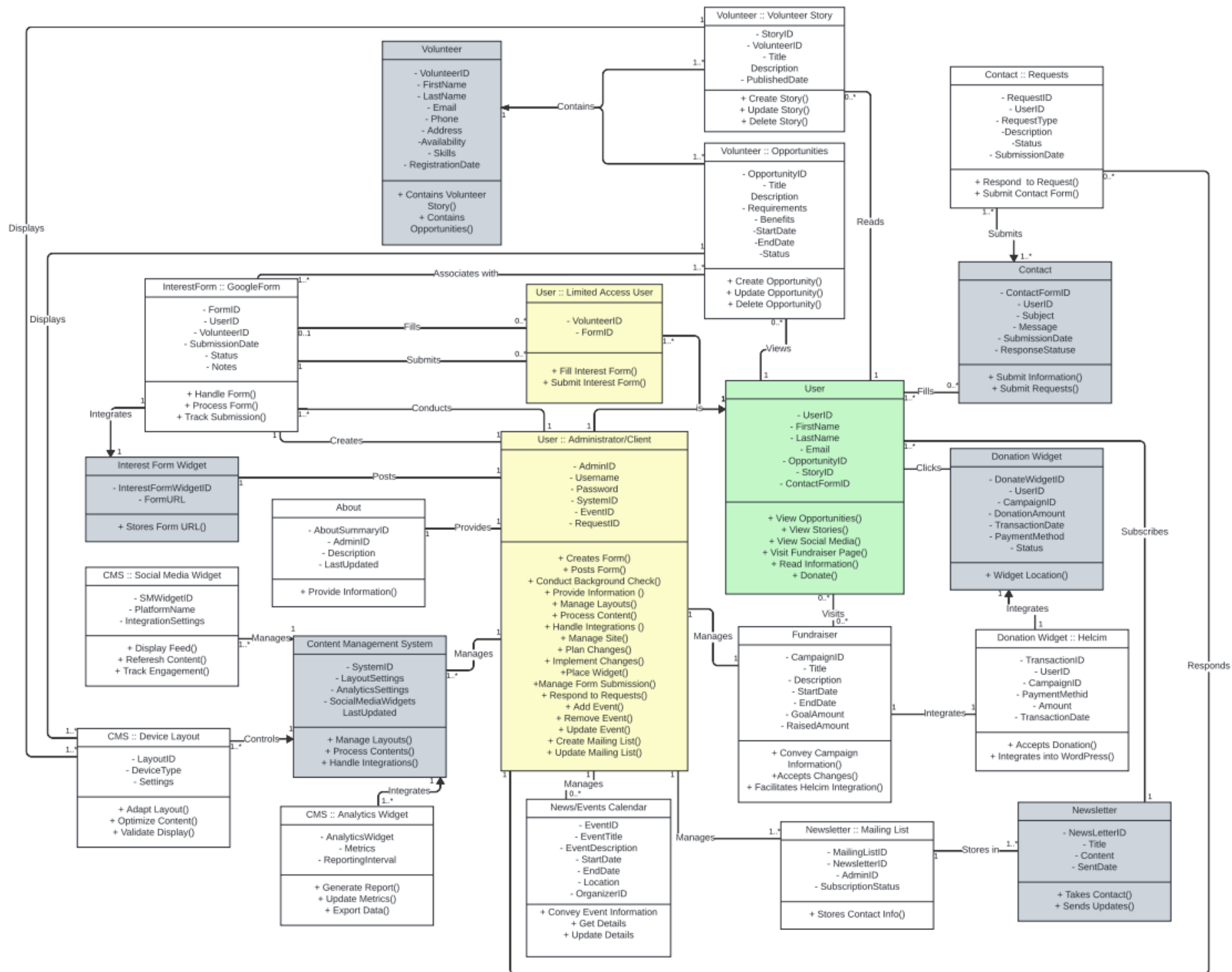




Class Diagram Narrative

The following Class Diagram shows all the necessary and major classes needed for our website. This diagram contains relationships, functions, connectivity, and attributes for the classes in this system. The best place to start in this diagram is the colored classes specifically the green User class. In the center, the User class is all users (limited access user and administrator/client) that have access to all tabs on a website which are the volunteer stories and opportunities, contact form, the donation widget, newsletter, and fundraiser. The yellow classes are the sub/child classes of the User class which are Limited Access User and Administrator/Client which are located in the center. These subclasses will have all of the User attributes as well as their attributes that the other subclass can't have. The grey classes are also parent classes that have their subclasses; these classes are the Interest Form Widget, Volunteer, Contact, Donation Widget, Newsletter, and Content Management System.

Starting in the top left, the Administrator/Client and Limited Access User class have relationships with the Interest Form/Google Form which is a subclass of the Interest Form Widget. The Administrator/Client can create an interest form/Google Form, then post the form's URL to the Interest Form Widget. A Limited Access User is a user intending to become a volunteer, so there will be a relationship between this class and the Google Form class. A Limited Access User will be able to fill out and submit an interest form/Google Form. The Admin/Client will also be able to view interest form submissions and conduct background checks for Limited Access Users who want to become a volunteer.

As mentioned before, the User (parent) class has access to all tabs on a website. The Volunteer parent class contains Volunteer Story and Opportunity subclasses that a user can access; a user can read Volunteer Stories and view Volunteer Opportunities. Specifically, the Opportunities class associated with the Interest Form/Google Form which integrates with the Google Form. Users can fill out a Contact Form which will be submitted to the Admin/Client who has access to view requests and respond to submitted requests.

Moving toward the right side of the diagram, the Donation Widget integrates to Helcim when a User clicks this widget. The Helcim class also connects to the Fundraiser class that Users can visit. In contrast, the

Admin/Client manages the Fundraiser page conveying information regarding campaigns and users can integrate to Helcim.

Users and Admin/Client have relationships with the Newsletter class. The Newsletter will send updates of the organization to all subscribed users. A User can subscribe to this Newsletter and their contact information will be stored in the Mailing List which is a subclass of the Newsletter. The Admin/Client has a relationship with the Mailing List; the Admin/Client manages the Mailing List, updating this list if it is needed.

The Admin/Client manages the Content Management System class (parent) of the subclasses Social Media Widget, Device Layout, and Analytics Widget showing the relationships between all five classes located in the bottom left. Through the Admin/Client, the CMS manages the Social Media Widget which will allow integration into a social media platform. The CMS class has a relationship with the Analytics Widget; the CMS handles data and the integration of the Analytics Widget. Lastly, the Device Layout is the different ways the website and content are presented on different devices, so in this case, the CMS class controls the Device Layout class. As said, the Device Layout class will be able to display content on different devices, so it will have relationships with the Volunteer Story and Opportunities. The Device Layout can display both the Volunteer Story and Opportunity classes.

Lastly, the Admin/Client class can provide information on Worth the Words in the About class giving a brief summary of the Organization and our Client.

Constructing this class diagram took teamwork from the whole team. The class diagram itself was derived using noun/verb analysis. First, we created all the system requirements using all the possible information each member found for our system, then using these requirements we created our use cases which used the verb/noun format (Create Interest Form). The classes were established using all the nouns that were in each use case. After gathering all the classes, they were given assigned attributes. The methods (operations) each class has were established from the verbs from the use cases. Using the operations, relationships will form describing the association between classes. As the diagram kept forming and getting larger, attributes were added to the table if it was needed. There were a few problems when building this diagram; some classes and

their attributes had to be either removed or done again due to some classes having incorrect attributes, classes that weren't relevant, and required associations that weren't included. Though this is only the first draft of the class diagram, we will make changes over time using the comments from our project sponsor and add the changes to our Elaboration Phase Specification.