

PET CARE SYSTEM

* **Faculty: Ho Hoan Kiem**
* **Batch:**
* **Team Member:**
  + - * **Nguyễn Minh Quân**
      * **Nguyễn Trung Nhân**

**Ho Chi Minh City, Oct 12 2017**

Table of Contents

1. Review 1

* **Problem Definition..…………………………………………………………………………………01**
* **Customer Request Specification……………………………………………………………… 02**

1. Review 2

* ***Use Case Diagram* 04**
* ***Class Diagram* 15**
* ***Sequence Diagram* 16**
* ***ERD* 18**
* ***Entity Description* 19**
* ***Table Relationship Diagram* 26**

1. Review 3

* Interface Design………………………………………………………………………………..... 27
* Site Map……………………………………………………………………………………………… 51
* Report ………………………………………………………………………………………………..53
* Tasksheet……………………………………………………………………………………………. 56

REVIEW 1

Problem Definition

1. **The Object Of The Project**

This project will aim at creation of a Pet Care System. This will be accessible to all carers who have a valid User Id and Password. This system provides the following facilities:

* Pet Management (Add, Search, Delete, Update, Change Password).
* Check information of pet.
* Return pet back to its owner.

Customer Request   
 Specification

1. **Functional Requirements**

Following is a list of functionalities of the system:

* Carers must have a valid User Id and Password to login to the system
* After the valid user login, system will show you the list of pets that having in the system.
* Carer can manage pets. Manage pets include add new pets, delete pets and update pets. If the action is successful a notification should appear to the carer, in case it is unsuccessful, a proper message should be given to the carer as to why it failed.
* Carer can return pets back to its owner. After that action, that pet will be disappear in the list of pets that having in the system. That pet will move to other list which include all pet that expired this pet care system.
* Carer can view the list of information about all pets which have been in the system.

# Hardware/ Software Requirements

By adding new technologies such as Windows Presentation Foundation and 3-layer architecture, the web site becomes more interactive with users, security and easily updated.

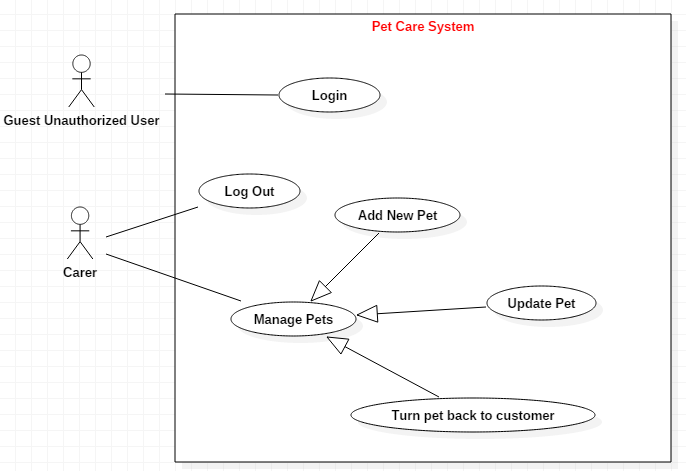
1. **System Requirement**

* **For Users**
  + **Hardware**
    - CPU: Pentium IV 2.0 GHz or higher
    - RAM: 512 MB or higher
  + **Software**
    - OS: Windows 7/XP Professional/Server 2003…
    - Browser: Internet Explorer 8.0 or higher
* **For Software Developers**
  + **Hardware**
    - CPU: Pentium IV 2.0 GHz or higher
    - RAM: 512 MB or higher
  + **Software**
    - OS: Windows XP Professional/Server 2003…
    - Environment: Microsoft .NET Framework 3.5
    - Database Management System: Microsoft SQL Server 2014
    - IDE: Microsoft Visual Studio 2016

|  |  |  |
| --- | --- | --- |
| **Date: Obt 12, 2017** | **Prepared by Project Group 6**  **Team Leader:** | **Approved by**  **Mr Ho Hoan Kiem** |
| **Signature** |  |  |

REVIEW 2

**USE CASE DIAGRAM**



**USE CASE DESCRIPTION**

1. **Actor:**

* **Carer :** include all carers that use this application. They are guests and members. ***Guest*** is the person that uses this application but doesn’t login. ***Member*** is the person that uses this application and login to use.
* **Usecase:**

|  |  |
| --- | --- |
| **Usecase** | Login |
| **Short Description** | Log in application to use any function of member. |
| **Pre-Conditions** |  |
| **Post-Conditions** |  |
| **Main flow** | \_Guest inputs the login information :   * Login ID. * Password.   \_Guest clicks on the Login button.  \_System check the Login ID and Password.  \_If Login ID and Password are valid, system will move member to Homepage.  \_If a wrong password, a proper message should be given to the carer as to why it failed and let them input again. |
| **Alternative flow(s)** | \_If guest clicks on “Reset” button, the Login ID field and Password field are blank.  \_If Login ID field or Password field is invalid form or blank when guest clicks on the “Login” button, system will notify the error.  \_If server notifies any error when it’s processing, system must be write report into Log file. |

|  |  |
| --- | --- |
| **Usecase** | Manage Pets |
| **Short Description** | Manage all information about pet. |
| **Pre-Conditions** |  |
| **Post-Conditions** |  |
| **Main flow** | \_Manage all information of pet. |
| **Alternative flow(s)** | \_If server notifies any error when it’s processing, system must be write report into Log file. |

|  |  |
| --- | --- |
| **Usecase** | View Pet’s Information |
| **Short Description** | Carer view pets’s information. |
| **Pre-Conditions** |  |
| **Post-Conditions** |  |
| **Main flow** | \_Carer choose the pet that carer wants to see information.  \_Carer clicks on the row of the pet which they want to see the information.  \_System shows the pet’s information to the carer. |
| **Alternative flow(s)** | \_If server notifies any error when it’s processing, system must be write report into Log file. |

|  |  |
| --- | --- |
| **Usecase** | Insert Pets |
| **Short Description** | Insert all information about a pet. |
| **Pre-Conditions** |  |
| **Post-Conditions** |  |
| **Main flow** | \_Carer can Insert pet’s information.  \_Carer chooses the correlative function. |
| **Alternative flow(s)** | \_If server notifies any error when it’s processing, system must be write report into Log file. |

-

|  |  |
| --- | --- |
| **Usecase** | Update pets |
| **Short Description** | Update all information about pet’s information. |
| **Pre-Conditions** |  |
| **Post-Conditions** |  |
| **Main flow** | \_Carer can update pets’s information.  \_Carer chooses the correlative function. |
| **Alternative flow(s)** | \_If server notifies any error when it’s processing, system must be write report into Log file. |

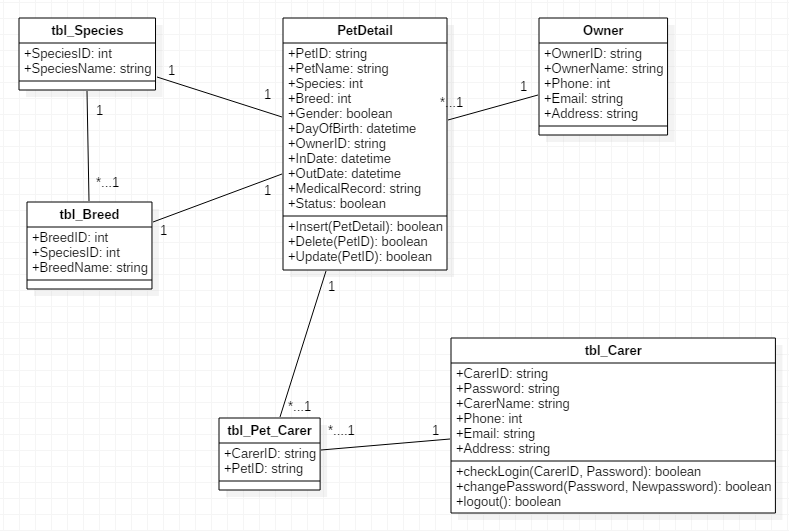
|  |  |
| --- | --- |
| **Usecase** | Search |
| **Short Description** | Search all information about one or many pets. |
| **Pre-Conditions** |  |
| **Post-Conditions** |  |
| **Main flow** | \_Carer inputs the information that wants to see in the search field.  \_Carer clicks on the “Search” button.  \_System shows the result to the carer. |
| **Alternative flow(s)** | \_If server notifies any error when it’s processing, system must be write report into Log file. |

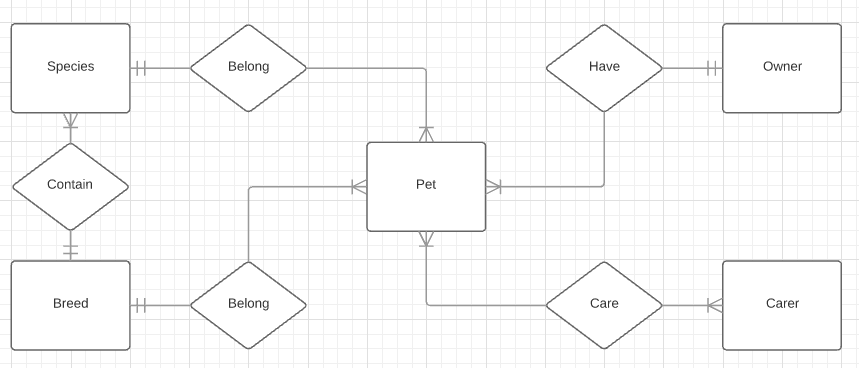
|  |  |
| --- | --- |
| **Usecase** | View Expiry Pet’s Information |
| **Short Description** | Carer can change password |
| **Pre-Conditions** |  |
| **Post-Conditions** |  |
| **Main flow** | \_Carer click the change password in the view list detail or expiry detail.  \_System shows the change password form |
| **Alternative flow(s)** | \_If server notifies any error when it’s processing, system must be write report into Log file. |

|  |  |
| --- | --- |
| **Usecase** | Return the pet back to its owner |
| **Short Description** | Move all information about pet’s information. |
| **Pre-Conditions** |  |
| **Post-Conditions** |  |
| **Main flow** | \_Carer can delete pets’s information  \_All information of that pet will be moved to Expiry Pet List. |
| **Alternative flow(s)** | \_If server notifies any error when it’s processing, system must be write report into Log file. |

|  |  |
| --- | --- |
| **Usecase** | View Expiry Pet’s Information |
| **Short Description** | Carer views all expiry pets’s information. |
| **Pre-Conditions** |  |
| **Post-Conditions** |  |
| **Main flow** | \_Carer views all pets which have been taken care of this system.  \_Carer clicks the button to see list of expiry pet.  \_System shows the pet’s information that have been taken care. |
| **Alternative flow(s)** | \_If server notifies any error when it’s processing, system must be write report into Log file. |

**CLASS DIAGRAM**



ENTITY RELATIONSHIP DIAGRAM (ERD)  
  
  
  
Note:

Entity

Entity Relationship

:One to  
many relationship

:One to

one realtionship

:Many to  
many  
relationship

ENTITY DESCRIPTION

**1.Breed :**

**Breed**

**2. PetDetail :  
3. Species :**

**Pet**

**Species**

**4. Owner :**

**Owner**

**5. Carer :**

**Carer**

**TABLE DESIGN**

1. **Breed:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Data Type** | **Size** | **Check** | **Description** |
| 1 | **🗝**BreedID | int | Identity | Not Null | ID of breed |
| 2 | SpeciesID | int |  | Not Null | ID of species |
| 3 | SpeciesName | Nvarchar | 50 | Not Null |  |

1. **Pet Detail :**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Data Type** | **Size** | **Check** | **Description** |
| 1 | **🗝**PetID | Varchar | 5 | Not Null | ID of pet |
| 2 | PetName | Nvarchar | 50 | Not Null |  |
| 3 | Species | int |  | Not Null |  |
| 4 | Breed | int |  | Not Null |  |
| 5 | Gender | Bit | 50 | Not Null |  |
| 6 | DayOfBirth | Date |  | Null | Birthday of pet |
| 7 | OwnerID | Nvarchar | 5 | Not Null | ID of pet’s owner |
| 8 | InDate | Date |  | Not Null | Date join |
| 9 | OutDate | Date |  | Null | Date expire |
| 10 | MedicalRecord | Nvarchar | 50 | Not Null | Record about medical of pet |
| 11 | Status | Bit |  | Null | 0: still in caring; 1:expired |

1. **Species:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Data Type** | **Size** | **Check** | **Description** |
| 1 | **🗝**SpeciesID | int |  | Not Null | ID of species |
| 2 | SpeciesName | Nvarchar | 50 | Not Null | Name of species |

1. **Owner:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Data Type** | **Size** | **Check** | **Description** |
| 1 | **🗝**OwnerID | Varchar | 5 | Not Null | Owner’s ID |
| 2 | OwnerName | Nvarchar | 50 | Not Null | Owner Name |
| 3 | Phone | Varchar |  | Not Null |  |
| 4 | Email | Datetime |  | Not Null |  |
| 5 | Address | Money |  | Not Null |  |

1. **Carer :**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Data Type** | **Size** | **Check** | **Description** |
| 1 | **🗝**CarerID | Varchar | 5 | Not Null | ID of carer |
| 2 | CarerName | Nvarchar | 50 | Not Null | Name of carer |
| 3 | Password | Varchar | 50 | Not Null | Password to login |
| 4 | Phone | Int |  | Not null |  |
| 5 | Email | Varchar | 50 | Not Null |  |
| 6 | Address | Varchar | 50 | Not Null |  |

1. **Pet\_Carer**

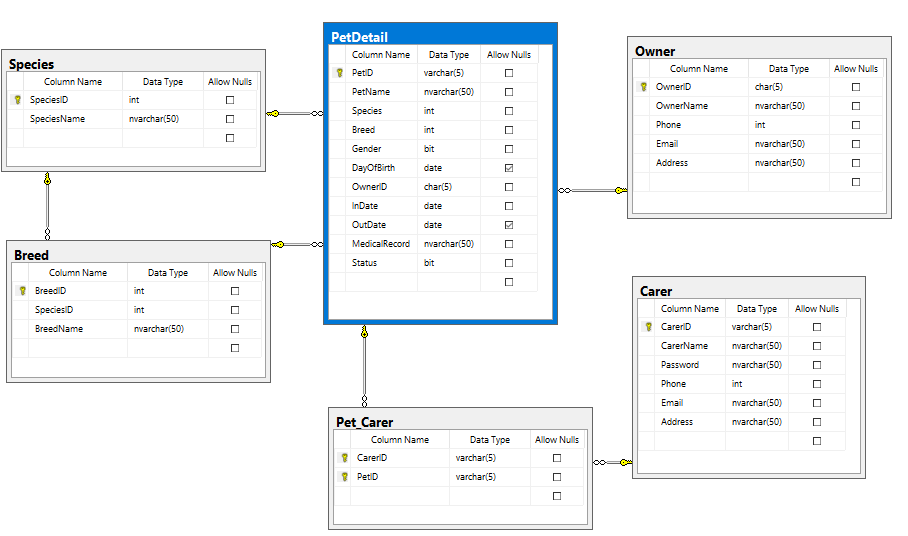
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Column Name** | **Data Type** | **Size** | **Check** | **Description** |
| 1 | **🗝**PetID | Varchar | 5 | Not Null | ID of pet |
| 2 | **🗝**CarerID | Varchar | 5 | Not Null | ID of carer |

**Note**

**🗝** Primary key

✓ Unique constraint

**TABLE RELATIONSHIP DIAGRAM**

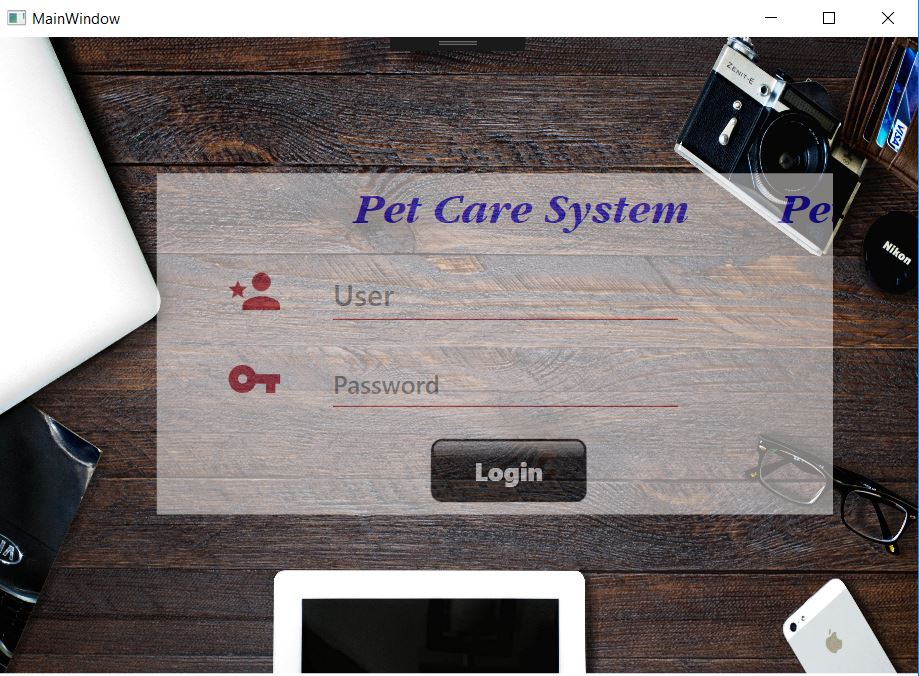


|  |  |  |
| --- | --- | --- |
| **Date: Obt 12, 2017** | **Prepared by Project Group 6**  **Team Leader:** | **Approved by**  **Mr. Ho Hoan Kiem** |
| **Signature** |  |  |

REVIEW 3

**INTERFACE DESIGN**

1. **Login:**

****

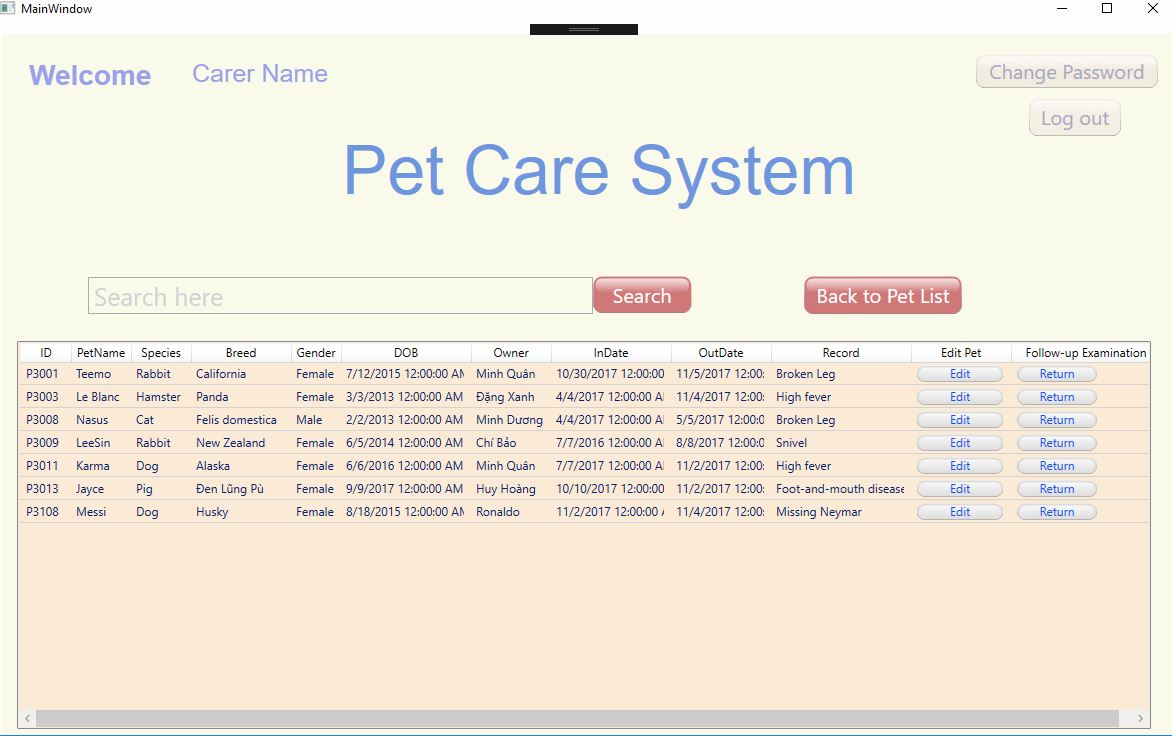
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| User Name | Text Box |  | String, not null | Input Carer’s user name |  |
| Password | Password Box |  | String, not null, | Input Carer’s password |  |
| Login | Button |  | Click | User press the button to login to system |  |

**2**. **List View All Pet:**



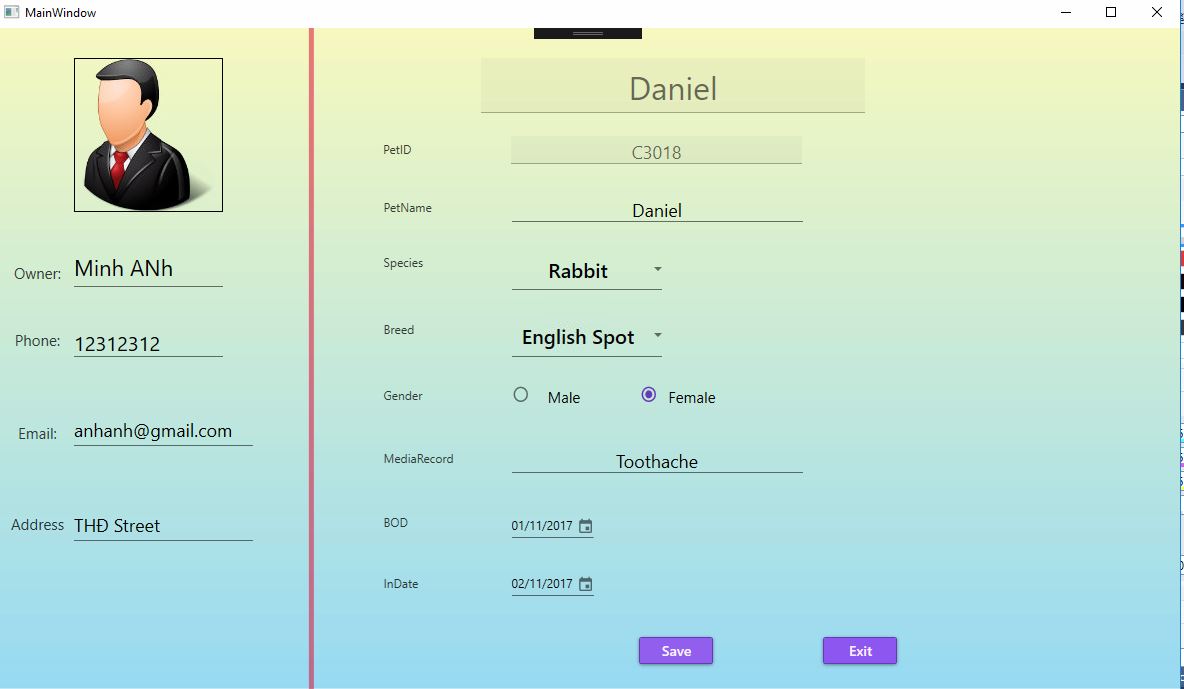
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| Username | Label |  |  | Get the user name |  |
| Change Password | Button |  |  | Go to the Change Password Form |  |
| Log out | Button |  |  | Go back to Login Form |  |
| txtSearch | TextBox |  |  | Search detail |  |
| Search | Button |  |  | When push the button the list of pet is shown |  |
| Add New Pet | Button |  |  | Go to the Add new pet Form |  |
| View History | Button |  |  | Go to the expiry Form |  |
| GridView Pet | Grid View |  |  | Show all Pet |  |
| Edit | Button |  |  | Go to the Edit Pet Form |  |
| Return | Button |  |  | Move pet to expiry form |  |

**3**. **Expiry Pet Detail:**



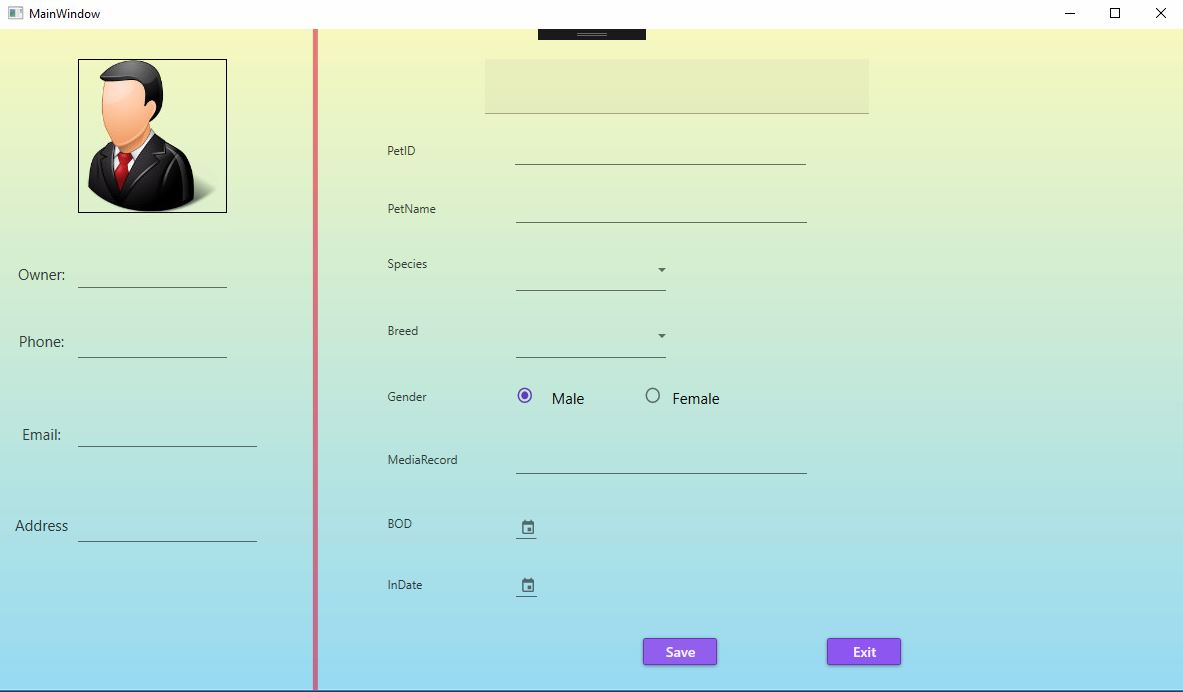
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** |
| Username | Label |  |  | Get the user name |
| Change Password | Button |  |  | Go to the Change Password Form |
| Log out | Button |  |  | Go back to Login Form |
| Search box | TextBox |  |  | Search detail |
| Search | Button |  |  | When push the button the list of pet is shown |
| Back To List | Button |  |  | Go Back to List View Detail Form |
| GridView Pet | Grid View |  |  | Show all Pet |
| Edit | Button |  |  | Go to the Edit Pet Form |
| Return | Button |  |  | Move pet back to List Detail Form |

**4**. **View Detail One Pet:**

****

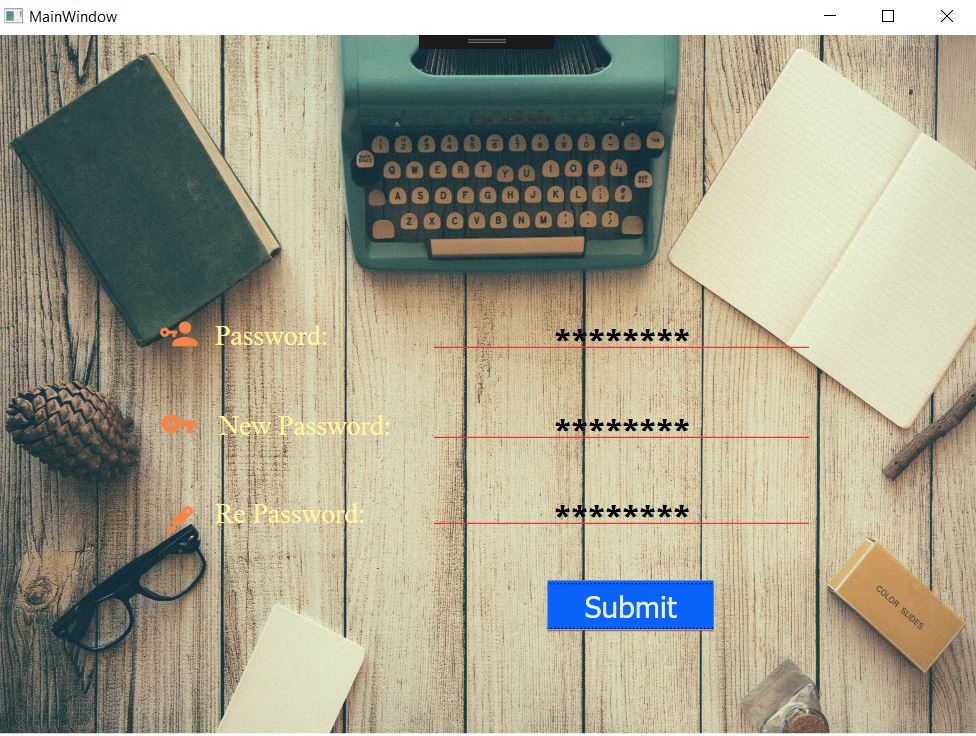
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** |
| txtOwnerName | TextBox |  | String, not null | Get the Owner Name |
| txtPhone | TextBox |  | Int, not null | Get the Phone of Owner |
| txtEmail | TextBox |  | String, not null | Get the Email of Owner |
| txtAddress | TextBox |  | String, not null | Get the Address of Owner |
| txtPetName | TextBox |  | String, not null | Get the Pet Name from the Pet Name inside the form |
| txtPetID | TextBox |  | String, not null | Get The Pet ID From The View Detail, User can’t update here |
| cbSpecies | ComboBox |  | Not Null | Choose the species of the pet |
| cbBreed | ComboBox |  |  | Choose the Breed of the pet depend on the species. |
| rbMale | RadioButton |  | Not null | Choose the gender for the pet |
| rbFemale | RadioButton |  | Not null | Choose the gender for the pet |
| txtMedicalRecord | TextBox |  | String, not null | Get the medical record of the pet |
| dpBOD | DatePicker |  |  | Get the DOB of the Pet |
| dpInDate | DatePicker |  | Not Null | Get the In Date of the Pet |
| btnSave | Button |  | Click | Save the detail of the pet |
| btnExit | Button |  | Click | Exit the form and go back to Detail List Form |

**5**. **Add New Pet:**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** |
| txtOwnerName | TextBox |  | String, not null | Get the Owner Name |
| txtPhone | TextBox |  | Int, not null | Get the Phone of Owner |
| txtEmail | TextBox |  | String, not null | Get the Email of Owner |
| txtAddress | TextBox |  | String, not null | Get the Address of Owner |
| txtPetName | TextBox |  | String, not null | Get the Pet Name from the Pet Name inside the form |
| txtPetID | TextBox |  | String, not null | Get The Pet ID From The View Detail, User can’t update here |
| cbSpecies | ComboBox |  | Not Null | Choose the species of the pet |
| cbBreed | ComboBox |  |  | Choose the Breed of the pet depend on the species. |
| rbMale | RadioButton |  | Not null | Choose the gender for the pet |
| rbFemale | RadioButton |  | Not null | Choose the gender for the pet |
| txtMedicalRecord | TextBox |  | String, not null | Get the medical record of the pet |
| dpBOD | DatePicker |  |  | Get the DOB of the Pet |
| dpInDate | DatePicker |  | Not Null | Get the In Date of the Pet |
| btnSave | Button |  | Click | Save the detail of the pet |
| btnExit | Button |  | Click | Exit the form and go back to Detail List Form |

**6**. **Change Password:**



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Type** | **Event** | **Validation** | **Description** | **Status** |
| Password | Password Box |  | Int, not null | User input the password |  |
| New Password | Password Box |  | int, not null, | User input the new password |  |
| Re Password | Password Box |  | Int, not null | User input the password again |  |
| Submit | Button |  | click | User accept the change password |  |

**TASK SHEET**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Project Group FFG** | | **Project Title**  **Pet Care Application** | | **Date of Preparation** | | **Activity Plan Prepared by**  **Nguyen Trung Nhan** |
| **No** | **Task Title** | | **Planned Start Date** | **Planned Completed Date** | **Team Mate Names** | **Status** |
| 1 | Design Login Form | | Oct 12, 2016 | Oct 12, 2016 | **Nguyen Trung Nhan** | Fix the scale |
| 2 | Login by Carer | | Oct 12, 2016 | Oct 17, 2016 | **Nguyen Trung Nhan** | Use EN with LinQ to connect to DB |
| 3 | Design the search form | | Oct 17, 2016 | Oct 22, 2016 | **Nguyen Trung Nhan** | Adjust the color for the main window |
| 4 | View list of Pet in system | | Oct 21, 2016 | Oct 24, 2016 | **Nguyen Trung Nhan** |  |
| 5 | Search Pet in system | | Oct 23, 2016 | Oct 25, 2016 | **Nguyen Minh Quan** | Search with autocomplete |
| 6 | Add new Pet | | Oct 26, 2016 | Nov 1, 2016 | **Nguyen Trung Nhan + Nguyen Minh Quan** |  |
| 7 | Delete the old Pet | | Change delete to return pet |
| 8 | Update the info of Pet | | Show form when onclick on pet |
| 9 | Logout Carer | | Nov 2, 2016 | Nov 3, 2016 | **Nguyen Minh Quan** |  |
| 10 | Design the View expiry form | | Nov 2, 2016 | Nov 2, 2016 | **Nguyen Minh Quan** | Use Light color and enlarge the text |
| 11 | View the expiry form detail | | Nov 4, 2016 | Nov 4, 2016 | **Nguyen Trung Nhan** |  |
| 12 | Design the Change Password Form | | Now 1, 2016 | Nov 1, 2016 | **Nguyen Trung Nhan** | Change the picture background |
| 13 | Validation for the Change Password Form | | Nov 3, 2016 | Nov 4, 2016 | **Nguyen Minh Quan** | Validation input user in password box |

|  |  |  |
| --- | --- | --- |
| **Date: Oct 10, 2012** | **Prepared by Project Group 4**  **Team Leader:** | **Approved by**  **Mr. Ho Hoan Kiem** |
| **Signature** |  |  |