**Test Scenarios**

**Customer Online Scenarios**

* **New customer creates an account, adds a pet, adds reservation**
  + User begins at login page
  + User clicks on the create new account button
  + User is brought to the form to create an account.
  + User fills out form.
  + User corrects errors and submits
  + User is brought to the appropriate form to add a pet.
  + User fills out form.
  + User corrects errors and submits
  + User is brought to the appropriate form to add a reservation
    - The pet they have added appears on the reservation page
  + User fills out form.
  + User corrects errors and submits
  + The user is brought to their home page where the reservation is displayed.
* **New customer creates an account, adds a pet, adds reservation with errors**
  + User begins at login page
  + User clicks on the create new account button
  + User is brought to the form to create an account.
  + User fills out form with errors
    - Form is not submitted and user is notified of errors
    - Refer to appendix A for validation tests
  + User corrects errors and submits
  + User is brought to the appropriate form to add a pet.
  + User fills out form with errors
    - Form is not submitted and user is notified of errors
    - Refer to appendix A for validation tests
  + User corrects errors and submits
  + User is brought to the appropriate form to add a reservation
    - The pet they have added appears on the reservation page
  + User fills out form with errors
    - Form is not submitted and user is notified of errors
    - Refer to appendix A for validation tests
  + User corrects errors and submits
  + The user is brought to their home page where the reservation is displayed.
* **New customer creates an account and presses log out**
  + User begins at login page
  + User clicks on the create new account button
  + User is brought to the form to create an account.
  + User fills out form correctly
  + User is brought to the appropriate form to add a pet.
  + The user presses log out
  + User is brought back to login page
* **New customer creates an account, does not add pet, or reservation views other pages**
  + User begins at login page
  + User clicks on the create new account button
  + User is brought to the form to create an account.
  + User fills out form correctly
  + User is brought to the appropriate form to add a pet.
  + User clicks on home page
  + Home page notifies user that they have no pets or reservations
  + User clicks on pets
  + Page displays form to add a pet
  + User clicks on reservations
  + Page displays that the user does not have any pets yet.
* **New customer creates an account, does not add pet, or reservation modifies account** 
  + User begins at login page
  + User clicks on the create new account button
  + User is brought to the form to create an account.
  + User fills out form correctly
  + User corrects errors and submits
  + User is brought to the appropriate form to add a pet.
  + User clicks on my account
  + Page displays form with owner’s information (in read-only mode)
  + User clicks on edit account
  + Form becomes editable
  + User changes information and clicks update account
  + Page validates form and updates the information (becomes read-only)
* **Returning customer logs in with errors**
  + User begins at login page
  + User enters their an invalid email
  + User is notified of invalid email, and cannot log in.
* **Returning customer modifies a pet**
  + User begins at login page.
  + User enters their email and submits.
  + User is brought to their home page and any upcoming reservation is displayed.
  + User clicks on pets.
  + A list of the owner’s pets is displayed (in read only).
  + User clicks on a pet to edit.
  + User is brought to the appropriate form with the chosen pet’s information.
  + User changes the pet’s information and clicks on the update button.
  + Form validates information and updates the information.
  + Form displays the lists of pets with the pet updated (in read only).
* **Returning customer adds a pet**
  + User clicks on pets
  + A list of the owner’s pets is displayed (in read only)
  + User clicks on add new pet
  + User is brought to the appropriate form to add a pet.
  + User fills out form correctly
  + The list of pets re-displays with the new pet included there (in read only)
* **Returning customer adds a pet with errors**
  + User clicks on pets
  + A list of the owner’s pets is displayed (in read only)
  + User clicks on add new pet
  + User is brought to the appropriate form to add a pet.
  + User fills out form incorrectly.
    - Form is not submitted and user is notified of errors.
    - Refer to appendix A for validation tests.
  + User fills out form incorrectly.
  + The list of pets re-displays with the new pet included there (in read only).
* **Returning customer adds a reservation**
  + User clicks on reservations
  + A list of the owner’s reservations is displayed (in read only)
  + User clicks on add new reservation
  + User is brought to the appropriate form to add a reservation
  + User fills out form correctly
  + The list of reservations is re-displayed with the new reservation
* **Returning customer adds a reservation with errors**
  + User clicks on reservations.
  + A list of the owner’s reservations is displayed (in read only).
  + User clicks on add new reservation.
  + User is brought to the appropriate form to add a reservation.
  + User fills out form incorrectly.
    - Form is not submitted and user is notified of errors.
    - Refer to appendix A for validation tests.
  + User fills out form incorrectly.
  + The list of reservations is re-displayed with the new reservation.
* **Returning customer modifies a reservation with errors**
  + User clicks on reservations.
  + A list of the owner’s reservations is displayed (in read only).
  + User clicks on a reservation to edit.
  + User is brought to the appropriate form with the chosen reservation’s information.
  + User changes the reservation’s information incorrectly and clicks on the update button.
    - Form is not submitted and user is notified of errors.
    - Refer to appendix A for validation tests.
  + Form validates information and updates the information.
  + Form displays the lists of reservations with the reservation updated (in read only)
* **Returning customer modifies a reservation**
  + User clicks on reservations
  + A list of the owner’s reservations is displayed (in read only)
  + User clicks on a reservation to edit
  + User is brought to the appropriate form with the chosen reservation’s information
  + User changes the reservation’s information and clicks on the update button
  + Form validates information and updates the information.
  + Form displays the lists of reservations with the reservation updated (in read only)
* **Returning customer modifies or adds a reservation and adds a pet midway**
  + User clicks on reservations
  + A list of the owner’s reservations is displayed (in read only)
  + User clicks on add new reservation
  + User is brought to the appropriate form to add a reservation
  + User clicks on add a pet
  + User is brought to the appropriate form to add a pet
  + The user fills out the pets information correctly and submits
  + The user is brought back to the reservation form where the information they had already filled out is still there and the new pet is displayed.
* **Returning customer chooses to modify their account but cancels midway**
  + User clicks my account
  + Account information is displayed in read only mode
  + User chooses to modify information
  + User makes changes but presses cancel
  + Form is redisplayed in read-only mode with the previous data before the changes
* **Returning customer modifies their account**
  + User clicks my account
  + Page displays form with owner’s information (in read-only mode)
  + User clicks on edit account
  + Form becomes editable
  + User changes information and clicks update account
  + Page validates form and updates the information (becomes read-only)
* **Returning customer cancels upcoming or active reservation**
  + User clicks on reservations
  + User selects the reservation they want to modify
  + User selects to cancel the reservation
* **Returning customer cancels past reservation**
  + User clicks on reservations
  + User selects the reservation they want to modify (in the past)
  + User selects to cancel the reservation
  + System notifies customer they can’t cancel a past reservation
* **Returning customer logs out**
  + The user presses log out
  + User is brought back to login page

**Employee Scenarios**

* **Employee logs into employee account**
  + Employee enters [reed@hvk.ca](mailto:reed@hvk.ca) with password 1234
  + Employee is brought to the home page where a search bar is displayed alongside a list of reservations ending today and a list of reservations starting today
* **Employee views customers, adds new customer**
  + Employee clicks on HVK Customers.
  + Page displays customers and a list of their pets.
  + Employee chooses to add a customer.
  + Page displays form to add a new customer.
  + Employee enters all information correctly.
  + Employee submits new customer.
  + Page displays customer page with the new customer.
* **Employee views customers, adds new customer with errors**
  + Employee clicks on HVK Customers.
  + Page displays customers and a list of their pets.
  + Employee chooses to add a customer.
  + Page displays form to add a new customer.
  + Employee enters information incorrectly.
    - Form is not submitted and user is notified of errors.
    - Refer to appendix A for validation tests.
  + Employee corrects information
  + Employee submits new customer.
  + Page displays customer page with the new customer.
* **Employee views customers, modifies a customer**
  + Employee clicks on HVK Customers.
  + Page displays a list of customers.
  + Employee selects a customer to modify.
  + Page displays form to modify customer information.
  + Employee enters all information correctly.
  + Employee submits modified customer information.
  + Page displays customer page with updated information.
* **Employee views customers, modifies a customer with errors**
  + Employee clicks on HVK Customers.
  + Page displays a list of customers.
  + Employee selects a customer to modify.
  + Page displays form to modify customer information.
  + Employee enters some information incorrectly.
    - Form is not submitted and user is notified of errors.
    - Refer to appendix A for validation tests.
  + Employee corrects information.
  + Employee submits modified customer information.
  + Page displays customer page with updated information.
* **Employee adds reservation for a customer with errors**
  + Employee clicks on reservations.
  + Employee clicks on add a reservation
  + Page displays a form to add a reservation.
  + Employee selects a customer from drop down list.
  + Page enables form to add a reservation.
  + Employee fills out form incorrectly.
    - Form is not submitted and user is notified of errors.
    - Refer to appendix A for validation tests.
  + Employee corrects information.
  + Page displays reservation information.
* **Employee adds reservation for a customer**
  + Employee clicks on reservations.
  + Employee clicks on add a reservation.
  + Page displays a form to add a reservation.
  + Employee selects a customer from drop down list.
  + Page enables form to add a reservation.
  + Employee fills out form correctly.
  + Page displays reservation information.
* **Employee modifies a reservation**
  + Employee clicks on reservations.
  + Employee selects to edit a reservation
  + Page displays a form to edit a reservation.
  + Employee selects a customer from drop down list.
  + Page enables form to add a reservation.
  + Employee fills out form correctly.
  + Page displays reservation information.
* **Employee modifies a reservation for a customer with errors**
  + Employee clicks on reservations.
  + Employee selects to edit a reservation
  + Page displays a form to add a reservation.
  + Employee selects a customer from drop down list.
  + Page enables form to add a reservation.
  + Employee fills out form incorrectly.
    - Form is not submitted and user is notified of errors.
    - Refer to appendix A for validation tests.
  + Employee corrects information.
  + Page displays reservation information.
* **Employee cancels a reservation**
  + Employee clicks on reservations
  + Employee selects a reservation to cancel
  + Page prompts employee for confirmation.
* **Employee cancels past reservation**
  + User clicks on reservations
  + User selects the reservation they want to modify (in the past)
  + User selects to cancel the reservation
  + System notifies customer they can’t cancel a past reservation
* **Employee verifies a pet’s vaccinations**
  + Employee clicks on HVK Customers
  + Employee clicks on the customer they want to choose a pet from
  + Employee clicks on the pet they want to modify
  + Page displays the appropriate form to display with the pets information
  + Employee checks off which vaccinations were modified
  + Employee updates changes
  + Employee is brought back to HVK Customers

**Appendix A**

**Manage Account**

*First Name, Last Name, Emergency First Name, Emergency Last Name Fields, Vet Name*

|  |  |
| --- | --- |
| Input | Result |
| Cody | Pass |
| cody | Pass |
| CODY | Pass |
| Marc-Andree | Pass |
| Marc Andree | Pass |
| CB | Pass |
| [empty string] | Error Message |
| C9dy | Error Message |
| C | Pass |
| Try length of 26 | Fail |

*Street, City, Province and Postal Code Fields*

|  |  |  |
| --- | --- | --- |
| Field | Input | Result |
| Street | 3 Place de la soeur-elizabeth-de-hongrie | Pass |
| Street | [Empty Space] city province and postal code are also blank | Pass |
| Street | [Empty Space] city province or postal code are not blank | Error Message |
| Street | Try length of 41 | fail |
| City | Hull | Pass |
| City | hull | Pass |
| City | HULL | Pass |
| City | Hull-gatineau | Pass |
| City | [Empty Space] street province and postal code are also blank | Pass |
| City | Hull9 | Error Message |
| City | [Empty String] street province or postal code are not blank | Error Message |
| City | Try length of 25 | Fail |
| Province | [Select One] street, city and postal code is also blank | Pass |
| Province | [Select One] street, city or postal code are not blank | Error Message |
| Postal Code | J0X 1V0 | Pass |
| Postal Code | J0X1V0 | Pass |
| Postal Code | JJX 1V0 | Error Message |
| Postal Code | JJx1V0 | Error Message |
| Postal Code | JJJ JJJ | Error Message |
| Postal Code | Try length of 8 | Fail |
| Postal Code | [Empty String] street, city, and province are also blank | Pass |
| Postal Code | [Empty String] street, city or province are not blank | Error Message |

*Phone and Email Fields*

|  |  |  |
| --- | --- | --- |
| Field | Input | Result |
| Phone | 1231231234 | Pass |
| Phone (Client Side) | [Empty String] | Pass |
| Phone (Clerk Side) | [Empty String] Email is not blank | Pass |
| Phone (Clerk Side) | [Empty String] Email is blank | Error Message |
| Phone | Try length of 11 | Fail |
| Phone | (819) 1F3-1234 | Error Message |
| Phone | (819) 1231234 | Error Message |
| Email | Berube.c@hotmail.ca | Pass |
| Email | Berube.c@hotm.ail.ca | Pass |
| Email | Berbechotm.ailca | Error Message |
| Email | Ber@ube.c@hotmail.ca | Error Message |
| Email (Clerk Side) | [Empty String] Phone is not blank | Pass |
| Email | Try length of 51 | Fail |
| Email (Client Side) | [Empty String] | Error Message |
| Email (Clerk Side) | [Empty String] Phone is blank | Error Message |

*Password and Confirm Password*

|  |  |  |
| --- | --- | --- |
| Field | Input | Result |
| Password > 8 char | 123Password | Pass |
| Confirm Password (matches) | 123Password | Pass |
| Password 8 char | Password | Pass |
| Confirm Password (no match) | password | Fail |
| Password < 8 char | Pass | Fail |
| Password | [Empty String] | Fail |
| Confirm Password | [Empty String] | Fail |

*Emergency First Name, Last Name, Phone Number*

|  |  |  |
| --- | --- | --- |
| Field | Input | Result |
| First Name | [Empty String] Emergency Last Name and Phone are blank | Pass |
| Last Name | [Empty String] Emergency First Name and Phone are blank | Pass |
| Phone | [Empty String] Emergency Last Name and First Name are blank | Pass |
| First Name | [Empty String] Emergency Last Name or Phone are not blank | Error Message |
| Last Name | [Empty String] Emergency First Name or Phone are not blank | Error Message |
| Phone | [Empty String] Emergency Last Name or First Name are not blank | Pass |

*Create Account Button*

|  |  |
| --- | --- |
| Scenario | Result |
| New customer is creating an account | Customer object is created with **valid** values. Session “Customer” is created with the object. User is then sent to the create pet page. |
| Existing customer updates their information | Session “Customer” is updated and the fields become disabled and displays the new information. |

*Edit Button*

|  |
| --- |
| Result |
| Fields become enabled where user can edit the information |

*Update Button*

|  |
| --- |
| Result |
| Updates the customer object with the new information |

*Clear Button*

|  |
| --- |
| Result |
| Fields become enabled where user can edit the information |

*Cancel Button*

|  |
| --- |
| Result |
| Cancels any changes made to the text boxes and sets them to the current values in the “Customer” session |

**Manage Pet**

*Gender Radio Buttons*

|  |  |
| --- | --- |
| Input | Result |
| Either is checked | Passed |
| Both are unchecked | Error Messages |

*Spayed Checkbox*

|  |  |
| --- | --- |
| Input | Result |
| Checked | Passed |
| Unchecked | Error Messages |

*Name Field*

|  |  |
| --- | --- |
| Input | Result |
| Any string | Passed |
| Empty string | Error Message |
| Try length of 26 | Fail |

*Birthday, Vaccination fields*

|  |  |
| --- | --- |
| Input | Result |
| Is a date in the past | Passed |
| Is the current date | Passed |
| Is a date longer than 30 years in the past | Error Message |
| Is a date in the future | Error Message |
| Empty string | Error Message |
| Is not a date | Error Message |

*Birthday*

|  |  |
| --- | --- |
| Input | Result |
| Is a date longer than 30 years in the past | Error Message |
| Is a date in the future | Error Message |

*Create Pet Button*

|  |  |
| --- | --- |
| Scenario | Result |
| customer is creating a new pet | Pet object is created with **valid** values and the Session “Pet” is created |
| customer updates existing pet | Session “Pet” is updated and the fields become disabled and displays the new information. |

**Manage Reservation**

*Start Date Field*

|  |  |
| --- | --- |
| Input | Result |
| Date in the future | Passed |
| Today’s Date | Passed |
| Date in the past | Error Message |
| Not a date | Error Message |

*End Date Field*

|  |  |
| --- | --- |
| Input | Result |
| Date in the future | Passed |
| Date before the start date | Error Message |
| Date as today’s date | Error Message |
| Is not a date | Error Message |

*Pet CheckBoxes*

|  |  |
| --- | --- |
| Input | Result |
| At least one is checked | Passed |
| None are checked | Error Message |

*Make Reservation Button*

|  |  |
| --- | --- |
| Scenario | Result |
| New customer is creating an account | Customer object is created with **valid** values. Session “Customer” is created with the object. User is then sent to the create pet page. |
| Existing customer updates their information | Session “Customer” is updated and the fields become disabled and displays the new information. |

*Add Pet Button*

|  |
| --- |
| Result |
| Customer is redirected to add a pet and once they finished adding the pet they are returned back to the reservation state |