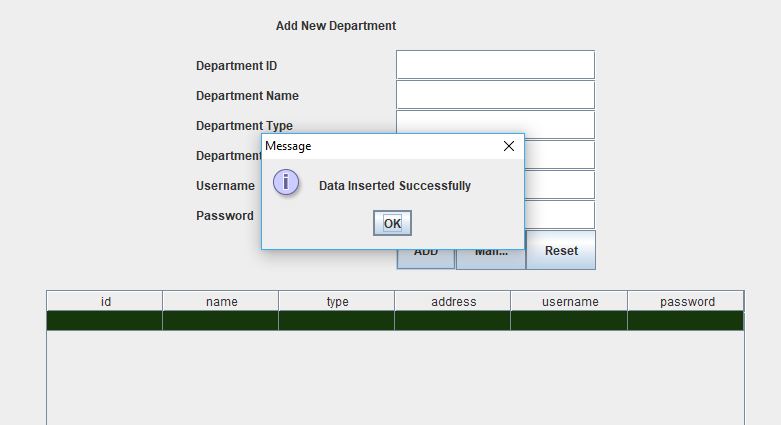


Figure 27 login with incorrect username or password (Secretary)

* ***Adding Department***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N.** | **Operations** | **Predicted Output** | **Actual Output** | **Results** |
| 1 | Admin clicks on add department button | Add department form is displayed | Add department form is displayed | Successful |
| 2 | Inserting valid data and clicking add button | Displays Department added successful message | Displays message “Data Inserted Successfully” | Successful |
| 3 | Inserting same ID again and again and clicking add button | Department is not supposed to be added | Displays message “Data Inserted Successfully” | Unsuccessful |
| 4 | Inserting no data and clicking add button | Department is not supposed to be added | Displays message “Data Inserted Successfully” | Unsuccessful |
| 5 | Inserting string value in ID and clicking add button | Lecturer is not supposed to be added and displays error message | Displays message “Data Inserted Successfully” and added department | Unsuccessful |



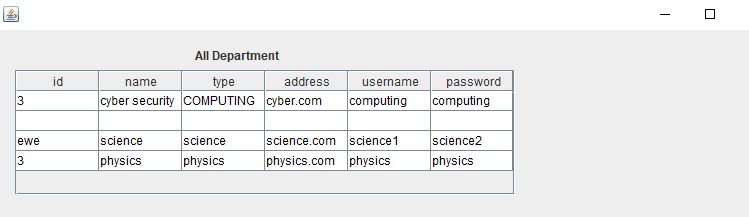
Figure 28 Empty data inserted successfully

Figure 29 Same ID and empty data inserted successfully

* **Adding lecturer**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N.** | **Operations** | **Predicted Output** | **Actual Output** | **Results** |
| 1 | Admin clicks on add lecturer button | Add lecturer form is displayed | Add lecturer form is displayed | Successful |
| 2 | Inserting valid data and selecting one of the lecturer options and clicking add button | Displays Lecturer added successful message | Displays message “Data Inserted Successfully” | Successful |
| 3 | Inserting same ID again and again and clicking add button | Lecturer is not supposed to be added | Displays message “Data Inserted Successfully” | Unsuccessful |
| 4 | Inserting no data and clicking add button | Lecturer is not supposed to be added and displays error message | Displays message “Data Inserted Successfully” | Unsuccessful |
| 5 | Inserting string data in contact and clicking add button | Lecturer is not supposed to be added and displays error message | Displays message “Data Inserted Successfully” | Unsuccessful |
| 6 | Inserting string value in ID and clicking add button | Lecturer is not supposed to be added and displays error message | Displays message “Data Inserted Successfully” | Unsuccessful |
| 7 | Inserting string value in Date and clicking add button | Lecturer is not supposed to be added and displays error message | Displays message “Data Inserted Successfully” | Unsuccessful |

* **Editing Department**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N.** | **Operations** | **Predicted Output** | **Actual Output** | **Results** |
| 1 | Admin clicks on edit button after clicking on view button in add department page | Edit department form is displayed | Edit department form is displayed | Successful |
| 2 | Inserting valid data and clicking edit button | Displays Department added successful message | Displays message “Edited Successfully” | Successful |
| 3 | Inserting empty data and clicking edit button | Displays error message and data should not be added | Displays message “Edited Successfully” | Unsuccessful |

* **Editing Lecturers**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N.** | **Operations** | **Predicted Output** | **Actual Output** | **Results** |
| 1 | Admin clicks on edit button after clicking on manage button in add lecturer page | Edit department form is displayed | Edit department form is displayed | Successful |
| 2 | Inserting valid ID and clicking edit button | Displays message Lecturer added successful | Displays message “Edited Successfully” | Successful |
| 3 | Inserting empty data and clicking edit button | Displays error message and data should not be added | Displays message “Edited Successfully” | Unsuccessful |
| 4 | Inserting invalid ID and clicking edit button | Lecturer should not be added | Displays error in console and data not edited | Successful |

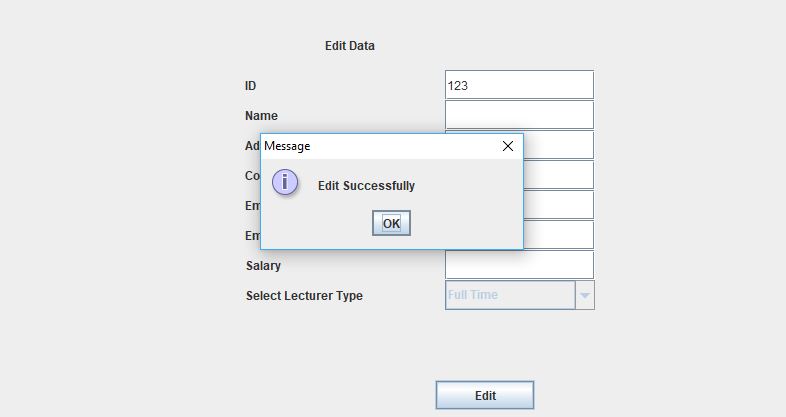


Figure 30 Empty data Edited Successfully

* **View Department**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N.** | **Operations** | **Predicted Output** | **Actual Output** | **Results** |
| 1 | Clicking view all button in manage department page | All the department details that are in the system should be viewed by admin | Obtained as predicted output | Successful |

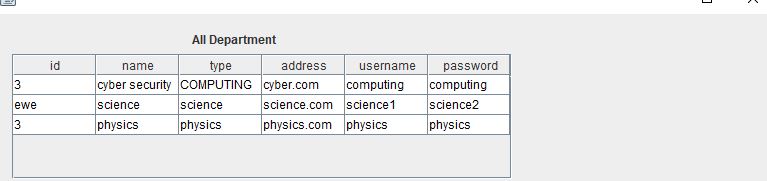
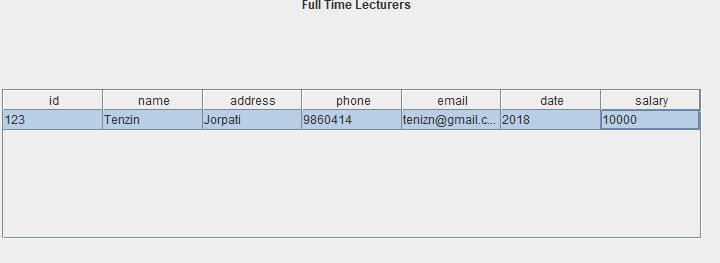
****

Figure 31 View All department details

* **View Lecturers**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N.** | **Operations** | **Predicted Output** | **Actual Output** | **Results** |
| 1 | Clicking view button in add lecturer page | Redirected to view lecturer page | Obtained as predicted output | Successful |
| 2 | Clicking view all button in view lecturer page | All the lecturer details that are in the system should be viewed by admin | Obtained as predicted output | Successful |
| 3 | Clicking view full time lecturer button | All the full-time lecturers details that are in the system are displayed | Obtained as predicted output | Successful |
| 4 | Clicking view part time lecturer button | All the part time lecturers’ details that are in the system are displayed | Obtained as predicted output | Successful |
| 5 | Clicking view contract lecturer button | All the contract lecturers’ details that are in the system are displayed | Obtained as predicted output | Successful |

****

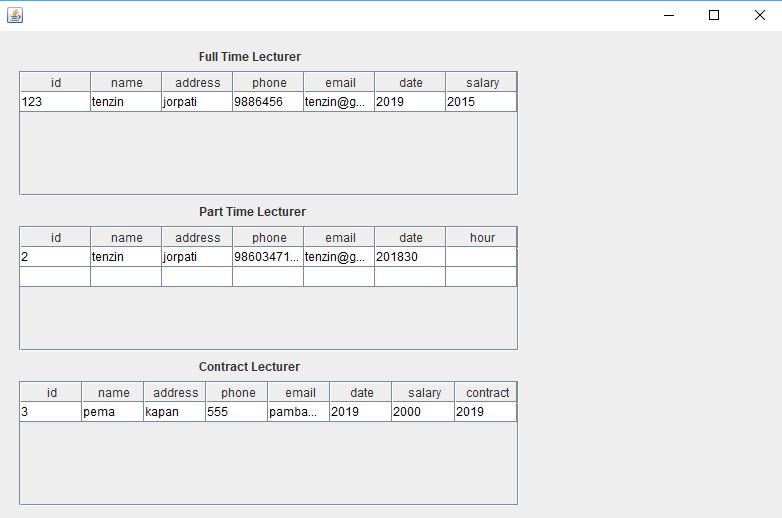
Figure 32 View Full time lecturers

Figure 33 View All Lecturers

* **Deleting Department**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.N. | Operation | Predicted Output | Actual Output | Results |
| 1. | Clicking delete button in manage department page | Redirect the Admin to deleting department page | Obtained as predicted output | Successful |
| 2. | Inserting the Department ID to delete that department | Department is deleted from the system | Obtained as predicted output | Successful |
| 3 | Inserting the incorrect Department ID to delete that department | Message should be displayed “invalid ID” | Throws error in console | Unsuccessful |

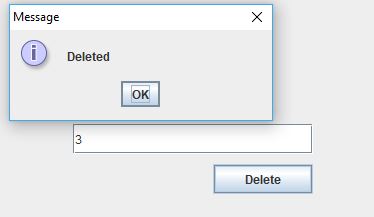
****

Figure 34 Deleting department with ID 3

* **Deleting Lecturer**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.N. | Operation | Predicted Output | Actual Output | Results |
| 1. | Clicking delete button in manage lecturer page | Redirect the Admin to manage lecturer page | Obtained as predicted output | Successful |
| 2. | Inserting the Lecturer ID to delete that lecture | Lecturer is to be deleted from the system | Throws input mismatch error in console | Unsuccessful |
| 3 | Inserting the incorrect Lecture ID to delete that lecturer | Message should be displayed “invalid ID” | Throws error in console | Unsuccessful |

#### **White Box Testing**

Another method for testing is white box testing. White box testing is the testing where the tester knows the structure of coding implementation that are implemented in the system.

Below is the white box texting for different functionality implemented in the system:

* **Login by Admin and Secretary**

**Admin**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N** | | **Path** | | **Test Case** |
| 1. | | Decision point 1 is true, Decision point 2 is false, Decision point 3 is false, Decision point 4 is false, Decision point 5 is false | | Admin is selected |
| 2. | | Decision point 1 is false, Decision point 2 is true, Decision point 3 is false, Decision point 4 is false, Decision point 5 is false | | Try statements block code executed |
| 3. | | Decision point 1 is false, Decision point 2 is false, Decision point 3 is true, Decision point 4 is false, Decision point 5 is false | | If the file is in system |
| 4. | | Decision point 1 is false, Decision point 2 is false, Decision point 3 is false, Decision point 4 is true, Decision point 5 is false | | If the File found has any data |
| 5. | Decision point 1 is false, Decision point 2 is false, Decision point 3 is false, Decision point 4 is false, Decision point 5 is true | | Admin is logged in and displays the message | |

**Department Secretary**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N** | | **Path** | | **Test Case** |
| 1. | | Decision point 1 is true, Decision point 2 is false, Decision point 3 is false, Decision point 4 is false, Decision point 5 is false | | Secretary is selected |
| 2. | | Decision point 1 is false, Decision point 2 is true, Decision point 3 is false, Decision point 4 is false, Decision point 5 is false | | Try statements block code executed |
| 3. | | Decision point 1 is false, Decision point 2 is false, Decision point 3 is true, Decision point 4 is false, Decision point 5 is false | | If the file is in system |
| 4. | | Decision point 1 is false, Decision point 2 is false, Decision point 3 is false, Decision point 4 is true, Decision point 5 is false | | If the File found has any data |
| 5. | Decision point 1 is false, Decision point 2 is false, Decision point 3 is false, Decision point 4 is false, Decision point 5 is true | | Admin is logged in and displays the message | |

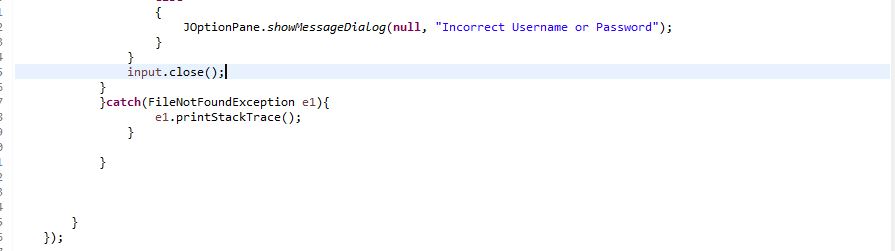
****

Figure 35 login to the system code implementation

Thus, the methods include one decision point and cyclomatic calculation is calculated below.

Cyclomatic Calculation of login of both admin and department secretary = 1+ Σ decision points

= 1+10=11 which is complex.

* **Add Department**

|  |  |  |
| --- | --- | --- |
| **S.N** | **Path** | **Test Case** |
| 1 | Decision point 1 is true | Try block code executed |
| 2 | Decision point 1 is false | Try block Exception is printed |

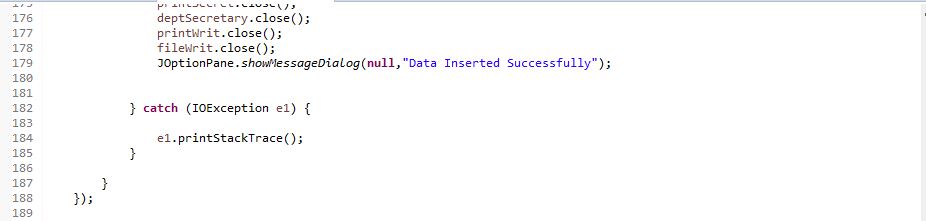
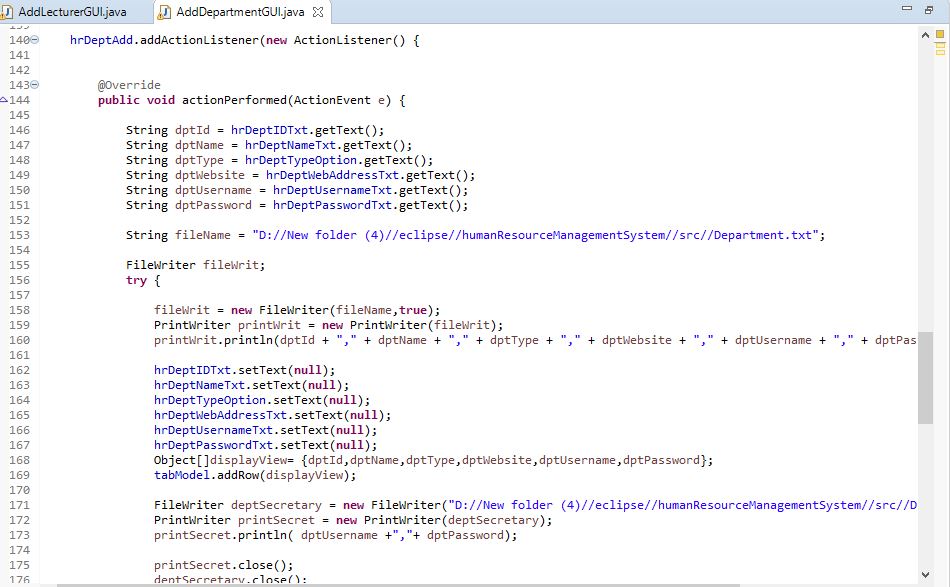
****

Figure 36 Adding new department code implementation

Thus, the methods include one decision point and cyclomatic calculation is calculated below.

Cyclomatic Calculation of adding new department = 1+ Σ decision points

= 1+2=3 which states that this method is not complex.

* **Edit department**

|  |  |  |
| --- | --- | --- |
| **S.N** | **Path** | **Test Case** |
| 1 | Decision point 1 is true | Try block code executed |
| 2 | Decision point 1 is false, Decision point 2 is true | If the file has any data in it |
| 3. | Decision point 1 is false, Decision point 2 is false, Decision point 3 is true | If id is equals to token[0] and is true |

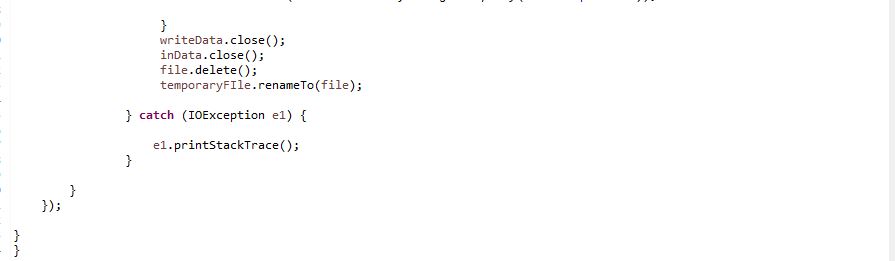
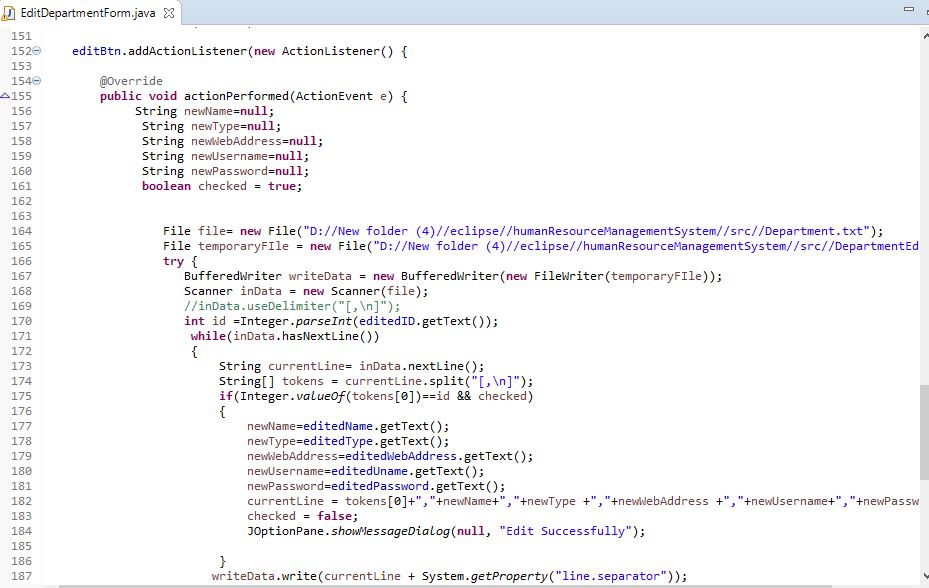
****

Figure 37 Edit Department code implementation

Thus, the methods include one decision point and cyclomatic calculation is calculated below.

Cyclomatic Calculation of adding new department = 1+ Σ decision points

= 1+3=4 which states that this method is not complex.

* **View Department**

|  |  |  |
| --- | --- | --- |
| **S.N** | **Path** | **Test Case** |
| 1 | Decision point 1 is true | Try block code executed |
| 2 | Decision point 1 is false | File not found exception displayed |

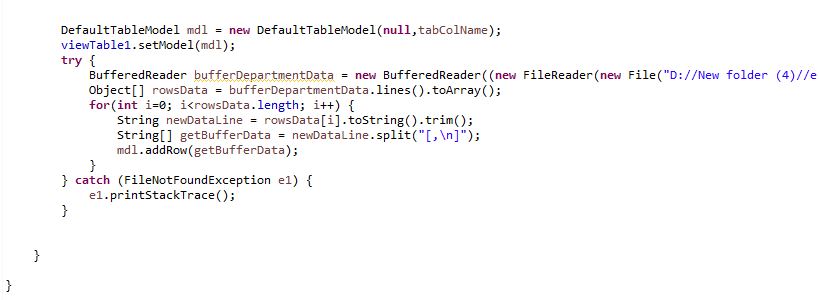


Figure 38 Viewing department code implementation

Thus, the methods include one decision point and cyclomatic calculation is calculated below.

Cyclomatic Calculation of adding new department = 1+ Σ decision points

= 1+2=3 which states that this method is not complex.

* **Deleting Department**

|  |  |  |
| --- | --- | --- |
| **S.N** | **Path** | **Test Case** |
| 1 | point | Try block code executed |
| 2 | Decision point 1 is false, Decision point 2 is true | If the file has any data |
| 3 | Decision point 1 is false, Decision point 2 is false, Decision point 3 is true | If Id from file is not equal to id provided by admin, write their details |

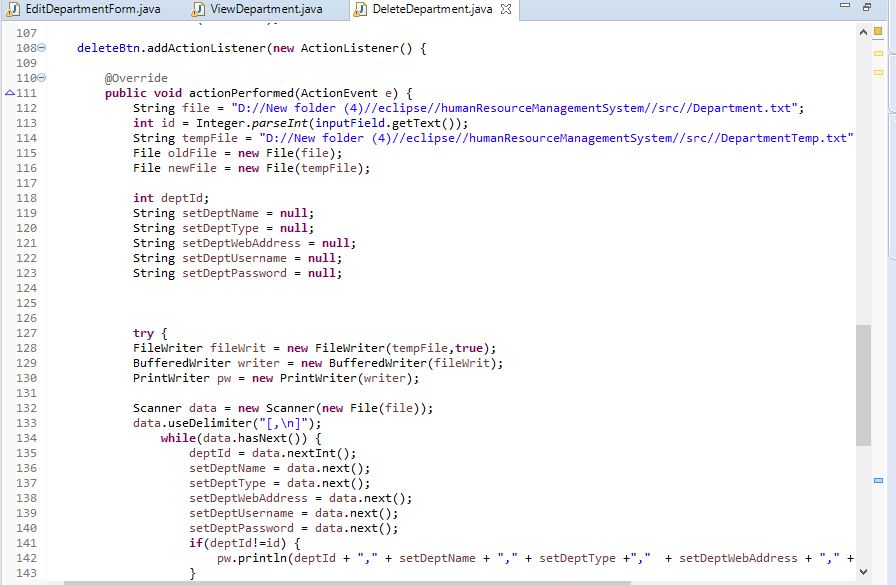
****

Figure 39 Code implementation for deleting a department

Thus, the methods include one decision point and cyclomatic calculation is calculated below.

Cyclomatic Calculation of adding new department = 1+ Σ decision points

= 1+3=4 which states that this method is not complex.

#### **Weakness or bugs found in the system**

#### Though the developer has successfully developed the software for human resource management system, there were several error and bugs that the developer has caught.

Below are some of list of weakness and bugs found in the current system:

* Since the developer used the .txt format to store and retrieved data, if the admin saved the empty data, then system throw input mismatch exception in the console.
* Because of storing data in .txt format, while logging in, the system checks all the data that are in the respective .txt file and for example, if there are 3 person data and if the person first is logging in to the system, the system checks all three data and display incorrect message for other two unmatched data.
* The other error found is that the Admin can enter text in place of number and number in place of text and Admin can also add and edit empty form and will be added.
* Similarly, while deleting the lecturers, when Admin clicks on delete button, it throws an input mismatch exception in console.
* The developer has not followed the Model View Controller (MVC) approach to build the system. Due to this, there is lot of redundancy of code.

# Conclusion and Recommendation

Lastly, the developer has successfully built the functionable system implementing most of the required features. The developer has built the system based on class diagram and use case diagram. Similarly, the developer has defined the system description defining how the system works and how to use the system by user. Furthermore, the developer has performed different types of testing to check if the function implemented works or not and how much the system complexity is. After testing the system, the developer has mentioned some of the weakness and bugs that is found in the current system.

However, if the developer is given the extra time, at first the developer would have implemented Model View Controller approach to develop the system so that there would be less redundancy of code. Also, developer would have overcome all the weaknesses that developer mentioned in weakness section of this report and would have added additional feature.

# Bibliography

*Guru99*. (2019, April 20). Retrieved from Guru99: https://www.guru99.com/black-box-testing.html

Stack Overflow. 2019. Editing existing txt file in java - Stack Overflow. [ONLINE] Available at: https://stackoverflow.com/questions/37263631/editing-existing-txt-file-in-java. [Accessed 24 April 2019].

Kode Java. 2019. How do I create and write data into text file? | Kode Java. [ONLINE] Available at: https://kodejava.org/how-do-i-create-and-write-data-into-text-file/. [Accessed 05 May 2019].