`1 q21Iteration 5

## Prepared by: Big B

November 30th, 2022

**Use Cases:**

Use cases are a way to represent how a particular “actor”, for example, the administrator, donor, etc. will interact with the system in question. Use cases are a way to let these actors know how our solution will respond to various actions that will take place within our system. After going through and deciding on all our system requirements, we created our use cases based on what actions our system would be required to perform. They are as follows.

**Use Case 1:** Create Donation

Description: Donations are processed using the payment system

Actor: Donor

Risk Level: High

**Use Case 2**: Edit Donation

Description: Donations can be edited or changed by an admin if needed

Actor: Administrator

Risk Level: High

**Use Case 3**: Delete Donation

Description: Admin can delete donation if needed

Actor: Administrator

Risk Level: High

**Use Case 4**: Create New Donor

Description: Donor puts information and registers before donating

Actor: Donor

Risk Level: Low

**Use Case 5**: Edit New Donor

Description: Donor can edit their account

Actor: Donor

Risk Level: Low

**Use Case 6**: Delete New Donor

Description: Admin can remove donor's information if needed

Actor: Administrator

Risk Level: High

**Use Case 7**: Create Existing Donor

Description: Existing Donor can sign in with existing information

Actor: Donor

Risk Level: Low

**Use Case 8**: Edit Existing Donor

Description: Existing Donors can change their account details

Actor: Donor

Risk Level: Low

**Use Case 9**: Delete Existing Donor

Description: Admin can delete existing donors if needed

Actor: Administrator

Risk Level: High

**Use Case 10:** Create FinancialReport

Description: Administrator can create financial reports

Actor: Administrator

Risk: High

**Use Case 11**: Edit Financial Report

Description: Administrator can edit financial reports

Actor: Administrator

Risk: High

**Use Case 12**: Delete Financial Report

Description: Administrator can remove financial reports

Actor: Administrator

Risk: High

**Use Case 13:** CreateFinancial Report ID

Description: Administrator can create financial report identification numbers

Actor: Administrator

Risk: High

**Use Case 14:** Delete FinancialReport ID’s

Description: Can delete financial report identification numbers

Actor: Administrator

Risk: High

**Use Case 15:** Edit Financial Report ID’s

Description: Administrator can edit the financial report identification numbers

Actor: Administrator

Risk: High

**Use Case 16:** Store Dates

Description: Administrator can store dates per report

Actor: Administrator

Risk: Low

**Use Case 17:** Edit Dates

Description: Administrator can edit dates for each report

Actor: Administrator

Risk: Low

**Use Case 18:** Remove Dates

Description: Administrator can delete dates for reports

Actor: Administrator

Risk: Low

**Use Case 19:** User Email Signup

Description: Users can use their email to sign up to receive updates from Blue Ridge

Actor: Users

Risk: Low

**Use Case 20:** User Email Edit

Description: Users can change the email address they used to sign up for updates

Actor: Users

Risk: Low

**Use Case 21:** User Email Remove

Description: Users can remove their email from Blue Ridge to stop receiving updates

Actor: Users

Risk: Low

**Use Case 22:** Creates User Volunteer data

Description: The website will allow users to input their information into the volunteer database.

Actor: Users

Risk: High

**Use Case 23:** Edit User Volunteer data

Description: Users will be able to edit their previously inputted data.

Actor: Users

Risk High

**Use Case 24:** Delete User Volunteer data

Description: User will be able to delete their previously inputted data.

Actor: Users

Risk: High

**Use Case 25**: Create Newsletter

Description: Newsletter will be created for Blue Ridge Animal Rescue & Sanctuary

Actor: Administrator

Risk Level: Low

**Use Case 26**: Edit Newsletter

Description: Newsletter can be edited if changes are required

Actor: Administrator

Risk Level: Low

**Use Case 27**: Delete Newsletter

Description: Old or outdated newsletters can be removed from the website

Actor: Administrator

Risk Level: Low

**Use Case 28**: Create an About Us post

Description: About Us post will be created for Blue Ridge Animal Rescue & Sanctuary

Actor: Administrator

Risk Level: Low

**Use Case 29**: Edit About Us post

Description: About us post can be edited if needed

Actor: Administrator

Risk Level: Low

**Use Case 30**: Delete the About Us post

Description: Delete or remove the About Us post

Actor: Administrator

Risk Level: Low

**Use Case 31: Access** Instagram Account

Description: Allows user to access link to external Blue Ridge Instagram account

Actor: User

Risk: Low

**Use Case 32: Access** Snapchat Account

Description: Allows user to access link to external Blue Ridge Snapchat account

Actor: User

Risk: Low

**Use Case 33: Access** Twitter Account

Description: Allows user to access link to external Blue Ridge Twitter account

Actor: User

Risk: Low

**Use Case 34: Access** Facebook Account

Description: Allows user to access link to external Blue Ridge Facebook account

Actor: User

Risk: Low

**Use Case 35: Access** TikTok Account

Description: Allows user to access link to external Blue Ridge TikTok account

Actor: User

Risk: Low

**Use Case 36:** Create Event post

Description: Administrator can create an event post

Actor: Administrator

Risk Level: Low

**Use Case 37:** Edit Event Post

Description: Administrator can edit an event post

Actor: Administrator

Risk Level: Low

**Use Case 38:** Delete an Event Post

Description: Administrator can delete an event post

Actor: Administrator

Risk Level: Low

**Use Case 39:** Create Volunteer opportunities post

Description: Administrator can create a volunteer opportunity post

Actor: Administrator

Risk: Low

**Use Case 40:** Edit Volunteer Opportunities Post

Description: Administrator can edit a volunteer opportunity post

Actor: Administrator

Risk Level: Low

**Use Case 41:** Delete Volunteer Opportunities Post

Description: Administrator can delete a volunteer opportunity post

Actor: Administrator

Risk Level: High

**Use Case 42**: Add Photos to Website

Description: Administrator can add photos to be posted

Actor: Administrator

Risk Level: Low

**Use Case 43**: Edit Photos on Website

Description: Administrator can update or change different photos to be posted

Actor: Administrator

Risk Level: Low

**Use Case 44**: Delete Photo Management

Description: Administrator can delete outdated photos from the website

Actor: Administrator

Risk Level: Low

**System Requirements**

The requirements below outline what is needed by Blue Ridge Pet Rescue and Sanctuary to succeed and

develop. Both Functional and Non-Functional requirements are listed, each with an individual

ID. A hierarchy is used to categorize the different requirements.

**Functional Requirements:**

**Donation Requirements**

ID: SR001 – The donation system will accept user payments

ID: SR002 - The donation system will allow users to customize entries

ID: SR003 – The donation system will allow users to delete entries

ID: SR004 – The donation system will allow new donor accounts

ID: SR005 – The donation system will allow the editing of donor accounts

ID: SR006 - The donation system will allow the deletion of donor accounts

**Database Requirements**

ID: SR007 - The database will allow for the creation of financial reports

ID: SR008 - The database will allow the customization of reports

ID: SR009- The da allow the deletion of reports.

ID: SR010 - The database will store specific report IDs

ID: SR011 - The database will edit specific report IDs

ID: SR012 - The database will delete specific IDs

ID: SR013 - The database will store specific dates per report

ID: SR014 - The database will allow editing of specific dates per report

ID: SR015 – The database will allow the deletion of specific dates per report

**Social Requirements**

ID: SR016 - The website will allow new user email signup

ID: SR017 - The website will allow user email signup editing

ID: SR018- The website will allow users to delete/unsubscribe from email signup

ID: SR019 - The website will accept volunteer data input

ID: SR020 - The website will allow volunteer data to be edited.

ID: SR021 - The website will allow volunteer data to be deleted.

ID: SR022 - The website will allow the creation of a newsletter

ID: SR023 - The website will allow the editing of a newsletter.

ID: SR024- The website will allow the deletion of a newsletter.

ID: SR025– The website will allow About Us post creation

ID: SR026 – The website will allow editing of About Us post

ID: SR027 – The website will allow the deletion of About Us post

ID: SR028– The website will allow access to organizations Instagram

ID: SR029– The website will allow access to organizations Snapchat

ID: SR030 - The website will allow access to organizations Twitter

ID: SR031 - The website will allow access to organizations Facebook

ID: SR032 - The website will allow access to organizations TikTok

ID: SR033 – The website will display event dates

ID: SR034 – The website will post volunteer opportunities

ID: SR035 – The website will have a photo display

ID: SR036 – The website will allow photo display editing

ID: SR037 - The website will allow photo display deletion

**Non-Functional Requirements:**

**Performance**

ID: SR038 - The website will run efficiently without error

ID: SR039 - The website will frequently be updated and maintained

ID: SR040 – The website will allow user interaction

ID: SR041 - The website will user friendly

**Security**

ID: SR042- The website will be protected against hackers, viruses, and phishers

ID: SR043 - The website will have third-party security

ID: SR044 – The website will have a backup

ID: SR045 – The user information will have embedded security

ID: SR046 - The donation information will have embedded security

ID: SR047 – User transactions and payment information will be secured

**Capacity**

ID: SR048- The database systems will have optimal storage

ID: SR050 -. Storage choice will be efficient

ID: SR051 – The donation systems will have optimal storage

ID: SR052 – The user information systems will have optimal storage

ID: SR053 – The email signup will have optimal storage.

**Availability**

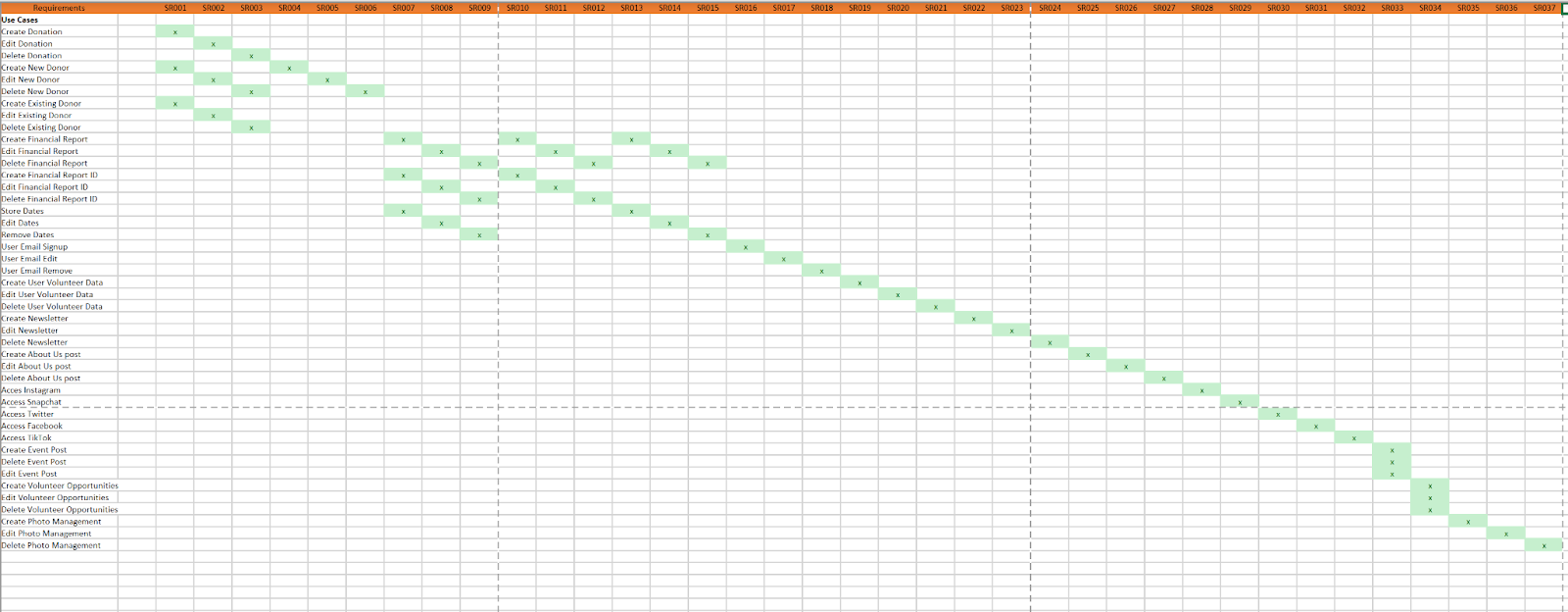
ID: SR054 – The website will be accessible to popular web browsers

ID: SR055 - The website will allow access from large amounts of users simultaneously

ID: SR056 – The website will allow easy navigation

ID: SR057 - The website will follow WC3 Standards and Policies

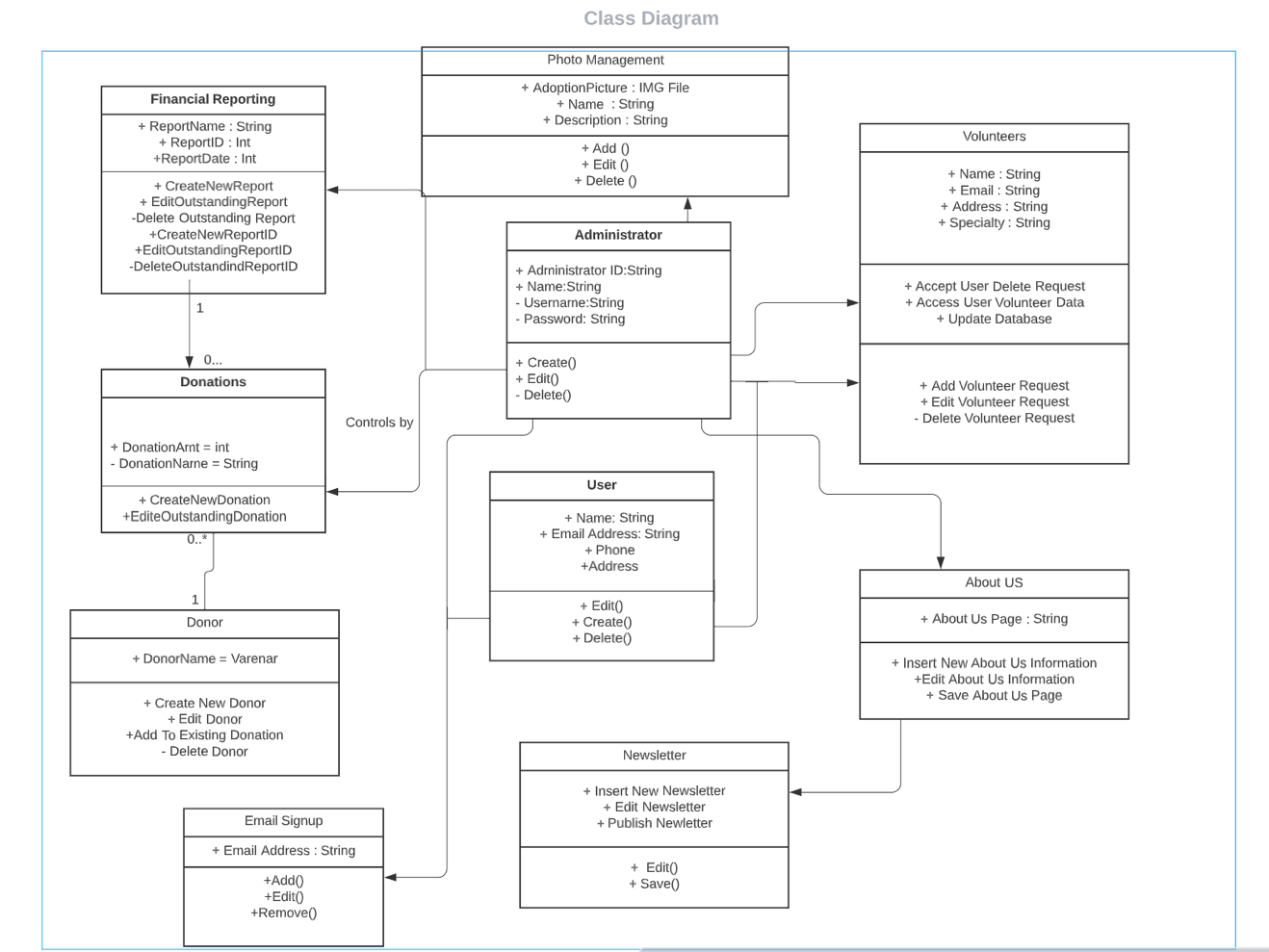
ID: SR058 – The website will be available 24/7 other than maintenance



**Class diagram:**

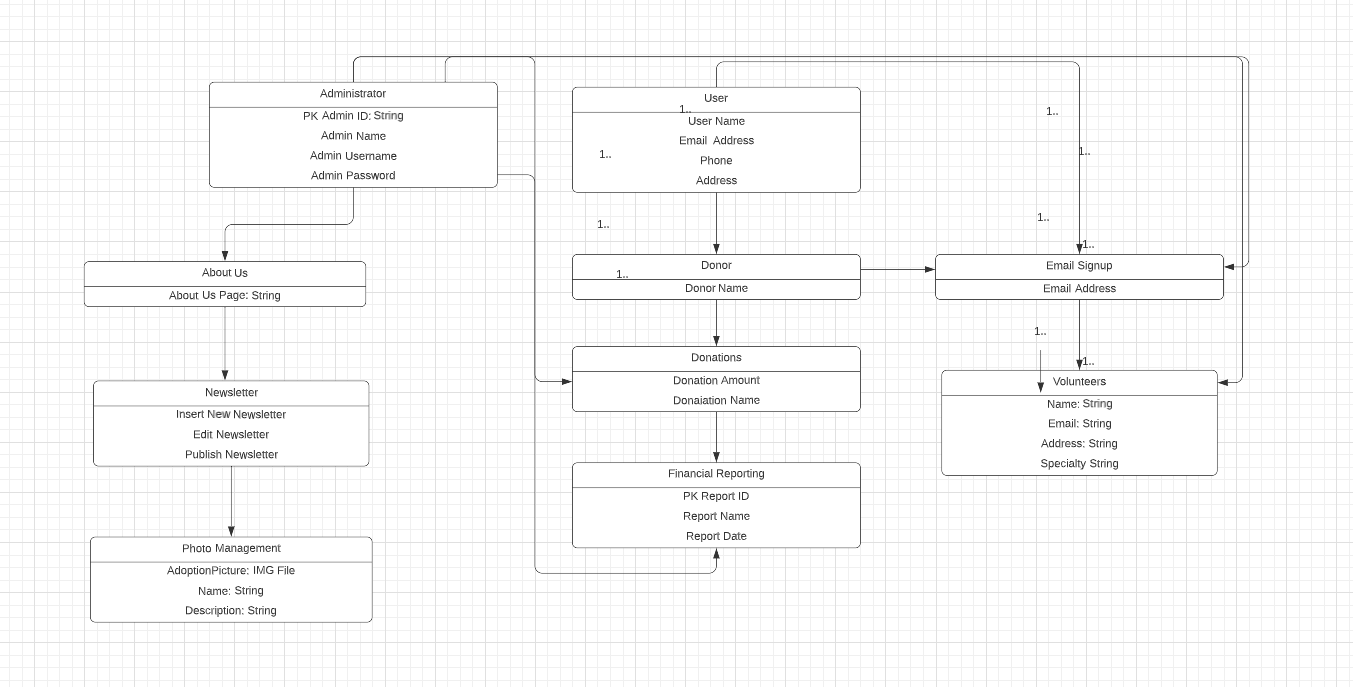
A class diagram is a collection of tables displaying the project's different classes. A class is used to create specific instances or objects, like a Volunteers or a news section. A class contains a class name, attribute list, and method list. An attribute list is what the user needs to input in order to get the class, like their name or email address. The methods are the buttons that the user clicks to perform what they want the class to do, like adding a new volunteer or deleting a report.

We used both verb-noun analysis and prototype analysis to create our class diagram. Verb-noun analysis looks at the use case basic flow and assigns the nouns as attributes and the verbs as methods. We also used prototype analysis, which looks at the prototypes we created to determine attributes and methods, and ensured we weren’t missing any information.



**Database design and data definitions:**

This is a diagram of the database design in 3NF. Having it in 3NF allows us to clearly show the features of the system along with their relationships. With it being in 3NF, we can ensure that all fields in the table are fully dependent on themselves and do not rely on primary key fields.



**User Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Attribute Definition | Attribute Size | Attribute Type | Primary or Foreign Key? |
| Name | User name | **Max 255 bytes** | string | **N/A** |
| EmailAddress | User email address | **Max 255 bytes** | string | N/A |
| Phone | User phone number | **Max 255 bytes** | varchar | N/A |
| Address | User address | **Max 255 bytes** | varchar | N/A |

**Administrator Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Attribute Definition | Attribute Size | Attribute Type | Primary or Foreign Key? |
| Administrator ID | Unique administrator identifier | **2 bytes** | uint | **Primary Key** |
| Name | Administrator name | **Max 255 bytes** | string | N/A |
| Username | Administrator username | **Max 255 bytes** | string | N/A |
| Password | Unique password | **Max 255 bytes** | string | N/A |

**Newsletter Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Attribute Definition | Attribute Size | Attribute Type | Primary or Foreign Key? |
| InsertNewNewsletter | Inserts a new newsletter | **Max 255 bytes** | string | **N/A** |
| EditNewsletter | Edits newsletter | **Max 255 bytes** | string | N/A |
| PublishNewsletter | Publishes newsletter | **Max 255 bytes** | string | N/A |

**About Us Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Attribute Definition | Attribute Size | Attribute Type | Primary or Foreign Key? |
| AboutUsPage | Webpage for about us | **Max 255 bytes** | string | **N/A** |

**Volunteers Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Attribute Definition | Attribute Size | Attribute Type | Primary or Foreign Key? |
| Name | Volunteer name | **Max 255 bytes** | string | **N/A** |
| Email | Volunteer email address | **Max 255 bytes** | string | N/A |
| Address | Volunteer address | **Max 255 bytes** | string | N/A |
| Specialty | Volunteer specialty | **Max 225 bytes** | string | N/A |

**Photo Management Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Attribute Definition | Attribute Size | Attribute Type | Primary or Foreign Key? |
| AdoptionPicture | Picture of adoptable pet | **Max 225 bytes** | IMG File | **N/A** |
| Name | Adoptable pets name | **Max 255 bytes** | string | N/A |
| Description | Adoptable pet description | **Max 255 bytes** | string | N/A |

**Financial Reporting Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Attribute Definition | Attribute Size | Attribute Type | Primary or Foreign Key? |
| ReportID | Unique report identifier | **2 bytes** | uint | **Primary Key** |
| ReportName | Reports name | **Max 255 bytes** | string | N/A |
| ReportDate | Date report was created | **Max 255 bytes** | string | N/A |

**Donations Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Attribute Definition | Attribute Size | Attribute Type | Primary or Foreign Key? |
| DonationAmt | Amount donated | **4 bytes** | int | **N/A** |
| DonationName | Name for donation | **Max 255 bytes** | string | N/A |

**Donor Table:**

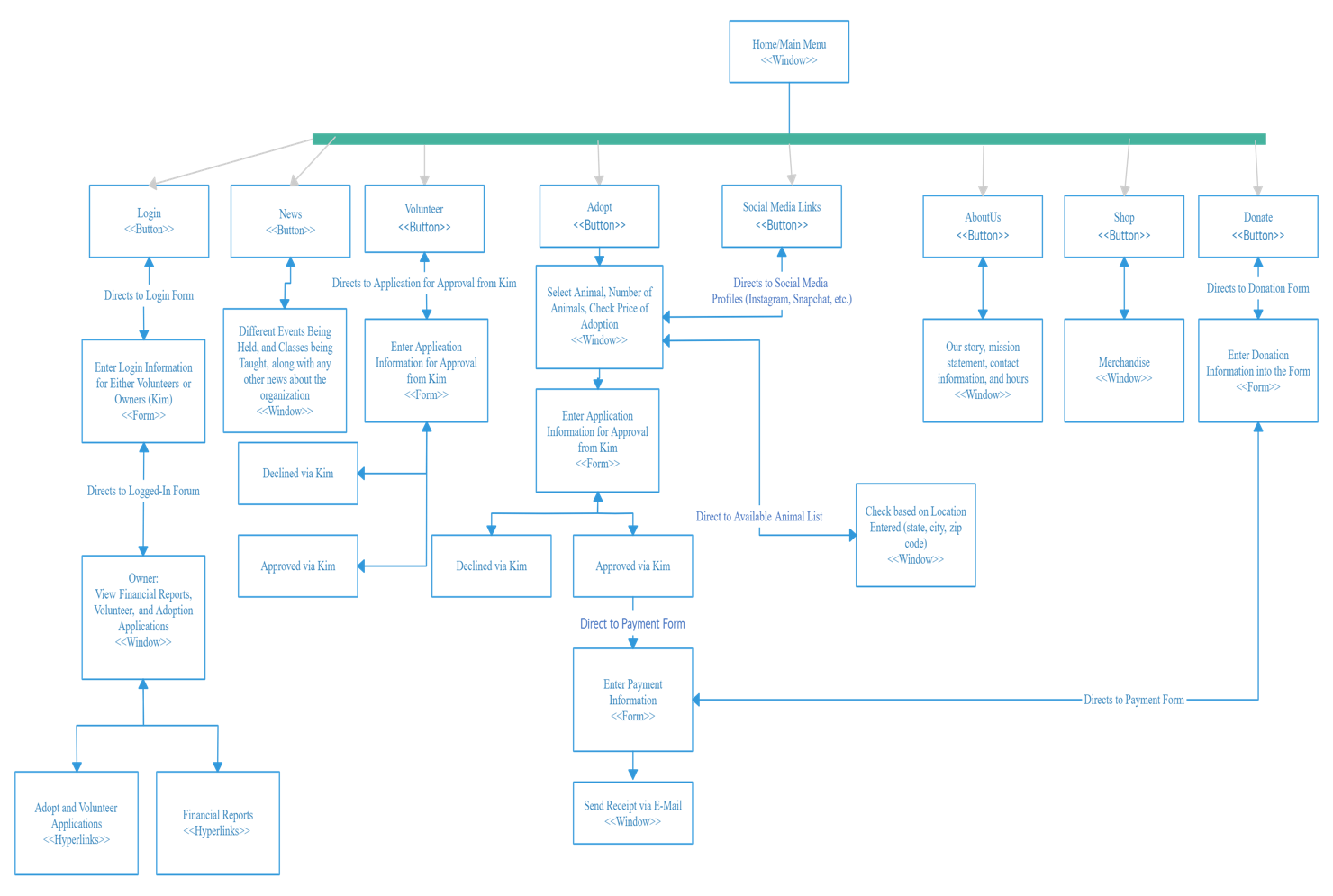
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Attribute Definition | Attribute Size | Attribute Type | Primary or Foreign Key? |
| DonorName | Name of donor | **Max 225 bytes** | string | **N/A** |

**Email Signup Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Attribute Definition | Attribute Size | Attribute Type | Primary or Foreign Key? |
| EmailAddress | User email address | **Max 225 bytes** | string | **N/A** |

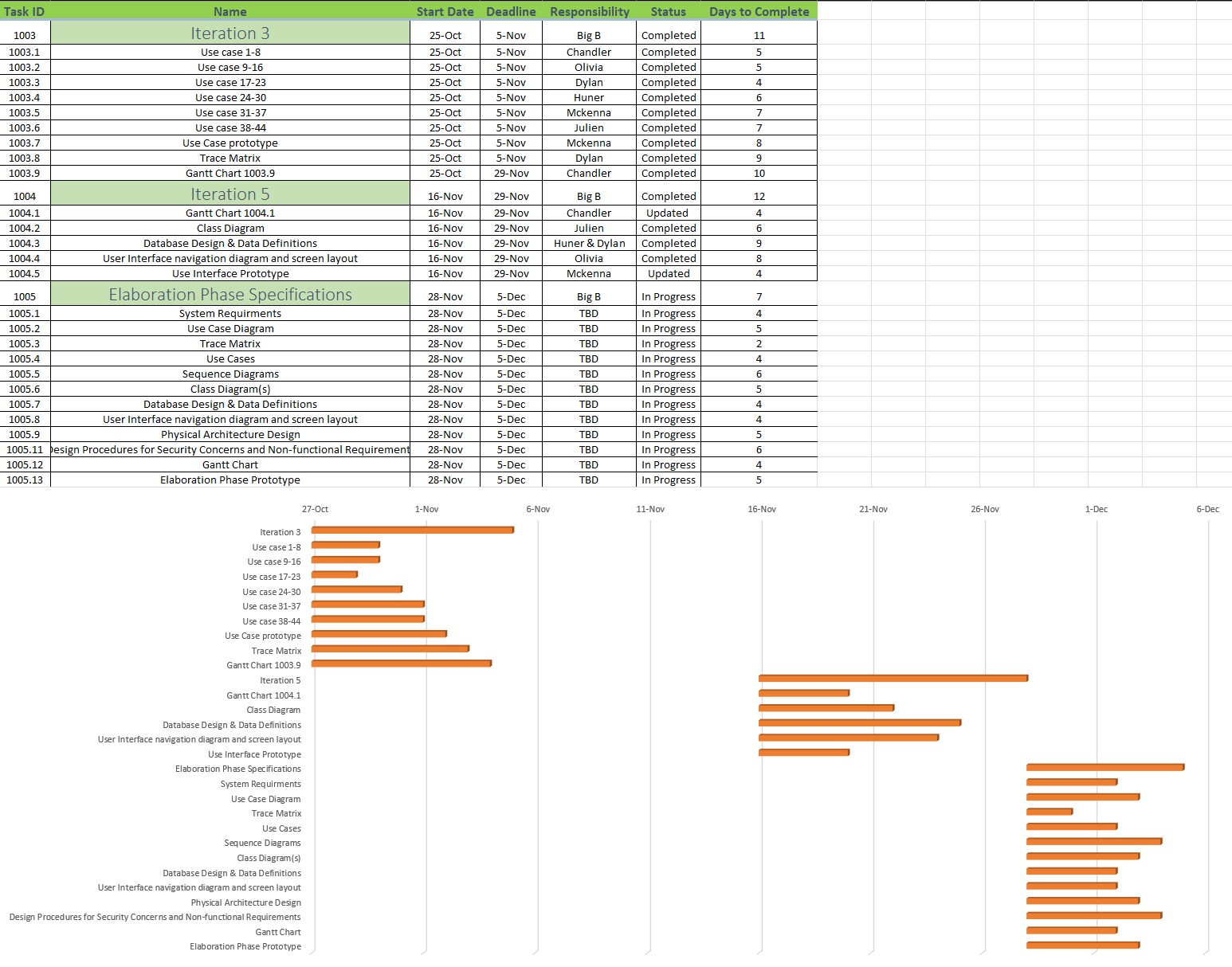
**Window Navigation Diagram:**

The window navigation diagram is a type of user interface diagram that allows us to model our different use cases and show the usability of our system. In our window navigation diagram, we depicted our Home Screen with all of the buttons that will appear right on the top banner of the screen when a user pulls it up. From these buttons, we have laid out where each button will take you, and how it will navigate you through our system. We have also described the different windows and forms that will be put to use throughout our system as well. Our buttons include login, news, volunteer, adopt, social media links, about us, shop, and donate.



**Gantt Chart:**

The Gantt Chart is a tool to help project managers track the time it takes for each task to be completed. It also shows which tasks are being worked on at the same time and if tasks are moving along faster than expected. In our chart, we have the deliverables for Iteration Three, Iteration 5, and Elaboration Phase Specifications. The days to complete come from the syllabus, it is the count of days from each due date. Team Responsibility is pulled from when each member chooses a deliverable to work on during the processes. Lastly, the status is based on the completion of a task, when it is Updated, the task has gained new information, In-progress means the task is still open and taking in new information and completed is when the information is finished, and the task is closed out of.



**User Interface Prototype:**

