Duran Moodley 13016335

Assignment 3

Portfolio of Evidence

Contents

[Problem Statement 1](#_Toc422924273)

[Purpose 1](#_Toc422924274)

[Functionality 2](#_Toc422924275)

[Programming structures that I have used 2](#_Toc422924276)

[Diagrams 3](#_Toc422924277)

[Class Diagrams 4](#_Toc422924278)

[Flowcharts 14](#_Toc422924279)

[How the Application Runs 22](#_Toc422924280)

[Screen shots 23](#_Toc422924281)

[Error Messages 31](#_Toc422924282)

[Lecturer Feed Back 32](#_Toc422924283)

[Corrections 32](#_Toc422924284)

[Enhancements 32](#_Toc422924285)

[References 33](#_Toc422924286)

# Problem Statement

## Purpose

A zoo is having trouble keeping track of all their animals and the feeding schedules of each animal as well as who feeds which animal. The zookeeper tends to get confused about which animals they should be looking after and at what times. The application needs to store all the different types of animals such has Lions, Elephants and Birds. Furthermore, information such as Name, unique ID, Gender, age, feeding schedules and zoo keeper in charge of that animal. Furthermore, the application is also required to store specific information for each animal. A lion, for example, will have its claw length stored and other information. Lastly, at the end of each week the application must generate schedule that randomly assigns each zoo keeper to an animal for that week. If an animal get moved to another zoo or dies. The animal can be deleted from the system.

## Functionality

* The application adds a Lion, Elephant and bird into the system.
* It deletes one of the above animal from the system.
* There are two type of reports that are shown : Animal List( Lion, Elephant and Bird), Schedule report
* Every week the system randomly assigns an animal to a zoo keeper.
* Randomly generates a unique ID (6 digits and a character value)
* Sends all records to a text file.
* Opens a notepad file to view deleted records.

## Programming structures that I have used

This application will be using a wide variety of text files to send and receive data. They are:

* Birds – Stores data related to birds
* Lion – stores data related to lions
* Elephant – stores data related to elephants
* EmployeeList – stores all employees
* RandomEmployeeSchedule – stores the weekly randomly generated employee schedule
* DateModified – stores the last date of when the RandomEmployeeSchedule was updated
  + If more the 7 days have passed, update the RandomEmployeeSchedule
  + If the RandomEmployeeSchedule is updated, store the date of when the file was updated in the DateModified text file.

2 dimensional Arrays will play a major part in the program. It will constantly be used to fetch data for certain processes and to display the text files.

**2 Dimensional Array**

**Array List**

**String Handling**

**Using Classes (objects)**

**File handling**

**Exception Handling**

**Threads**

**Random Generator**

**Interface Class**

**Sealed Classes**

**Abstract Classes**

**Dynamic & Var types**

**Polymorphism**

# Diagrams

## Class Diagrams





















## Flowcharts



















# How the Application Runs

**For Testing Purposes: username : Administrator and Password 5**

First the user is greeted with a Splash screen. Then the user is taken to a Login Form where a valid user name and password needs to be entered. Thereafter, the user is taken to the menu which contains the following options: Add Animal, Delete Animal and View Report.

If the user selects the Add Animal option. He/she will need to select the type of animal that is being added into the system (Lion, Elephant, Bird).

Thereafter, the user needs to enter in generic/basic information about that animal. Things like: name of the animal, gender, age, weight and the feeding times.

Depending on the animal that is selected. Specific information needs to be added. This will be different on every type of animal. For instance, specific information for a lion will be its claw length, does it have mane, its skin colour or skin type. However for an elephant, specific information will be: its tusk length, tusk weight, tusk shape, ear length and trunk length.

Once all relevant information has been entered in correctly. The user can save the record. Once the user clicks save, the computer will randomly generate a unique ID for that type of animal. The ID will contain 6 digits and a character. The character will depend on which type of animal is being added. If a Lion is being added, the character “L” will appear at the end of the ID. This will then get sent to a text file.

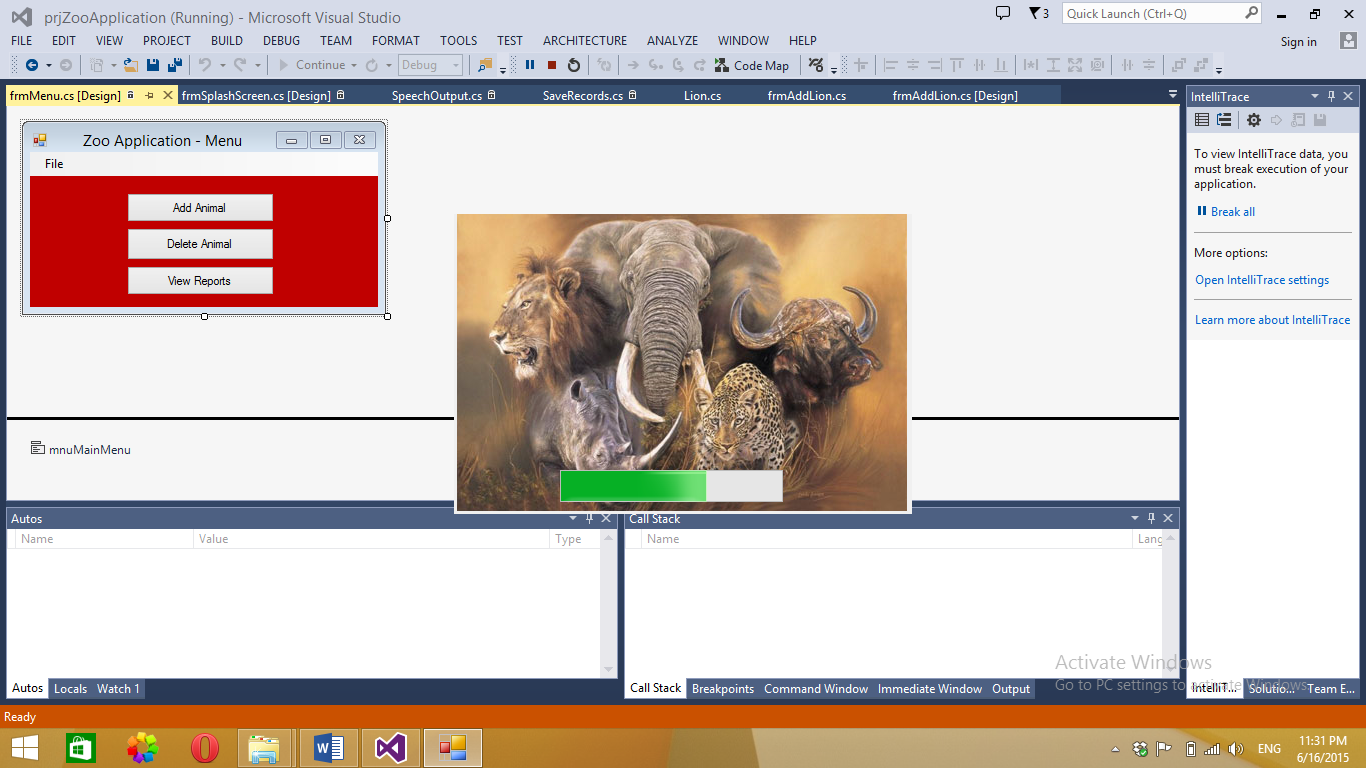
In the unfortunate event that the animal dies or leaves the zoo. He/she can be deleted from the system by simply selecting the Delete option. Thereafter, the user needs to select the unique ID of the animal being deleted from a combo box and state the reason for its deletion. All deleted records will be sent to a backup file, in case an error is made. The user can also view all deleted records by clicking on the View Delete records List. This will open a notepad file containing the dates and the unique ID of the animal.

If a new zoo keeper is employed into the zoo. He/she can be added into the system by selecting the Add Zoo keeper. The user will need to enter in the name of the new employee. Once the Add Zoo keeper button is clicked, the computer will randomly generate a password consisting of the ‘’Z” character and a random 3 digit number.

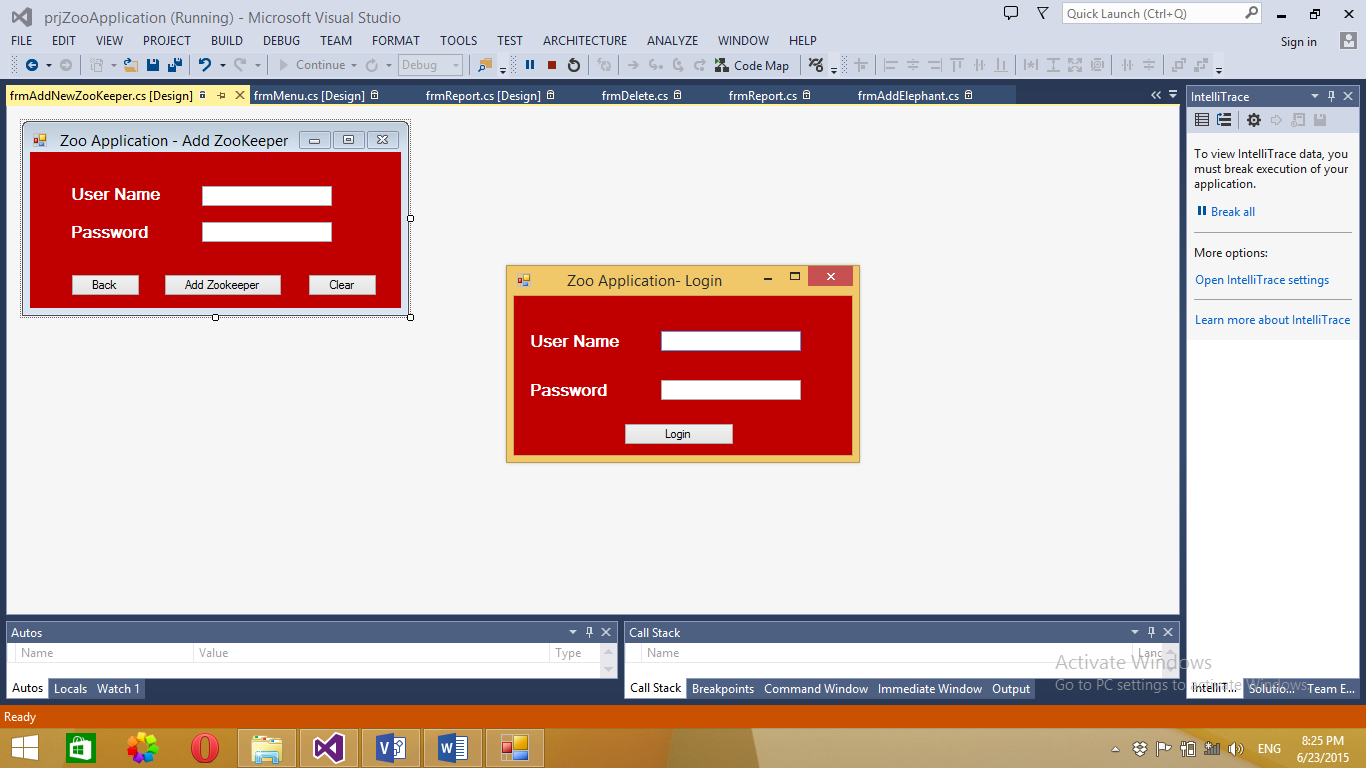
Lastly, by clicking on the View Reports option. The user can view all the different type of animals and the zoo Keeper schedules for each week. After a week, a new randomly generated schedule will be generated and saved to a text file. This means, that each zoo keeper will be assigned to different animals after each week has passed.

# Screen shots

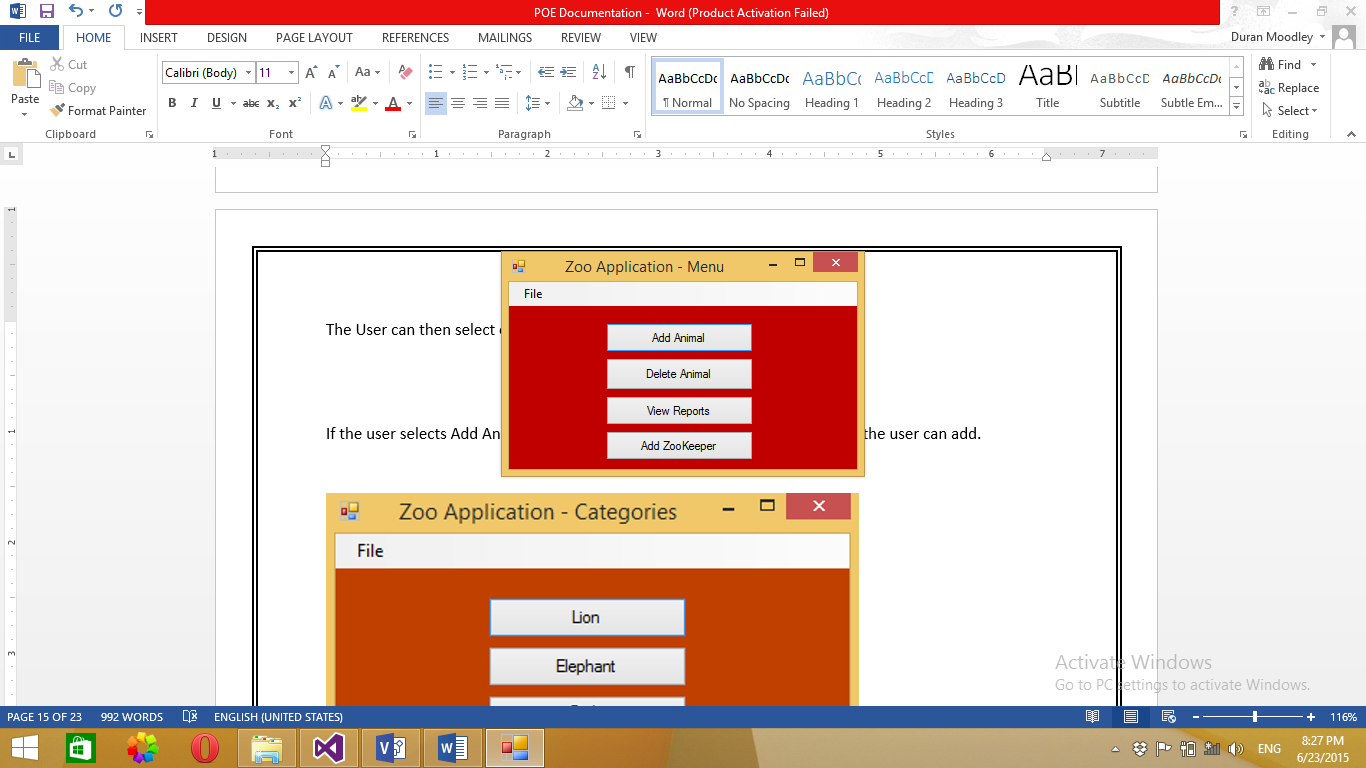
First the user is greeted with a **Splash screen**.



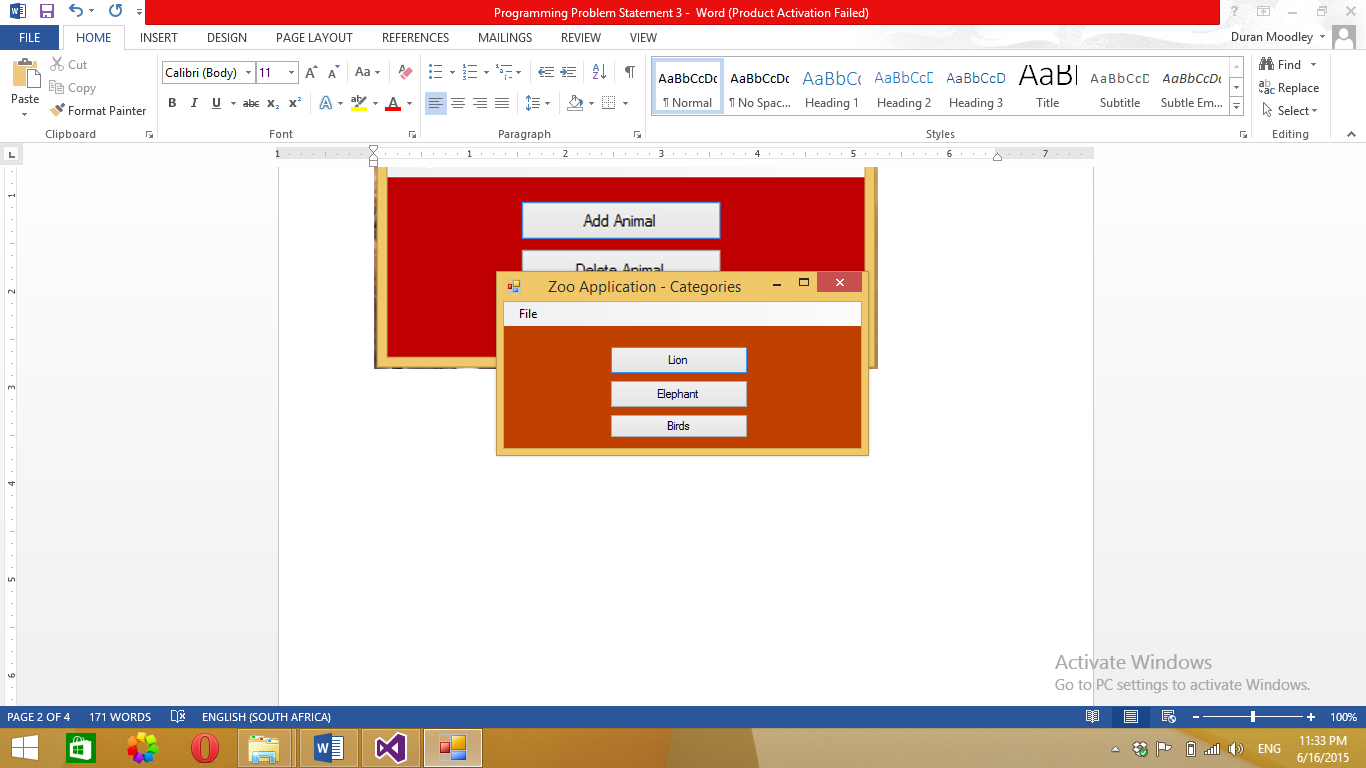
**Login Form**, will require the user to enter in a valid username and password.



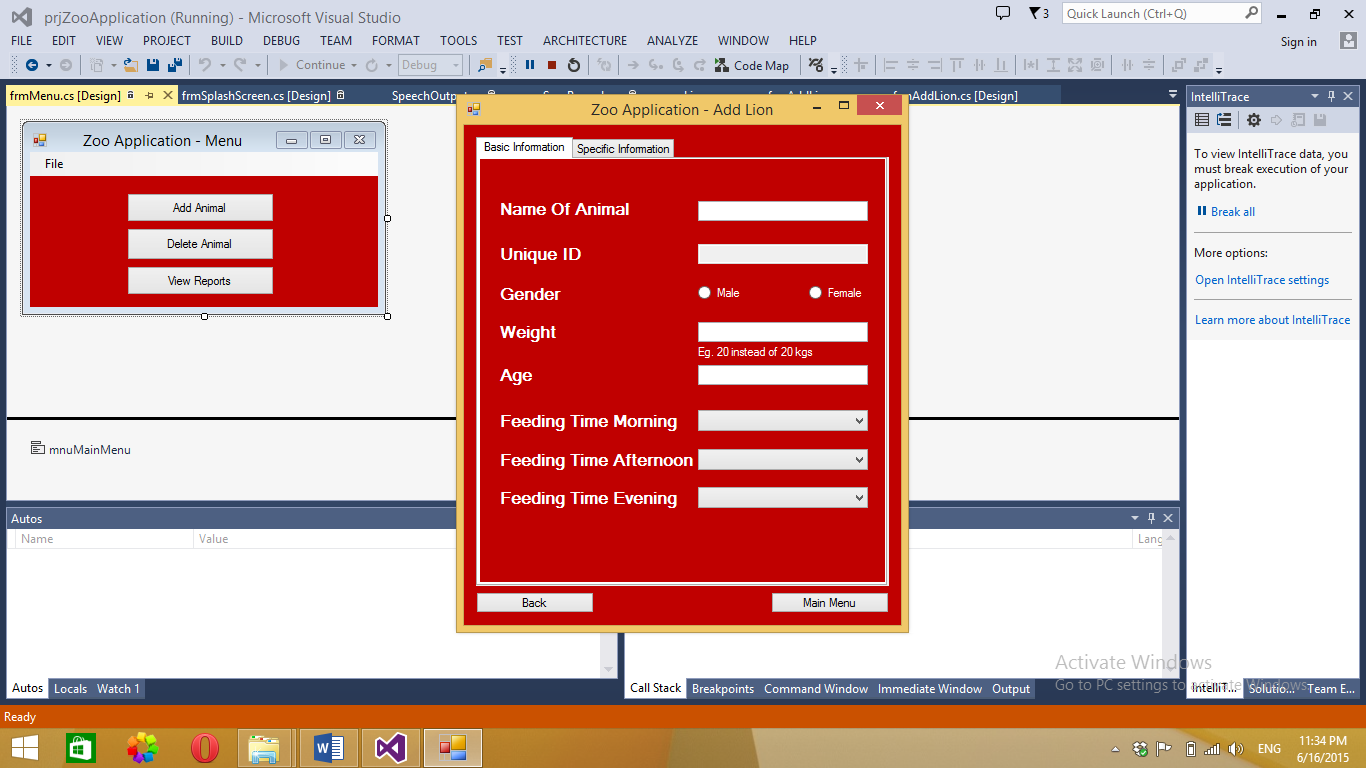
The User can then select options from the **Menu Form.**



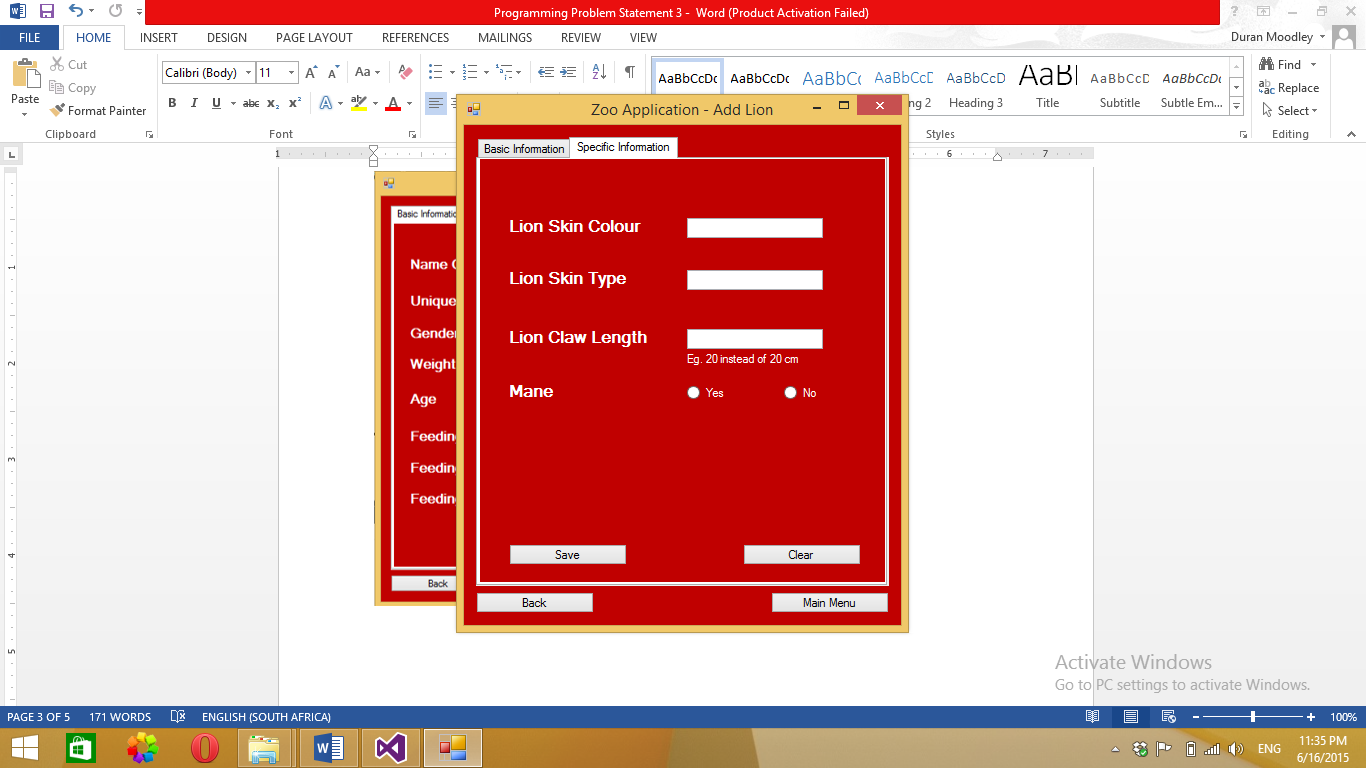
If the user selects Add Animal, the **Categories Form** will contain which animals the user can add.



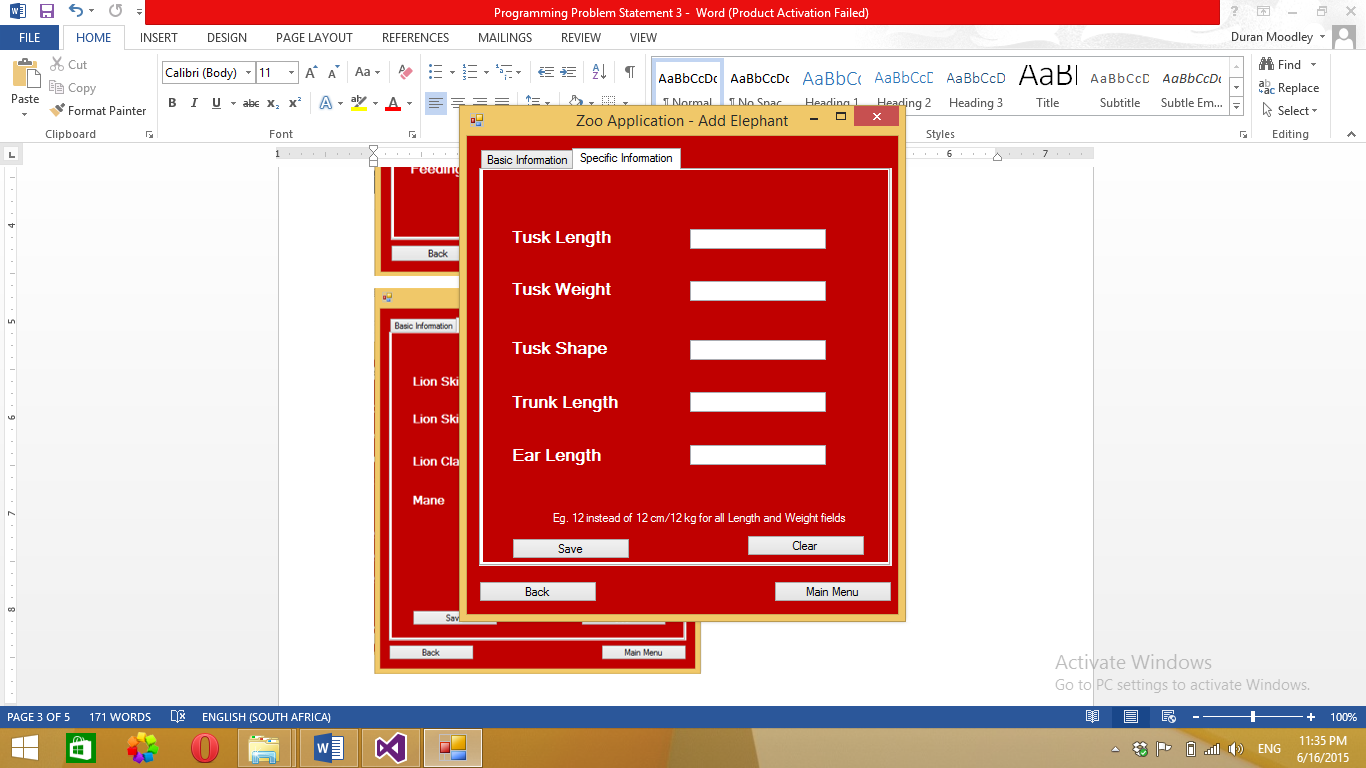
**Add Lion Form**. Selecting Lion will open a form that will require the user to enter in basic information about that Animal. This form will open with every Animal (Lion, Elephant, Bird) being added.



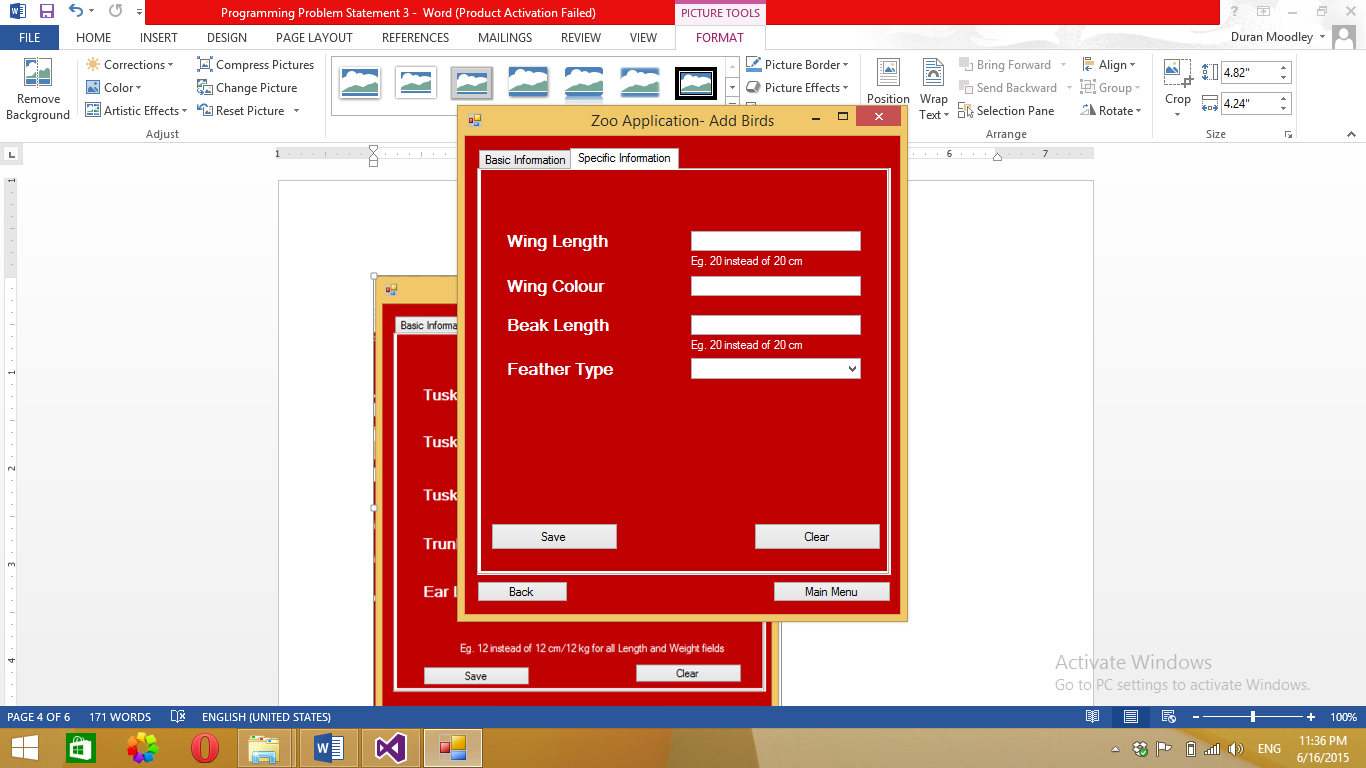
Once the Basic information is completed. The user will need to enter specific information relating to that type of animal.



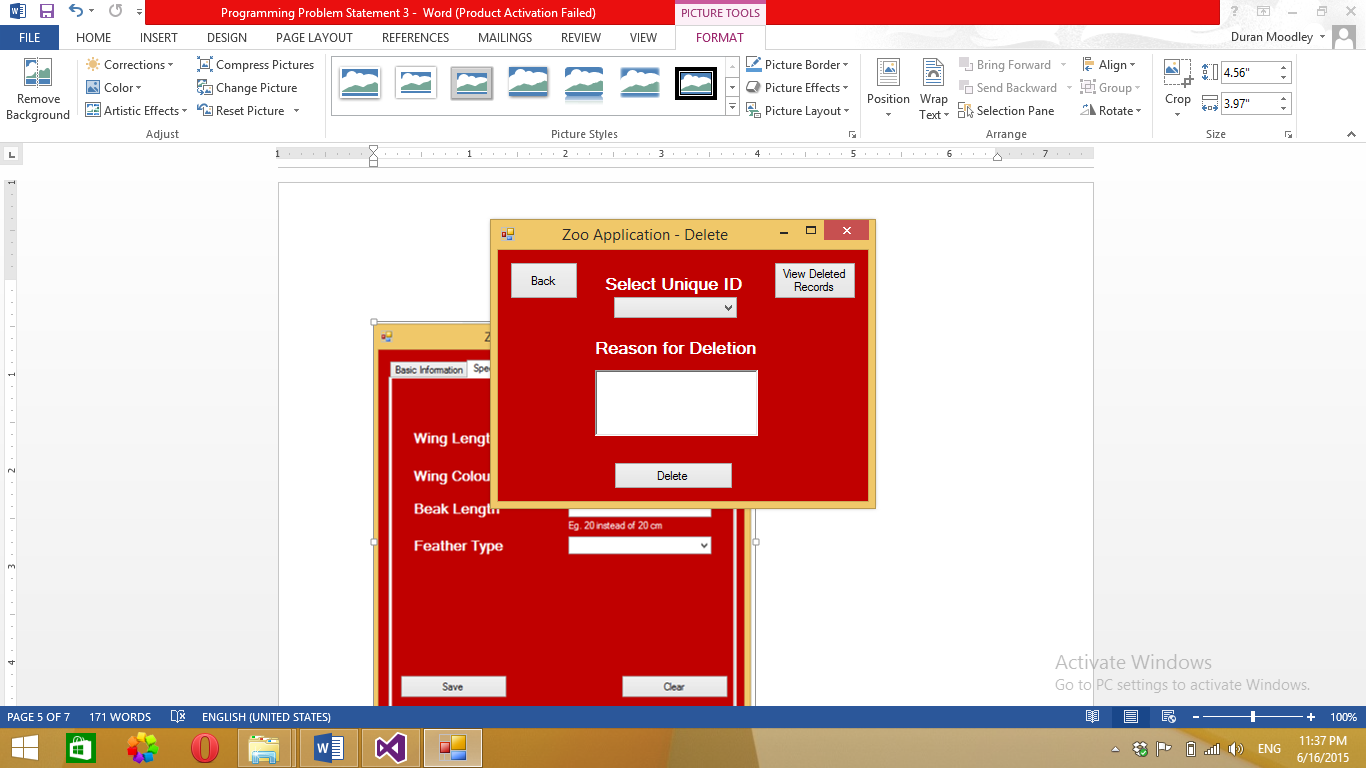
**Add Elephant.** Like mentioned before, once the Basic information is completed. The user will need to enter specific information relating to that type of animal.

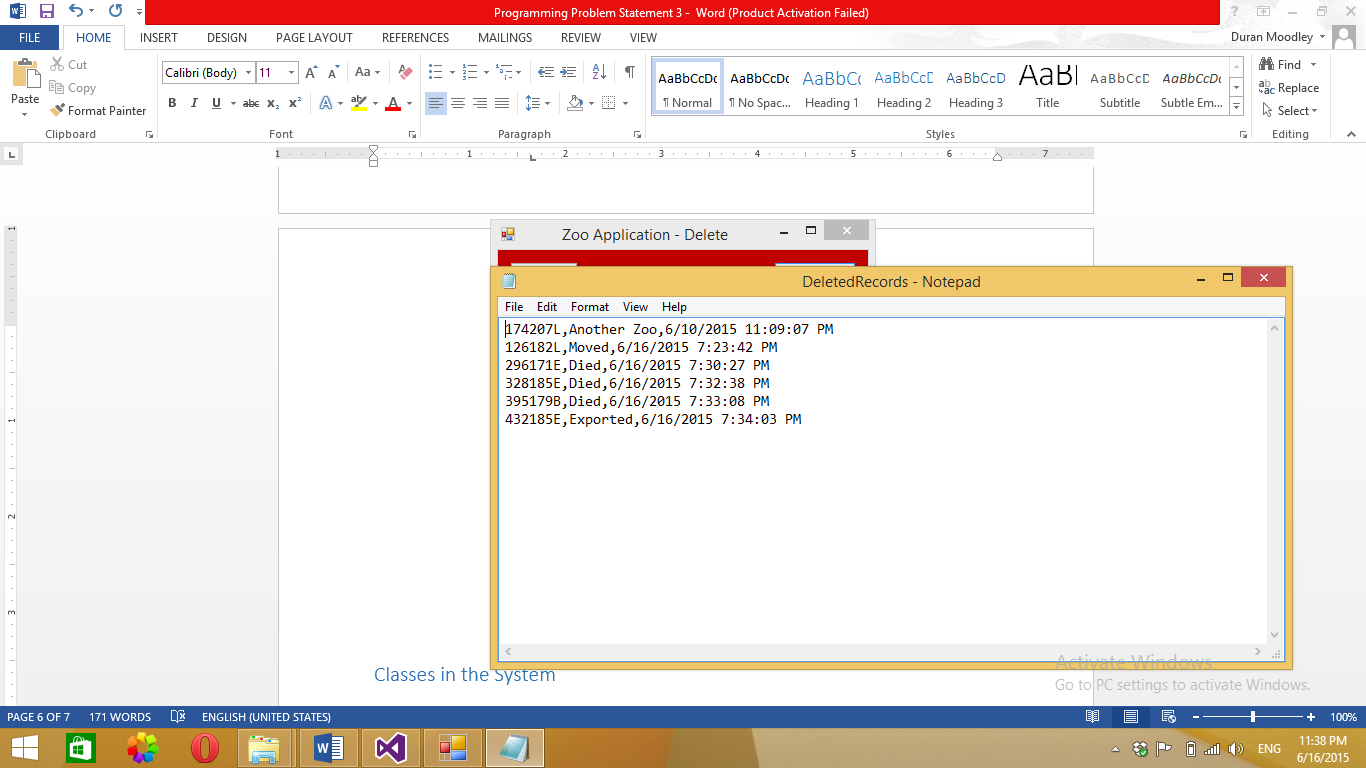


**Add Birds Form**, adds a new bird. Refer to previous explanations for more details.



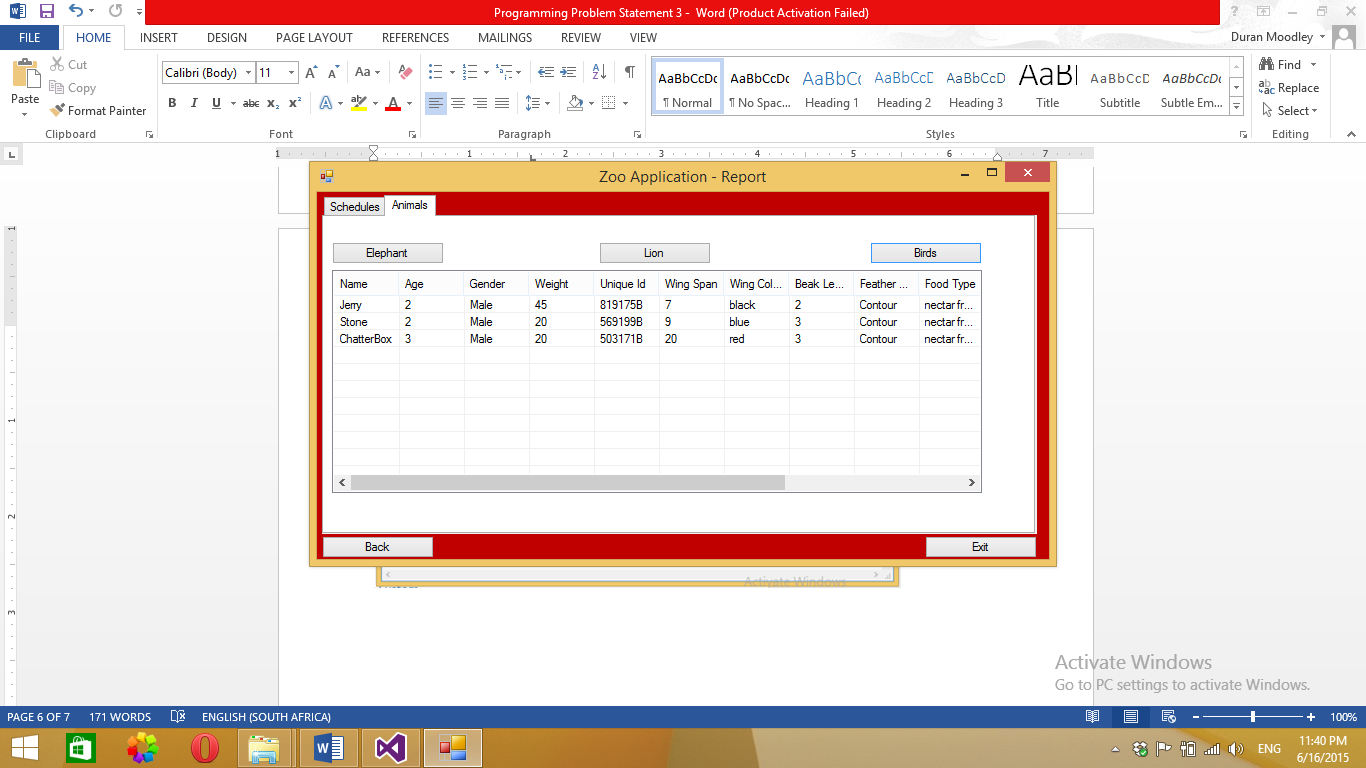
If selects the Delete option. The **Delete Form** will open requiring the user to select the unique ID of the animal and enter in the reason for deletion. The user can also view all deleted records. By clicking on the view deleted button.

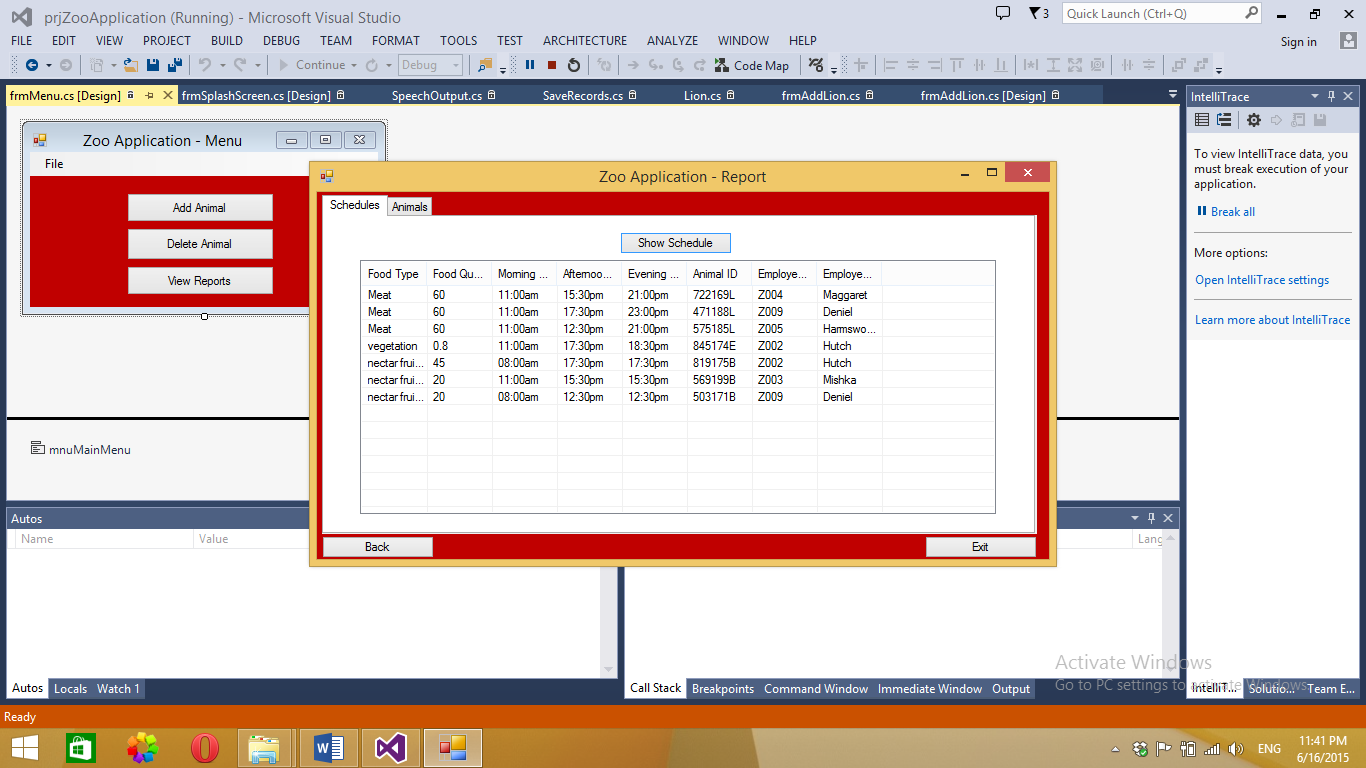




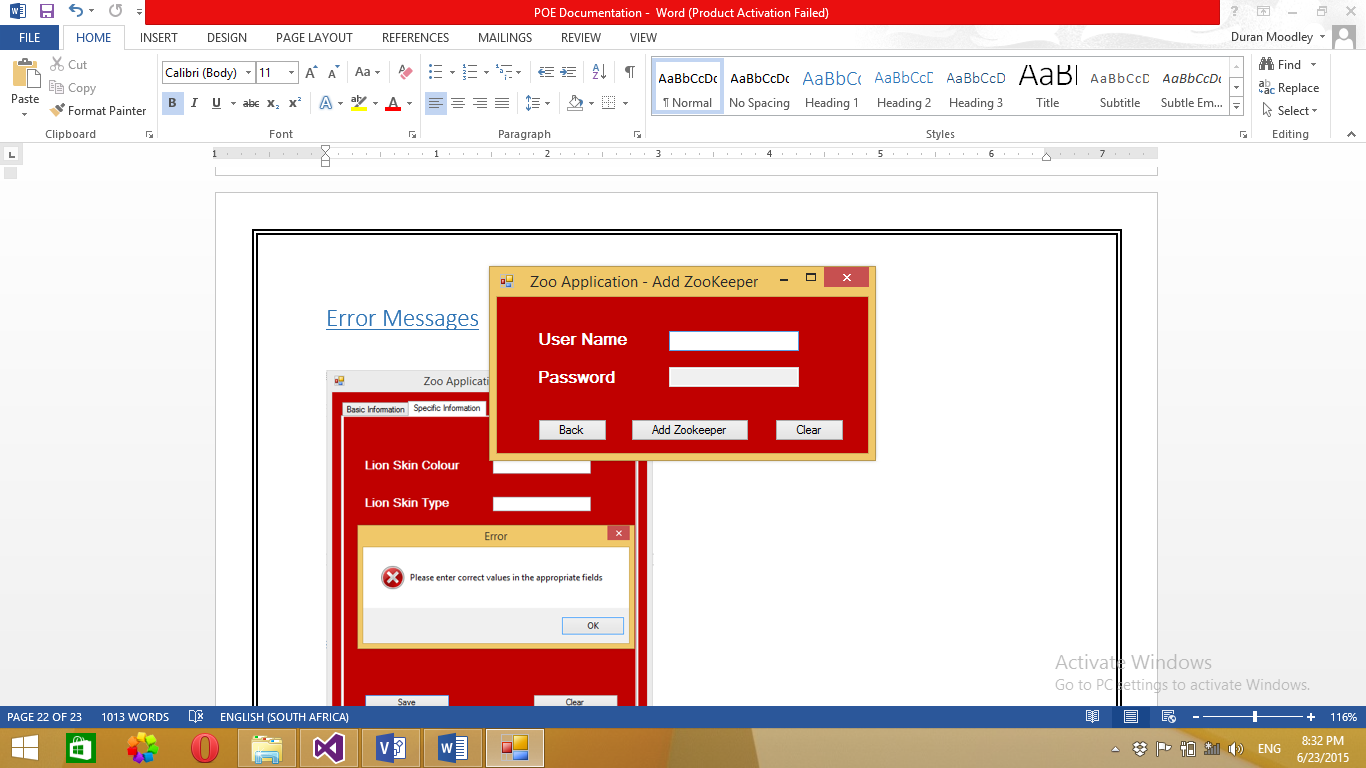
**Report Form**

The user can view 2 different types of reports. One with all the animal information and the other containing the schedules of each zoo keeper.

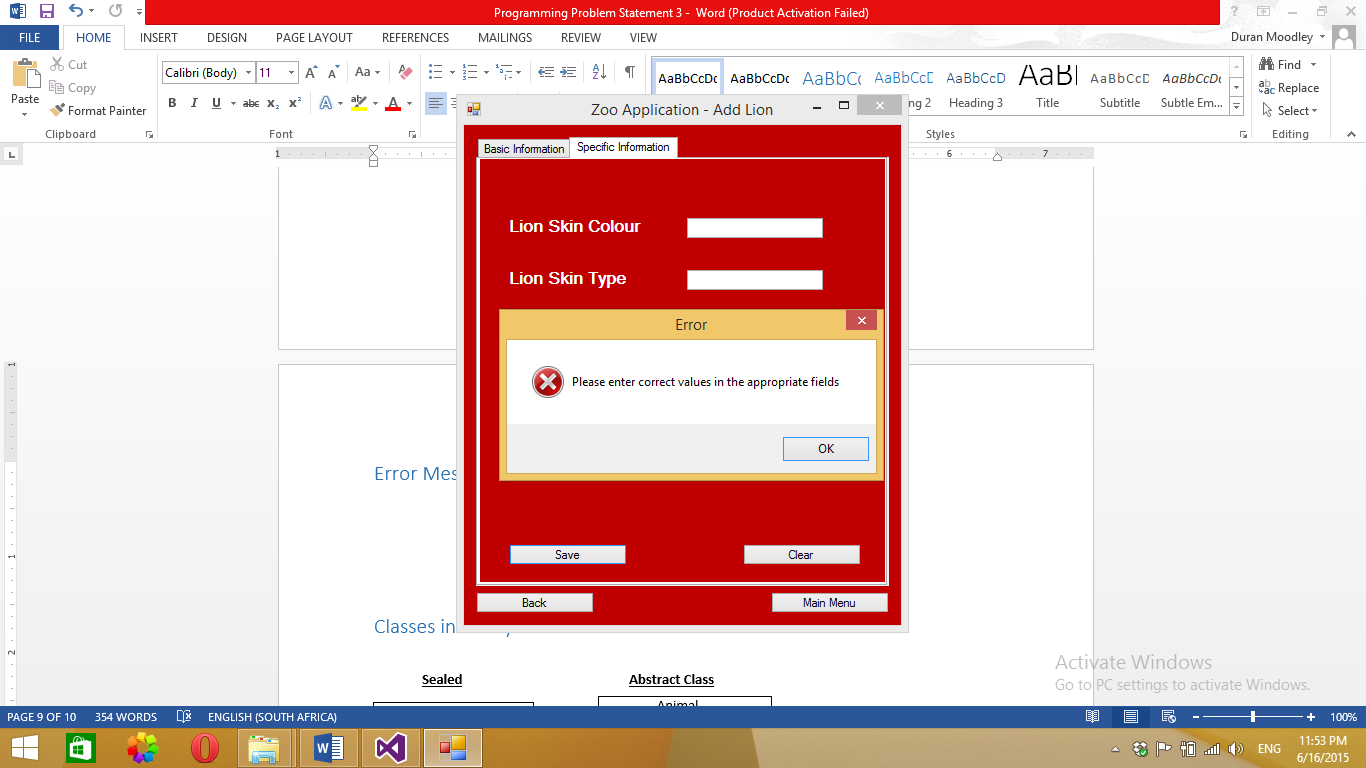




**Add Zookeeper**, adds a newly employed zoo keeper into the system.



# Error Messages



I have also incorporated Speech output when these errors occur.

# Lecturer Feed Back

* Duplicated Code
  + frmAddBirds Lines 157- 175 , 123-149
  + frmAddElephants Lines 169 – 188, 118-140
  + frmAddLions 176- 193, 79-95
  + Validation Lines 35 to 56
  + 239 – 5 groups
* Use properties for getFoodType
* Why is the form maximised, it looks funny?
* Splash Screen should only start once at the beginning of the program.

# Corrections

* Made getFoodType a property and changed the name of the Methods in the IFood interface class. (GetFeedingTimeMorning to MorningFeedTime)
* All cloned code has been corrected.
* Splash screen only appear once in the program
* Minimizing the Report form will make the records being viewed too clustered. This has not been changed.

# Enhancements

* Added a Login Form. Therefore, only the administrator and selected employers are allowed access to the system.
* Adding a new zoo keeper has been a feature added to the program. I have added the Add zoo keeper form. Any newly employed members can now be added.
* When an employee is added, I have used the random generated to create a password. This password consists of 3 random digits and the “Z” character.
* I have also created an additional text file which stores the employee schedules.
* I have added additional methods to the Validation class. One method checks for all null values and the other checks if the formatting of a variable is correct eg. 23 instead of 23 cm.

# References

<http://stackoverflow.com/questions/473148/c-sharp-listview-how-do-i-add-items-to-columns-2-3-and-4-etc>

Adding Items to columns in a List view display