

CIS 320-01

Dr. Robert Barker

May 1, 2018



Elaboration Spec

FINE EQUINE:

Abdulaziz Arrak, Madeline Cowgill, Diana Durig,
Vin Lam, Shivani Patel, and Henry Wang

Table of Contents

System Requirements.....	2
Use Case Diagrams.....	4
Trace Matrix.....	11
Use Cases & Sequence Diagrams.....	13
Class Diagram.....	97
Database Design Decisions.....	99
Data Dictionary.....	100
Entity Relationship Diagram.....	108
User Interface Navigation Diagram.....	110
Screen layouts	112
Physical Architecture Design Decisions.....	118
Physical Architecture Diagram.....	119
Design Procedures for Security Concerns and Non-Functional Requirements...	120
Gantt Chart.....	125
Prototypes.....	126

System Requirements

The system requirements are the requirements that we have identified the proposed system to need. Included are all potential requirements that we identified while collecting information from HOOF.

Functional requirements

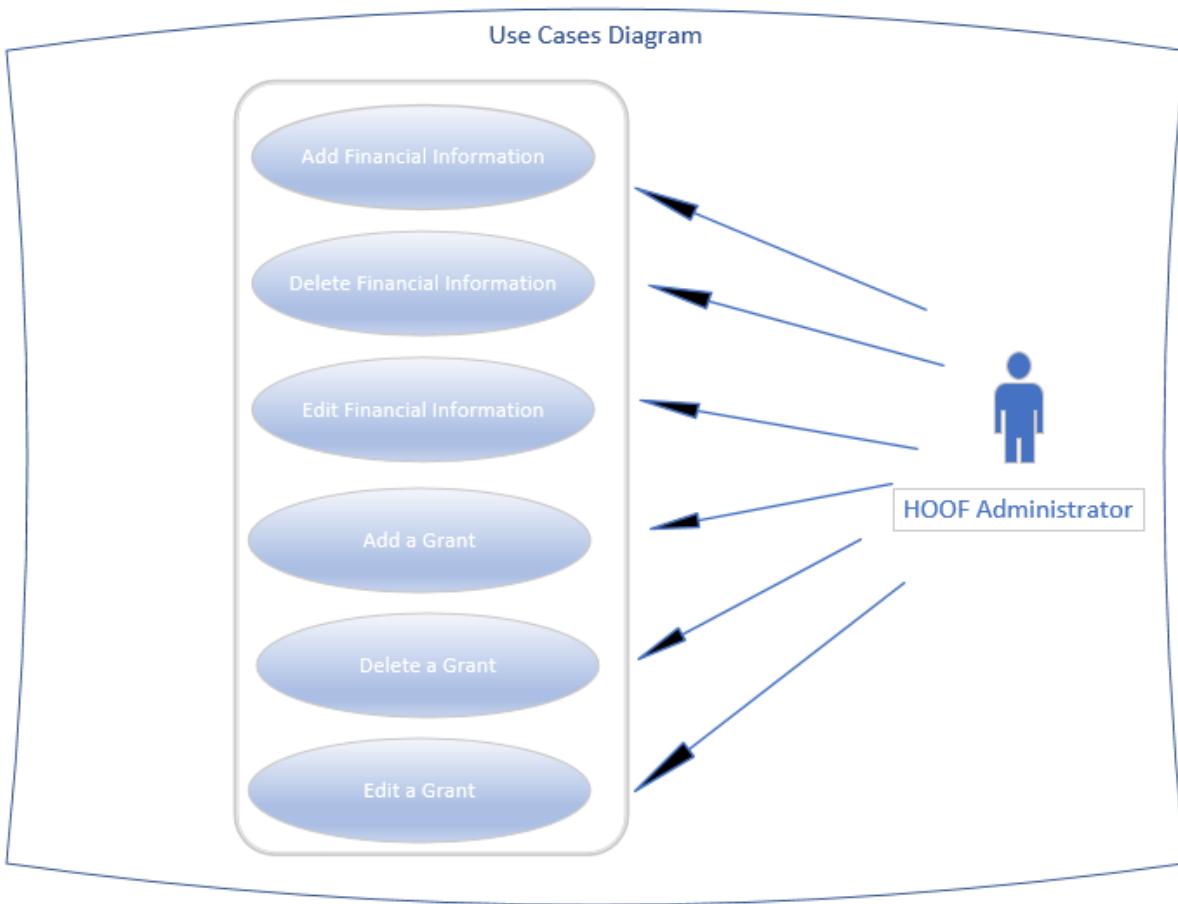
1. Manage Volunteers data:
 - a. The system shall collect volunteers information.
 - b. The system shall store volunteers information.
 - c. The system shall be able to delete/modify volunteer information.
2. Manage participants data:
 - a. The system shall track participants information.
 - b. The system shall store participants information.
 - c. The system shall be able to delete/modify participants information.
3. Manage donors data:
 - a. The system shall collect donor information.
 - b. The system shall store donor information.
 - c. The system shall be able to delete/modify donor information.
4. Produce events:
 - a. The system shall allow Auction director to manage events.
 - b. The system shall allow director of public relations to manage public relationships and marketing.
 - c. The system shall support the social media features
5. Manage website:
 - a. The HOOF administrators shall be able to add content on the website.
 - b. The HOOF administrators shall be able to update content on the website.
 - c. The HOOF administrators shall be able to delete/modify content on the website.

Non-functional requirement

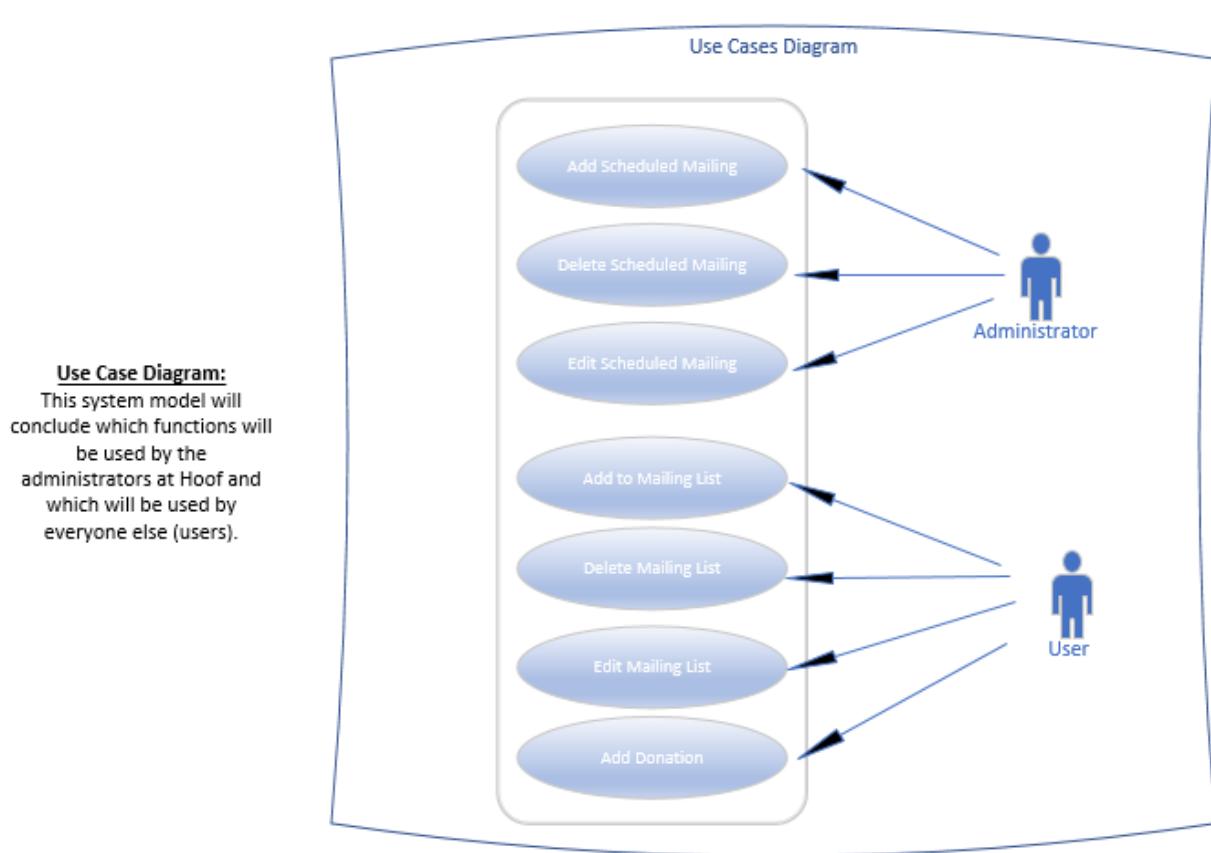
1. Operational requirements
 - a. The system shall operate in windows environment.
 - b. The system should be able to connect to peripheral devices.

- c. The system should have the capability to perform routine backup.
 - d. The system shall centralize data that can be stored on a cloud platform.
 - e. The system shall use Microsoft office suite application for basic business processes.
2. Performance requirements.
 - a. The system shall store new applications.
 - b. The system shall retrieve new applications.
 - c. The system shall allow data to be easily shared amongst users.
 - d. The system should be able to track volunteers hours
 - e. The system shall allow users to collect financial data via Excel
 3. Security requirements.
 - a. Users shall be granted permissions as operate.
 - b. All devices should be password protected.
 - c. The system should be able to track users activities on the website

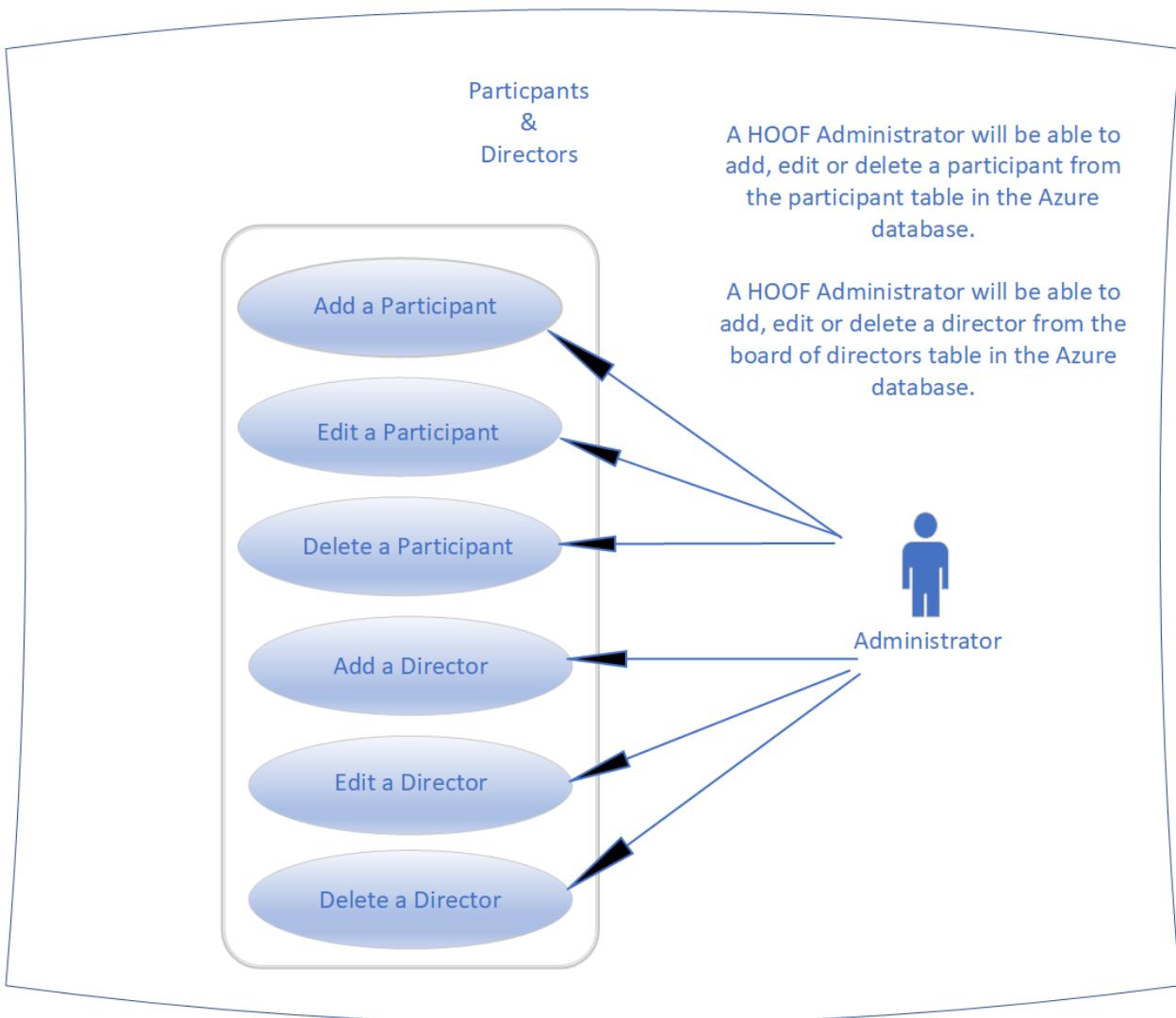
Financial Information and Grants Use Case Diagram



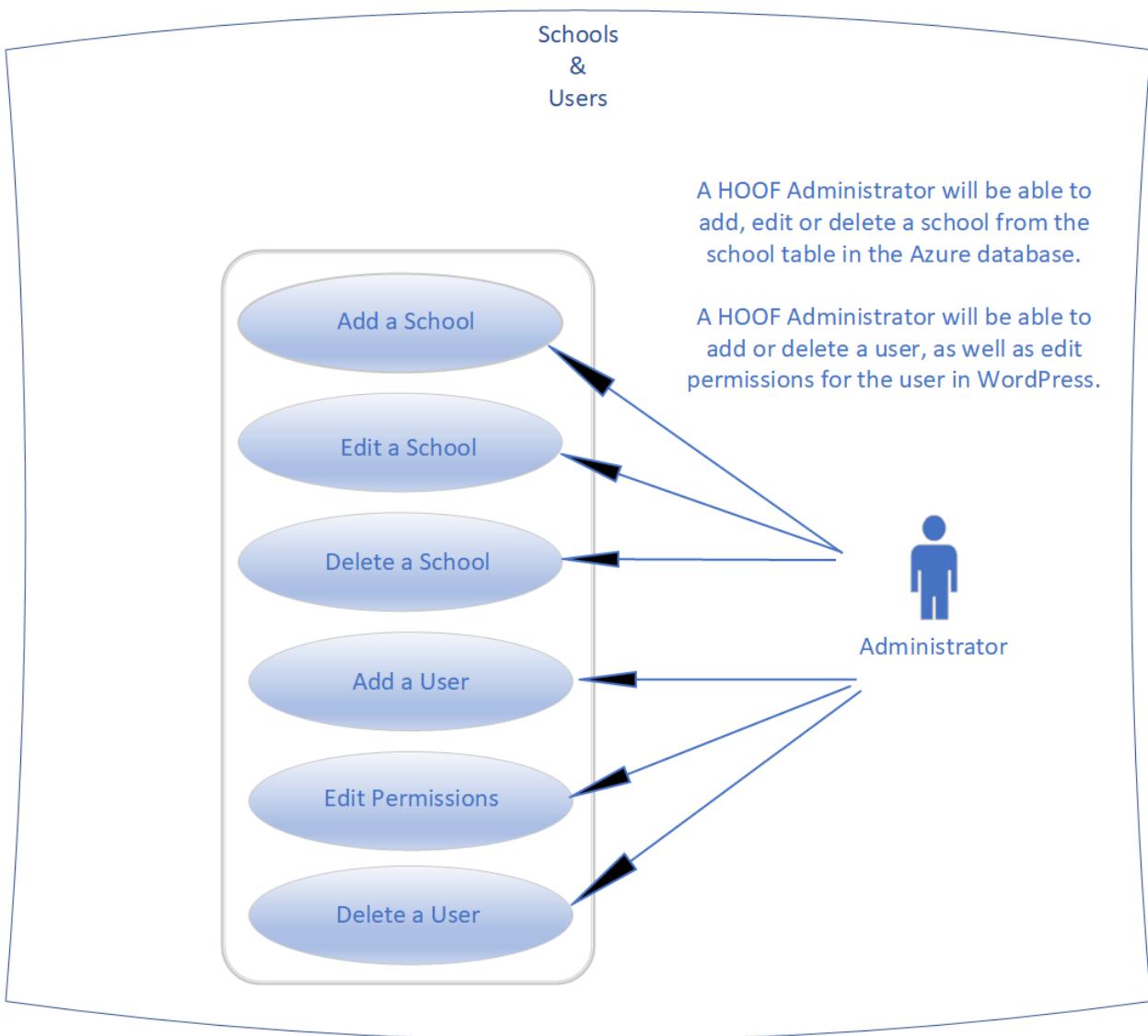
Scheduled Mailing and Mailing List Use Case Diagram



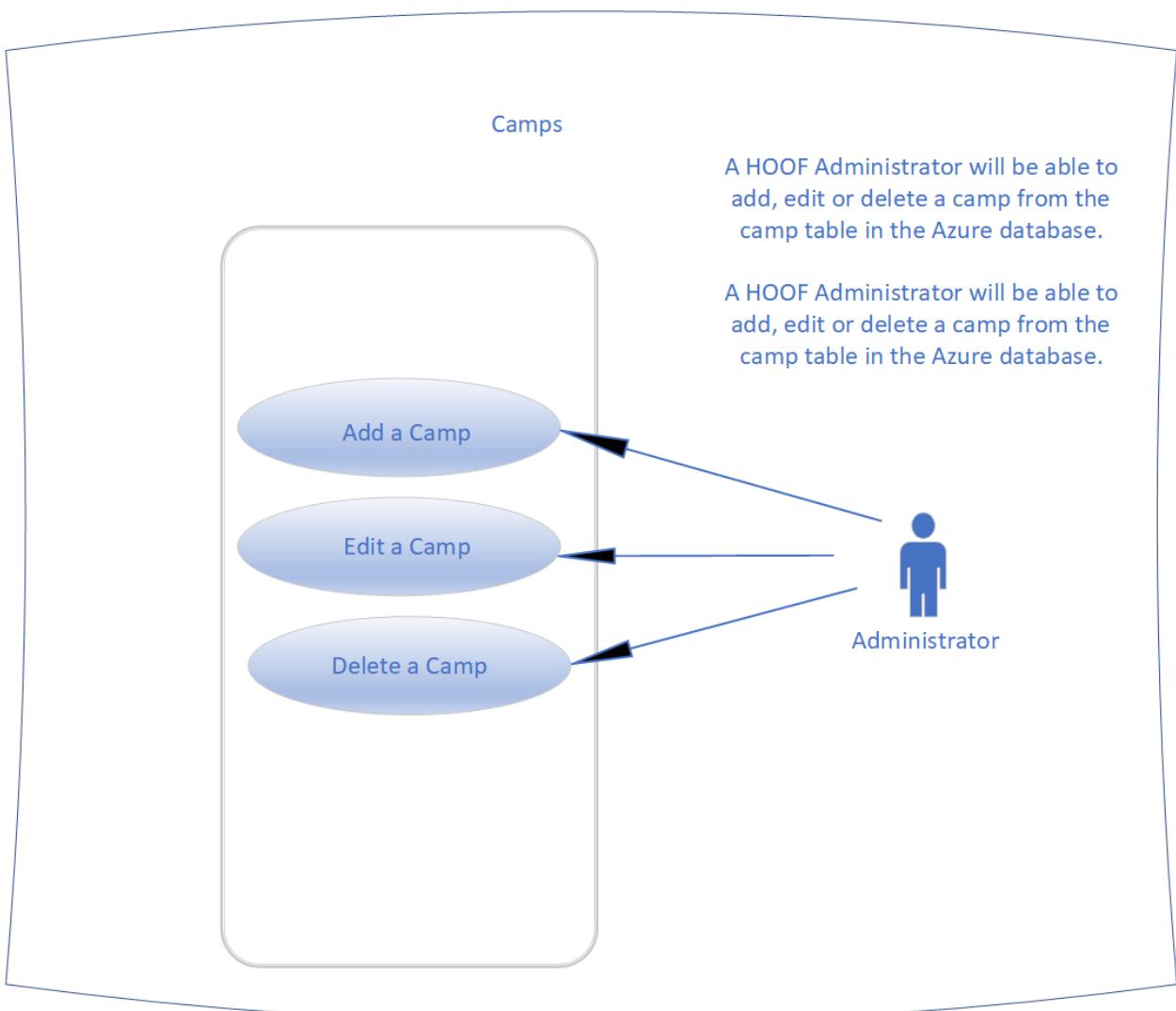
Participant and Director Use Case Diagram



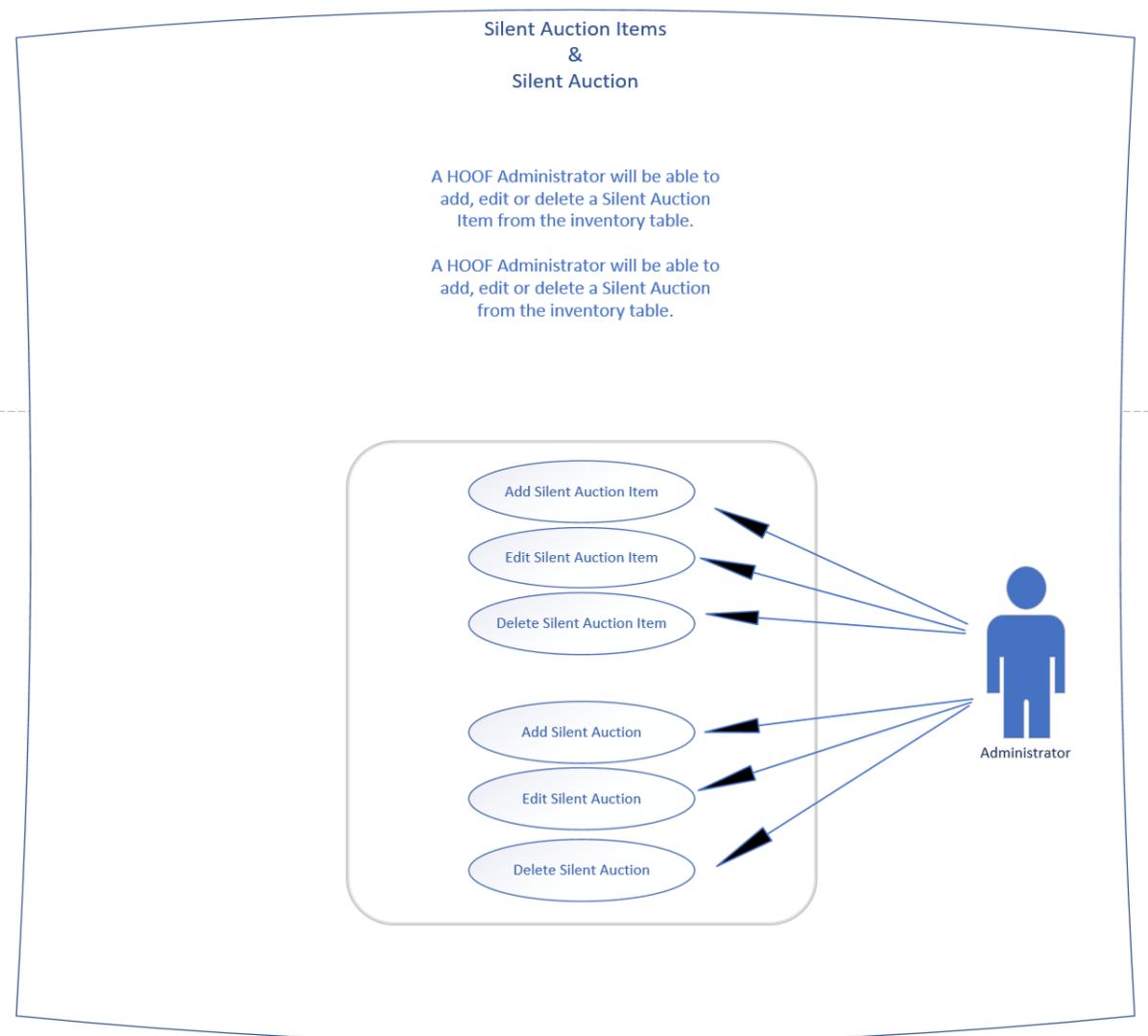
School and User Use Case Diagram



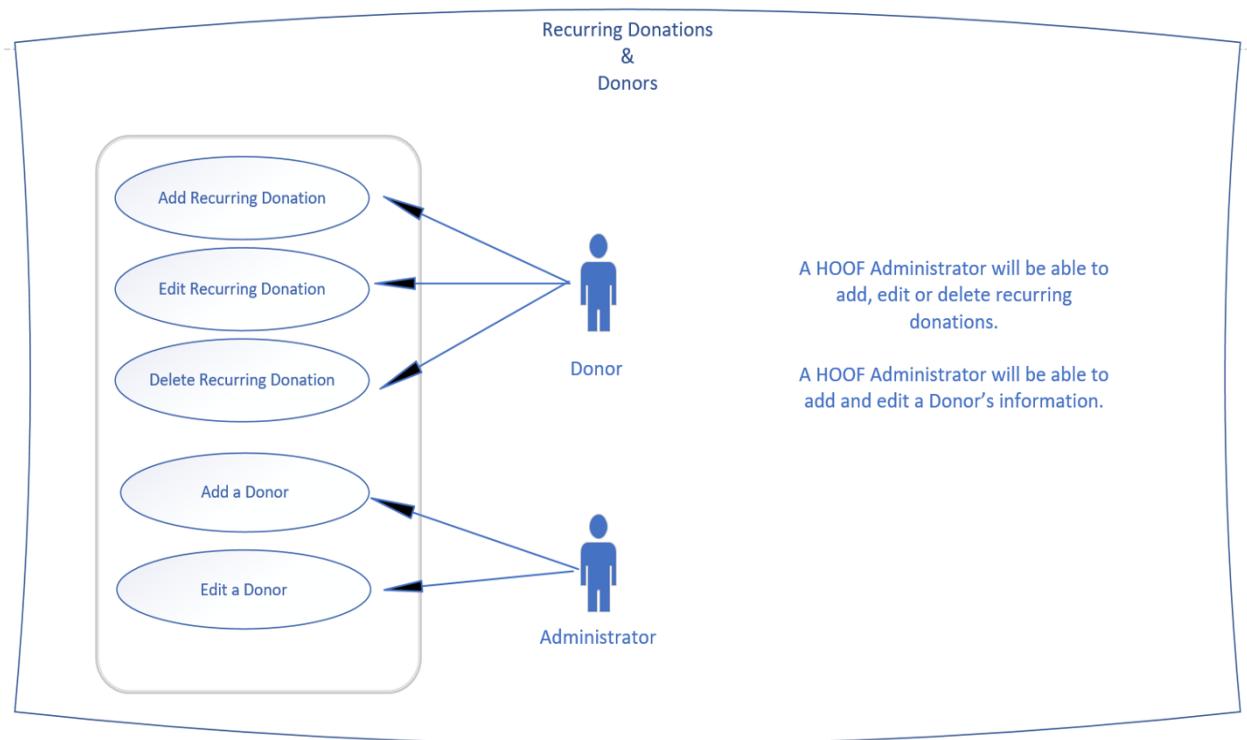
Camp Use Case Diagram



Silent Auction Items and Silent Auction



Recurring Donations and Donors



Trace Matrix

A trace matrix ensures that the project's scope, requirements and deliverables are within the business rules and are satisfied. This trace matrix shows system requirements are met by the high risk use cases. The requirements addressed are denoted by an x within the spreadsheet.

HIGH RISK USE CASES:	Store Donations	Store Donor Information	Excel for Financials	Add Applications	Update Applications
<i>Add a Donation</i>	x	x	x		
<i>Add a Donor</i>	x	x	x		
<i>Edit a Donor</i>	x	x	x		
<i>Add Grant Application</i>			x		x
<i>Update a Grant Application</i>			x	x	
<i>Add Silent Auction Item</i>				x	
<i>Update Silent Auction Item</i>				x	
<i>Add Silent Auction</i>				x	
<i>Update Silent Auction</i>				x	
<i>Add a Recurring Donation</i>	x	x	x		
<i>Drop a Recurring Donation</i>	x	x	x		
<i>Update a Recurring Donation</i>	x	x	x		
<i>Add an Event</i>					
<i>Update an Event</i>					
<i>Login</i>					

HIGH RISK USE CASES:	Manage Events	Support Social Media	Manage Public Relations	Manage Marketing	Use of Windows OS
<i>Add a Donation</i>					x
<i>Add a Donor</i>					x
<i>Edit a Donor</i>					x
<i>Add Grant Application</i>					x
<i>Update a Grant Application</i>					x
<i>Add Silent Auction Item</i>					x
<i>Update Silent Auction Item</i>					x
<i>Add Silent Auction</i>	x	x	x	x	x
<i>Update Silent Auction</i>	x	x	x	x	x
<i>Add a Recurring Donation</i>					x
<i>Drop a Recurring Donation</i>					x
<i>Update a Recurring Donation</i>					x
<i>Add an Event</i>	x	x	x	x	x
<i>Update an Event</i>	x	x	x	x	x
<i>Login</i>					

HIGH RISK USE CASES:	Add Auctions	Update Auctions	Grant Permissions	Manage Website Content	Store Volunteer Info
Add a Donation					x
Add a Donor					x
Edit a Donor					
Add Grant Application			x		
Update a Grant Application			x		
Add Silent Auction Item	x				
Update Silent Auction Item		x			
Add Silent Auction	x			x	
Update Silent Auction		x		x	
Add a Recurring Donation					
Drop a Recurring Donation					
Update a Recurring Donation					
Add an Event				x	x
Update an Event				x	x
Login					

HIGH RISK USE CASES:	Use of Cloud Database	Devices are Password Protected	Securely Share Data	Routine Backups	Use of MS Office
Add a Donation	x				x
Add a Donor	x				x
Edit a Donor	x				x
Add Grant Application	x		x	x	x
Update a Grant Application	x		x	x	x
Add Silent Auction Item	x				x
Update Silent Auction Item	x				x
Add Silent Auction	x				x
Update Silent Auction	x				x
Add a Recurring Donation	x				
Drop a Recurring Donation	x				
Update a Recurring Donation	x				
Add an Event	x				
Update an Event	x				
Login		x			

HIGH RISK USE CASES:	Devices are Password Protected	Securely Share Data	Routine Backups	Use of MS Office	Website Activity Log
Add a Donation				x	
Add a Donor				x	
Edit a Donor				x	
Add Grant Application		x	x	x	
Update a Grant Application		x	x	x	
Add Silent Auction Item				x	
Update Silent Auction Item				x	
Add Silent Auction				x	
Update Silent Auction				x	
Add a Recurring Donation					
Drop a Recurring Donation					
Update a Recurring Donation					
Add an Event					
Update an Event					
Login	x				x

Use Case Specification: Add a Grant Application

1. Add a Grant Application

2. Brief Description

A HOOF administrator or designated grant writer will be able to add a grant to the database.

3. Flow of Events

4. Basic Flow

- Input name of grant
- Input grant contact information
- Input grantor or agency
- Input grant writer first name
- Input grant writer last name
- Input grant writer middle initial
- Input grant amount requesting
- Input grant due date
- Input grant start date
- Input grant end date
- Check box whether complete, in progress, or funded
- Input grant amount awarded
- Input grant description
- Upload documents
- Click submit

4.1 Alternative Flows

<Null Error Message>

- Any fields left null other than the middle initial, an error message will be displayed to the user prompting them to input a value into each required field.

5. Special Requirements

- Must have admin access
- Must have grant writing permissions
- Must have login access
- Must have internet access

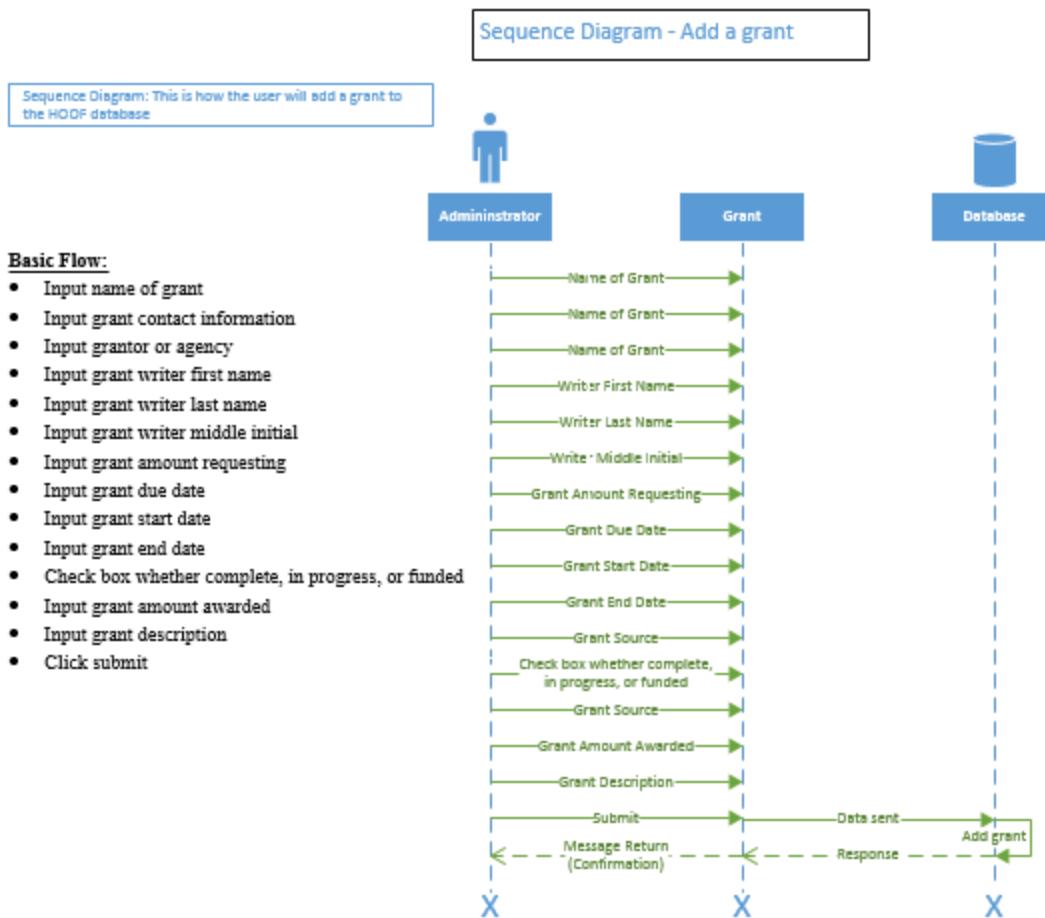
6. Pre-conditions

- Administrator/Writer has navigated to HOOF's website.
- Admin must be logged into Azure database

7. Post-conditions

- Grant application has been added to the database.

Add a Grant Sequence Diagram



Use Case Specification: Edit a Grant Application

1. Edit a Potential Grant

2. Brief Description

A HOOF administrator or designated grant writer will be able to edit a potential grant to the list.

3. Flow of Events

4. Basic Flow

- Edit name of grant
- Edit grant contact information
- Edit grantor or agency
- Edit grant writer first name
- Edit grant writer last name
- Edit grant writer middle initial
- Edit grant amount requesting
- Edit grant due date
- Edit grant start date
- Edit grant end date
- Check box whether complete, in progress, or funded
- Edit grant amount awarded
- Edit grant description
- Upload documents
- Click submit

5. Special Requirements

- Must have admin access
- Must have grant writing permissions
- Must have login access
- Must have internet access

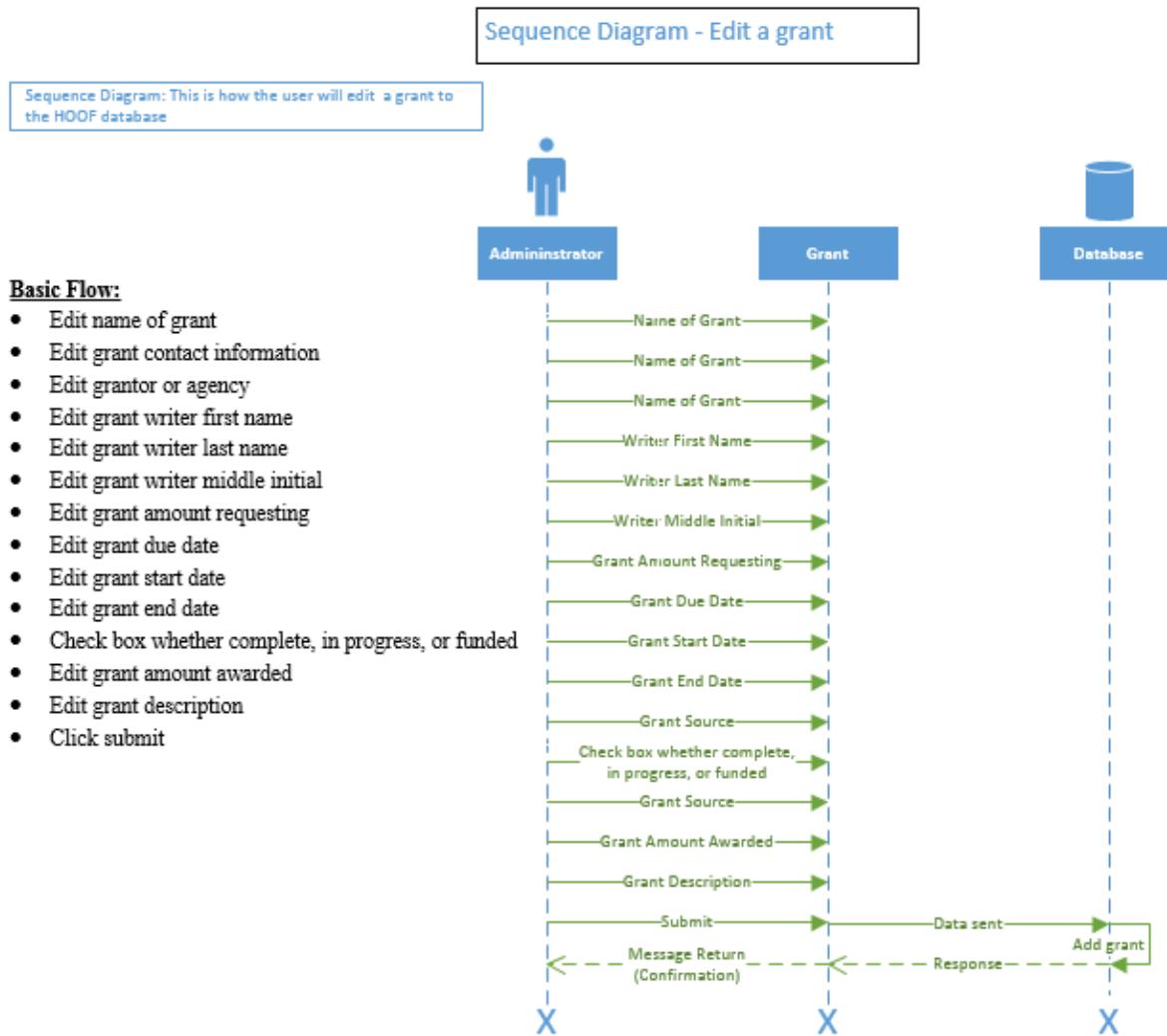
6. Pre-conditions

- Administrator/Writer has navigated to HOOF's website.
- Admin must be connected to Azure database

7. Post-conditions

- Potential grant application has been modified in the database.

Edit a Grant Sequence Diagram



Use Case Specification: Delete a Grant Application

1. Delete a Grant Application

2. Brief Description

A HOOF administrator or designated grant writer will be able to delete a grant from the database.

3. Flow of Events

4. Basic Flow

- Select grant
- Click on delete icon
- Click submit

5. Special Requirements

- Must have admin access
- Must have grant writing permissions
- Must have login access
- Must have internet access

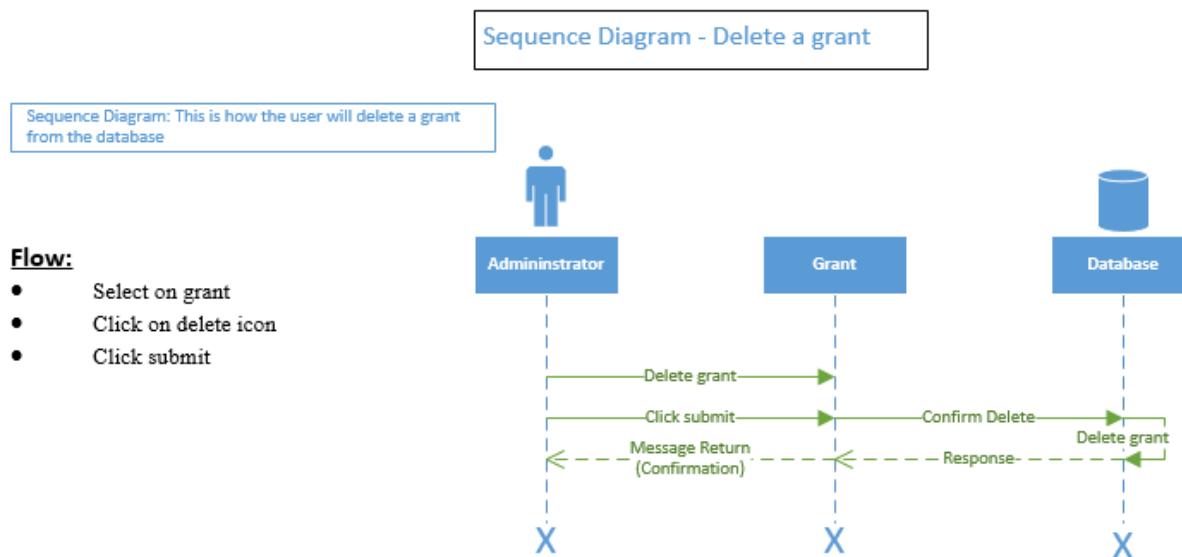
6. Pre-conditions

- Administrator/Writer has navigated to HOOF's website.
- Admin must be logged into Azure database

7. Post-conditions

- Grant Application has been deleted from the database.

Delete a Grant Sequence Diagram



Use Case Specification: Add to Mailing List

1. Add to Mailing List

2. Brief Description

A hoof administrator will be able to add a new member to the mailing list.

3. Flow of Events

- Input member's first name
- Input member's middle initial
- Input member's last name
- Input member's title
- Input member's email address
- Input member's street address
- Input member's city
- Input member's state
- Input member's zip code
- Input member's school or organization affiliations
- Member clicks the submit button
- An update query is ran to store a new member to the mailing list

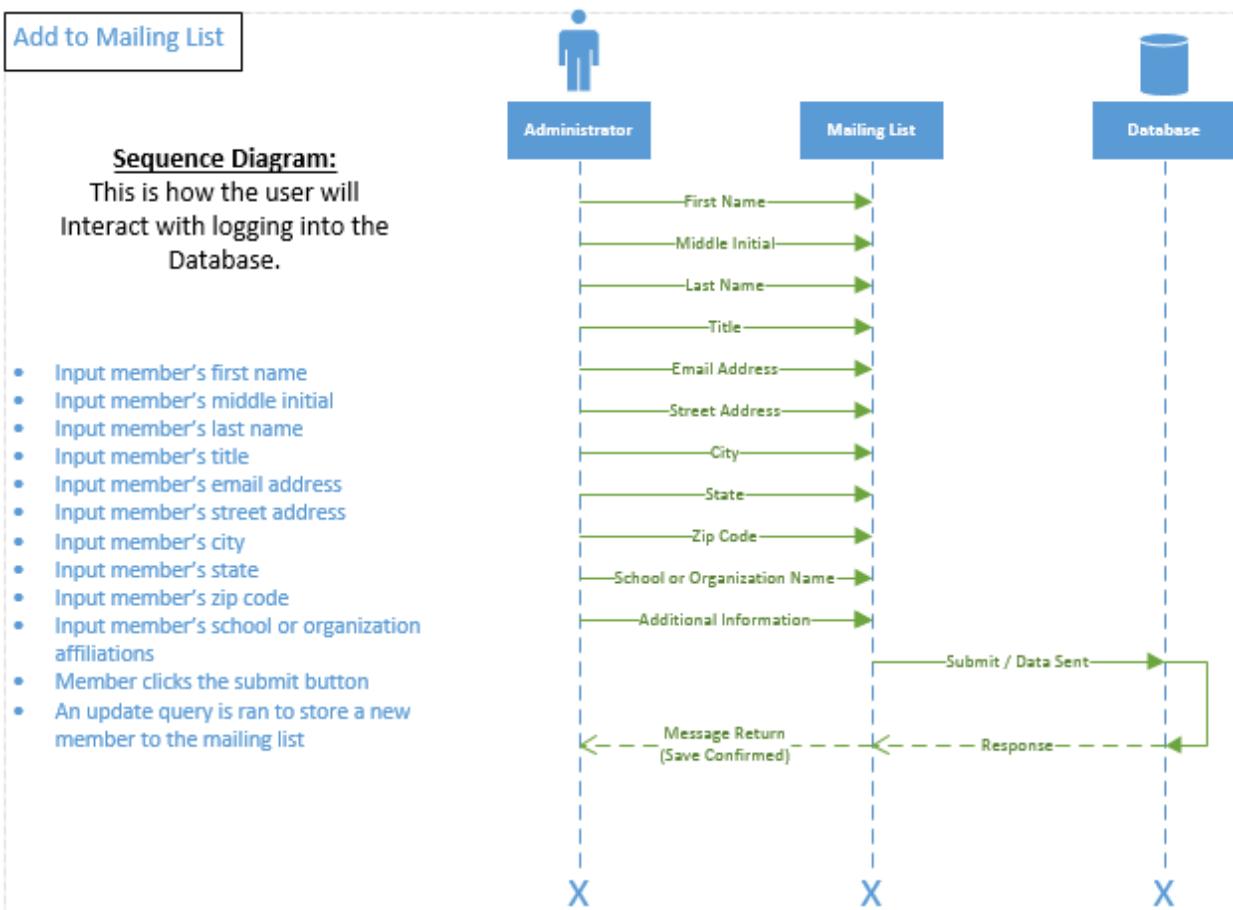
4. Pre-conditions

- Member must use a computer with internet access
- Member must open HOOF.org webpage and click on Mailing List
- Member will input their data onto the website and complete registration for the mailing list

5. Post-conditions

- Member must save submission to mailing list on webpage
- The new user will receive an email at the previously specified email address to confirm subscription to mailing list

Add a Mailing List Sequence Diagram



Use Case Specification: Edit Mailing List

1. Edit Mailing List

2. Brief Description

A hoof administrator will be able to edit information on the mailing list.

3. Flow of Events

4. Basic Flow

- Edit member's first name
- Edit member's last name
- Edit member's middle initial
- Edit member's email address.
- Edit member's street address
- Edit member's city
- Edit member's state
- Edit member's zip code
- Edit member's phone number
- Edit member's school or organization affiliations
- User presses the update button to update the record
- An update query is ran to store updated member information

5. Special Requirements

- Administrator must use a computer with internet access
- A record for the member must exist in order to be edited

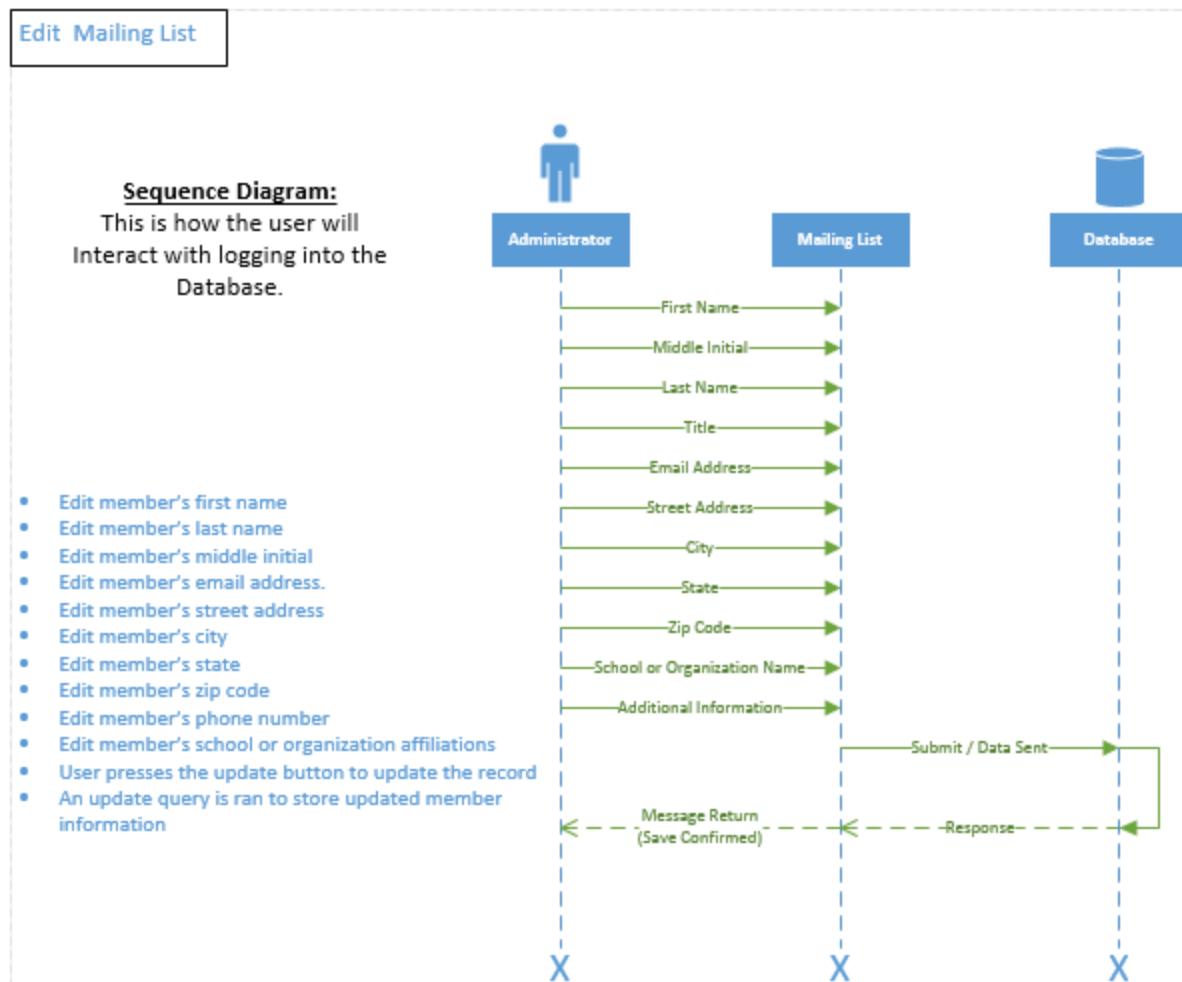
6. Pre-conditions

- Administrator must be *logged in* the Azure database in order to access the database

7. Post-conditions

- The additions to the Azure database are saved

Edit a Mailing List Sequence Diagram



Use Case Specification: Delete from Mailing List

1. Delete from Mailing List

2. Brief Description

A hoof administrator will be able to delete a contact on the mailing list.

3. Basic Flow

- Delete member's information from database.
- Administrator locates existing member record that is to be deleted
- Administrator deletes member record
- System asks Administrator to confirm the deletion of the record
- Administrator confirms deletion
- System deletes the Member Record

4. Special Requirements

- Administrator must use a computer with internet access

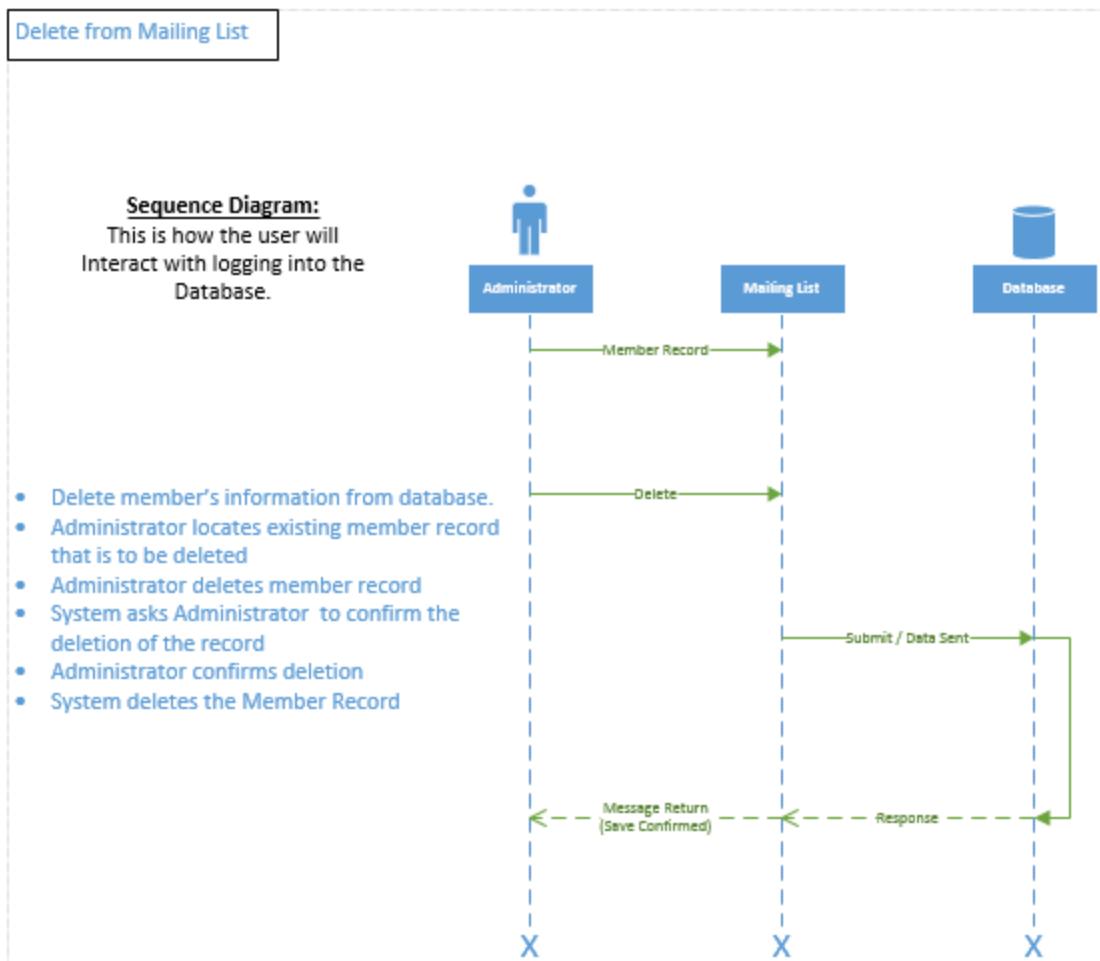
5. Pre-conditions

- Administrator must be logged into the Azure database
- Administrator must have permission to delete member's information from the database
- There must be an existing blank or filled record in order to be deleted

6. Post-conditions

- The table with deleted record(s) is saved to the Azure database

Delete from a Mailing List Sequence Diagram



Use Case Specification: Add Scheduled Mail

1. **Add Scheduled Mail**

2. **Brief Description**

A hoof administrator will be able to add a scheduled email to the mailing list.

3. **Flow of Events**

4. **Basic Flow**

- Input message title
- Input message body
- Input users to send mail to
- Upload Image
- Administrator will click send/submit to send out mail to members

5. **Special Requirements**

- Administrator must use a computer with internet access
- Administrator has navigated to the website

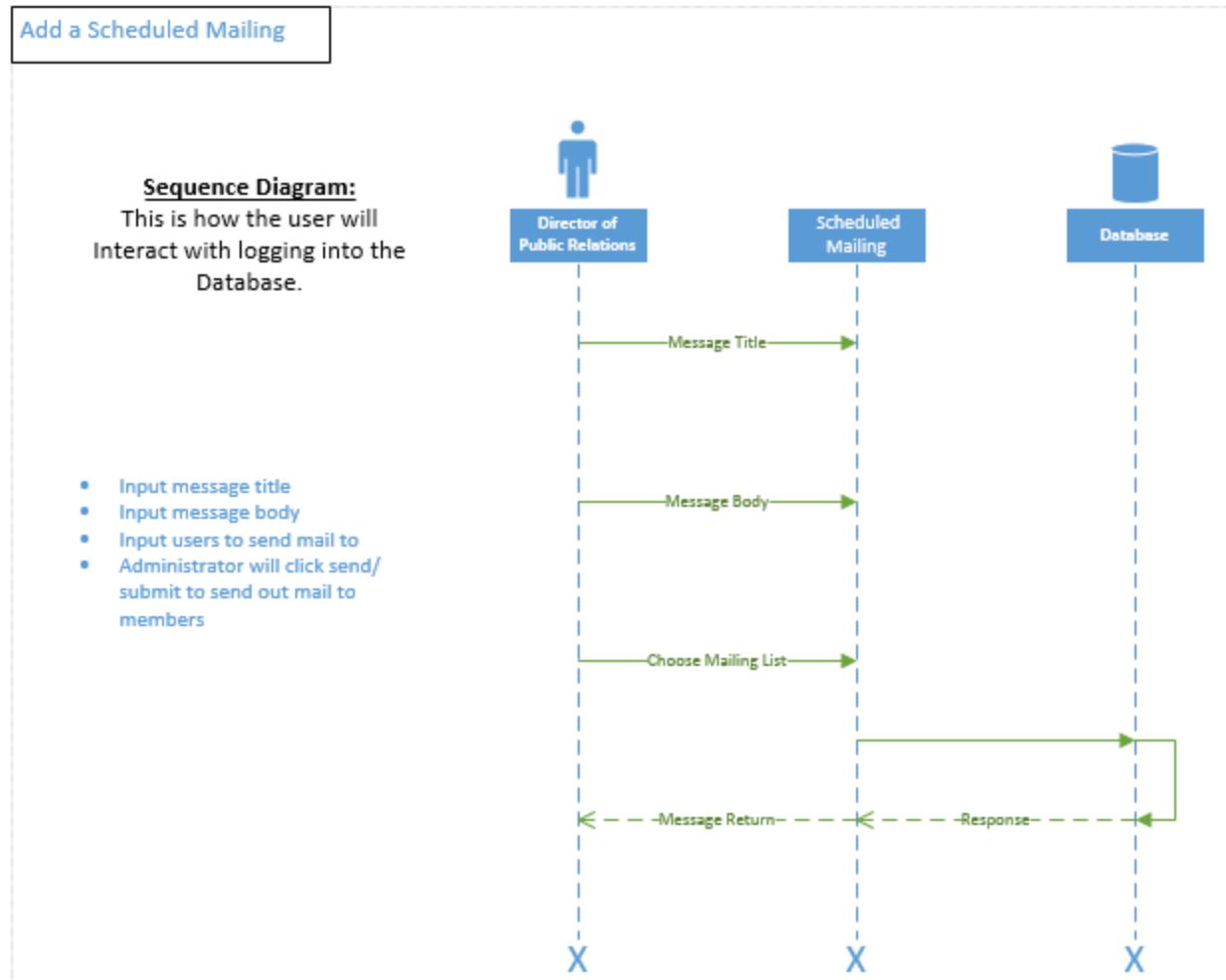
6. **Pre-conditions**

- Administrator must be logged into the wordpress

7. **Post-conditions**

- Administrator must save the additions to the azure database
- Confirmation email will be sent to administrator that mailer has been sent out

Add a Scheduled Mailing Sequence Diagram



Use Case Specification: Edit Scheduled Mail

1. Edit Scheduled Mail

2. Brief Description

A hoof administrator will be able to Edit scheduled emails to the mailing list.

3. Flow of Events

4. Basic Flow

- Edit message title
- Edit message body
- Edit mailing list
- Upload Image
- Click submit to confirm edit

5. Special Requirements

- Administrator must use a computer with internet access

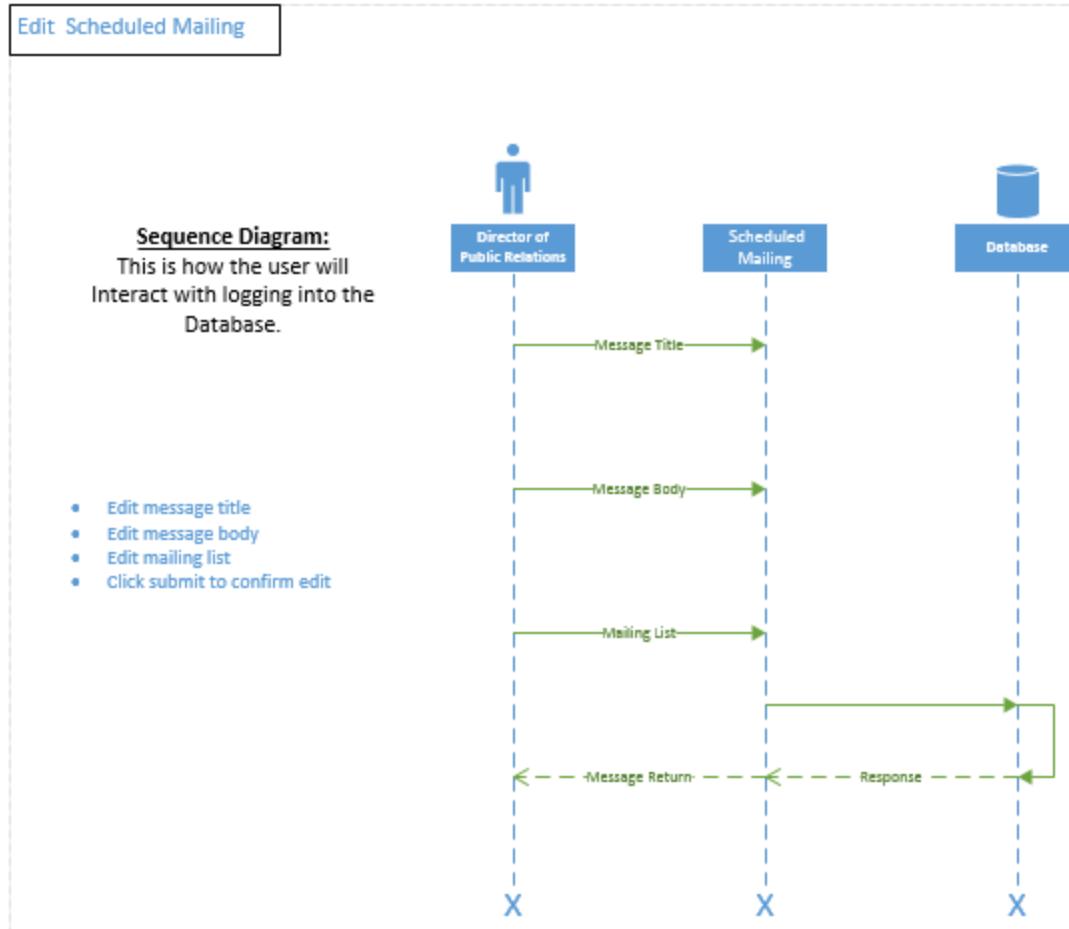
6. Pre-conditions

- Administrator must be *logged in* the Azure database in order to access the database

7. Post-conditions

- The additions to the Azure database are saved

Edit a Scheduled Mailing Sequence Diagram



Use Case Specification: Delete Scheduled Mail

1. **Delete Scheduled Mail**

2. **Brief Description**

A hoof administrator will be able to Delete a scheduled email to the mailing list.

3. **Basic Flow**

- Select scheduled mailing to delete
- Click submit to confirm deletion

4. **Special Requirements**

- Administrator must use a computer with internet access

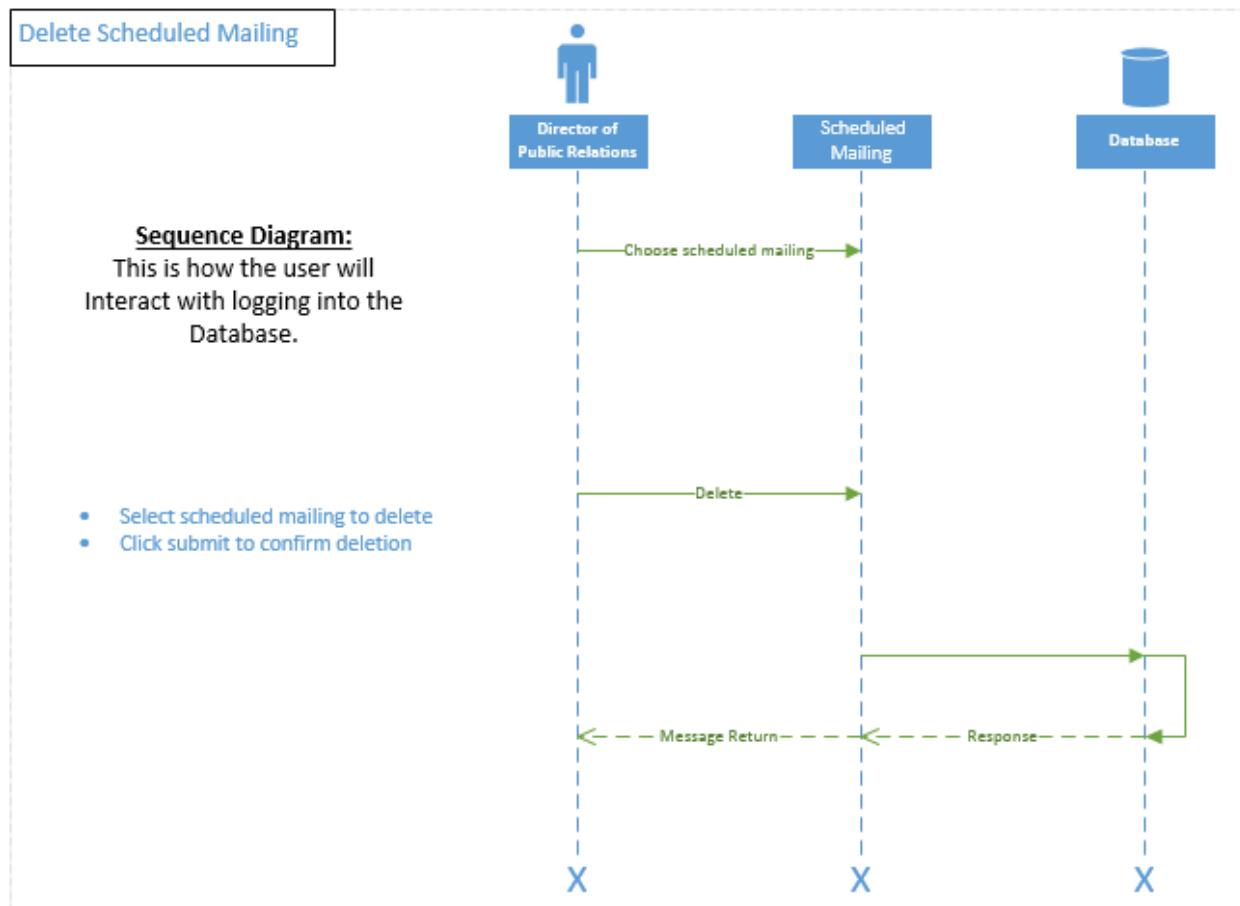
5. **Pre-conditions**

- Administrator must be logged into the Azure database

6. **Post-conditions**

- The table with deleted record(s) is saved to the Azure database

Delete Scheduled Mailing Sequence Diagram



Use Case Specification: Add a Camp

1. Add a Camp

1.1 Brief Description

A HOOF administrator will be able to add a camp to the camp table.

1. Flow of Events

2.1 Basic Flow

- Input camp id
- Input event id
- Input camp name
- Input start date
- Input end date
- Input location
- Click submit

2.2 Alternative Flows

2.2.1 < Null Start Date, End Date and Location>

- If the start date, end date or location have not been determined, they may be null.

1. Special Requirements

- Administrator must use a computer with internet access

1. Pre-conditions

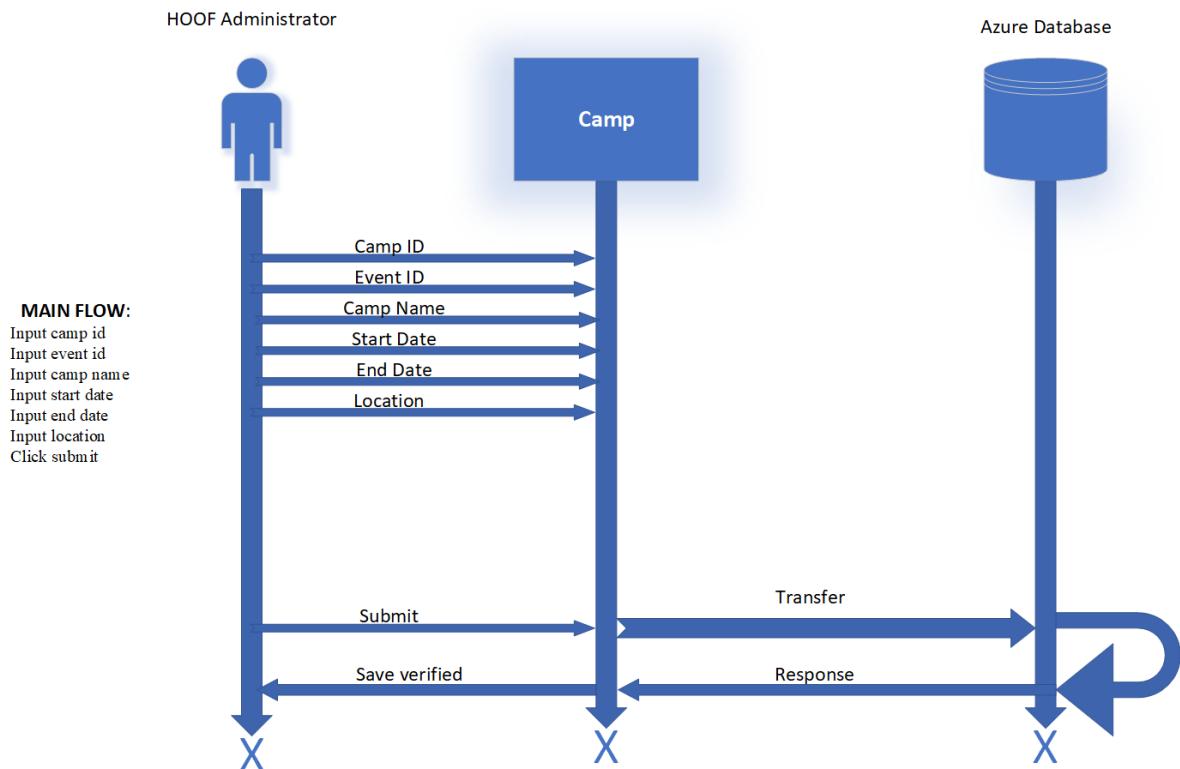
- Administrator must be *logged into* the Azure database

1. Post-conditions

- The additions to the participant table in the Azure database are saved

Add a Camp Sequence Diagram

Add A Camp



Use Case Specification: Edit a Camp

1. Edit a Camp

1.1 Brief Description

A HOOF administrator will be able to edit a camp to the camp table.

1. Flow of Events

2.1 Basic Flow

- Edit camp id
- Edit event id
- Edit camp name
- Edit start date
- Edit end date
- Edit location
- Click submit

2.2 Alternative Flows

2.2.1 < Null Start Date, End Date and Location>

- If the start date, end date or location have not been determined, they may be null.

1. Special Requirements

- Administrator must use a computer with internet access

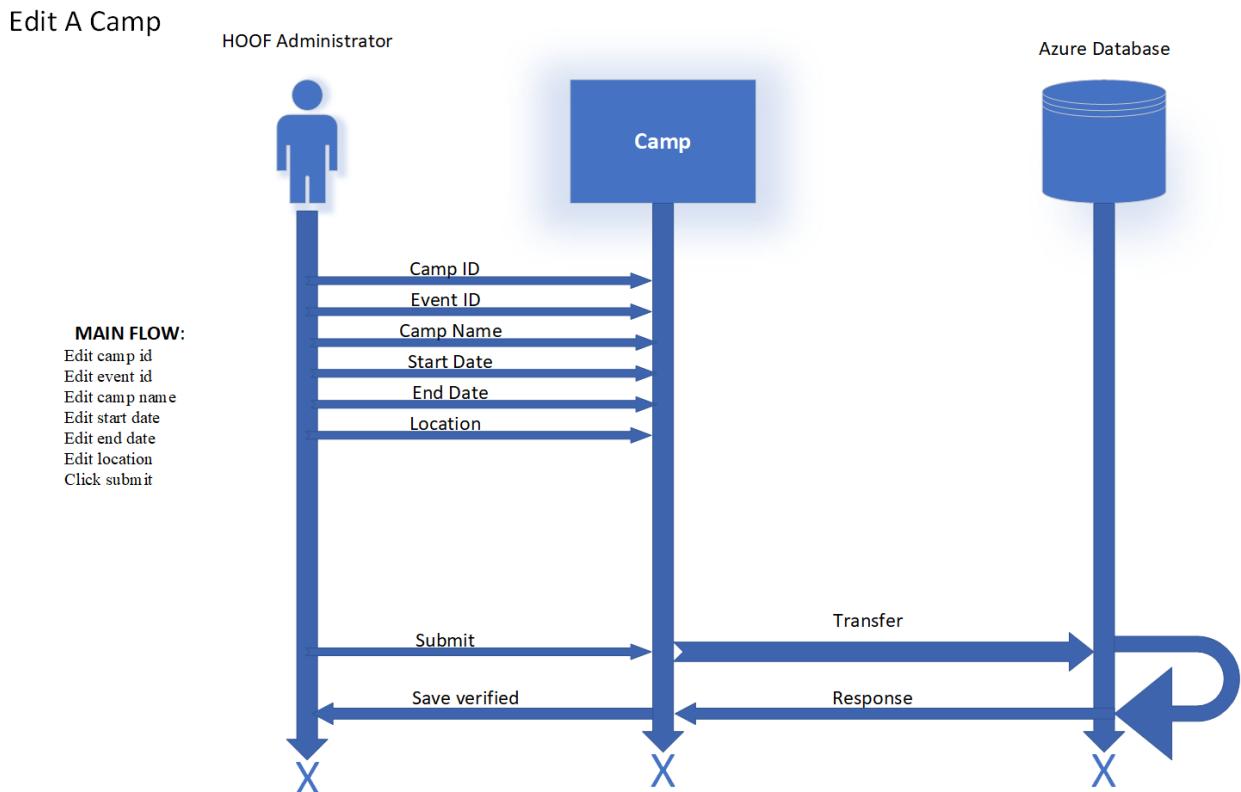
1. Pre-conditions

- Administrator must be *logged into* the Azure database

1. Post-conditions

- The additions to the participant table in the Azure database are saved

Edit a Camp Sequence Diagram



Use Case Specification: Delete a Camp

1. Delete a Camp

1.1 Brief Description

A HOOF administrator will be able to delete a camp to the camp table.

1. Flow of Events

2.1 Basic Flow

- Input camp row to be deleted in 'search for record'
- Choose record from dropdown option(s)
- Click delete record switch for 'YES'
- Click submit record

1. Special Requirements

- Administrator must use a computer with internet access

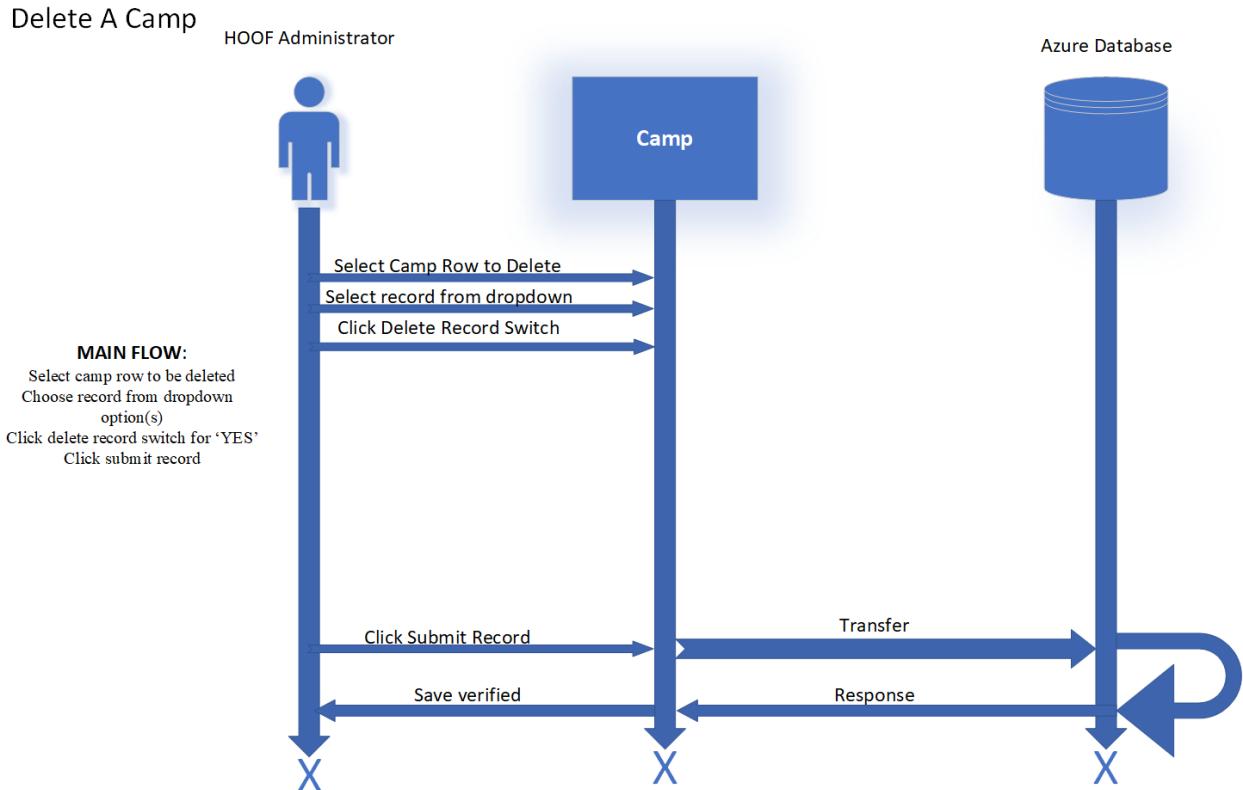
1. Pre-conditions

- Administrator must be *logged into* the Azure database

1. Post-conditions

- The additions to the participant table in the Azure database are saved

Delete a Camp Sequence Diagram



Use Case Specification: Add a Director

1. Add a Director

1.1 Brief Description

A HOOF administrator will be able to add a member to the board of directors table

1. Flow of Events

2.1 Basic Flow

- Input director id
- Input director first name
- Input director middle initial
- Input director last name
- Input director street address
- Input director city
- Input director state
- Input director zip code
- Input director phone number
- Input director email
- Input director specialties
- Click submit

2.2 Alternative Flows

2.2.1 <No Specialties>

If the new director has no specialties, this field can be null.

2.2.2 <Null Middle Initial>

- If there is no middle initial for the participant, the middle initial field may be null

2.2.2a <Null Error Message>

- If any fields other than middle initial or specialties are null, an error message will be displayed to the user prompting them to input a value into each required field

1. Special Requirements

- Administrator must use a computer with internet access

1. Pre-conditions

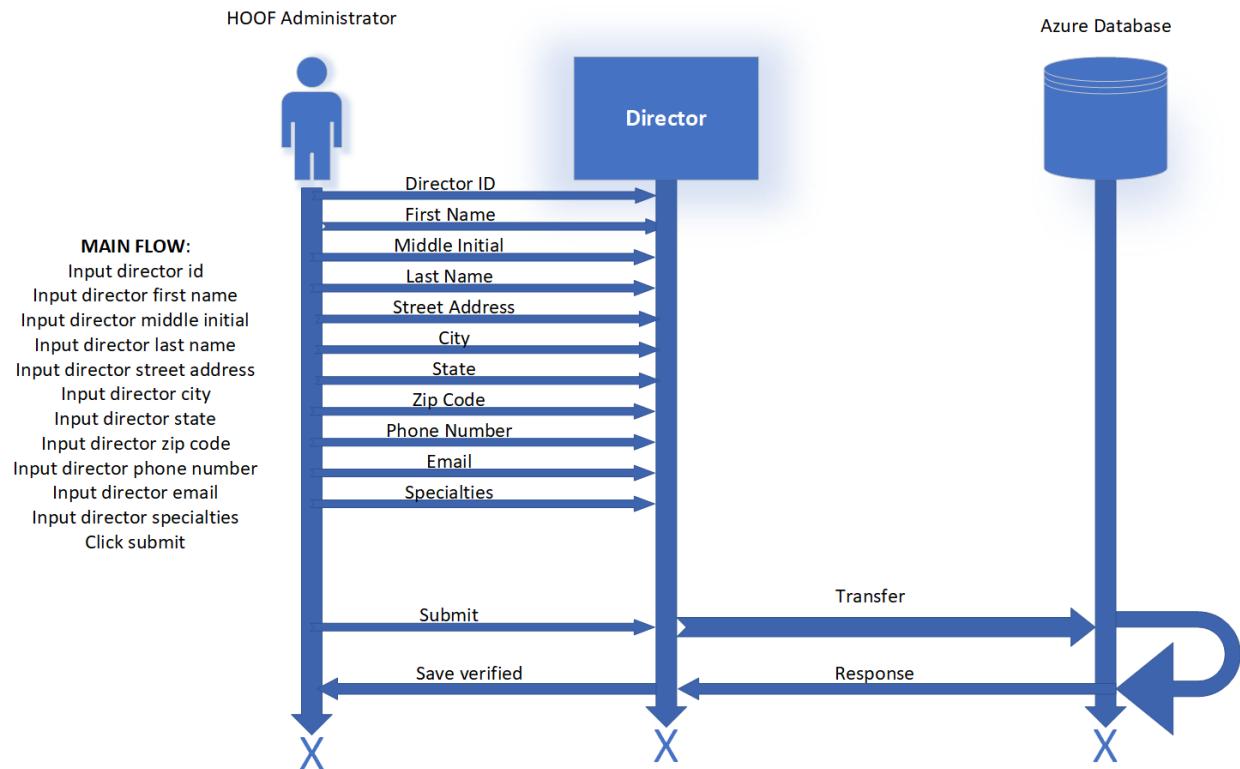
- Administrator must be logged into the Azure database

1. Post-conditions

- The additions to the board of directors table in the Azure database are saved

Add a Director Diagram

Add A Director



Use Case Specification: Edit a Director

1. Edit a Director

1.1 Brief Description

A HOOF administrator will be able to edit a director in the board table

1. Flow of Events

2.1 Basic Flow

- Edit director id
- Edit director first name
- Edit director middle initial
- Edit director last name
- Edit director street address
- Edit director city
- Edit director state
- Edit director zip code
- Edit director phone number
- Edit director email
- Edit director specialties
- Click submit

2.2 Alternative Flows

2.2.1 < No Specialties >

If the new director has no specialties, this field can be null.

2.2.2 <Null Middle Initial>

- If there is no middle initial for the participant, the middle initial field may be null
 - 2.2.2a <Null Error Message>
 - If any fields other than middle initial or specialties are null, an error message will be displayed to the user prompting them to input a value into each required field

7. Special Requirements

- The administrator must use a computer with internet access

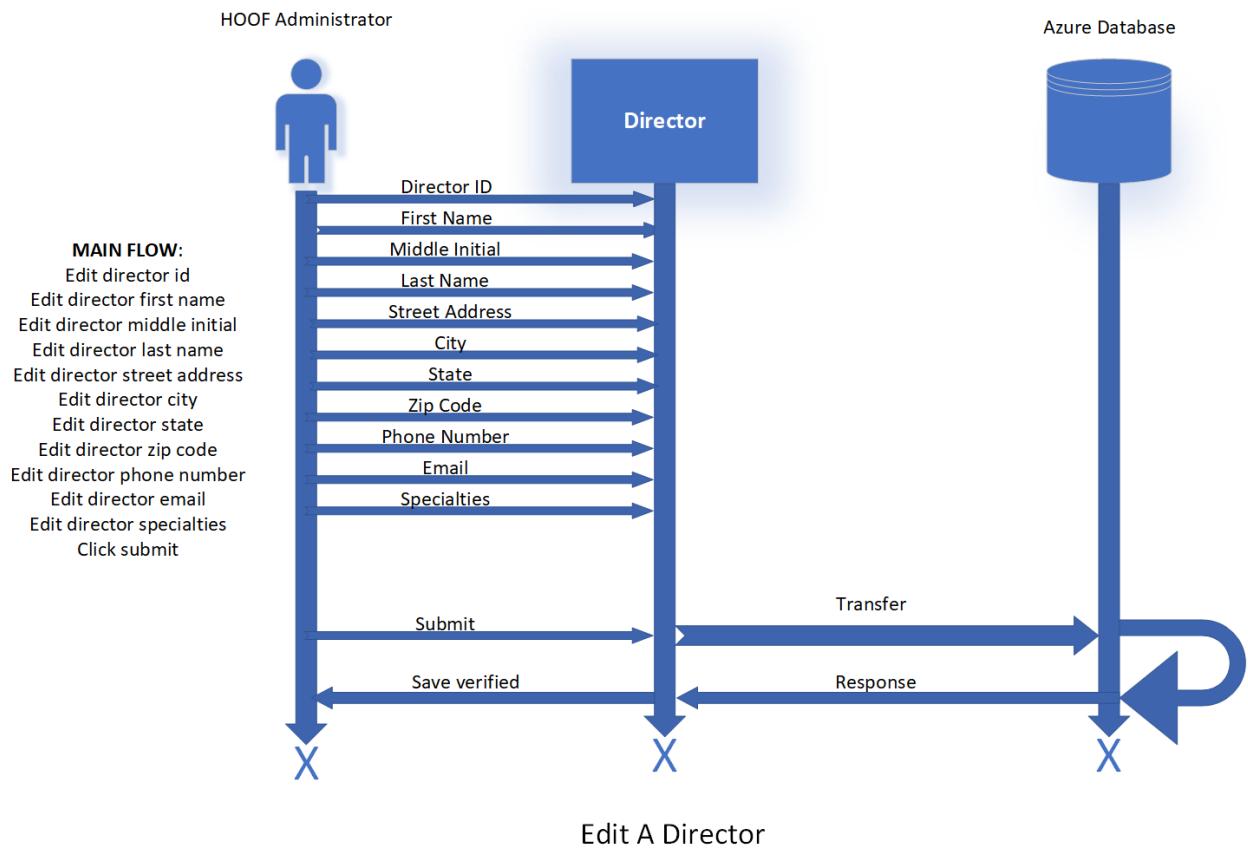
8. Pre-conditions

- Administrator must be logged into the Azure database
- A record for a director must exist

9. Post-conditions

- The edits to the board of directors table will be saved in the Azure database

Edit a Director Sequence Diagram



Use Case Specification: Delete a Director

1. Delete a Director

1.1 Brief Description

A HOOF administrator will be able to delete a director from the board of directors table

1. Flow of Events

2.1 Basic Flow

- Select director row to be deleted
- Right click
- Click delete
- Click submit

3. Special Requirements

- Administrator must use a computer with internet access

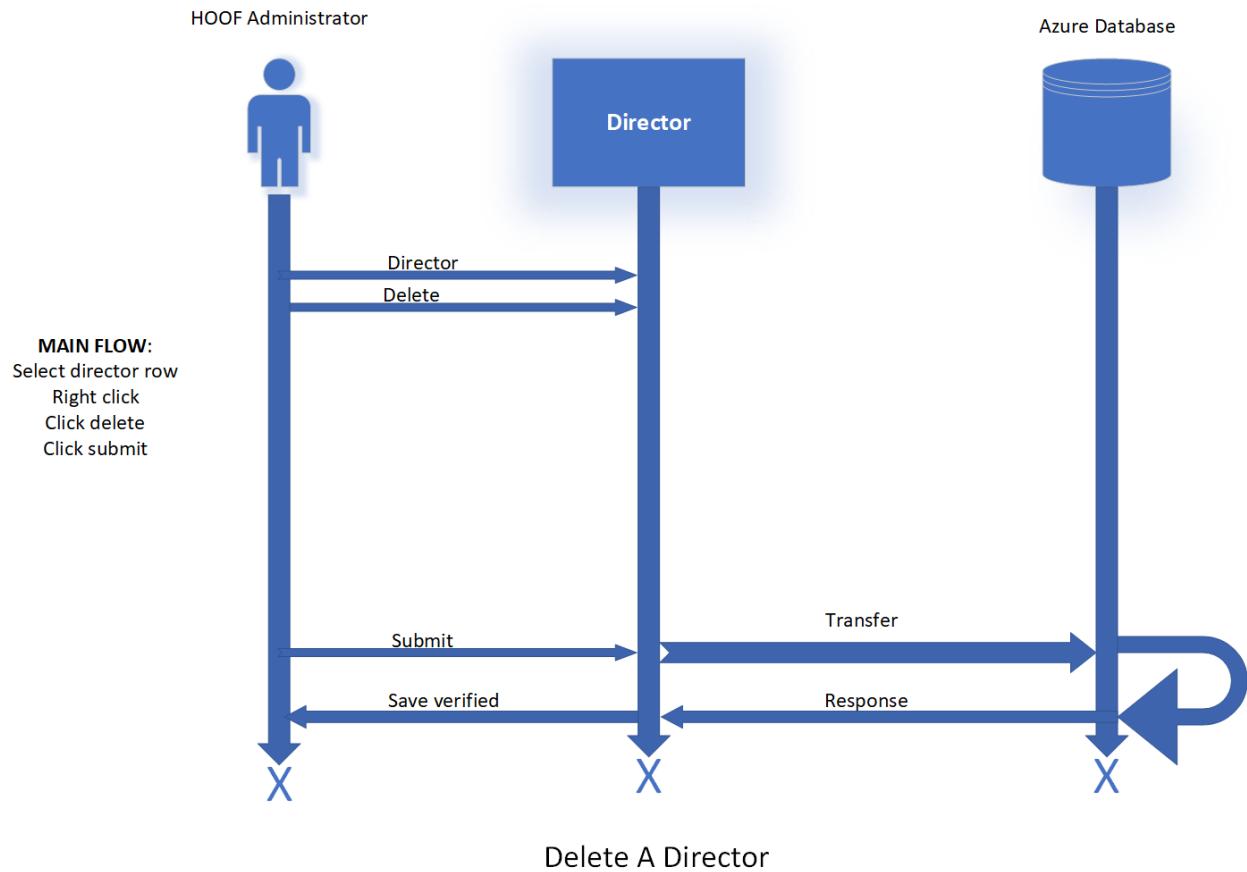
4. Pre-conditions

- Administrator must be logged into the Azure database
- A record for a director must exist

5. Post-conditions

- The deletions to the board of directors table in the Azure database are saved

Delete a Director Sequence Diagram



Use Case Specification: Add a Participant

1. Add a Participant

1.1 Brief Description

A HOOF administrator will be able to add a participant's information to the participant table

1. Flow of Events

2.1 Basic Flow

- Input participant id
- Input participant first name
- Input participant middle initial
- Input participant last name
- Input participant street address
- Input participant city
- Input participant state
- Input participant zip code
- Input participant school id
- Input participant grade
- Input guardian first name
- Input guardian last name
- Input guardian phone number
- Input guardian email
- Input emergency contact first name
- Input emergency contact last name
- Input emergency contact phone number
- Click submit

2.2 Alternative Flows

2.2.1 <Matching Guardian and Emergency Contact>

- If the emergency contact first name, last name and phone number are the same as the guardian's first name, last name and phone number only emergency contact phone number field is required.

2.2.2 <Null Middle Initial>

- If there is no middle initial for the participant, the middle initial field may be null

2.2.2a <Null Error Message>

- If any fields other than middle initial, guardian first name, last name or phone number are null, an error message will be displayed to the user prompting them to input a value into each required field

1. Special Requirements

- Administrator must use a computer with internet access

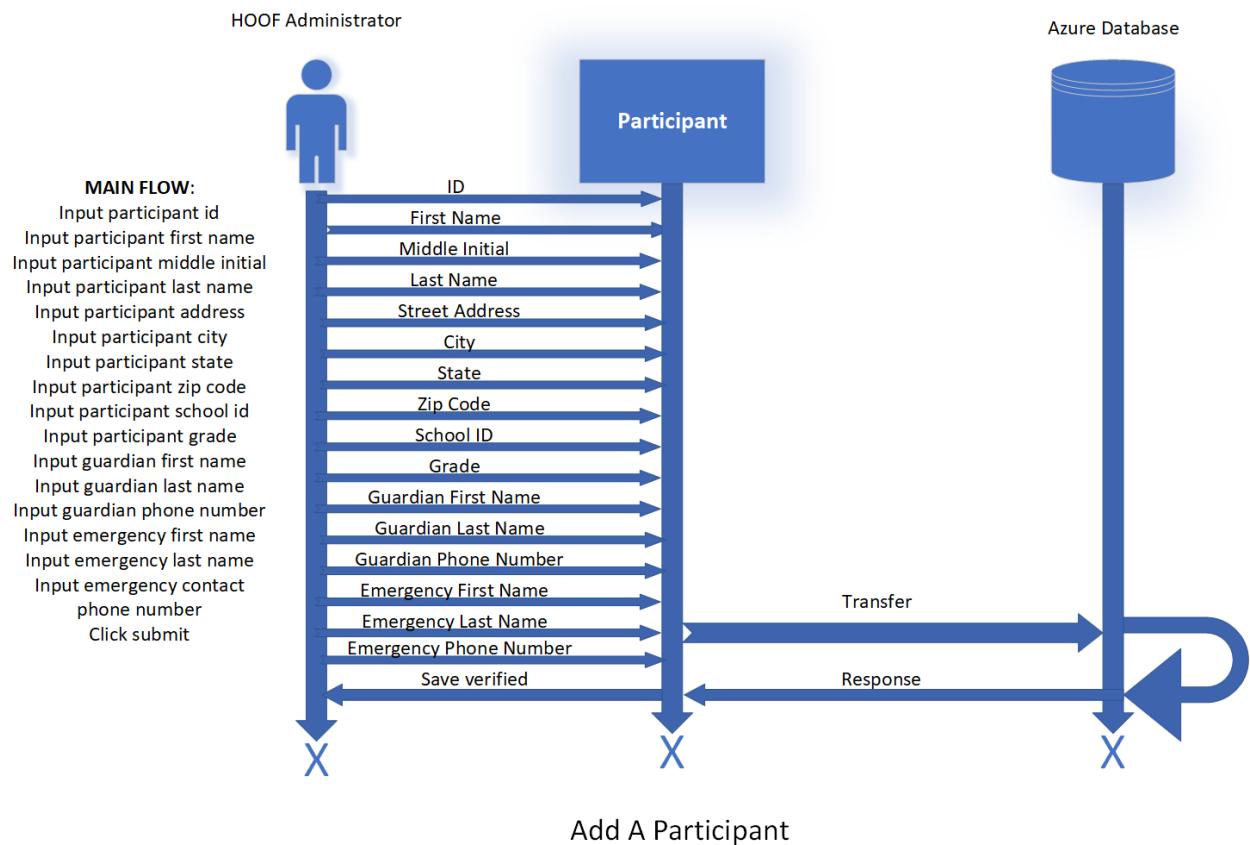
1. Pre-conditions

- Administrator must be *logged into* the Azure database

1. Post-conditions

- The additions to the participant table in the Azure database are saved

Add a Participant Sequence Diagram



Use Case Specification: Edit a Participant

1. Edit a Participant

1.1 Brief Description

A HOOF administrator will be able to edit a participant's information in the participant table

1. Flow of Events

2.1 Basic Flow

- Edit participant id
- Edit participant first name
- Edit participant middle initial
- Edit participant last name
- Edit participant street address
- Edit participant city
- Edit participant state
- Edit participant zip code
- Edit participant school id
- Edit participant grade
- Edit guardian first name
- Edit guardian last name
- Edit guardian phone number
- Edit emergency contact first name
- Edit emergency contact last name
- Edit emergency contact phone number
- Click submit

2.2 Alternative Flows

2.2.1 < Matching Guardian and Emergency Contact>

- If the emergency contact first name, last name and phone number are the same as the guardian's first name, last name and phone number only emergency contact phone number field is required.

2.2.2 <Null Middle Initial>

- If there is no middle initial for the participant, the middle initial field may be null

2.2.2a <Null Error Message>

- If any fields other than middle initial, guardian first name, last name or phone number are null, an error message will be displayed to the user prompting them to input a value into each required field

1. Special Requirements

- The administrator must use a computer with internet access

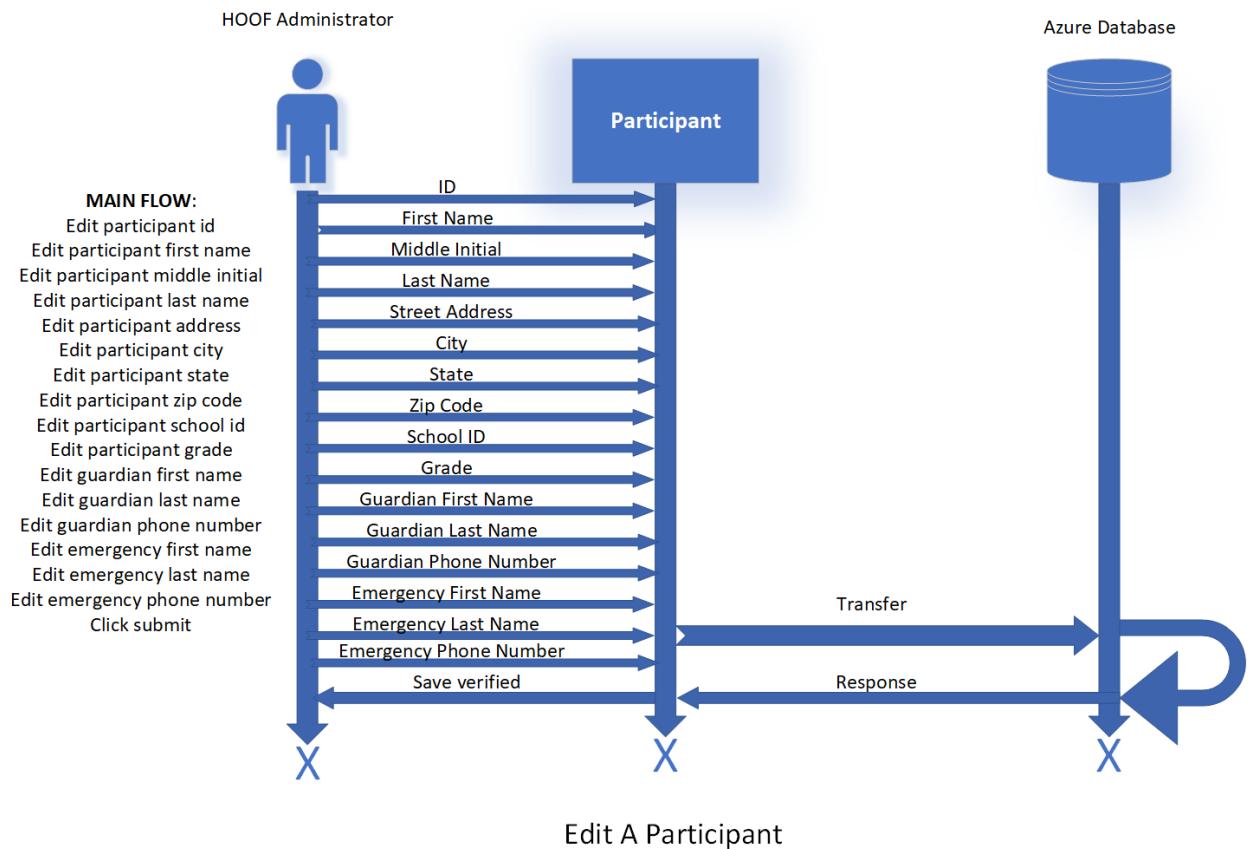
1. Pre-conditions

- Administrator must be *logged in* the Azure database
- A record for a child must exist

1. Post-conditions

- The edits to the participant table in the Azure database are saved

Edit a Participant Sequence Diagram



Use Case Specification: Delete a Participant

1. Delete a Participant

1.1 Brief Description

A HOOF administrator will be able to delete a participant's information from the participant table

1. Flow of Events

2.1 Basic Flow

- Select participant row to be deleted
- Right click
- Click delete
- Click submit

3. Special Requirements

- Administrator must use a computer with internet access

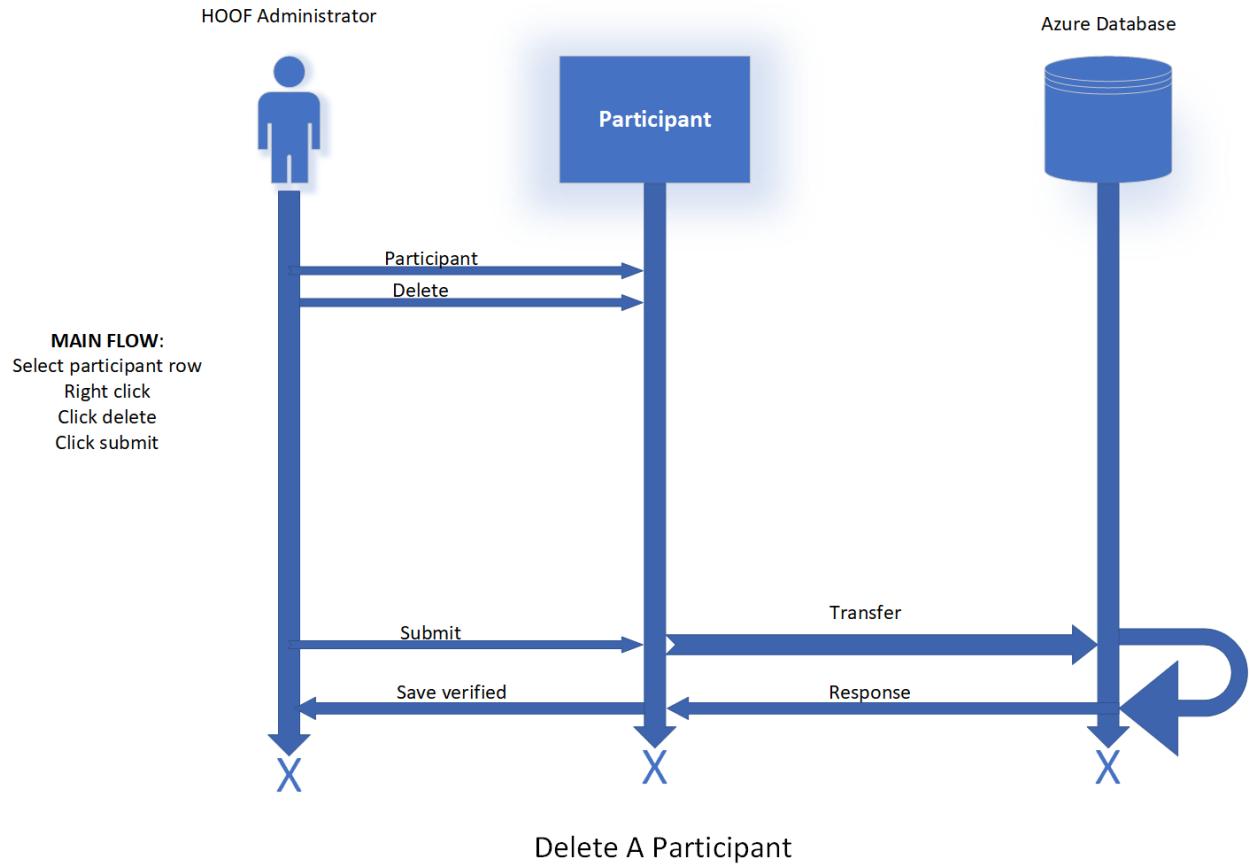
4. Pre-conditions

- Administrator must be *logged into* the Azure database
- A record for a participant must exist

5. Post-conditions

- The deletions to the participant table in the Azure database are saved

Delete a Participant Sequence Diagram



Use Case Specification: Add a User

1. Add a User

1.1 Brief Description

A HOOF administrator will be able to add a WordPress user

1. Flow of Events

2.1 Basic Flow

- Click on 'users'
- Click on 'add new'
- Input first name
- Input middle initial
- Input last name
- Input email address
- Input user name (this can never be changed)
- Choose a temporary password for the user
- Check box that says: "Send this password to the new user by email"
- Click dropdown to select user type
- Click on Submit

2.2 Alternative Flows

2.2.1 <Login Prompt>

The new user will receive an email at the previously specified address. This email will instruct them on how to log in.

2.2.2 <Updating Profile>

Once the user has logged in, they will be prompted to change their password and profile information.

1. Special Requirements

- User must have access to a computer with internet access

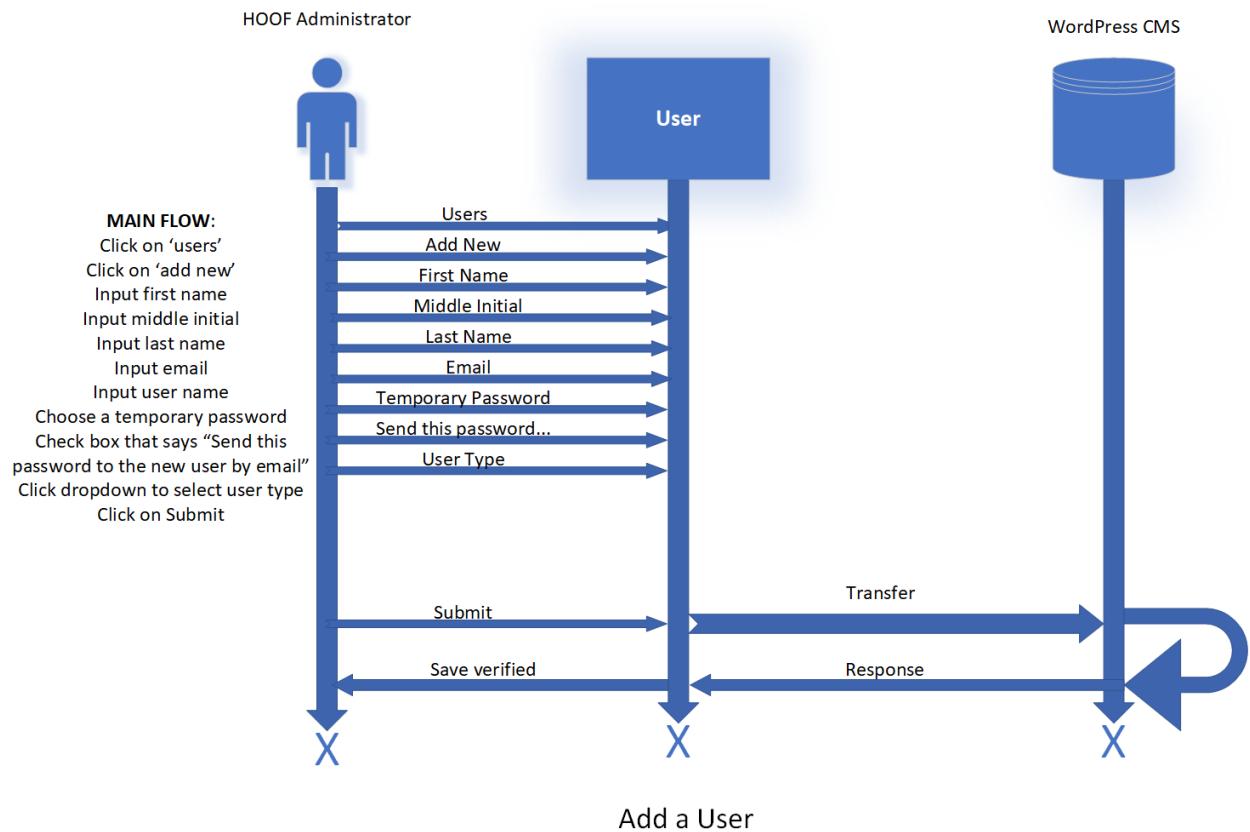
1. Pre-conditions

- Administrator must be logged into WordPress

1. Post-conditions

- Added user is saved

Add a User Sequence Diagram



Use Case Specification: Delete a User

1. Delete a User

1.1 Brief Description

A HOOF administrator will be able to delete a WordPress user

1. Flow of Events

2.1 Basic Flow

- Click on 'users'
- Click on 'all users'
- Check the box next to the user avatar
- Click on trash can icon to delete
- Click submit

1. Special Requirements

- User must have access to a computer with internet access

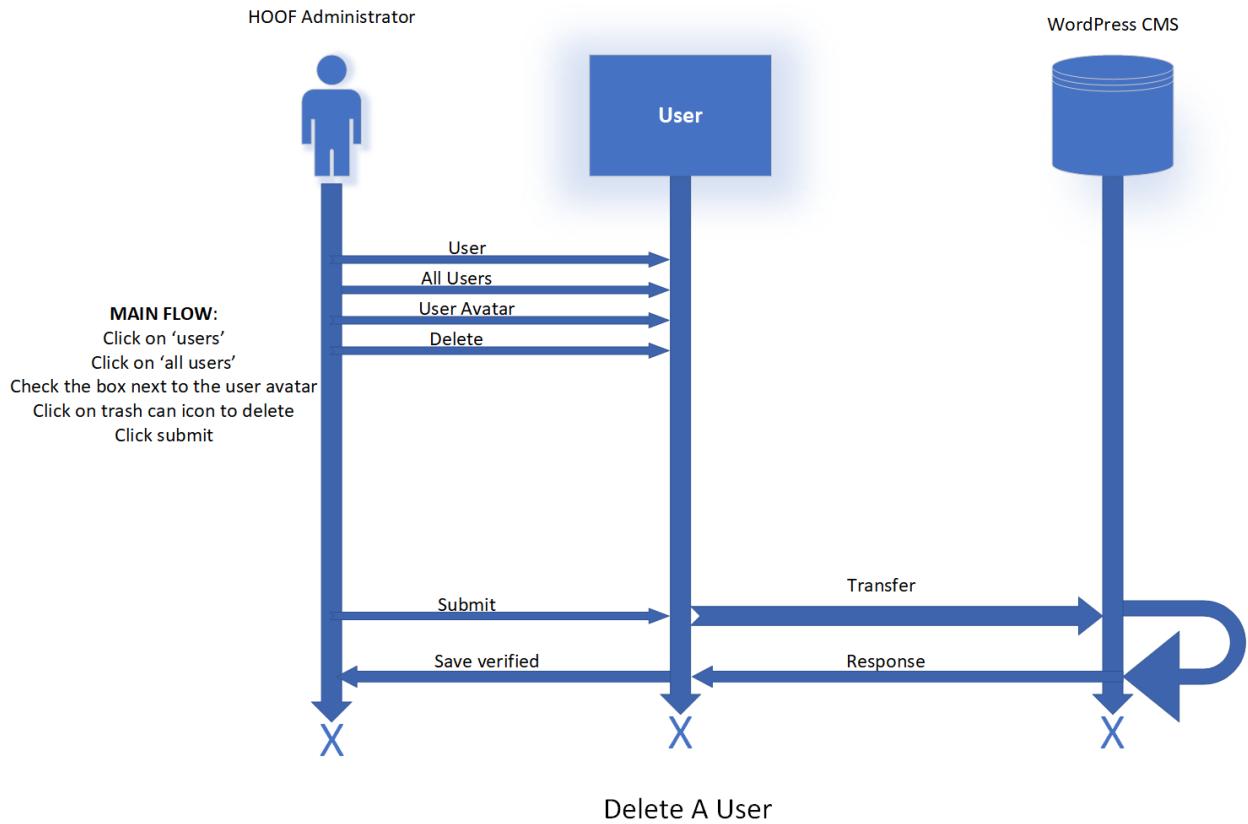
1. Pre-conditions

- Administrator must be logged into WordPress

1. Post-conditions

- Deleted user is saved

Delete a User Sequence Diagram



Use Case Specification: Edit Permissions

1. Edit Permissions

1.1 Brief Description

A HOOF administrator will be able to edit a WordPress user's permissions

1. Flow of Events

2.1 Basic Flow

- Click YES or NO for 'can access wp-admin'
- Choose YES or NO for 'force hide adminbar in frontend'
- Choose YES or NO for 'can edit other member accounts'
- Choose YES or NO for 'can delete other members accounts'
- Choose YES or NO for 'can edit their profile'
- Choose YES or NO for 'can delete their account'
- Choose YES or NO for 'can view other member profiles'
- Input user roles that are allowed
- Choose YES or NO for 'can make their profile private'
- Choose YES or NO for 'can view/access private profiles'
- Click submit

1. Special Requirements

- User must have access to a computer with internet access

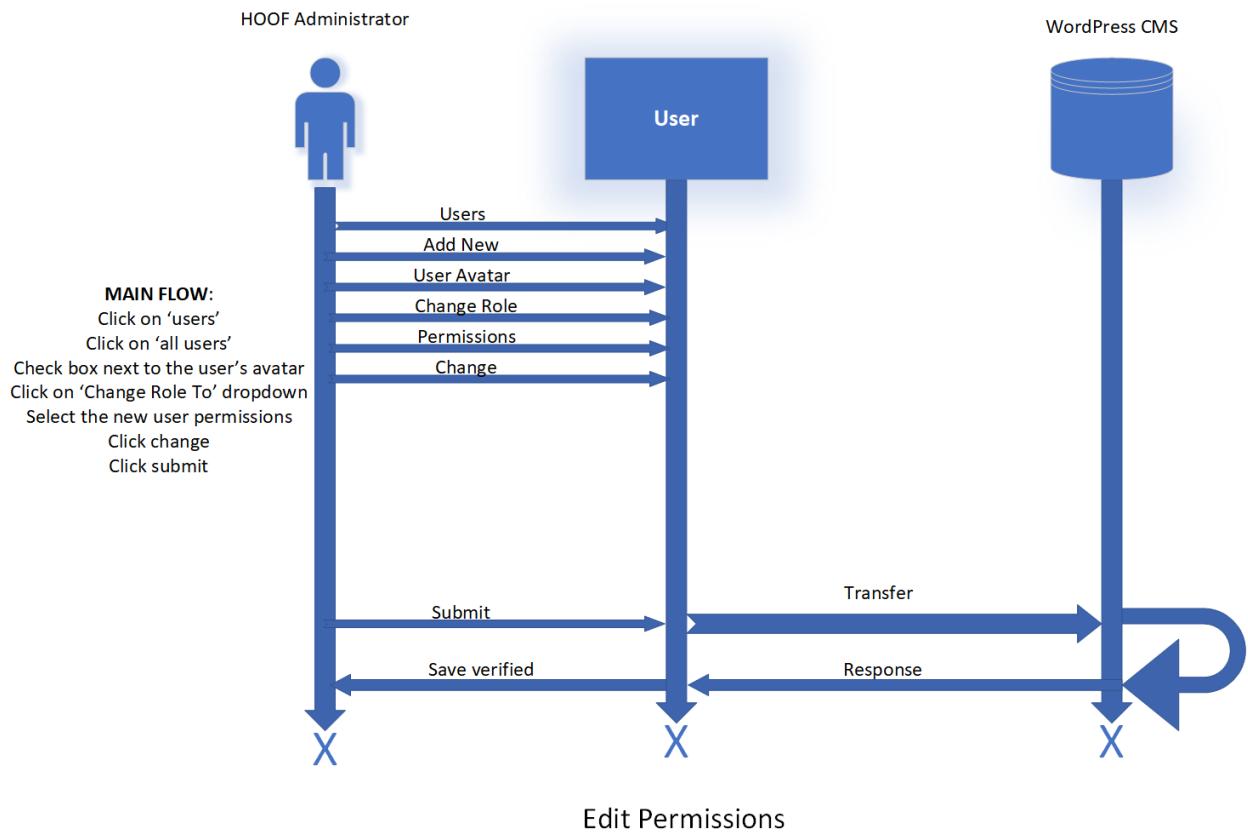
1. Pre-conditions

- Administrator must be logged into WordPress

1. Post-conditions

- Edited permissions are saved

Edit Permissions Sequence Diagram



Use Case Specification: Add a School

1. Add a School

1.1 Brief Description

A HOOF administrator will be able to add a school to the schools table

1. Flow of Events

2.1 Basic Flow

- Input school id
- Input school name
- Input school street address
- Input school city
- Input school state
- Input school zip code
- Input school contact name
- Input school phone number
- Input school contact email
- Click submit

2.2 Alternative Flows

2.2.1 <Null Error Message>

- If any fields are null, an error message will be displayed to the user prompting them to input a value into each required field

3. Special Requirements

- Administrator must use a computer with internet access

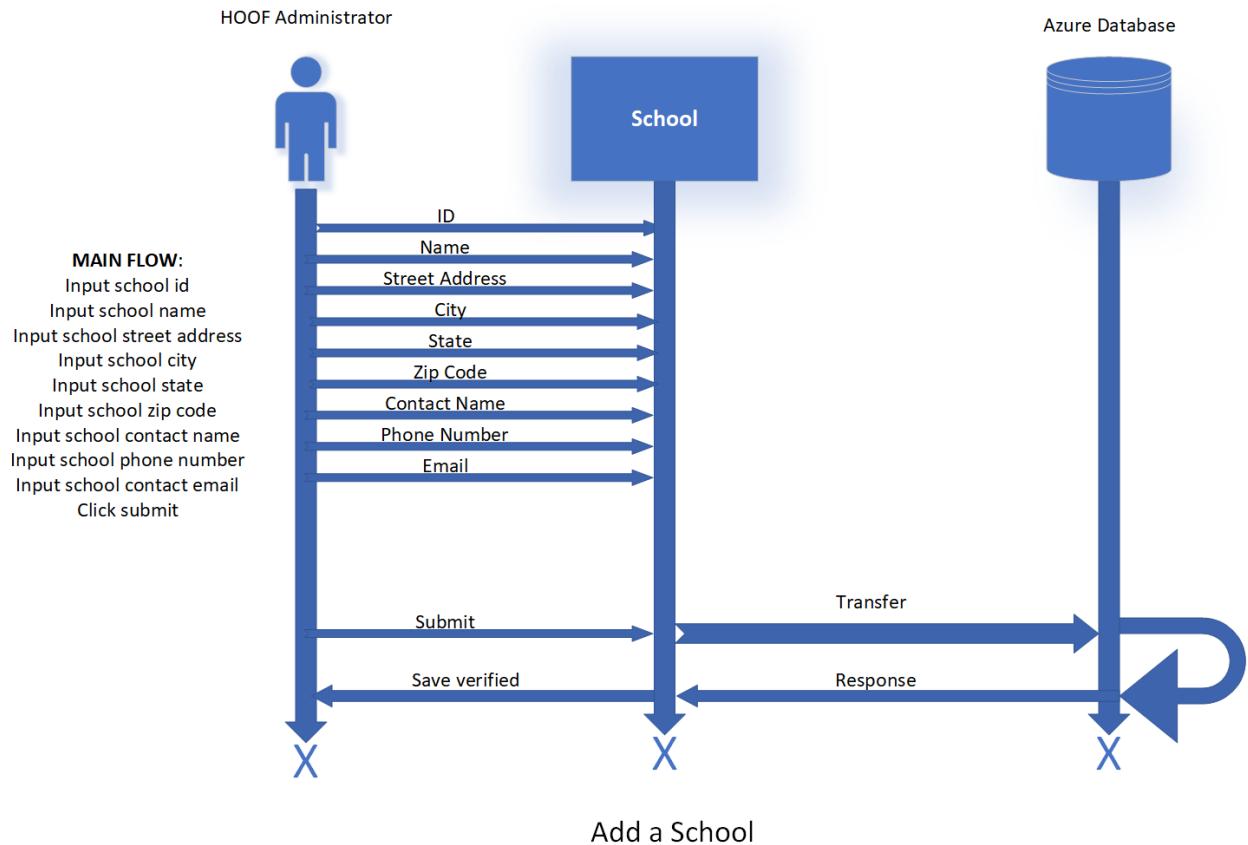
4. Pre-conditions

- Administrator must be logged into the Azure database

5. Post-conditions

- The additions to the school table will be saved in the Azure database

Add a School Sequence Diagram



Use Case Specification: Edit a School

1. Edit a School

1.1 Brief Description

A HOOF administrator will be able to edit a school in the schools table

1. Flow of Events

2.1 Basic Flow

- Edit school id
- Edit school name
- Edit school street address
- Edit school city
- Edit school state
- Edit school zip code
- Edit school contact name
- Edit school phone number
- Edit school contact email
- Click submit

2.2 Alternative Flows

2.2.1 <Null Error Message>

- If any fields are null, an error message will be displayed to the user prompting them to input a value into each required field

3. Special Requirements

- Administrator must use a computer with internet access

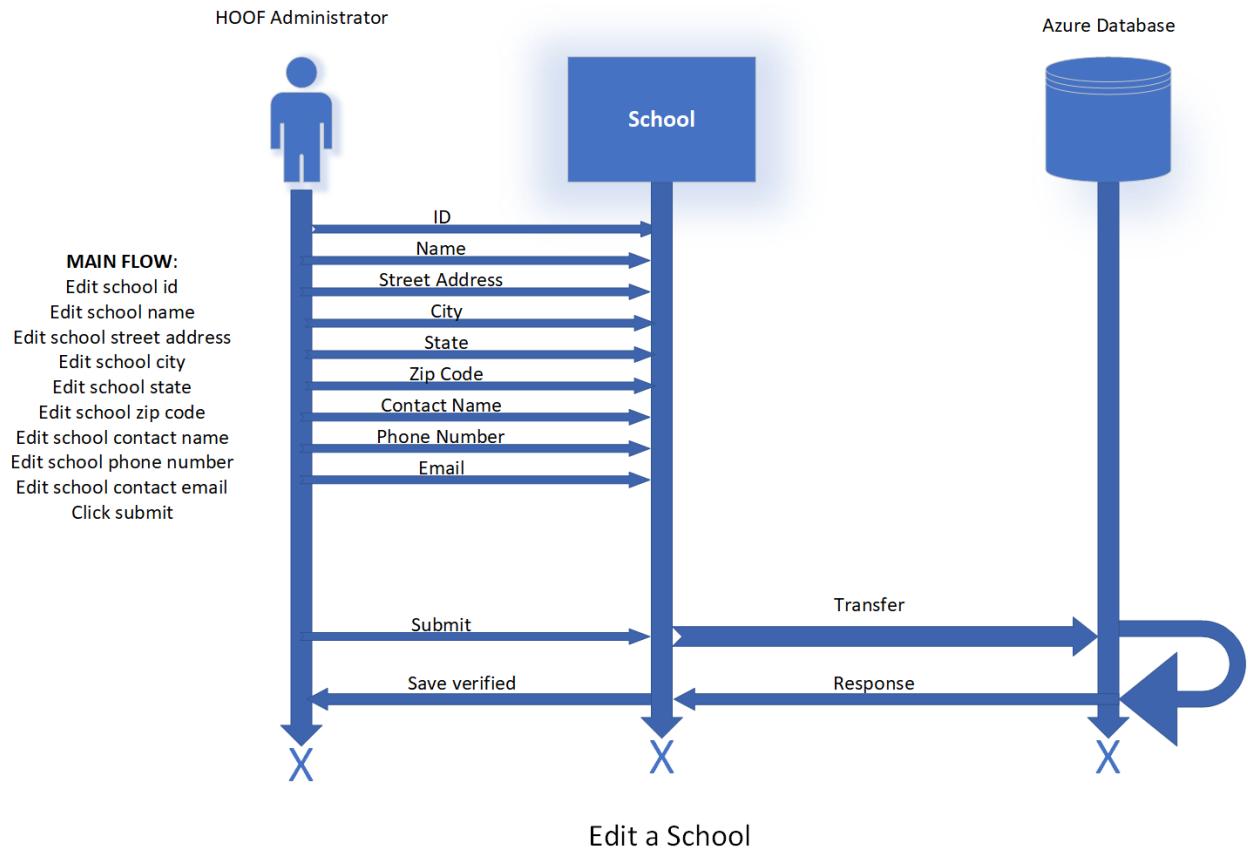
4. Pre-conditions

- Administrator must be logged into the Azure database
- A record for a school must exist

5. Post-conditions

- The edits to the schools table will be saved in the Azure database

Edit a School Sequence Diagram



Use Case Specification: Delete a School

1. Delete a School

1.1 Brief Description

A HOOF administrator will be able to delete a school from the schools table

1. Flow of Events

2.1 Basic Flow

- Select director row to be deleted
- Right click
- Click delete
- Click submit

1. Special Requirements

- Administrator must use a computer with internet access

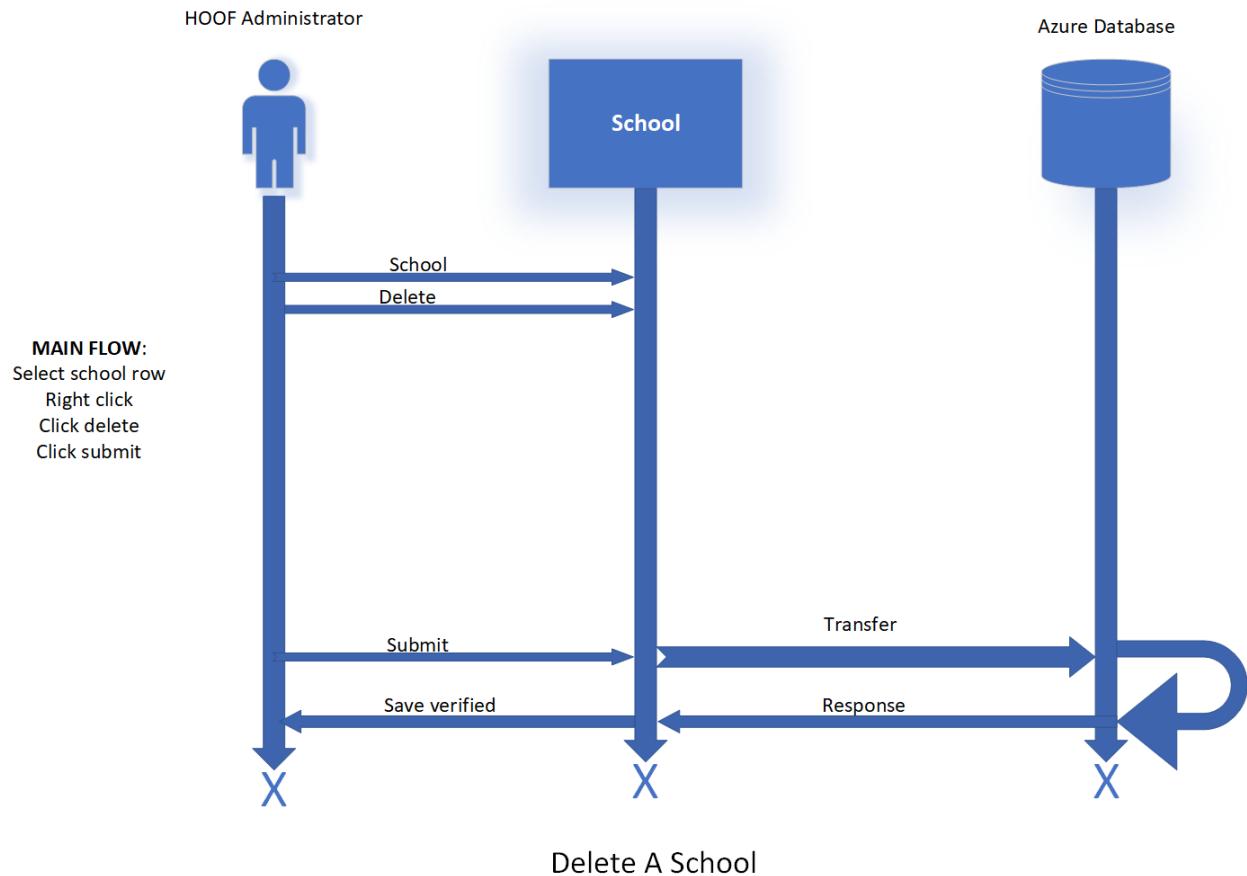
1. Pre-conditions

- Administrator must be logged into the Azure database
- A record for a school must exist

1. Post-conditions

- The deletions to the schools table in the Azure database are saved

Delete a School Sequence Diagram



Use Case Specification: Add a Potential Grant

1. Add a Potential Grant Application

2. Brief Description

A HOOF administrator or designated grant writer will be able to add a potential grant to the list.

3. Flow of Events

4. Basic Flow

- Input potential grant name
- Input grant writer first name
- Input grant writer last name
- Input grant writer middle initial
- Input potential grant amount
- Input potential grant source
- Input description
- Click submit

4.1 Alternative Flows

<Null Error Message>

- Any fields left null other than the middle initial, an error message will be displayed to the user prompting them to input a value into each required field.

5. Special Requirements

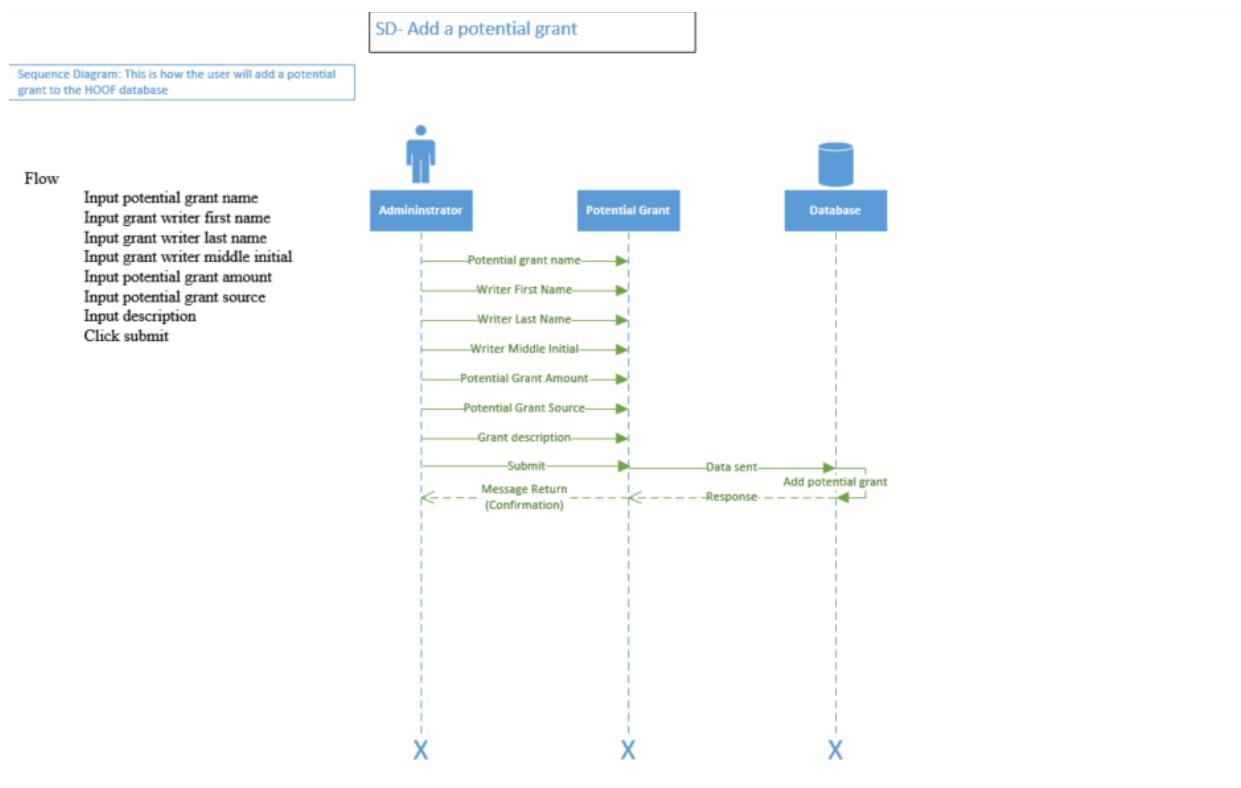
- Must have admin access
- Must have grant writing permissions
- Must have login access
- Must have internet access

6. Pre-conditions

- Administrator/Writer has navigated to HOOF's website.
- Admin must be logged into Azure database

7. Post-conditions

- Potential grant application has been added to the database.



Use Case Specification: Edit a Potential Grant

1. Edit a Potential Grant

2. Brief Description

A HOOF administrator or designated grant writer will be able to edit a potential grant to the list.

3. Flow of Events

4. Basic Flow

- Edit potential grant name
- Edit grant writer first name
- Edit grant writer last name
- Edit grant writer middle initial
- Edit potential grant amount
- Edit potential grant source
- Edit description
- Click submit

5. Special Requirements

- Must have admin access
- Must have grant writing permissions
- Must have login access
- Must have internet access

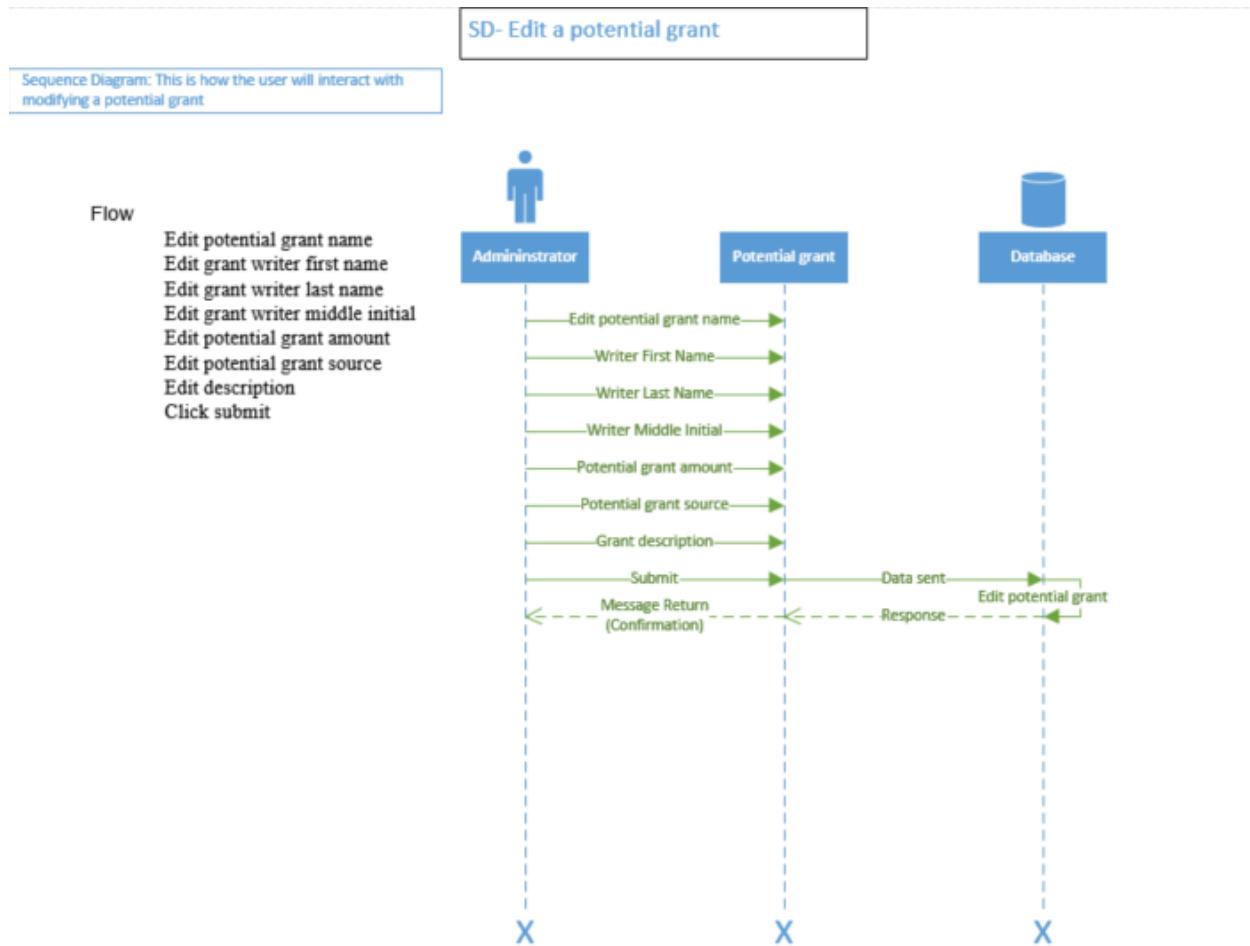
6. Pre-conditions

- Administrator/Writer has navigated to HOOF's website.
- Admin must be connected to Azure database

7. Post-conditions

- Potential grant application has been modified in the database.

Edit a potential grant Sequence Diagram



Use Case Specification: Delete a Potential Grant

1. Delete a Potential Grant

2. Brief Description

A HOOF administrator or designated grant writer will be able to delete a potential grant from the list.

3. Flow of Events

4. Basic Flow

- Select on potential grant
- Click on delete icon
- Click submit

5. Special Requirements

- Must have admin access
- Must have grant writing permissions
- Must have login access
- Must have internet access

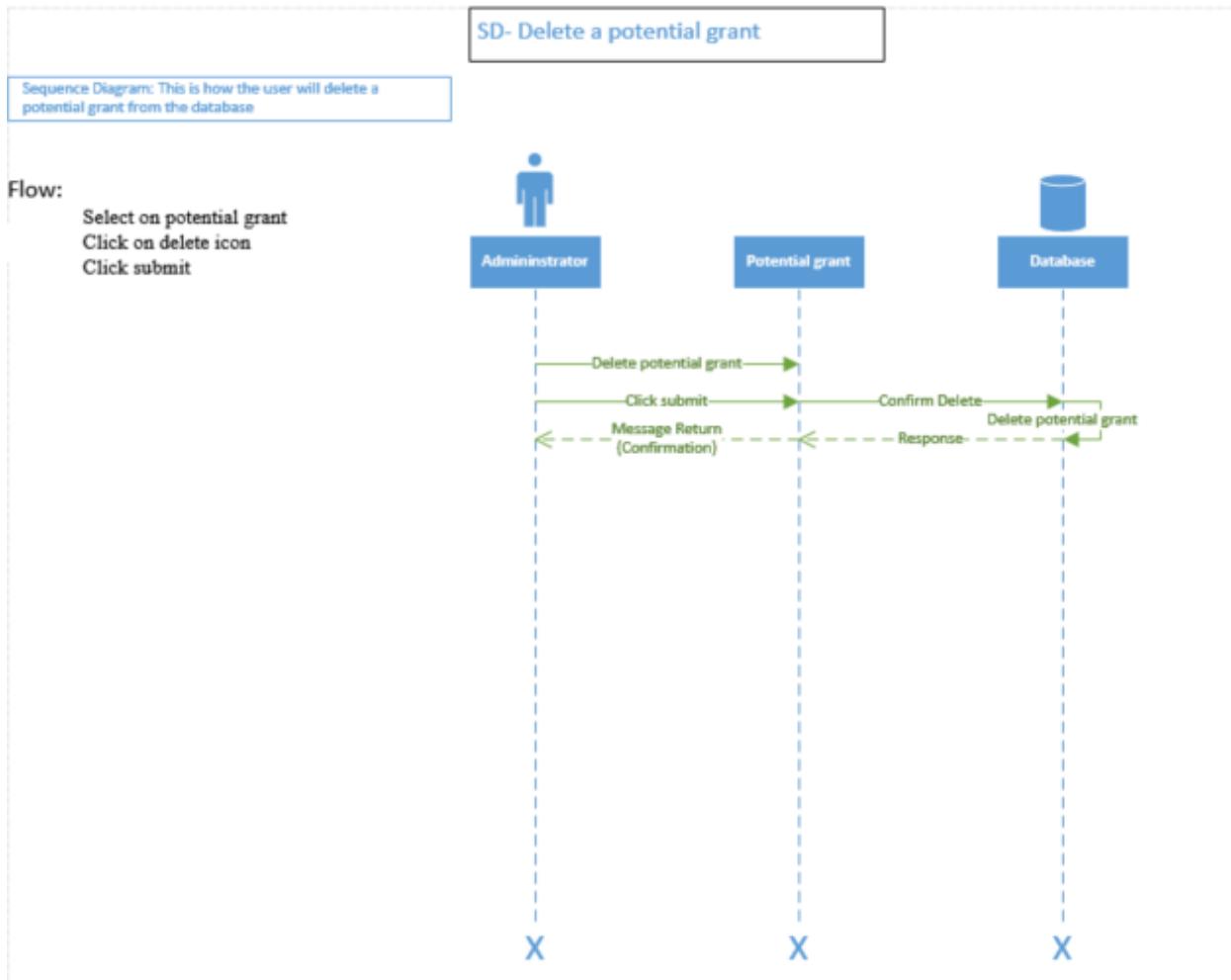
6. Pre-conditions

- Administrator/Writer has navigated to HOOF's website.
- Admin must be logged into Azure database

7. Post-conditions

- Potential grant application has been deleted from the database

Delete a Potential Grant Sequence Diagram



Use Case Specification: Add Silent Auction Item

1. Add Silent Auction Item

1.1 Brief Description

A HOOF administrator will be able to add silent auction items to the inventory table

2. Flow of Events

2.1 Basic Flow

- Input inventory item name
- Input inventory item description
- Input inventory item quantity
- Input inventory sale status
- Input donor name
- Input donor address
- Input item value
- Input final sale value
- Click on Submit

2.2 Alternative Flows

2.2.1 <Item Sold>

- If the inventory item has been sold, input the auction id into the sale status

2.2.1a <Item Not Sold>

- If the inventory item has not been sold, input null for sale status

2.2.2 <Null Error Message>

- If any fields other than sale status are null, an error message will be displayed to the user prompting them to input a value into each required field

2.2.2 <Export to Spreadsheet>

- If administrator wishes to view it as an Excel spreadsheet, they would click on the Export to Spreadsheet button.

3. Special Requirements

- Administrator must use a computer with internet access

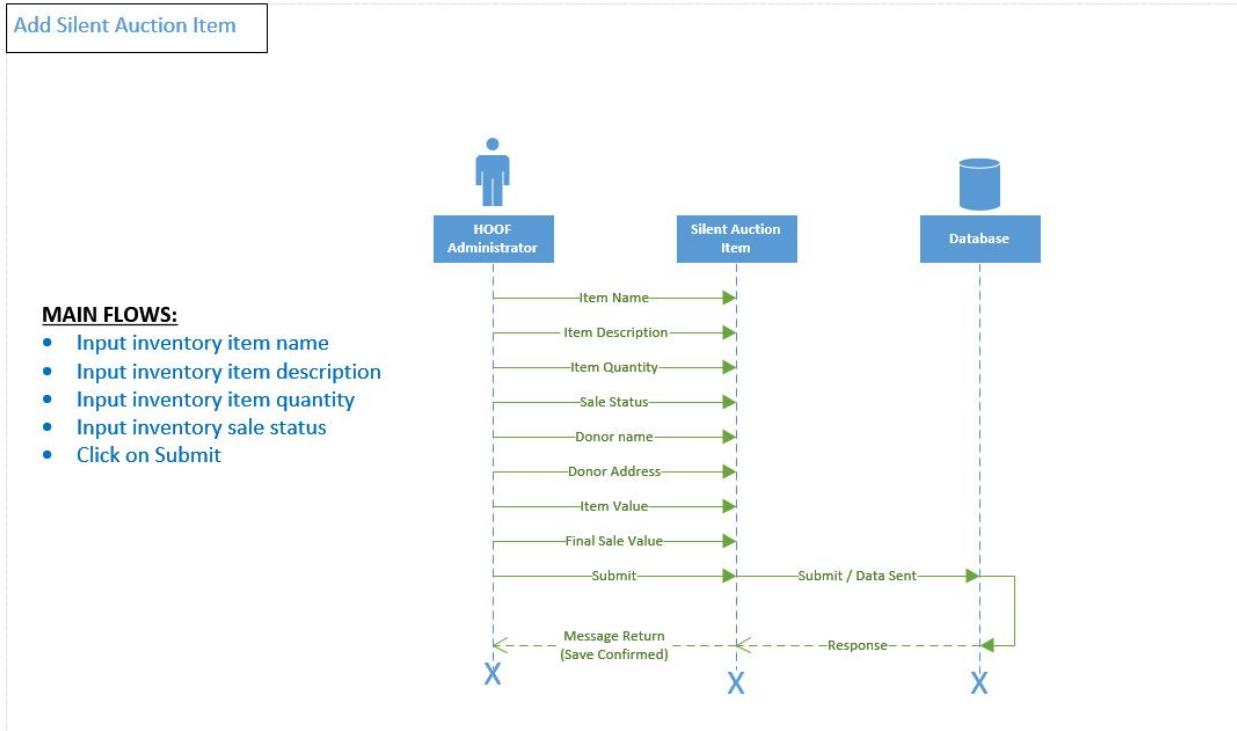
4. Pre-conditions

- Administrator must be *logged in* to the Azure database in order to access the database

5. Post-conditions

- The additions to the inventory table will be saved in the Azure database

Add Silent Auction Item Sequence Diagram



Use Case Specification: Edit Silent Auction Item

1. Edit Silent Auction Item

1.1 Brief Description

A HOOF administrator will be able to edit silent auction items in the inventory table

2. Flow of Events

2.1 Basic Flow

- Edit inventory item name
- Edit inventory item description
- Edit inventory item quantity
- Edit inventory item price paid
- Input donor name
- Input donor address
- Input item value
- Input final sale value
- Click on Submit

3. Alternative Flows

3.3.1 < Item Sold >

- If the inventory item has been sold, input the auction id into the sale status

3.3.1a < Item Not Sold >

- If the inventory item has not been sold, input null for sale status

3.3.2 < Null Error Message >

- If any fields other than sale status are null, an error message will be displayed to the user prompting them to input a value into each required field

3.3.3 < Export to Spreadsheet >

- If administrator wishes to view it as an Excel spreadsheet, they would click on the Export to Spreadsheet button.

4. Special Requirements

- Administrator must use a computer with internet access

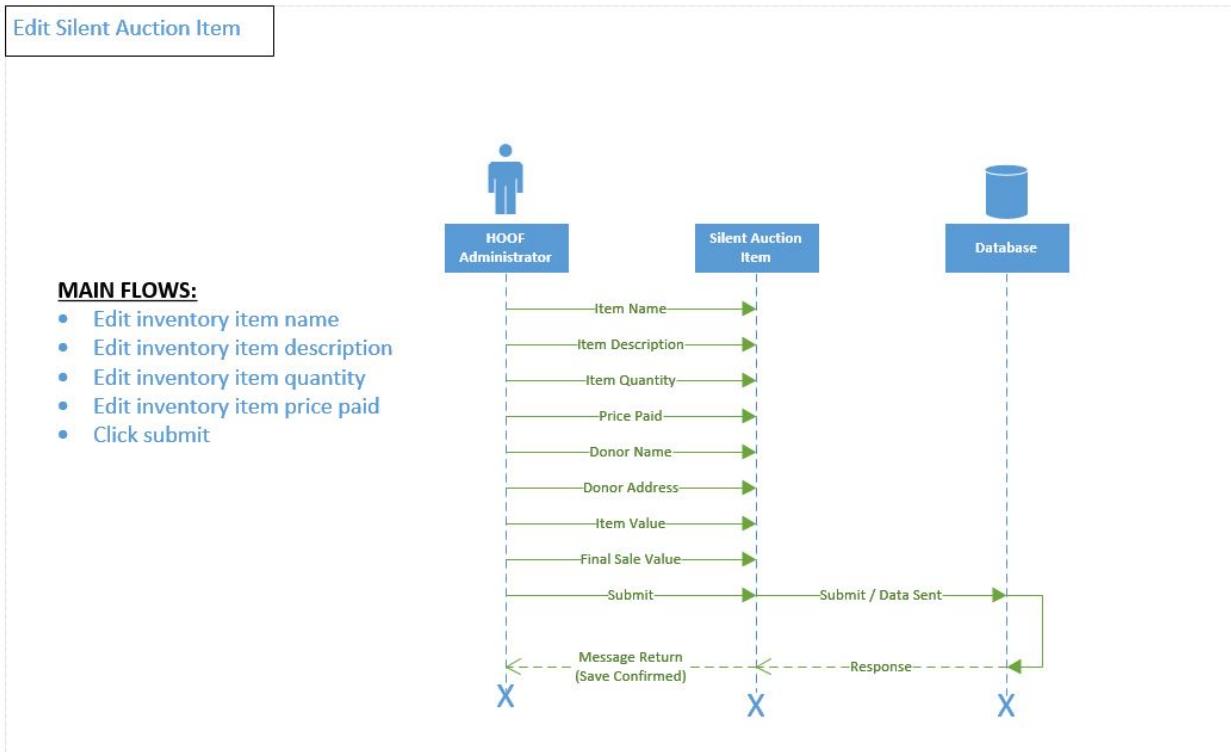
5. Pre-conditions

- Administrator must be *logged into* the Azure database
- A record for a silent auction item must exist

6. Post-conditions

- The edits to the inventory table in the Azure database are saved

Edit Silent Auction Item Sequence Diagram



Use Case Specification: Delete Silent Auction Item

1. Delete Silent Auction Item

1.1 Brief Description

A HOOF administrator will be able to delete silent auction item from the inventory table

2. Flow of Events

2.1 Basic Flow

- Select silent auction row to be deleted
- Right click
- Click delete
- Click submit

3. Special Requirements

- Administrator must use a computer with internet access

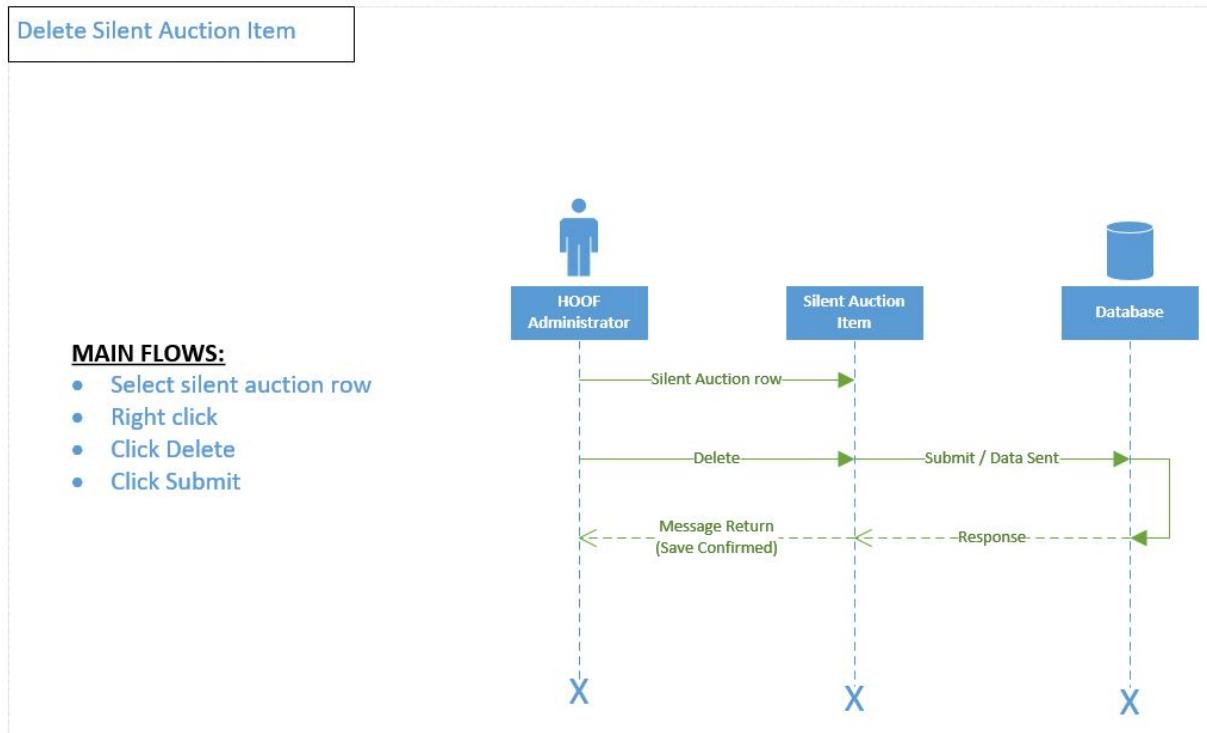
4. Pre-conditions

- Administrator must be *logged into* the Azure database
- A record for a silent auction item must exist

5. Post-conditions

- The deletions to the inventory table in the Azure database are saved

Delete Silent Auction Item Sequence Diagram



Use Case Specification: Add Silent Auction

1. Add Silent Auction

1.1 Brief Description

A HOOF administrator will be able to add silent auctions to the silent auctions table

2. Flow of Events

2.1 Basic Flow

- Input silent auction id
- Input silent auction name
- Input silent auction date
- Input silent auction time
- Input silent auction venue
- Input silent auction manager
- Click submit

2.2 Alternative Flows

2.2.1 <Missing Date, Time and Venue>

If the silent auction has yet to establish a set date, time or venue these fields may be null

2.2.1a <Null Error Message>

- If any fields other than date, time and venue are null than a error message will be displayed to the user, prompting them to input a value into each required field

3. Special Requirements

- Administrator must use a computer with internet access

4. Pre-conditions

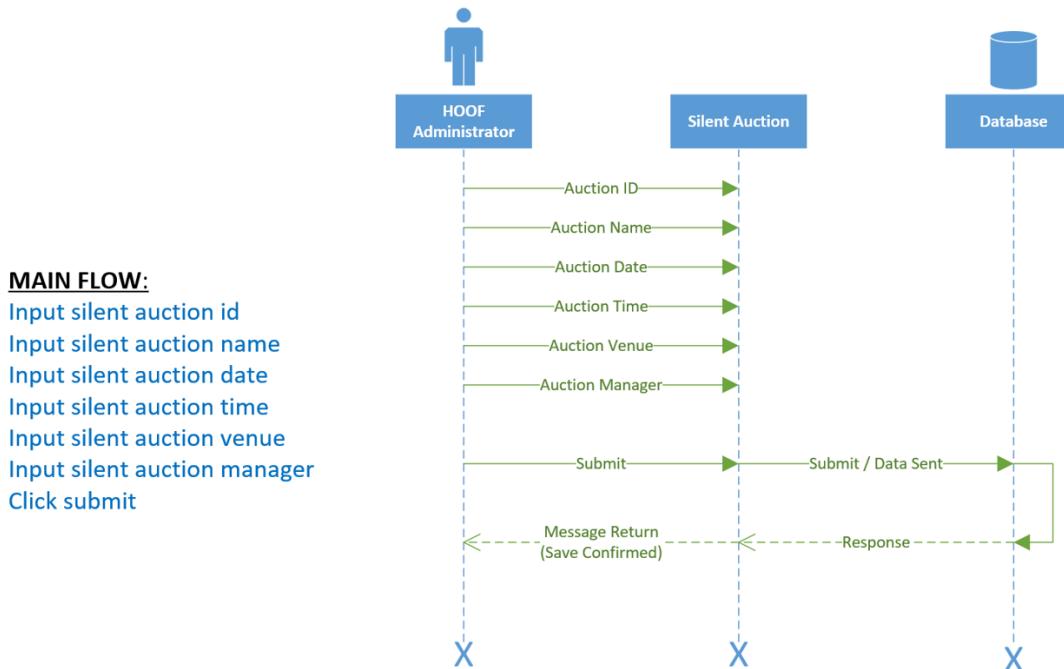
- Administrator must be *logged in* the Azure database

5. Post-conditions

- The additions to the silent auctions table will be saved in the Azure database

Add Silent Auction Sequence Diagram

Add Silent Auction



Use Case Specification: Edit Silent Auction

1. Edit Silent Auction

2. Brief Description

A HOOF administrator will be able to edit silent auction information in the auctions table

3. Flow of Events

4. Basic Flow

- Edit silent auction id number
- Edit silent auction name
- Edit silent auction date
- Edit silent auction time
- Edit silent auction venue
- Edit silent auction manager
- Click submit

2.2.1 <Missing Date, Time and Venue>

If the silent auction has yet to establish a set date, time or venue these fields may be null

2.2.1a <Null Error Message>

- If any fields other than date, time and venue are null than a error message will be displayed to the user, prompting them to input a value into each required field

5. Special Requirements

- Administrator must use a computer with internet access

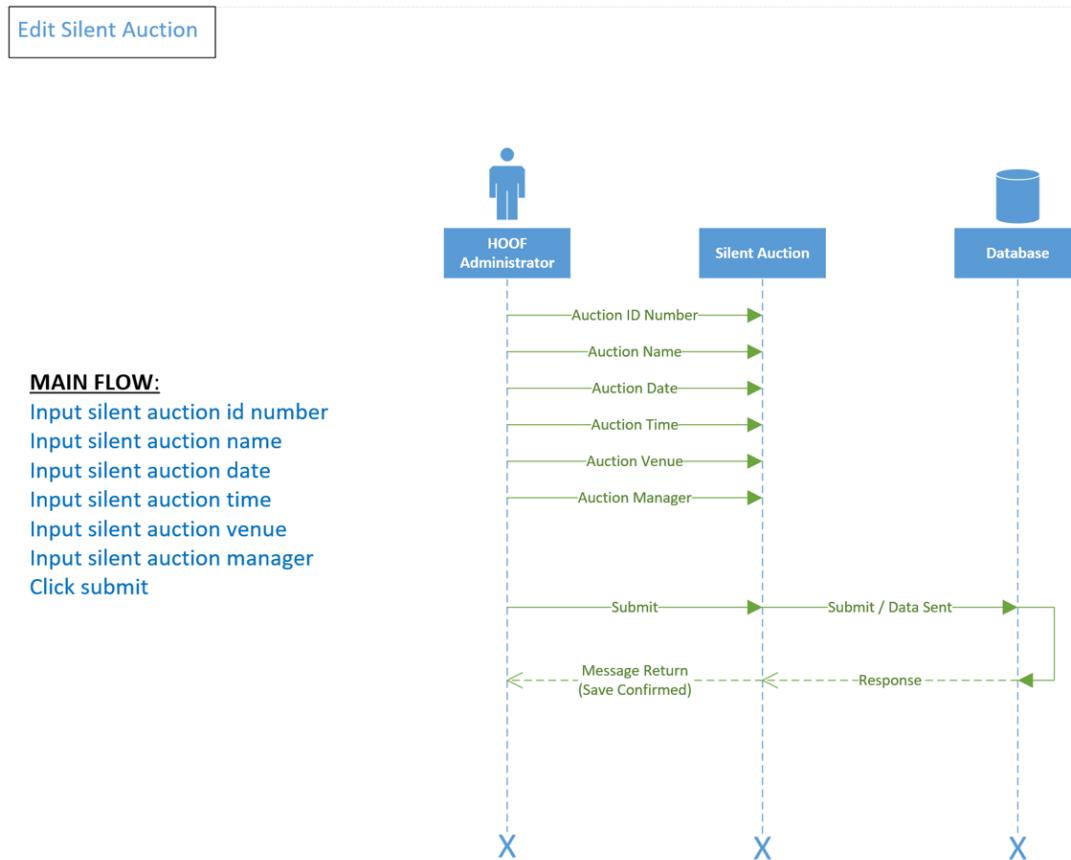
6. Pre-conditions

- Administrator must be *logged into* the Azure database
- A record for a silent auction must exist

7. Post-conditions

- The edits to the auctions table in the Azure database are saved

Edit Silent Auction Sequence Diagram



Use Case Specification: Delete Silent Auction

1. Delete Silent Auction

1.1 Brief Description

A HOOF administrator will be able to delete silent auction information from the auction table

2. Flow of Events

2.1 Basic Flow

- Select silent auction row to be deleted
- Right click
- Click delete
- Click submit

3. Special Requirements

- Administrator must use a computer with internet access

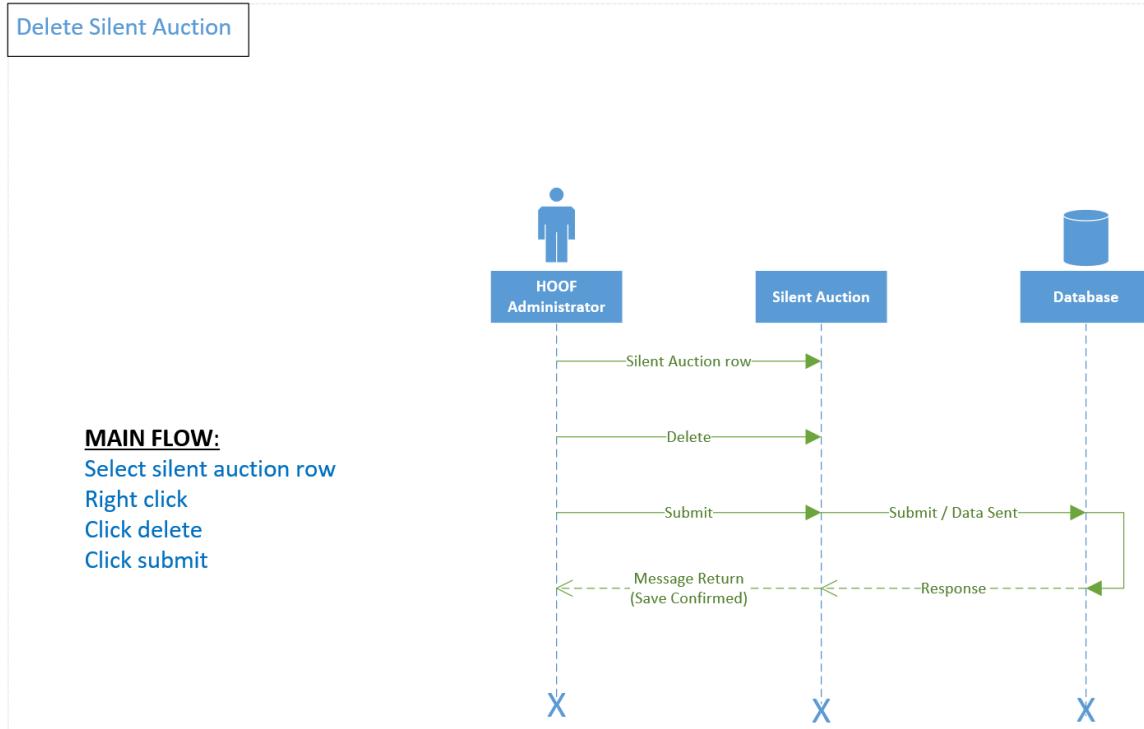
4. Pre-conditions

- Administrator must be *logged into* the Azure database
- A record for a silent auction must exist

Post-conditions

- The deletions to the auction table in the Azure database are saved

Delete Silent Auction Sequence Diagram



Use Case Specification: Add Recurring Donations

1. Add Recurring Donation

2. Brief Description

The donor will be able to add recurring donations.

3. Flow of Events

4. Basic Flow

- Donor enters Donor ID
- Donor enters First Name
- Donor enters Last Name
- Donor enters Middle initial
- Donor enters email
- Donor enters street address
- Donor enters city
- Donor enters state
- Donor enters zip code
- Donor enters amount in the Donation Amount field
- Donor selects one recurring option (One-Time, Weekly, Bi-Weekly, Monthly, Quarterly, Yearly)
- Donor enters date to start donating
- Donor enters Cardholder Name
- Donor enters Card Number
- Donor enter Card Expiration Month
- Donor enters Card Expiration Year
- Donor enters CVC
- Donor clicks submit

5. Alternative Flows

5.1 < Null error message>

If any fields are null a message will be displayed to the user prompting them to fill in the missing field(s).

5.2 < Credit Card Information >

An error message will be displayed if the Card Information is incorrect.

6. Special Requirements

- Donor must use a computer with internet access

7. Pre-conditions

- Donor must be on the donor page.

8. Post-conditions

- Donation is saved.

Add Recurring Donations Sequence Diagram



Use Case Specification: Edit Recurring Donations

1. Edit Recurring Donation

2. Brief Description

The donor will be able to edit a recurring donations.

3. Flow of Events

4. Basic Flow

- Donor edits First Name
- Donor edits Last Name
- Donor edits Middle initial
- Donor edits email
- Donor edits street address
- Donor edits city
- Donor edits state
- Donor edits zip code
- Donor edits amount in the Donation Amount field
- Donor selects one recurring option (One-Time, Weekly, Bi-Weekly, Monthly, Quarterly, Yearly)
- Donor edits date to start donating
- Donor edits Cardholder Name
- Donor edits Card Number
- Donor edits Card Expiration Month
- Donor edits Card Expiration Year
- Donor edits CVC
- Donor clicks submit

5. Alternative Flows

5.1 < Null error message>

If any fields are null a message will be displayed to the user prompting them to fill in the missing field(s).

5.2 <Left as is>

If editing a field is not necessary it can be left as is.

5.3 < Credit Card Information >

An error message will be displayed if the Card Information is incorrect.

6. Special Requirements

- Donor must use a computer with internet access

7. Pre-conditions

- Donor must be on payment portal.
- A record for a donor must exist

8. Post-conditions

- The edits to the payment portal will be saved.

Edit Recurring Donations Sequence Diagram



Use Case Specification: Delete Recurring Donations

1. Delete Recurring Donations

2. Brief Description

A donor will be able to delete recurring donations.

3. Flow of Events

4. Basic Flow

- Administrator receives request to deactivate Donor
- Administrator clicks on “Donors” tab
- Administrator finds specified Donor
- Administrator clicks “Deactivate”
- Administrator clicks yes on dialog box next to are you sure you want to deactivate donor?

5. Special Requirements

- Donor must be logged in under an administrator account
- User must be logged in under a donor account

6. Pre-conditions

- Administrator must be *logged into* the Azure database
- A record for a donor must exist

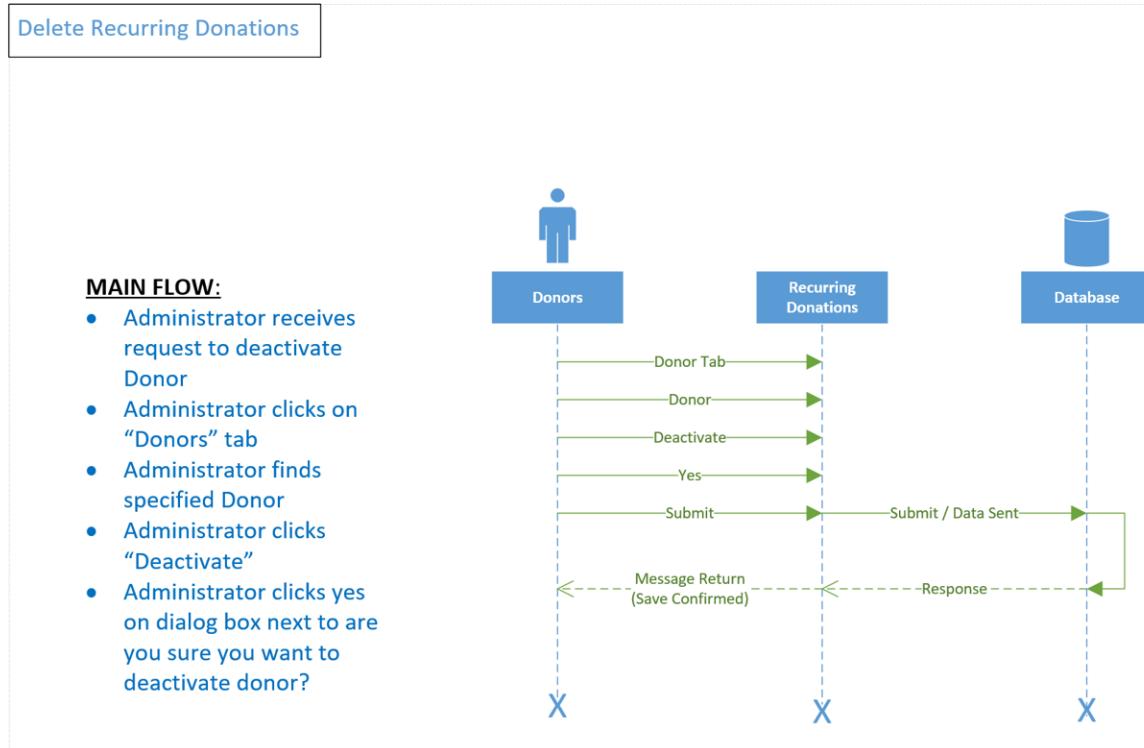
7. Post-conditions

- The deletions in the Azure database are saved

7.1 Donor Deleted

Once the administrator deactivates the account, the login no longer works. The administrator can reactivate the account at the request of the donor at any point in time and functionality returns and the donor will be prompted to update his or her information.

Delete Recurring Donations Sequence Diagram



Use Case Specification: Add a Donor

1. **Add a Donor**

2. **Brief Description**

A HOOF administrator will be able to add a Donor's information.

3. **Flow of Events**

4. **Basic Flow**

- Input donor ID
- Input donor first name
- Input donor middle initial
- Input donor last name
- Input donor email
- Input donor street address
- Input donor city
- Input donor state
- Input donor zip code
- Click Submit

5. **Alternative Flows**

6. < Middle Initial can be null >

If there is no middle initial, the middle initial field can be left null.

< Null error message>

If any fields other than the middle initial are null a message will be displayed to the user prompting them to fill in the missing field(s).

7. **Special Requirements**

- Administrator must use a computer with internet access

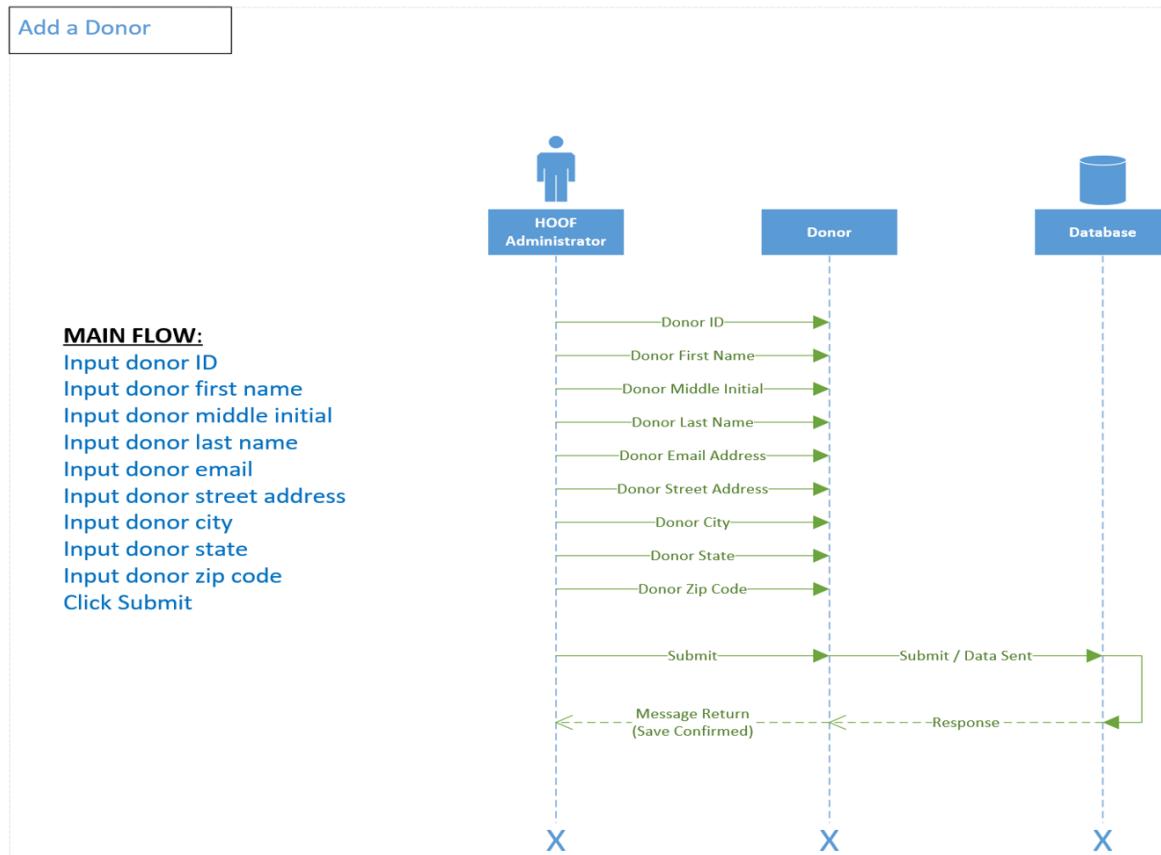
8. **Pre-conditions**

- Administrator must be logged in to Azure database.

9. **Post-conditions**

- Donor is saved.

Add a Donor Sequence Diagram



Use Case Specification: Edit a Donor

1. Edit a Donor

2. Brief Description

A HOOF administrator will be able to edit a donor's information.

3. Flow of Events

4. Basic Flow

- Edit donor ID
- Edit donor first name
- Edit donor middle initial
- Edit donor last name
- Edit donor email
- Edit donor street address
- Edit donor city
- Edit donor state
- Edit donor zip code
- Click submit

5. Alternative Flows

6. < Middle Initial can be null >

If there is no middle initial, the middle initial field can be left null.

< Null error message>

If any fields other than the middle initial are null a message will be displayed to the user prompting them to fill in the missing field(s).

7. Special Requirements

- Administrator must use a computer with internet access

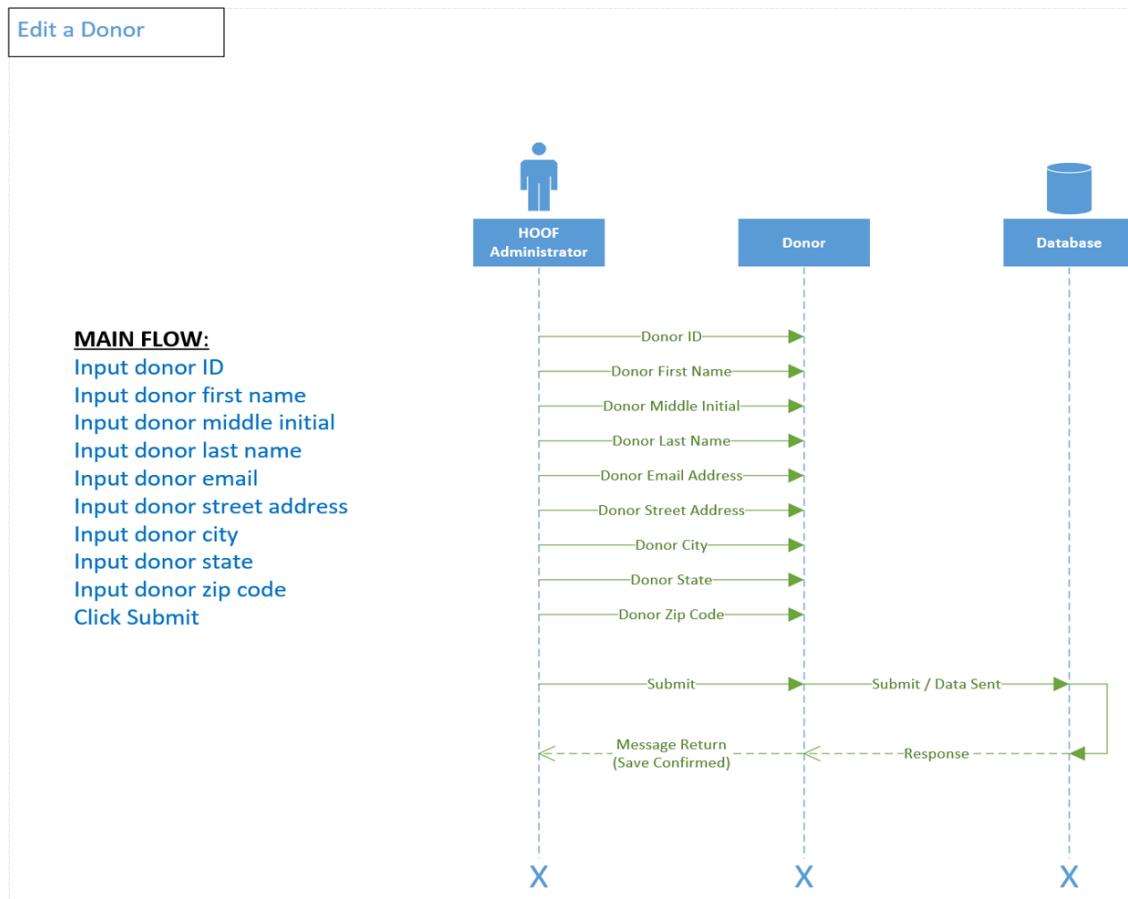
8. Pre-conditions

- Administrator must be logged into the Azure database
- A record for a donor must exist

9. Post-conditions

- The edits to the donor table will be saved in the Azure database

Edit a Donor Sequence Diagram



Use Case Specification : Add Donation

1. Add Donation

2. Brief Description

A hoof administrator will be able to add a donation from the website

3. Flow of Events

- Input first name
- Input last name
- Input middle initial
- Input email address
- Select Country from dropdown
- Input street address
- Input city
- Input state
- Input zip code
- Input Cardholder name
- Input Card number
- Input Card Expiration month
- Input Card Expiration year
- Input Card CSC
- Keep my donations anonymous checkbox blank
- Click whether donation if recurring or not
- Input donation amount
- Choose optional set donation amount from the dropdown
- Member clicks the submit button

4. Alternative Flows

< Anonymous Donation >

If the donor wishes to remain anonymous, the check box for keep my donation anonymous should be checked.

<Recurring Donations>

Click the link click here for Recurring Donations if the donor wishes to create recurring donations.

<Set Donations>

If the donor does not wish to donate a set amount, they can leave the set donation amount blank.

<Donate by check>

If the donor wishes to donate by check, they can send the check to the designated address listed on the donation page.

5. Pre-conditions

- Member must use a computer with internet access
- Member must open HOOF.org webpage and click on Donations
- Member will input their data onto the website and complete donation

6. Post-conditions

- Member must submit donation on webpage
- The new donation will receive an email at the previously specified email address to confirm

Add Donation Sequence Diagram



Use Case Specification: Add Financial Information

1. Add Financial Information

2. Brief Description

The accountant will be able to add revenues and expenses.

3. Flow of Events

4. Basic Flow

- Input foundation revenue
- Input private donations revenue
- Input online donations revenue
- Input business revenue
- Input auction revenue
- Input parking revenue
- Input miscellaneous revenue
- Input office supplies expense
- Input insurance expense
- Input program fees expense
- Input memberships expense
- Input online expenses
- Input Secretary of state tax expense
- Input miscellaneous expenses
- Click submit for expenses

5. Special Requirements

- Must have admin access
- Must have internet access
- Must have a working login
- Must have accounting permissions from admin

6. Pre-conditions

- Accountant/Administrator has navigated to HOOF's website.

7. Post-conditions

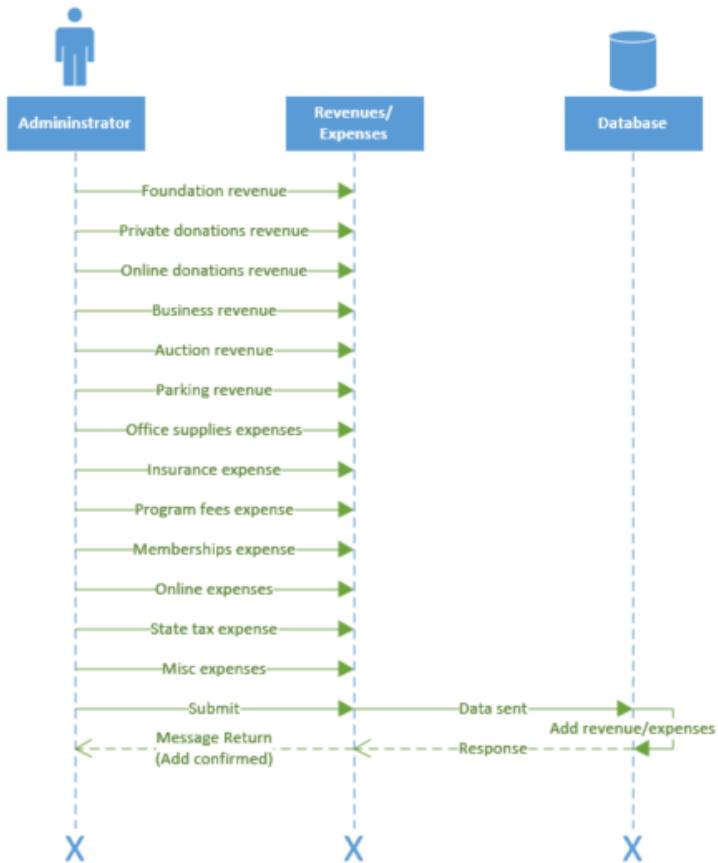
- Spreadsheet is updated with current revenues and expenditures.

SD- Add Financial information

Sequence Diagram: This is how the accountant will add financial expenses and revenue to the database

Flow

- Input foundation revenue
- Input private donations revenue
- Input online donations revenue
- Input business revenue
- Input auction revenue
- Input parking revenue
- Input miscellaneous revenue
- Input office supplies expense
- Input insurance expense
- Input program fees expense
- Input memberships expense
- Input online expenses
- Input Secretary of state tax expense
- Input miscellaneous expenses
- Click submit for expenses



Use Case Specification: Edit Financial Information

1. Edit Financial Information

2. Brief Description

The accountant will be able to edit revenues and expenses.

3. Flow of Events

4. Basic Flow

- Edit foundation revenue
- Edit private donations revenue
- Edit online donations revenue
- Edit business revenue
- Edit auction revenue
- Edit parking revenue
- Edit miscellaneous revenue
- Click submit for revenues
- Edit office supplies expense
- Edit insurance expense
- Edit program fees expense
- Edit memberships expense
- Edit online expenses
- Edit Secretary of state tax expense
- Edit miscellaneous expenses
- Click submit for expenses

5. Special Requirements

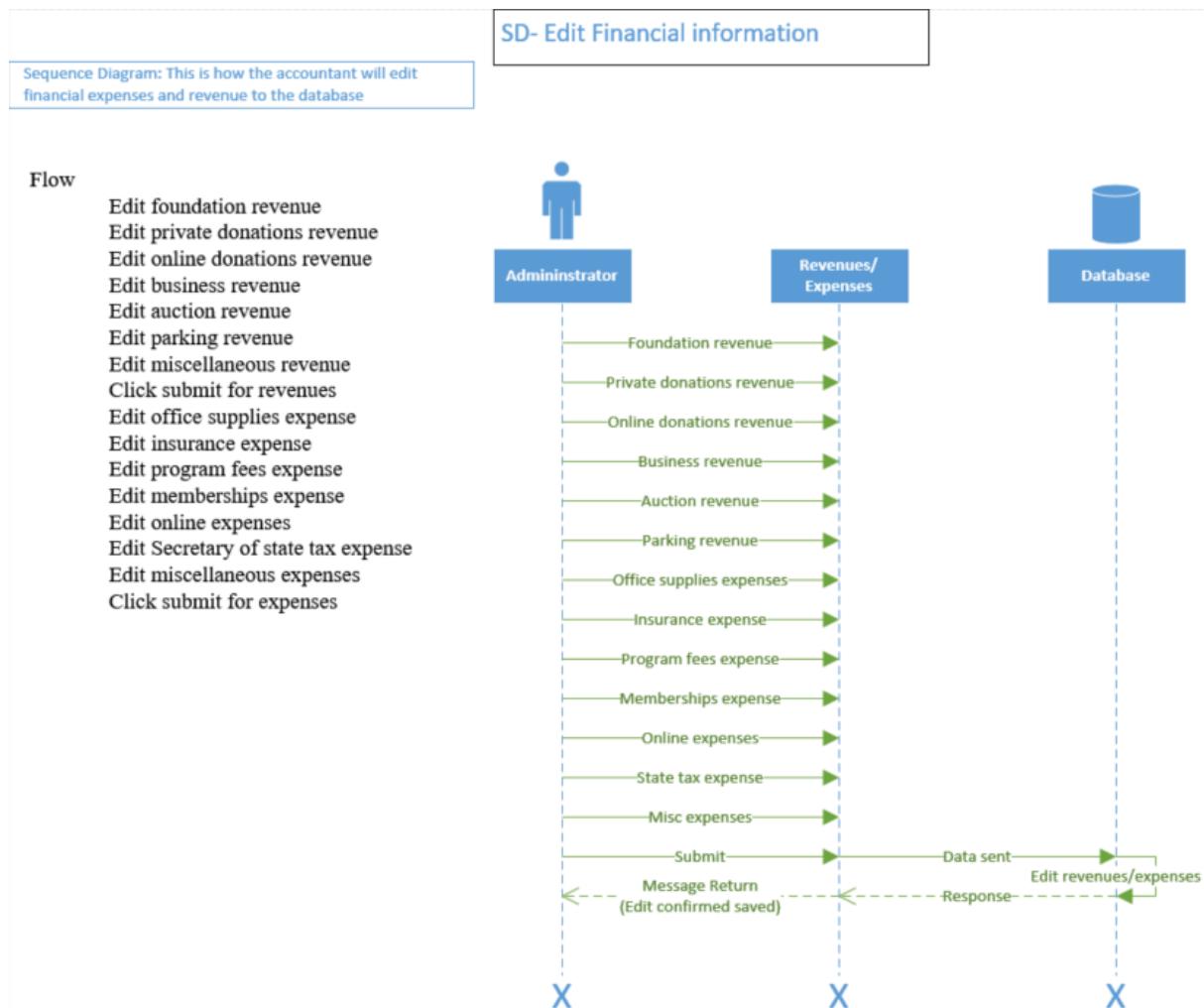
- Must have admin access
- Must have internet access
- Must have a working login
- Must have accounting permissions from admin

6. Pre-conditions

- Accountant/Administrator has navigated to HOOF's website.

7. Post-conditions

- Spreadsheet is modified with current revenues and expenses



Use Case Specification: Delete Financial Information

1. Delete Financial Information

2. Brief Description

The social media manager will be able to delete revenues and expenses.

3. Flow of Events

4. Basic Flow

- Delete foundation revenue
- Delete private donations revenue
- Delete online donations revenue
- Delete business revenue
- Delete auction revenue
- Delete parking revenue
- Delete miscellaneous revenue
- Click submit for revenues
- Delete office supplies expense
- Delete insurance expense
- Delete program fees expense
- Delete memberships expense
- Delete online expenses
- Delete Secretary of state tax expense
- Delete miscellaneous expenses
- Click submit for expenses

5. Special Requirements

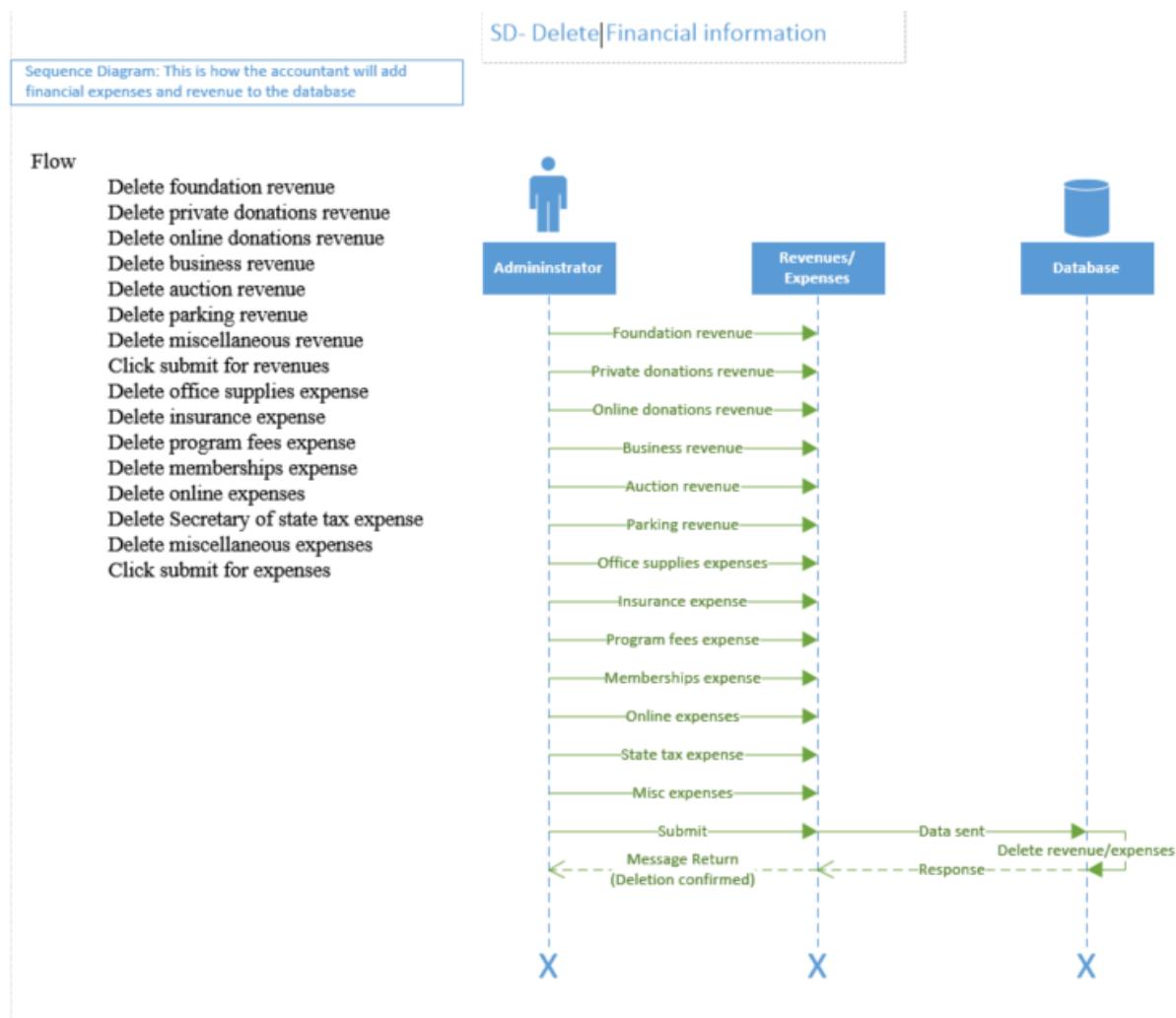
- Must have admin access
- Must have internet access
- Must have a working login
- Must have accounting permissions from admin

6. Pre-conditions

- Accountant/Administrator has navigated to HOOF's website.

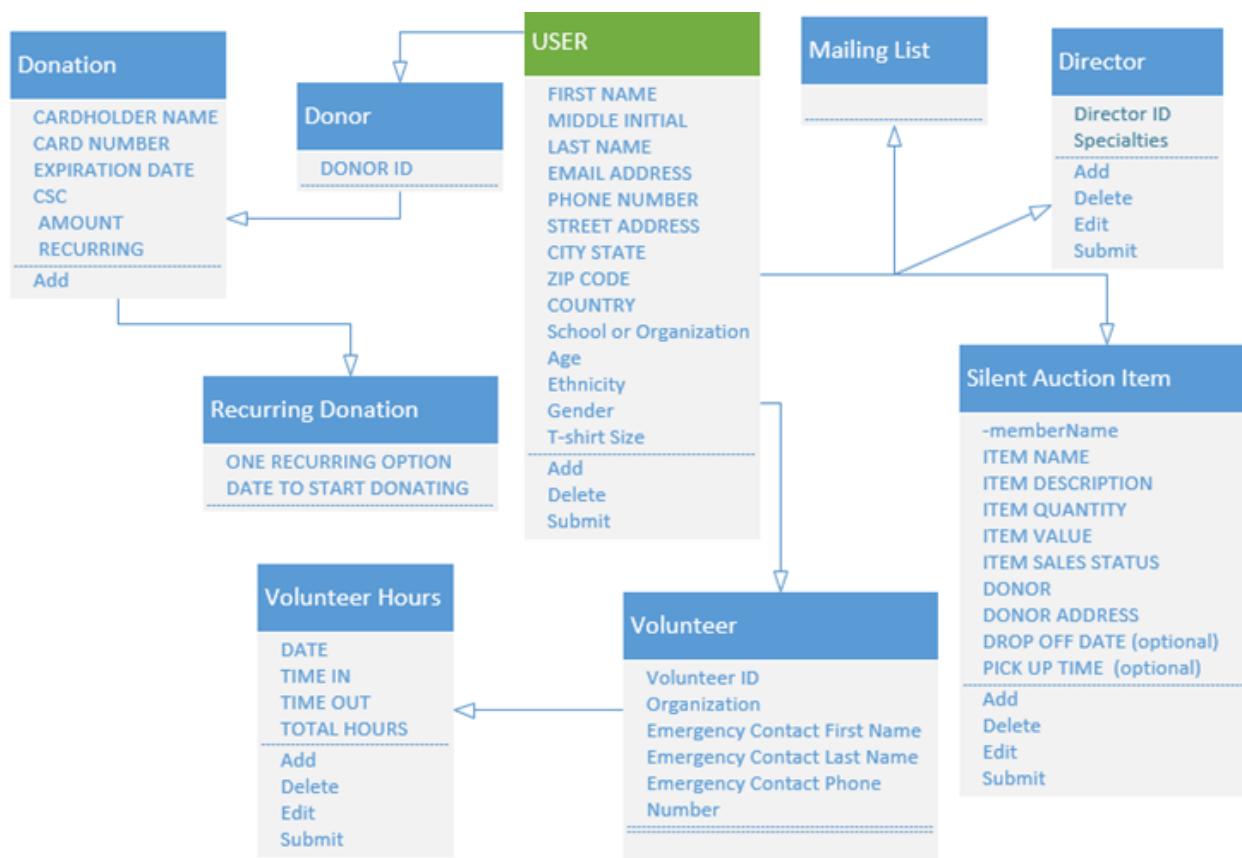
7. Post-conditions

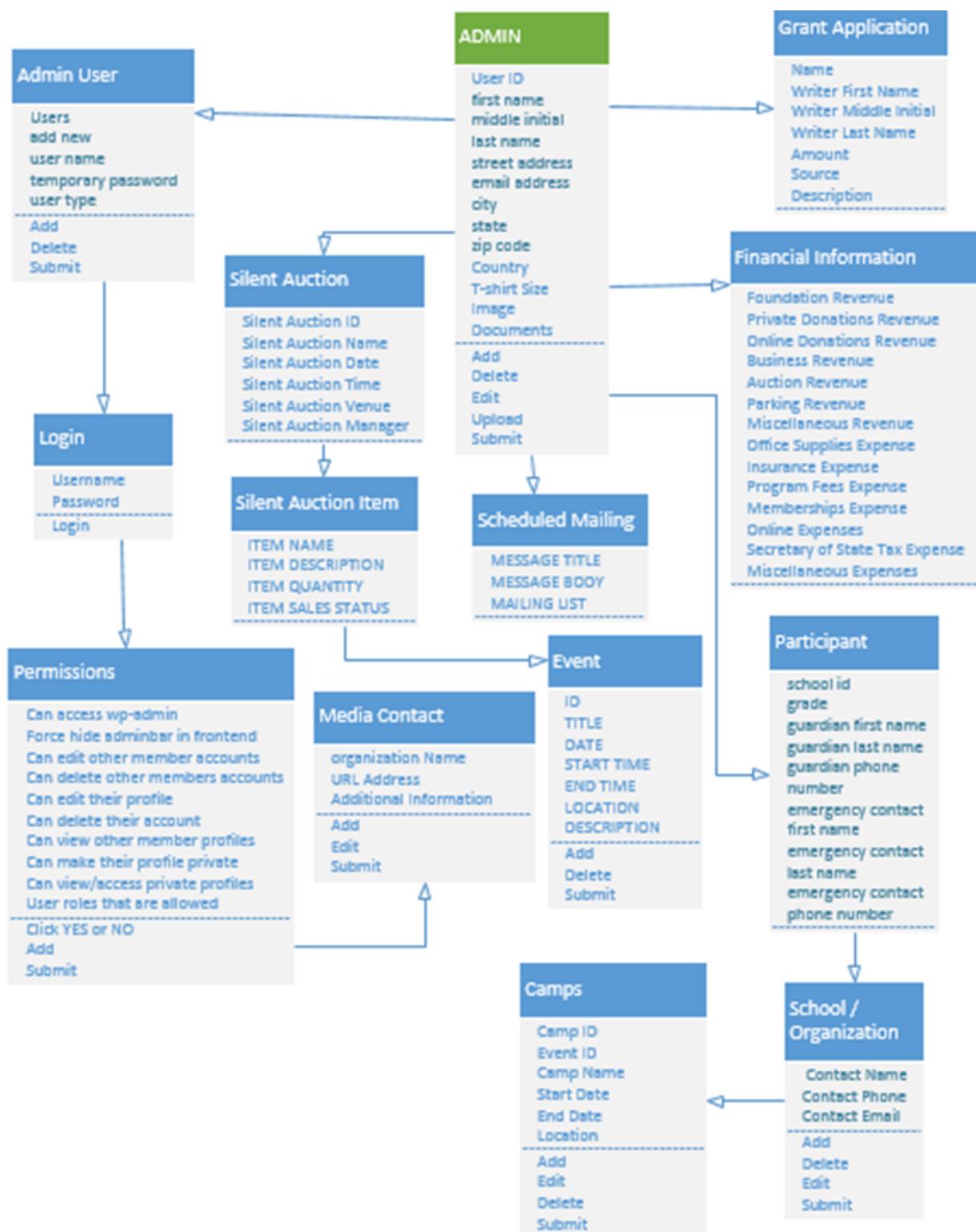
- Spreadsheet is updated with removed revenues and/or expenses.



Class Diagram

This class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations or methods, and the relationships among objects.





Database Design Decisions

The financial information class was first addressed by creating three tables to generate financial statements. Next, the super/abstract classes were used to create tables. The relationships between the class diagrams and the database tables are similar, but we adjusted the relationships for the database design according to how the table's attributes are related.

3NF Database Design Decisions

Designing the database in 3NF format involved two major changes. First, the address and contact information in the 'Donations' and 'Recurring Donations' tables was removed. This information is already in the 'Donors' table. The 'Donors' table is linked to the 'Donations' and 'Recurring Donations' tables by the 'Donors' table's primary ID: Donor ID. Having the address and contact information in all three tables is redundant and one change requires multiple updates in different locations, which increases the chance of an anomaly occurring.

The second change to place our database design in 3NF form was to create an 'Emergency Contact' table. The 'Volunteers' and 'Participants' tables contain emergency contact information: First Name, Last Name and Phone Number. It is possible for a volunteer and a participant to have the same contact information. Having emergency contact information in separate tables increases the likelihood of an anomaly, should the Last Name or Phone Number of a contact be changed.

Data Dictionary

Volunteers			
Attribute Name	Definition	Type	Size
Volunteer ID	Unique code that identifies each volunteer	Char	5 bytes
User ID - FK	Unique code that identifies each user	Char	5 bytes
First Name	First name of volunteer	Char	25 bytes
Middle Name	Middle name of volunteer	Char	2 bytes
Last Name	Last name of volunteer	Char	25 bytes
Date of birth	Date of birth for volunteer	Date	10 bytes
Street Address	Street address of volunteer	Varchar	50 bytes
City	City of volunteer	Char	25 bytes
State	State of volunteer	Char	2 bytes
Zip Code	Zip code of volunteer	Char	9 bytes
Organization	Organization for volunteer	Char	50 bytes
Phone Number	Phone number of volunteer	Char	12 bytes
Email	Email address of volunteer	Varchar	40 bytes
Total			260 bytes

Events			
Attribute Name	Definition	Type	Size
Event ID	Unique code that identifies each event	Char	5 bytes
Manager ID - FK	User ID for the manager of event	Char	5 bytes
Title	Event title	Char	25 bytes
Date	Event date	Char	10 bytes
Start time	Event start time	Char	6 bytes
End time	Event end time	Date	6 bytes
Location	Event location	Varchar	30 bytes
Description	Event description	Char	50 bytes
Total			137 bytes

Data Dictionary

<u>Donors</u>			
<u>Attribute Name</u>	<u>Definition</u>	<u>Type</u>	<u>Size</u>
<u>Donor ID</u>	Unique code that identifies each donor	Char	5 bytes
First Name	First name of donor	Char	25 bytes
Middle Name	Middle name of donor	char	2 bytes
Last Name	Last name of donor	Char	25 bytes
Email	Email address of donor	Varchar	40 bytes
Street Address	Address of donor	Varchar	50 bytes
City	City of donor	Char	25 bytes
State	State of donor	Char	2 bytes
Zip Code	Zip code of donor	Char	9 bytes
		Total	183 bytes

<u>Silent Auctions</u>			
<u>Attribute Name</u>	<u>Definition</u>	<u>Type</u>	<u>Size</u>
<u>Silent Auction ID</u>	Unique code that identifies each silent auction	Char	5 bytes
<u>Event ID - FK</u>	Event ID of the silent auction	Char	5 bytes
Name	Name of silent auction	Char	25 bytes
Date	Date of silent auction	Date	10 bytes
Time	Time of silent auction	Time	6 bytes
Venue	Venue of silent auction	Char	25 bytes
		Total	76 bytes

Data Dictionary

Silent Auction Items			
Attribute Name	Definition	Type	Size
Silent Auction items ID	Unique code that identifies each silent auction item	Char	5 bytes
Donor ID - FK	Unique code that identifies Each donor	Char	5 bytes
Silent Auction ID - FK	Unique code that identifies each silent auction	Char	5 bytes
Name	Name of item	Char	25 bytes
Description	Item's description	Varchar	50 bytes
Quantity	Quantity of item	Varchar	5 bytes
Sale Status	Sale status of item	Char	9 bytes
Sale Price	Price item sold for	Currency	7 bytes
Photo	Photo of item	Char	25 bytes
		Total	136 bytes

Data Dictionary

Directors			
Attribute Name	Definition	Type	Size
Director ID	Unique code that identifies each Director	Char	5 bytes
User ID - FK	Unique code that identifies each user	Char	5 bytes
First Name	First name of director	Char	25 bytes
Middle Name	Middle name of director	Char	2 bytes
Last Name	Last name of director	Char	25 bytes
Street Address	Address of director	Varchar	50 bytes
City	City of director	Char	25 bytes
State	State of director	Char	2 bytes
Zip Code	Zip code of director	Char	9 bytes
Phone Number	Phone number of director	Char	12 bytes
Email	Email address of director	Varchar	40 bytes
Specialty	Specialty of director	Char	50 bytes
			Total 250 bytes

Schools			
Attribute Name	Definition	Type	Size
School ID	Unique code that identifies each school	Char	5 bytes
Name	Name of school	Char	20 bytes
Street Address	Address of school	Varchar	50 bytes
City	City of school	Char	25 bytes
State	State of school	Char	2 bytes
Zip Code	Zip code of school	Char	9 bytes
Contact First Name	Contact first name	Char	25 bytes
Contact Last Name	Contact last name	Char	25 bytes
Contact Phone Number	Contact phone number	Char	12 bytes
Email	Email address of contact	Varchar	40 bytes
			Total 213 bytes

Data Dictionary

<u>Participants</u>			
<u>Attribute Name</u>	<u>Definition</u>	<u>Type</u>	<u>Size</u>
<u>Participant ID</u>	Unique code that identifies each participant	Char	5 bytes
<u>School ID - FK</u>	Unique code that identifies each school	Char	5 bytes
First Name	First name of participant	Char	25 bytes
Middle Name	Middle name of participant	Char	2 bytes
Last Name	Last name of participant	Char	25 bytes
Zip Code	Zip code of participant	Char	9 bytes
Gender	Gender of participant	Char	25 bytes
Ethnicity	Ethnicity of participant	Char	25 bytes
T-Shirt Size	T-Shirt size of participant	Char	25 bytes
Waiver Received	Indicates whether waiver for participant has been received	Char	12 bytes
		Total	158 bytes

<u>Users</u>			
<u>Attribute Name</u>	<u>Definition</u>	<u>Type</u>	<u>Size</u>
<u>User ID</u>	Unique code that identifies each	Char	5 bytes
<u>Role ID - FK</u>	Role ID of user	Char	10 bytes
First Name	First name of user	Char	25 bytes
Middle Name	Middle name of user	Char	2 bytes
Last Name	Last name of user	Char	25 bytes
Email	Email address of user	Varchar	40 bytes
Username	Username of user	Char	20 bytes
		Total	127 bytes

Data Dictionary

<u>Grants</u>			
<u>Attribute Name</u>	<u>Definition</u>	<u>Type</u>	<u>Size</u>
<u>Grant ID</u>	Unique code that identifies each grant	Char	5 bytes
<u>Writer ID - FK</u>	Writer's user ID	Char	10 bytes
Name of Grantor	Name of each grant	Char	25 bytes
Name of Grant	Amount of each grant	Char	7 bytes
Amount	Source of each grant	Char	20 bytes
Minimum Due Date	Description of each grant	Char	50 bytes
Completed or In Progress	Whether the grant is complete or in progress	Char	3 bytes
Funded	Whether the grant has been funded	Char	3 bytes
Start Date	Date the grant begins	Char	10 bytes
End Date	Date the grant ends	Char	10 bytes
Amount Awarded	Amount of Grant Funded	Char	6 bytes
			Total 149 bytes

<u>Logins</u>			
<u>Attribute Name</u>	<u>Definition</u>	<u>Type</u>	<u>Size</u>
<u>Login ID</u>	Unique code that identifies each login	Char	5 bytes
<u>User ID - FK</u>	Unique code that identifies each user	Char	5 bytes
Username	Username of each user	Char	20 bytes
Password	Password of each user	Char	15 bytes
			Total 45 bytes

Data Dictionary

<u>Mailing List</u>			
<u>Attribute Name</u>	<u>Definition</u>	<u>Type</u>	<u>Size</u>
<u>Mailing ID</u>	Unique code that identifies each mailing list.	Char	5 bytes
<u>Donor ID - FK</u>	Unique code that identifies each donor.	Char	5 bytes
First Name	First name of donor	Char	25 bytes
Middle Name	Middle name of donor	Char	2 bytes
Last Name	Last name of donor	Char	25 bytes
Email Address	Email address of donor	Varchar	40 bytes
Street Address	Address of donor	Char	50 bytes
City	City of donor	Char	25 bytes
State	State of donor	Char	2 bytes
Zip Code	Zip code of donor	Char	9 bytes
		Total	188 bytes

<u>Media Contacts</u>			
<u>Attribute Name</u>	<u>Definition</u>	<u>Type</u>	<u>Size</u>
<u>Media Contact ID</u>	Unique code that identifies each media contact	Char	5 bytes
<u>Liaison ID - FK</u>	User ID for liaison	Char	5 bytes
Organization	Organization of contact	Char	20 bytes
First Name	First name of contact	Char	25 bytes
Last Name	Last name of contact	Char	25 bytes
Title	Title of contact	Char	20 bytes
Phone Number	Phone number of contact	Char	12 bytes
Email Address	Email address of contact	Varchar	40 bytes
Website	Website of contact	Char	25 bytes
		Total	177 bytes

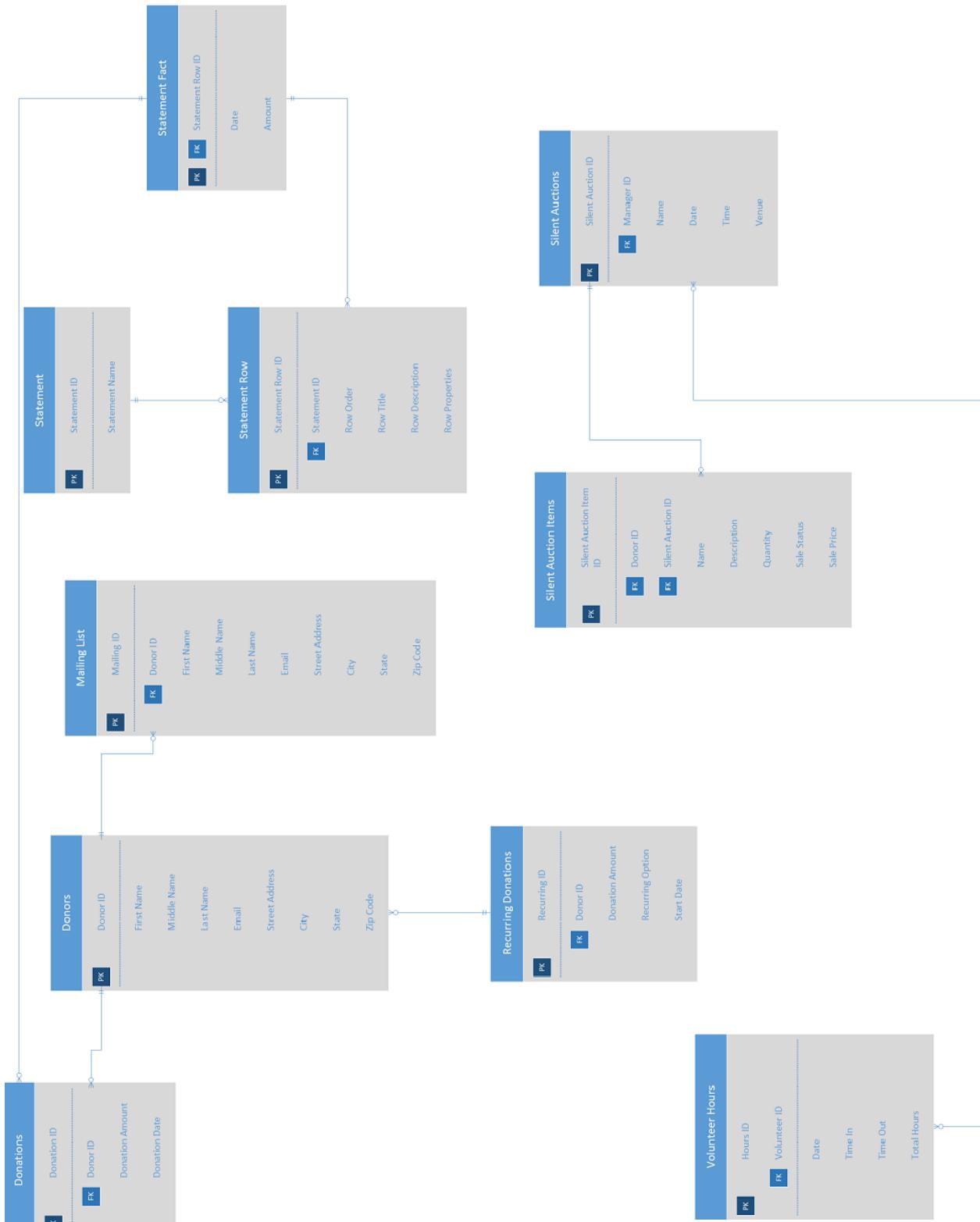
Data Dictionary

<u>Donations</u>			
<u>Attribute Name</u>	<u>Definition</u>	<u>Type</u>	<u>Size</u>
<u>Donation ID</u>	Unique code that identifies each donation	Char	5 bytes
<u>Donor ID - FK</u>	Unique code that identifies each donor	Char	5 bytes
Donation Amount	Amount of donation	Char	20 bytes
Donation Date	Date of donation	Date	10 bytes
		Total	40 bytes

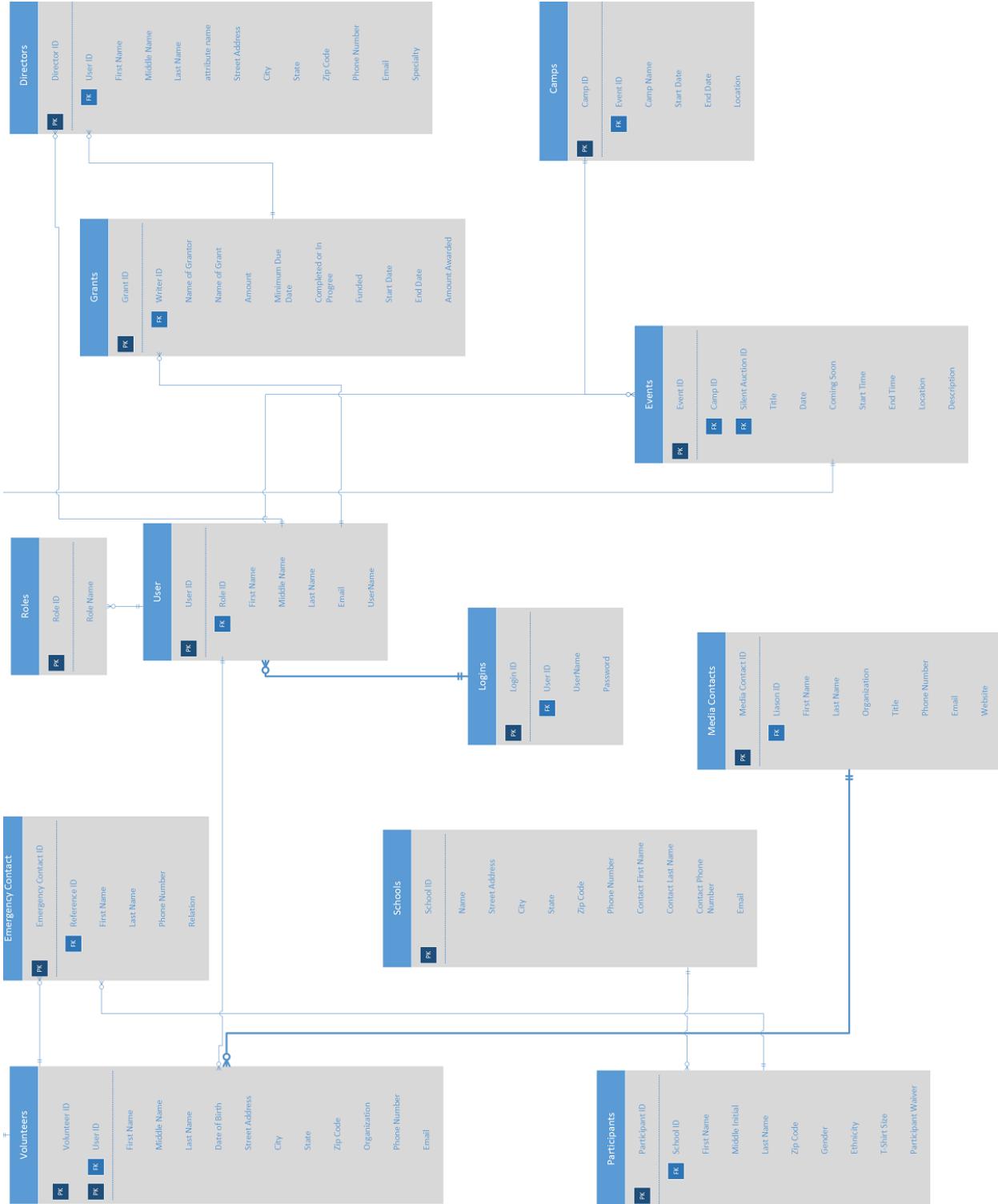
<u>Roles</u>			
<u>Attribute Name</u>	<u>Definition</u>	<u>Type</u>	<u>Size</u>
<u>Role ID</u>	Unique code that identifies each role	Char	5 bytes
Role Name	Name of role	Char	20 bytes
			Total 25 bytes

<u>Camps</u>			
<u>Attribute Name</u>	<u>Definition</u>	<u>Type</u>	<u>Size</u>
<u>Camp ID</u>	Unique code that identifies each camp	Char	5 bytes
<u>Event ID - FK</u>	Unique code that identifies each event	Char	5 bytes
Camp Name	Name of camp	Char	20 bytes
Start Date	Date camp starts	Char	10 bytes
End Date	Date camp ends	Char	10 bytes
Location	Location of camp	Char	20 bytes
			Total 70 bytes

Entity Relationship Diagram (ERD)



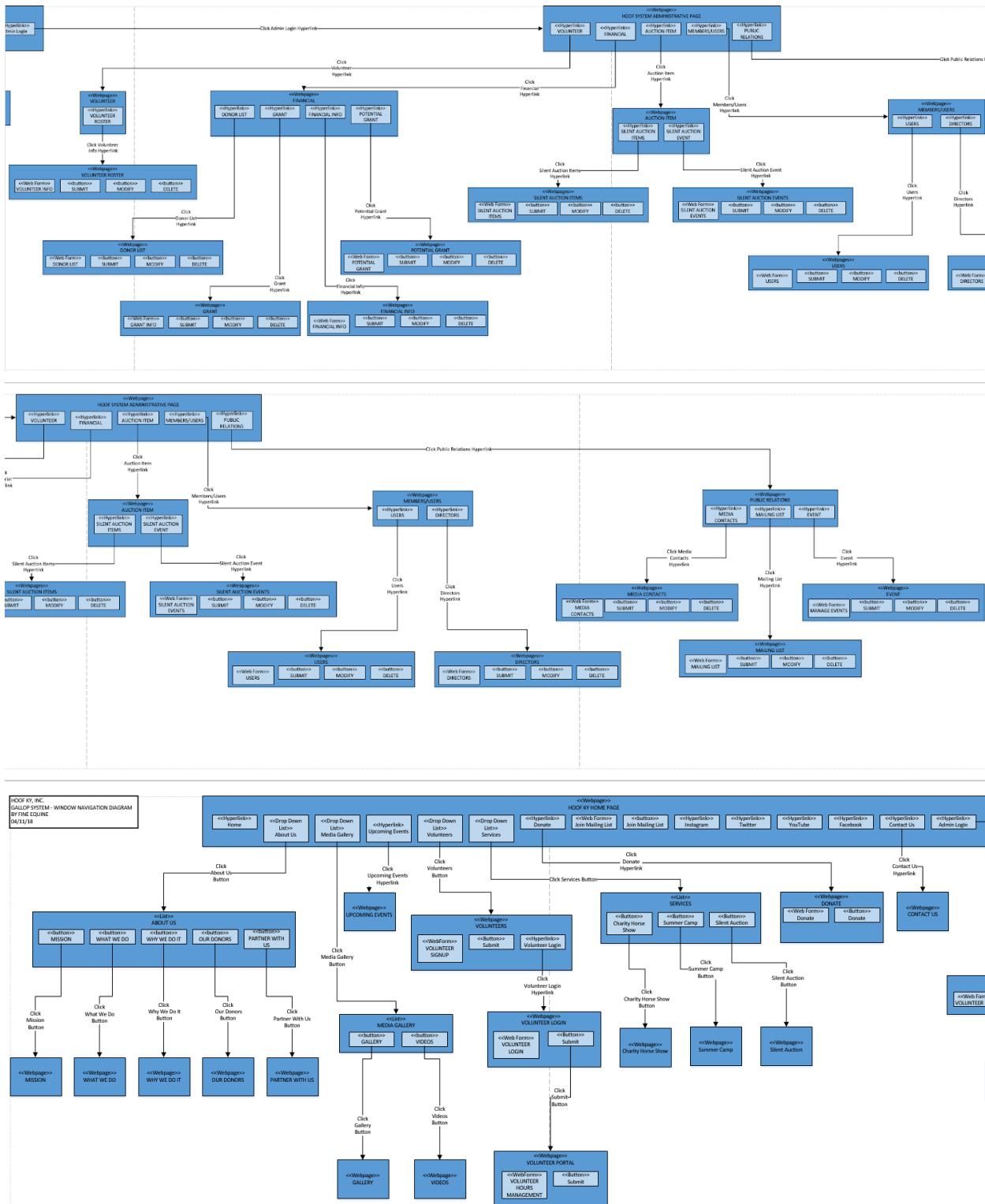
Entity Relationship Diagram (ERD)



User Interface Navigation Design Narrative

This diagram is a representation of the website and how a user will navigate through the website. Each box of this diagram represents a screen that we believe needs to be featured on the HOOF website. The arrows represent the navigation through the website via a click of the mouse. The screens mentioned in this user interface navigation diagram can either already be found in the HOOF website or can be found in the prototypes that are featured in this iteration.

User Interface Navigation Diagram



Screen Layouts

We have included several prototypes that illustrate how we intend to meet the requirements that we have listed in the system requirements of this specification. We will list the prototypes here:

About us Page - This is the 'About Us' Page. If you hover towards About Us, it will show Mission, What we Do?, Why We Do?, Partner with Us, Our Donors tabs and you can select one to know in detail about a particular tab.

Media Page - This is the 'Media Page' of the website where the viewers can see the photos and videos for the events. You can view pictures and videos from past years as well.

Calendar Prototype - This prototype displays the upcoming events in the given month in a calendar format for an easy view for the viewers. This prototype is displayed under the "Upcoming Events" tab in the website. Events will be found under the corresponding day in the calendar that is hyperlinked to a more detailed webpage of the event. Also, the location of the event will be displayed alongside the calendar.

Volunteer Prototype - This prototype displays a volunteer form for the new volunteers to sign up for HOOF. The volunteers who have been volunteering or who has already signed up once can easily sign in under the "Already a volunteer? Click here" link where they can store in their days and hours they worked. Potential volunteers can easily and safely input their information into the form and submit it to the HOOF database that will be managed. This will allow volunteer information to be transferred from the web page to the address with ease and simplicity.

Services Prototype - This prototype has three tabs which shows 'Charity Horse Show' , 'Silent Auction' and 'Summer Camp' that are the services HOOF provides. You can select a tab if you want to know in detail about a particular service.

Donate Prototype - This prototype displays the donation form. This method allows potential donors to donate by providing their name, email, country, address, city, state, zip and payment information. It will ask for the amount once you will click the donate button. This prototype will be found under the "Donate" tab in the website. You can also share your donation on social media and let your friends know about your donation.

The screenshot shows the homepage of the Horses Offering Opportunities for the Future, Inc. (HOOF KY) website. At the top left is the HOOF KY logo, which features a blue outline of Kentucky with a silhouette of a person riding a horse inside it. To the right of the logo is the organization's name in large, bold, blue capital letters: "HORSES OFFERING OPPORTUNITIES FOR THE FUTURE, INC.". Below the name is a navigation bar with links: HOME, ABOUT US, MEDIA, UPCOMING EVENTS, VOLUNTEER PORTAL, SERVICES, and DONATE. The "ABOUT US" menu is currently open, showing sub-links: MISSION, WHAT WE DO?, WHY WE DO?, PARTNER WITH US, and OUR DONORS. The main content area features a photograph of two young people riding horses. On the left, a boy in a blue shirt rides a brown horse. On the right, a girl in a red t-shirt with "ETID C 82" printed on it rides a white horse. To the right of the photo is a section titled "Horses Offering Opportunities for the Future, Inc." with a brief mission statement: "The mission of Horses Offering Opportunities for the Future, Inc. (HOOF KY) is to connect at-risk, disadvantaged, and trauma-stricken youth with horses and let horses work their magic." Below this is another text block: "Horses have the ability to touch the human spirit in ways not found in other relationships." At the bottom right of the content area is a large orange button with the text "DONATE NOW". At the very bottom of the page are social media icons for Instagram, Twitter, and Facebook, along with input fields for an email address and a "Join our mailing list" button, and contact information: P.O.Box 1303, Prospect, KY 40059, 502-558-732, Contact Us, and Admin Login.

This screenshot shows the same website layout as the first one, but with three photographs displayed below the main content area. The first photograph on the left shows a group of five young women standing outdoors next to two horses, one white and one dark-colored. The second photograph in the center shows a young boy wearing a helmet and a woman in a red t-shirt with the HOOF KY logo riding a brown horse. The third photograph on the right shows a large group of approximately 20 people, mostly young, all wearing matching blue t-shirts, posing for a group photo in a grassy field. Below each photograph is a link: "Click for more pictures 2018" under the first photo, "Click for more pictures 2017" under the second photo, and "Click for more pictures 2016" under the third photo.

The screenshot shows the website's main menu at the top with links for HOME, ABOUT US, MEDIA, UPCOMING EVENTS, VOLUNTEER PORTAL, SERVICES, and DONATE. Below the menu, there are two tabs: "All Videos" and "VIDEOS", with "VIDEOS" being the active tab. A grid of video thumbnails is displayed, each with a play button, duration (e.g., 0:43, 3:54, 0:40, 0:09), views (e.g., 121 views - June 30, 2017, 149 views - June 30, 2017, 14 views - June 30, 2017, 165 views - June 23, 2017), and a timestamp.

Thumbnail	Duration	Views	Date
	0:43	121 views	June 30, 2017
	3:54	149 views	June 30, 2017
	0:40	14 views	June 30, 2017
	0:09	165 views	June 23, 2017
	0:11	7 views	June 23, 2017
	0:10	7 views	June 23, 2017
	0:15	8 views	June 23, 2017
	0:15	6 views	June 23, 2017
	0:13	9 views	June 23, 2017
	0:26	8 views	June 23, 2017
	0:14	26 views	June 23, 2017
	0:12	25 views	June 23, 2017

The screenshot shows the website's main menu at the top with links for HOME, ABOUT US, MEDIA, UPCOMING EVENTS, VOLUNTEER PORTAL, SERVICES, and DONATE. To the right of the menu, there is a sidebar with the text "Horses Offering Opportunities for the Future, Inc." and "Upcoming Events". Below that, there are sections for "Date/Time:", "Meet the instructors" events, and "Location:".

2018 MARCH						
Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
				1	2	3
4	5	6	7	8	9	10 Meet the instructors
11	12	13	14	15	16	17
18 Meet the Horses	19	20	21	22	23	24
25	26	27	28	29	30	31

© BlankCalendarPages.com

Horses Offering Opportunities for the Future, Inc.

Upcoming Events

Date/Time:
Date: 3/10/2018
Time: 2:00 pm - 5:00 pm
Type of Event: Meet the instructors

Date/Time:
Date: 3/18/2018
Time: 2:00 pm - 5:00 pm
Type of Event: Meet the Horses

Location:
Horses Offering Opportunities for the Future, Inc. -HOOF KY
40059 Prospect, United States

 **HORSES OFFERING OPPORTUNITIES FOR THE FUTURE, INC.**

HOME ABOUT US MEDIA UPCOMING EVENTS VOLUNTEER PORTAL SERVICES **DONATE**

Volunteer Form

Thank you for your interest in volunteering with Horses Offering Opportunities for the Future, Inc. Please take a moment to fill out our volunteer application. Once your application is received you may be contacted by HOOF staff with more information. Information is collected in a secure way and is not shared with anyone outside of our organization.

[Already a Volunteer? Click here](#)

First Name

Last Name

Home Address

City, State, Zip

Email Address

Phone

School Name/
Organization (Optional)

Date 1/5/2013 12 ▾ 00 ▾ AM ▾ Now

Start Time: 12/29/2012 9 ▾ 07 ▾ AM ▾ Now

End Time: 12/29/2012 9 ▾ 07 ▾ AM ▾ Now

[Click here to receive HOOF news and upcoming events.](#)

If you have any questions, please email hoofky@gmail.com


**Horses Offering Opportunities
for the Future, Inc.**



HORSES OFFERING OPPORTUNITIES FOR THE FUTURE, INC.

[HOME](#)
[ABOUT US](#)
[MEDIA](#)
[UPCOMING EVENTS](#)
[VOLUNTEER PORTAL](#)
[SERVICES](#)
[DONATE](#)

[CHARITY HORSE SHOW](#)
[SILENT AUCTION](#)
[SUMMER CAMP](#)

JULY 21, 2018 KENTUCKY STATE FAIRGROUNDS!

ACADEMY DIVISION STARTS AT 9:00 A.M.

- 1) Academy Pleasure Driving
- 2) Academy Pleasure 14-17 WTC
- 3) Academy Equitation 14-17 WTC
- 4) Academy Pleasure Adult WTC
- 5) Academy Equitation Adult WTC
- 6) Academy Pleasure 13 & under WTC
- 7) Academy Equitation 13 & under WTC
- 8) Academy Equitation Work-Out WTC (all ages)
- 9) Academy Pleasure 14-17 WT
- 10) Academy Equitation 14-17 WT
- 11) Academy Pleasure Adult WT
- 12) Academy Equitation Adult WT
- 13) Academy Pleasure 11-13 WT
- 14) Academy Equitation 11-13 WT
- 15) Academy Pleasure 9-10 WT
- 16) Academy Equitation 9-10 WT
- 17) Academy Pleasure 8 & under WT
- 18) Academy Equitation 8 & under WT
- 19) Academy Equitation Work-Out WT (all ages)
- 20) Academy Lead Line Horsemanship WT
- 21) Academy Lead Line Equitation WT (with cone pattern)

ALL Academy entries will be due on July 18 to Betsy Webb at louisvillequestriancenter@gmail.com

Academy Work-Out Classes will be judged solely on a Work-Out only—Riders will enter the arena and line-up and then will be called out individually to perform work-out. All riders entered in Work-Out class will perform individual work-out. Work-Out will be posted on Face Book page by July 1st.

Academy High Point Awards will be given out to each Academy division. Points will be tabulated for each rider entered in their 3 respective classes (Equitation, Pleasure and Work-Out). This will be similar to how the Hunter riders are given out awards at their shows. Points are awarded for each placing in each of the rider's 3 respective classes, the rider with the highest points/core will be named High Point rider for their division. In both the WTC and the WT divisions, a High Point Award and Reserve High Point award will be awarded. For WTC there will be 14-17, 13 & under and Adult, High Point and Reserve High Point awards (for any ties, both riders will receive awards and recognition!). For the WT division, there will be High Point and Reserve High Point awards for the following divisions: 14-17, Adult, 11-13, 9-10, and 8 and under.

Awards and recognition will be given out for the WTC High Point divisions at the secretary's desk within 30 minutes of the last WTC class (to give time for tabulation), and the WT High Point Awards will be given out at the secretary's desk within 30 minutes of the last WT class. There are no High Point Awards given for the lead line riders.

All Stalls: \$45 must be prepaid by July 15
Academy Entries: \$20
Performance Entries
Box Seats: (There is limit seating, it is recommended stables purchase a Box Seat of 8 Seats)

PERFORMANCE DIVISION STARTS AT 3:00 P.M.

- 22) Walk-Trot Pleasure, Rider 12 years and under
- 23) Country Pleasure Driving
- 24) Open Road Pony
- 25) 3-Gaited Show Pleasure, Novice Rider
- 26) Hunt Seat Pleasure (all breeds)
- 27) Novice Saddle Seat Equitation (with work-out posted at show)
- 28) Adult 3-Gaited Show Pleasure
- 29) 3-Gaited Park Pleasure
- 30) Country Pleasure, Novice Rider
- 31) Saddle and Bridle Hunt Seat Classic
- 32) Bluegrass Select 3-Gaited Pleasure
- 33) Open Fine Harness
- 34) Hackney Pony Pleasure Driving
- 35) Show Pleasure Driving
- 36) Western Country Pleasure (all breeds)
- 37) Bluegrass Select Country Pleasure
- 38) Open 3-Gaited Park
- 39) Classic Pleasure (the purpose of this class is to create a division to offset the progression of the Country Pleasure division which then left a void for the not so animated horse and the entry level rider—this class is designed for the horse that would likely not ribbon in the country pleasure class) this class will flat walk, pleasure trot, extend the trot, canter and halt and back—horses may cross enter if they meet the class requirements
- 40) Open 3-Gaited
- 41) Shetner Western Pleasure
- 42) Open Harness Pony
- 43) Adult Country Pleasure
- 44) Juvenile 3-Gaited Pleasure
- 45) Open 5-Gaited
- 46) Open English Pleasure (all breeds, horses will be asked to walk, pleasure trot, extend the trot, canter and back in line-up—horses may cross enter from other classes—open to amateur and juvenile riders only)
- 47) Open Hackney Pony
- 48) Senior Horse English Pleasure (all breeds, horses must be at least 17 years old, horses will be judged on suitability as a pleasure horse at the flat walk, pleasure trot and canter and will back in line up, open to amateur and juvenile riders only)
- 49) 5-Gaited Pleasure

There will be a break approximately mid-point of show schedule for Arena maintenance.

ALL Proceed Benefit HOOF KY
(Horses Offering Opportunities for the Future)

For More Info and Stall Reservations
Show Manager: Betsy Webb (502) 477-0830

Fine Equine

CIS 320-02

Dr. Barker

The screenshot shows the homepage of the Horses Offering Opportunities for the Future, Inc. website. At the top, there is a logo featuring a horse head and the text "HOOF KY". Below the logo, the main title "HORSES OFFERING OPPORTUNITIES FOR THE FUTURE, INC." is displayed in large blue letters. A navigation bar with links for "HOME", "ABOUT US", "MEDIA", "UPCOMING EVENTS", "VOLUNTEER PORTAL", "SERVICES", and "DONATE" is visible. The "DONATE" button is highlighted in orange. On the left side, there are three photographs of groups of people, likely camp participants, wearing blue t-shirts. The central content area features a blue header box with the text "SUMMER CAMP". Below this, a section titled "HOOF KY partners with the Louisville Equestrian Center to provide:" lists various program details. To the right of this list are three small boxes: "CHARITY HORSE SHOW", "SILENT AUCTION", and "SUMMER CAMP".

SUMMER CAMP

HOOF KY partners with the Louisville Equestrian Center to provide:

- Weekly summer Equine Academies each serving 30 at-risk youth ages 9-18. Academies run from 10 AM- 4:00 PM daily.
- Horses, ponies, and their tack chosen for safety, manners and suitability to riders.
- Safety riding helmets.
- Highly qualified American Riding Instructors Association (ARIA) certified instructors who have years of experience working with all ages and abilities of riders.
- Rotating stations whereby participants learn riding, equine education and barn management. Competitive equine games and horse themed crafts are included.
- Daily recorded individual participant competency evaluations.
- Spacious indoor and outdoor arenas.
- Logo T-shirts for each participant.
- Awards Ceremony with certificates, individual pictures, and equine mementos.
- Safe facility, First Aid trained counselors, and liability insurance.

Participants will:

- Follow all counselor instructions to ensure safety for participants and horses.
- Wear appropriate clothing, shoes, and safety gear.
- Refrain from aggression and bullying.
- Learn to respect and care for other living creatures and mother earth.

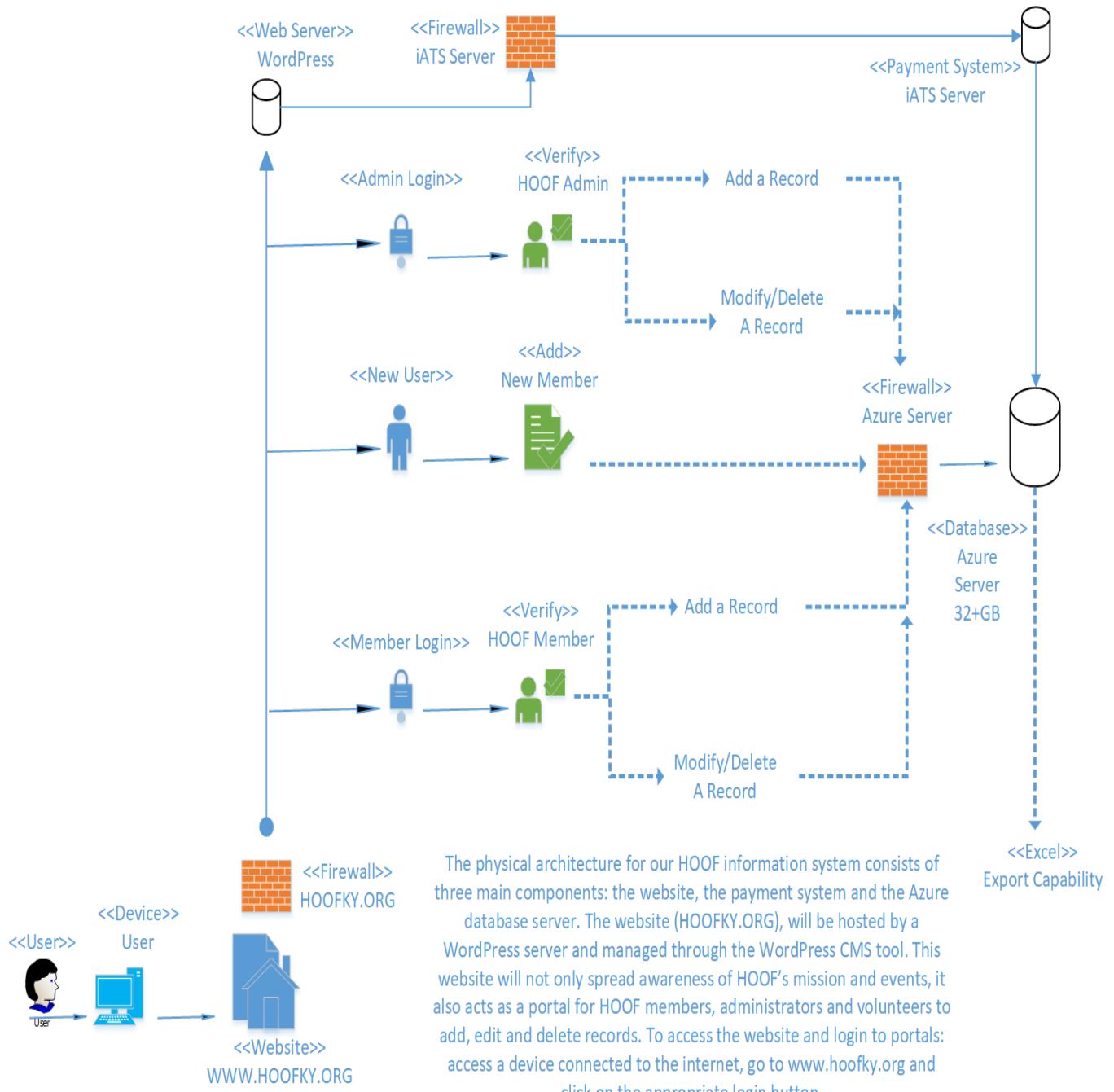
Physical Architecture Design Decisions

The physical architecture for our HOOF information system consists of three main components. First, the website (HOOFKY.ORG), will be hosted by a WordPress server and managed through the WordPress CMS tool. This website will not only spread awareness of HOOF's mission and events, it also acts as a portal for HOOF members, administrators and volunteers to add, edit and delete records. To access the website and login to portals: access a device connected to the internet, go to www.hoofky.org and click on the appropriate login button.

Instead of hosting important data for HOOF on the WordPress server, we decided to integrate a Microsoft Azure cloud database. (The second main component of our physical architecture design) The HOOFKY.ORG website has multiple forms that will allow data to be input into the Azure database. By logging into the Azure database directly, HOOF will be able to generate reports based on the data in the database. The use of the Azure database server will allow for better quality business analytics and allows HOOF information to be more secure than the WordPress server, at a minimal cost.

The third main component is the iATS Payment system, who have specialized in processing donations for non-profit organizations for the past twenty years. The donors will go to the iATS payment portal from the HOOFKY.ORG website and the donor data will be sent to the Azure database for storage on donor and donation information. Credit card information will never be stored by HOOF.

Physical Architecture Diagram



Design Procedures for Security Concerns and Non-functional Requirements

This is a general overview of how the security requirements will be constructed and interspersed into the proposed system. It will account for premeditated as well as random event circumstances.

Type of Requirement	Definition	Implementation
System Value Estimate	Estimated business value of the system and its data	<ul style="list-style-type: none">Since the system data will be backed up to a Cloud server, it will be relatively inexpensive to recover it.
Access Control Requirements	Limitations on who can access what data	<ul style="list-style-type: none">Only site admins will be able to make changes to pictures, member data, newsletters, etc.
Encryption/Authentication Requirements	Defines what data will be encrypted, and where and when authentication will be needed for user access	<ul style="list-style-type: none">Data will be encrypted from the user's computer to the website to provide secure payments and form ordersAnyone logging into the site will be required to authenticate.
Virus Control Requirements	Requirements to control the spread of viruses	<ul style="list-style-type: none">All uploaded files and forms will be scanned for viruses before being saved into the system.

Database Backups

The GallOp system database is to be backed up routinely to mitigate against any circumstances that could jeopardize the business operations of HOOK KY if the operational database were to be compromised. For example, natural disasters, database corruption, hardware failure by the cloud host could result in lost data that would likely have an adverse and costly outcome for any organization.

Fine Equine has researched the backup capabilities of Microsoft Azure whom it has recommended as a reliable host for cloud database services. Microsoft Azure automatically performs a full backup of the database it hosts on a weekly basis. It performs database transaction log backups on much more frequent basis; occurring every 5-10 minutes considering the activity levels of the database.. These database files are retained for a period of time depending on the contractual service level between the service and HOOF KY.

These database backups provide HOOF KY with different options in case of a failure. If the need arises for the restoring of a database backup, HOOF KY has the option of restoring the database to a specific point in time within the retention period mentioned above. For example, if the retention period is one month, HOOF KY would be able to restore the system to the state that the system was in within that period.

In simpler terms, if a backup occurred 27 days ago and every 7 days after that, it would be available for restoration for HOOF's use, as would a backup from 20, 13, and 6 days ago. This gives a great degree of flexibility for the recovery of a database system. For example, if a database is corrupted 14 days ago, the database backup from 13 and 6 days ago may no longer be desirable unless the data can be analyzed and corrected. If it is beyond correction, then there is the option of resuming operations with the database from 20 or 27 days ago.

A strong feature of selecting Microsoft Azure as the cloud database host, is their robust geographical footprint. They have locations geographically dispersed which allows HOOF the option of relocating the actual server location easily if the primary and initial location were to be compromised. For example, if the primary location is in California and the Azure server location is compromised by an earthquake, HOOF could relocate its database to a server in Kansas. If the Kansas location is compromised by a tornado, HOOF could then relocate its database to another location that is perhaps not as prone to natural disasters.

The backup service is a built in feature of the Azure SQL Database service and is included in the overall price. Additional features can be added and may be contingent upon the frequency, size, or retention period of the data. Because it is automated, it is relatively low maintenance, but a HOOF administrator should keep in mind that back ups are occurring and are available if needed and be knowledgeable about the scheduling and frequency of the process and how to implement a restoration if needed.

Database Encryption

Encrypting databases for Hoof KY is our utmost priority. For many organizations, data encryption is a compulsory step towards data privacy, compliance and data sovereignty. Encryption will be a crucial concern for Hoof KY because this organization will be handling very sensitive information including volunteer's, donors, and especially children's data.

WordPress Encryption: Every WordPress site should use an SSL certificate and force the use of HTTPS. This is a crucial first step to securing data. Also, this is going to require an encryption key that needs to be warehoused somewhere, probably wp-config.php. Ensuring no one can access it is very imperative and requires appropriate security for your server. We will take several steps to ensure the WordPress site is encrypted including but not limited to these 4 main steps:

1. Securing the login page and averting attacks by adding “/wp-login.php” or “/wp-admin/” at the end of domain name:
 - a. setting up website lock-down and forbidding users.
 - b. using 2-factor-authentication.
 - c. altering passwords and prompting changing them frequently.
2. Locking admin dashboard:
 - a. safeguarding wp-admin directory.
 - b. using SSL to encrypt documents.
 - c. adding user accounts with vigilance.
 - d. fluctuating the admin username and using an iThemes Security plugin to stop hacking endeavors with IP addresses trying to use username “admin”.
 - e. keep an eye on files.
3. Fortifying the database:
 - a. systematic backups.
 - b. shifting WordPress database table prefix to something unique like “mywp-“ or “wpnew-“ instead of “wp-“. This can prevent against “SQL injection hacks.”
 - c. heavy-duty passwords.
 - d. defending “wp-config.php” file
 - e. prohibiting file editing.
 - f. connecting the server accurately.
 - g. setting directory permissions meticulously.
 - h. restricting directory listing with “.htaccess” to prevent visitors from gaining access to full directory.
4. Locking your WordPress themes and plugins:
 - a. updating consistently.

- b. eliminating your WordPress version number to make hacking harder.

Azure Disk Encryption: Encrypting Azure Virtual Machines is our main priority. Azure Disk Encryption is a new facility that helps you encrypt your Windows virtual machine disks. Azure Disk Encryption controls the industry standard BitLocker feature of Windows to deliver volume encryption for the OS and the data disks. The solution is assimilated with Azure Key Vault to help you regulate and manage the disk-encryption keys and secrets in your key subscription. The solution also guarantees that all data on the virtual machine disks are encrypted at rest in your Azure storage. Azure disk encryption for Windows is now in General Availability in all Azure public regions and AzureGov regions for Standard VMs and VMs with superior storage.

Data Recovery

In the event where data loss has occurred, there is a way for HOOF to recover organizational data. Data loss could be data that's been accidentally deleted such as child records, grants, auction events or data that's lost from power outages, theft, hard drive failures or natural disasters.

Data will be routinely backed up to the Azure cloud server. In the event of a data loss, Microsoft offers with Azure the MARS (Microsoft Azure Recovery services) agent, which is their recovery tool allows you to restore data to a different laptop/desktop or restore data from the original source. You can restore files instantly from the cloud from previous recovery points and allows HOOF to copy the files over to the main workstation. This restore mechanism comes with the Azure cloud service so there won't be an additional charge for this recovery service.

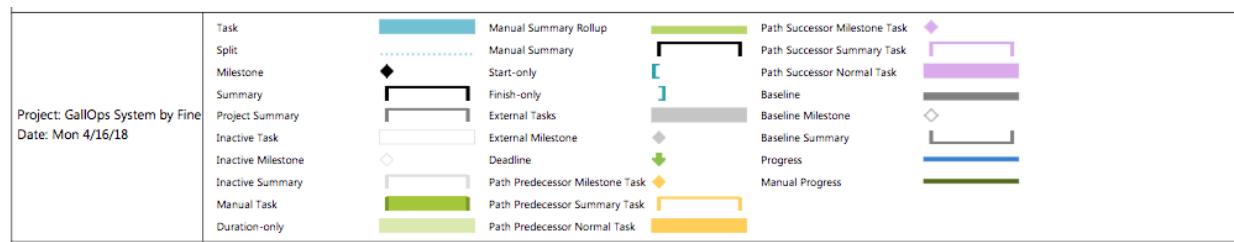
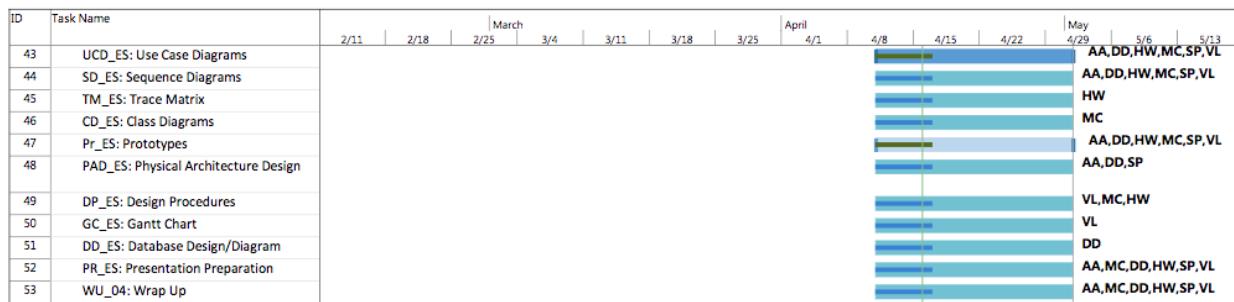
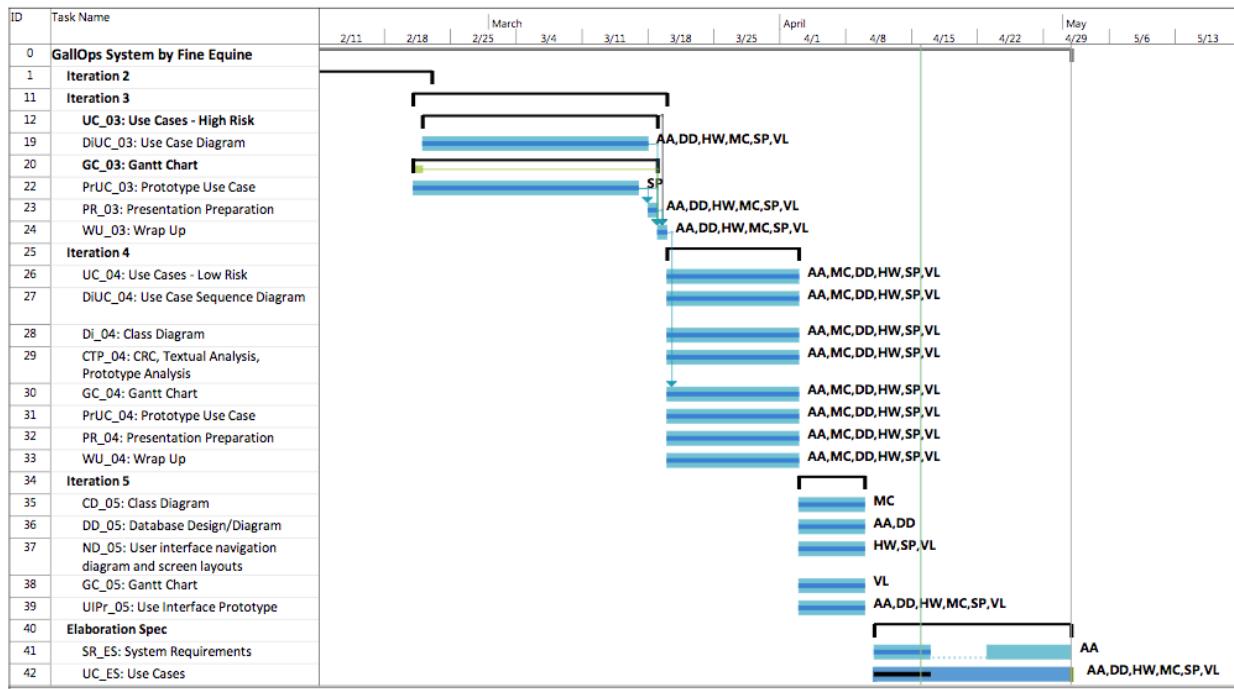
To recover data from Azure on the same machine:

1. Administrator opens the Azure Backup snap in tool.
2. Select recover data on the wizard
3. Select HOOF's administrator workstation or organizational laptop
4. Select the recovery mode and choose the individual files/folders they need to recover
5. Select the volume and the date. For the date selection, it allows administrators to go back to recovery points that were made from previous backups. Since backups are going to be set to run at least once a day, there should be a recovery point for every day on the calendar which they can recover from.
6. After the date has been confirmed, they would select mount which uses the current system storage and uploads the recovered data onto.
7. Confirm that the files have been recovered onto the system
8. Unmount the recovery storage and confirm that the files are recovered correctly.

For more help with recovering data, administrators can go to Microsoft Azure's support website for more data recovery guides.

There are companies that offer data recovery software that can recover data from hard drives in case the information was not backed up or saved successfully, they can access the hard drive disks and read each partition and save it to a functional hard drive. That way, any critical or sensitive information such as record of a child or a grant will be recovered and fully wiped off the hard drive with tools such as DBAN in case it needs to be disposed.

Gantt Chart



Elaboration Phase Prototypes

Add a Grant Prototype

HORSES OFFERING OPPORTUNITIES THE FUTURE, INC.

Add a grant

Grant Name	First Name	Last Name	M.I.
Grant Amount			
Grant source			

Description

Add  **Modify** **Cancel**

Edit a Grant Mailing Prototype

HORSES OFFERING OPPORTUNITIES THE FUTURE, INC.

Edit a grant

Camp Grant	Robert	Barker	T.
\$999.99			
GoFundMe Grant Inc.			

Need to fund camp program

Confirm Changes?

Confirm  **Cancel**

Add  **Modify**  **Cancel**

This use case shows how a HOOF member would edit a grant. The user would pull up an existing grant in the database and then click on modify. It would then prompt the user to confirm the changes. The data would then be updated in the database.

Delete a Grant Prototype

HORSES OFFERING OPPORTUNITIES THE FUTURE, INC.

Delete a grant

Grant Name
First Name Last Name M.I.
Grant Amount
Grant Source
Description

Add Modify Delete Grant

Confirmation

Are you sure you want to delete application?

This use case prototype shows how a HOOF member would delete a grant. The member would select an existing grant in the database and then click on delete grant. The grant would then be removed from the database.

Add to Mailing List Prototype

HORSES OFFERING OPPORTUNITIES FOR THE FUTURE, INC.

Add to Mailing List

First Name M.I. Last Name
Organization Name Title
Email Address
Street Address
City State Zip Code
Country
Add Cancel

Edit a Mailing List Prototype

HORSES OFFERING OPPORTUNITIES FOR THE FUTURE, INC.

Edit Mailing List

First Name	M.I.	Last Name
Organization Name	Title	
Email Address		
Street Address		
City	State	Zip Code
Country		

Add/Delete a Scheduled Mailing Prototype

HORSES OFFERING OPPORTUNITIES FOR THE FUTURE, INC.

Add Scheduled Mailing



HORSES OFFERING OPPORTUNITIES FOR THE FUTURE, INC.

Delete From Mailing List

Select Member Record

Member 1

Member 2

Member 3

Edit a Scheduled Mailing Prototype

HORSES OFFERING OPPORTUNITIES FOR THE FUTURE, INC.

Edit Scheduled Mailing

Select Scheduled Mailing

Delete a Scheduled Mailing Prototype

HORSES OFFERING OPPORTUNITIES FOR THE FUTURE, INC.

Delete Scheduled Mailing

Select Scheduled Mailing

Scheduled Mailing 1

Scheduled Mailing 2

Scheduled Mailing 3

Add a Director Prototype



Directors

Director ID #: * Search for Record:

Director Name:

Director Address:

Director Phone: Director Email:

Specialties:

Delete Record

Report Abuse Terms of Service

This is a prototype for the Add a Director use case. This data entry form enables a HOOF administrator to add records for directors into the director table in the Azure database. 'Search for Record' and 'Delete Record' are only applicable when editing or deleting an existing director from the director table.

Edit a Director Prototype



Directors

Director ID #: * Search for Record:

Director Name:

Director Address:

Director Phone: Director Email:

Specialties:

Delete Record

Report Abuse Terms of Service

This is a prototype for the Edit a Director use case. This data entry form enables a HOOF administrator to edit records for directors in the director table in the Azure database. 'Search for Record' must be used to find the record to be edited, before editing.

Delete a Director Prototype



Directors

Director ID #: * Search for Record:

Director Name:

Director Address:

Director Phone: Director Email:

Specialties:

[Report Abuse](#) [Terms of Service](#)

This is a prototype for the Delete a Director use case. This data entry form enables a HOOF administrator to delete records for directors in the director table in the Azure database. 'Search for Record' must be used to find the record to be deleted, before checking 'Delete Record' to 'Yes'.

Add a Participant Prototype



Participants

Participant ID #: *

School ID Search for Record:

Participant Name:

Zip Code Gender Ethnicity

Emergency Contact Name: (optional)

Emergency Contact Phone: (optional)

T-Shirt Size

Waiver Received
 Yes No

Delete Record
 No

This is a prototype for the Add a Participant use case. This data entry form enables a HOOF administrator to add records for participants into the participant table in the Azure database. 'Search for Record' and 'Delete Record' are only applicable when editing or deleting an existing participant from the participant table.

Edit a Participant Prototype



Participants

Participant ID #: *

School ID Search for Record:

Participant Name:

Zip Code Gender Ethnicity

Emergency Contact Name: (optional)

Emergency Contact Phone: (optional)

T-Shirt Size

Waiver Received
 Yes No

Delete Record
 No

This is a prototype for the Edit a Participant use case. This data entry form enables a HOOF administrator to edit records for participants into the participant table in the Azure database. 'Delete Record' is only applicable when deleting an existing participant from the participant table.

Delete a Participant Prototype



Participants

Participant ID #:* Search for Record:

Participant Name:
First
Middle
Last

Participant Address:
Address Line 1
City State Zip Code

Guardian Name:
First Last

Guardian Phone: Guardian Email:

Emergency Contact Name: (optional)
First Last

Emergency Contact Phone: (optional)

Yes

This is a prototype for the Delete a Participant use case. This data entry form enables a HOOF administrator to delete records for participants in the participant table in the Azure database. 'Search for Record' must be used to find the record to be deleted, before checking 'Delete Record' to 'Yes'.

Add a School Prototype



Schools

School ID #: * **School Name:** **Search for Record:**

School Address:

Contact Name:

Contact Phone:

Contact Email:

Delete Record

[Report Abuse](#) [Terms of Service](#)

This is a prototype for the Add a School use case. This data entry form enables a HOOF administrator to add records for schools into the school table in the Azure database. 'Search for Record' and 'Delete Record' are only applicable when editing or deleting an existing school from the school table.

Edit a School Prototype



Schools

School ID #: * **School Name:** **Search for Record:**

School Address:

Contact Name:

Contact Phone:

Contact Email:

Delete Record
 Yes No

Submit Record

[Report Abuse](#) [Terms of Service](#)

This is a prototype for the Edit a School use case. This data entry form enables a HOOF administrator to edit records for schools in the school table in the Azure database. 'Search for Record' must be used to find the record to be edited, before editing.

Delete a School Prototype



Schools

School ID #: * **School Name:** **Search for Record:**

School Address:

Contact Name:

Contact Phone:

Contact Email:

Delete Record
 Yes

Submit Record

[Report Abuse](#) [Terms of Service](#)

This is a prototype for the Delete a School use case. This data entry form enables a HOOF administrator to delete records for schools in the school table in the Azure database. 'Search for Record' must be used to find the record to be deleted, before checking 'Delete Record' to 'Yes'.

Add a User Prototype

The screenshot shows the 'Add New User' page in the WordPress admin dashboard. The left sidebar is dark with white text and icons, showing links like Dashboard, Addons, Posts, Portfolio, Media, Forms, Pages, Comments, Appearance, Plugins, Users (which is selected and highlighted in blue), All Users, Add New (with a red circle containing the number 1), Your Profile, Tools, and Settings.

The main content area has a light background. At the top, it says 'Add New User' and 'Create a brand new user and add them to this site.' Below this is a form with several input fields:

- Username (required)** (Field 2)
- E-mail (required)**
- First Name**
- Last Name**
- Website**
- Password (required)**
- Repeat Password (required)**

Below the password fields is a 'Strength indicator' bar and a hint: 'Hint: The password should be at least seven characters long. To make it stronger, use upper and lower case letters, numbers, and symbols like ! " ? \$ % ^ &).'

Underneath the form are two checkboxes: 'Send Password?' and 'Send this password to the new user by email.'

A dropdown menu labeled 'Role' (Field 3) is set to 'Subscriber'. A red box highlights this dropdown, and a red circle with the number 3 is placed above it.

At the bottom left is a blue button labeled 'Add New User' (Field 4), with a red circle containing the number 4 placed below it.

At the very bottom of the page, there is a small message: 'Thank you for creating with WordPress.'

In the bottom right corner, it says 'Version 4.2.2'.

This is a prototype for the Add a User use case. This area of the WordPress CMS allows HOOF administrators to add users for the WordPress CMS tool.

Edit Permissions Prototype

The screenshot shows the 'Edit User Role' page in the WordPress admin. The left sidebar has a 'User Roles' section selected under 'Ultimate Member'. The main area shows the 'Member' role being edited. It includes sections for 'Administrative Permissions', 'General Permissions', and 'Profile Access', each with several checkboxes for 'Yes' or 'No'.

Administrative Permissions	
Can access wp-admin?	<input type="checkbox"/> No
Force hiding adminbar in frontend?	<input checked="" type="checkbox"/> Yes
Can edit other member accounts?	<input type="checkbox"/> No
Can delete other member accounts?	<input type="checkbox"/> No

General Permissions	
Can edit their profile?	<input checked="" type="checkbox"/> Yes
Can delete their account?	<input checked="" type="checkbox"/> Yes

Profile Access	
Can view other member profiles?	<input checked="" type="checkbox"/> Yes
Can view these user roles only?	<input type="checkbox"/>
Can make their profile private?	<input type="checkbox"/> No
Can view/access private profiles?	<input type="checkbox"/> No

This is the prototype for the Edit Permissions use case. This section of the WordPress CMS tool allows a HOOF administrator to control administrative, general and profile permissions of each user.

Delete a User Prototype

The screenshot shows the WordPress User Management interface. The left sidebar has a 'Users' section selected, which is highlighted with a grey arrow. The main area displays a table of users:

Username	Status	Name	E-mail	Role	Posts
admin	ONLINE		test@test.com	Administrator	1
test-1	SUSPENDED	test test	test1@test.com	Subscriber	0
test-2	OFFLINE	test test	test2@test.com	Subscriber	0
test-3	OFFLINE	test test	test3@test.com	Subscriber	0
user-4	OFFLINE	test test	test4@test.com	Subscriber	0
user-5	OFFLINE	test test	test5@test.com	Subscriber	0
user-6	OFFLINE	test test	test6@test.com	Subscriber	0

At the bottom of the table, there is a note: "Thank you for creating with WordPress." and the version information "Version 3.6.1".

This is a prototype for the Delete a User use case. It allows a HOOF administrator to delete users of the WordPress CMS tool.

Add A Camp Prototype



Camps

Camp ID: * Event ID: Search for Record:

000

Camp Name:

Start Date End Date 

Location:

Delete Record

This is a prototype for the Add a Camp use case. This data entry form enables a HOOF administrator to add records for camps in the camp table in the Azure database.

Edit a Camp Prototype



Camps

Camp ID: * Event ID: Search for Record:

000

Camp Name:

Start Date End Date

Location:

Delete Record No

This is a prototype for the Edit a Camp use case. This data entry form enables a HOOF administrator to edit records for camps in the camp table in the Azure database. 'Search for Record' must be used to find the record to be edited.

Delete a Camp Prototype



Camps

Camp ID: * Event ID: Search for Record:

000

Camp Name:

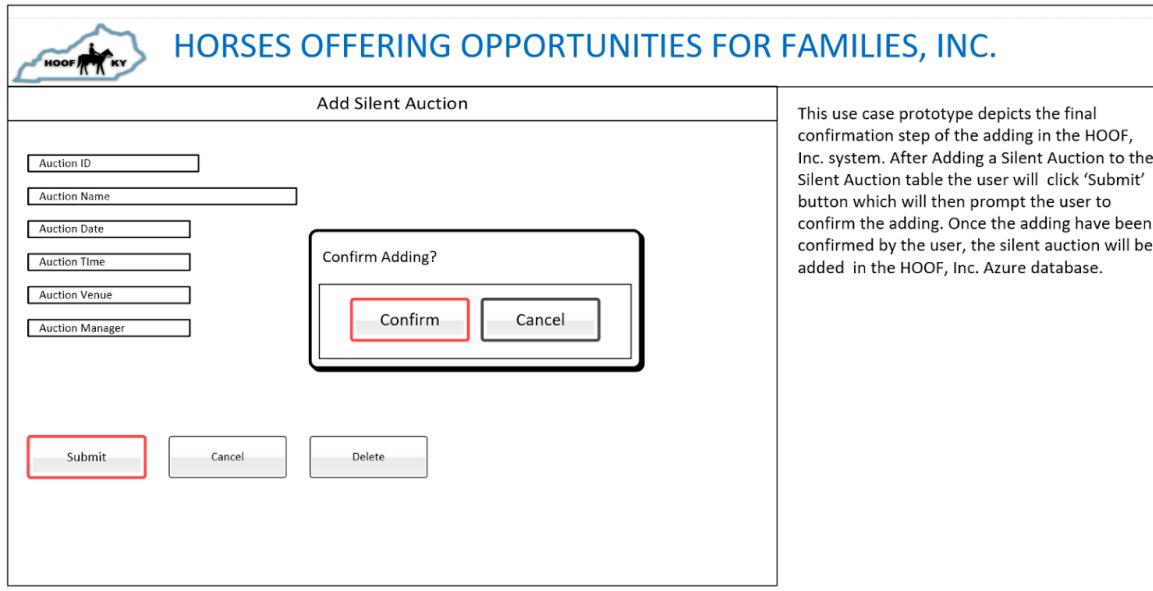
Start Date End Date

Location:

Delete Record

This is a prototype for the Delete a Camp use case. This data entry form enables a HOOF administrator to delete records for camps in the camp table in the Azure database. 'Search for Record' must be used to find the record to be deleted, before checking 'Delete Record' to 'Yes'.

Silent Auction Item Prototypes



HORSES OFFERING OPPORTUNITIES FOR FAMILIES, INC.

Add Silent Auction

Auction ID

Auction Name

Auction Date

Auction Time

Auction Venue

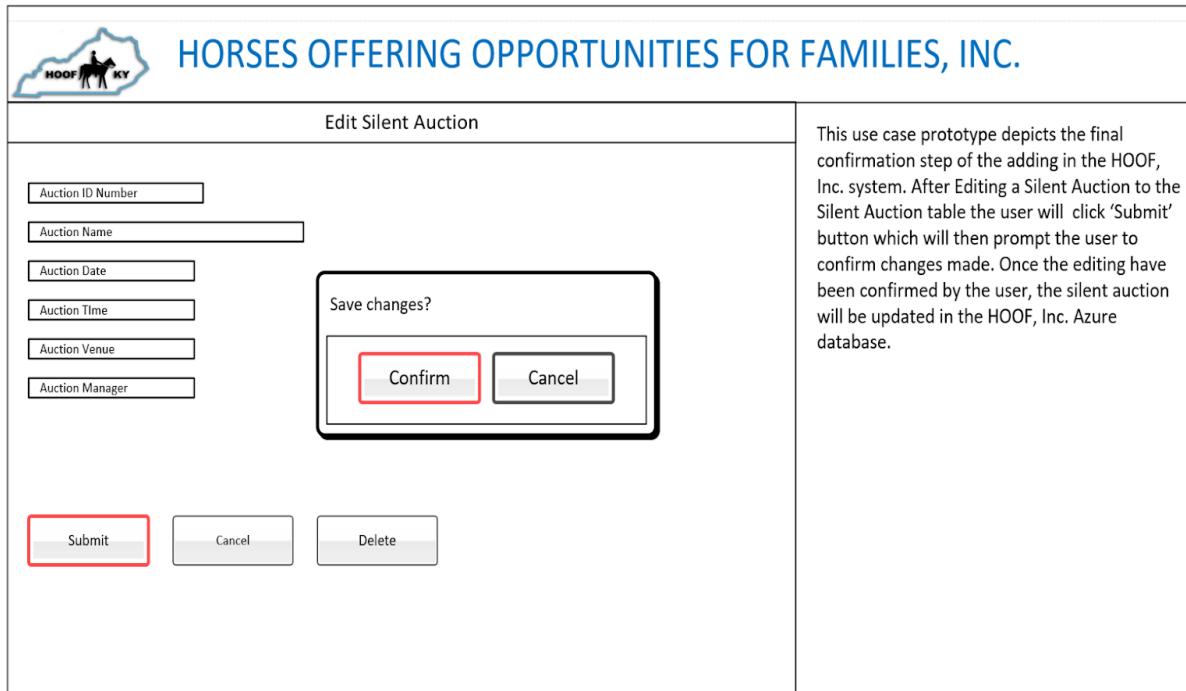
Auction Manager

Confirm Adding?

Confirm Cancel

Submit Cancel Delete

This use case prototype depicts the final confirmation step of the adding in the HOOF, Inc. system. After Adding a Silent Auction to the Silent Auction table the user will click 'Submit' button which will then prompt the user to confirm the adding. Once the adding have been confirmed by the user, the silent auction will be added in the HOOF, Inc. Azure database.



HORSES OFFERING OPPORTUNITIES FOR FAMILIES, INC.

Edit Silent Auction

Auction ID Number

Auction Name

Auction Date

Auction Time

Auction Venue

Auction Manager

Save changes?

Confirm Cancel

Submit Cancel Delete

This use case prototype depicts the final confirmation step of the adding in the HOOF, Inc. system. After Editing a Silent Auction to the Silent Auction table the user will click 'Submit' button which will then prompt the user to confirm changes made. Once the editing have been confirmed by the user, the silent auction will be updated in the HOOF, Inc. Azure database.

The screenshot shows a web-based application for managing silent auctions. At the top, there is a logo for "HORSES OFFERING OPPORTUNITIES FOR FAMILIES, INC." featuring a horse and jockey silhouette within a blue cloud-like shape. Below the logo, the text "Delete Silent Auction" is displayed. Two input fields are present: "Auction ID" and "Auction Name". A modal dialog box titled "Delete Silent Auction?" contains two buttons: "Confirm" and "Cancel". At the bottom of the page, there are three buttons: "Submit", "Cancel", and "Delete", where "Delete" is highlighted with a red border.

This use case prototype depicts the final confirmation step of the deleting in the HOOF, Inc. system. After Deleting a Silent Auction to the Silent Auction table the user will click 'Delete' button which will then prompt the user to confirm the deleting. Once the deleting have been confirmed by the user, the silent auction will be deleted in the HOOF, Inc. Azure database.



HORSES OFFERING OPPORTUNITIES FOR FAMILIES, INC.

Add Recurring Donation

Donor ID

First Name M.I. Last Name Email Address

Street Address City State Zip Code

Date of birth

Thank you for your generous gift

Donation Amount * \$ 500

Recurring * One Time Weekly Bi-Weekly Monthly Quarterly Yearly

Start gift on: 

Cardholder Name

Card Number Expiration MM/YY Security Code (CVC)

Confirm Adding?

Confirm **Cancel**

Add **Update** **Delete** **Cancel**

This use case prototype depicts the final confirmation step of the adding in the HOOF, Inc. system. After 'Adding a Recurring donation' to the donation table the donor will click 'Add' button which will then prompt the donor to confirm the adding. Once the adding have been confirmed by the donor, the record will be added in the HOOF, Inc. Azure database.



HORSES OFFERING OPPORTUNITIES FOR FAMILIES, INC.

Edit Recurring Donation

First Name _____ M.I. _____ Last Name _____ Email Address _____
 Street Address _____ City _____ State _____ Zip Code _____
 Date of birth _____

Thank you for your generous gift

Donation Amount * \$ 500
 Recurring * One Time
 Weekly
 Bi-Weekly
 Monthly
 Quarterly
 Yearly
 Start gift on: 31-October-2017

Apply Changes?

Cardholder Name _____
 Card Number _____ Expiration MM/YY _____ Security Code (CVC) _____

Add

This use case prototype depicts the final confirmation step of the adding in the HOOF, Inc. system. After 'Editing a Recurring donation' to the donation table the donor will click 'Update' button which will then prompt the donor to supply changes. Once the changes have been confirmed by the donor, the record will be updated in the HOOF, Inc. Azure database.



HORSES OFFERING OPPORTUNITIES FOR FAMILIES, INC.

Delete Recurring Donation

Donor ID _____
 First Name _____ M.I. _____ Last Name _____ Email deleteress _____
 Street deleteress _____ City _____ State _____ Zip Code _____
 Date of birth _____

Thank you for your generous gift

Donation Amount * \$ 500
 Recurring * One Time
 Weekly
 Bi-Weekly
 Monthly
 Quarterly
 Yearly
 Start gift on: 31-October-2017

Confirm Deletion !!

Data will be permanently lost.
 Are you sure you want to delete this record?

Cardholder Name _____
 Card Number _____ Expiration MM/YY _____ Security Code (CVC) _____

Delete

This use case prototype depicts the final confirmation step of the deleting in the HOOF, Inc. system. After 'deleting a Recurring donation' to the donation table the donor will click 'delete' button which will then prompt the donor to confirm the deleting. Once the deleting have been confirmed by the donor, the record will be deleted in the HOOF, Inc. Azure database.

Add Donors

DONOR ID: SEARCH FOR DONOR:

DONOR NAME:

DONOR ADDRESS:

DONOR EMAIL:

This is a prototype for the Add, Edit and Delete a Donor use cases. This data entry form enables a HOOF administrator to add/edit/delete records for donors into the Donor table in the Azure database. 'Search for Donor', 'Delete donor', 'Update Donor' are only applicable when editing or deleting an existing donor from the Donor table.

Edit Donors

DONOR ID: SEARCH FOR DONOR:

DONOR NAME:

DONOR ADDRESS:

DONOR EMAIL:

Confirm Changes?

This is a prototype for the Add, Edit and Delete a Donor use cases. This data entry form enables a HOOF administrator to add/edit/delete records for donors into the Donor table in the Azure database. 'Search for Donor', 'Delete donor', 'Update Donor' are only applicable when editing or deleting an existing donor from the Donor table.

The screenshot shows the homepage of the Horses Offering Opportunities for the Future, Inc. website. At the top left is the logo 'HOOF KY' with a silhouette of Kentucky and a horse. The main title 'HORSES OFFERING OPPORTUNITIES FOR THE FUTURE, INC.' is centered in large blue letters. Below the title is a navigation bar with links: HOME, ABOUT US, MEDIA, UPCOMING EVENTS, VOLUNTEER PORTAL, SERVICES, and a prominent orange DONATE button. Two photographs are displayed: one showing a woman interacting with a young child who is petting a dark horse, and another showing a young boy in a camouflage shirt petting a horse's neck. To the right of the photos is a donation form with fields for Name (first name and last name), Email, Country (United States), Address, City, State & zip (state and zip), Cardholder name, Card number, and Expire date (month and year). CSC and a help icon are also present. Below the form is a 'Donation Amount' field set to '\$0.00' and another orange 'DONATE' button.

HOME ABOUT US MEDIA UPCOMING EVENTS VOLUNTEER PORTAL SERVICES **DONATE**

Name: first name last name

Email:

Country: United States

Address:

City:

State & zip: state zip

Cardholder name:

Card number:

Expiration: month year CSC: ?

Donation Amount: \$0.00

DONATE

HORSES OFFERING OPPORTUNITIES FOR FAMILIES, INC.

Add Financial Information

Revenues		Expenses	
Foundation	\$ <input type="text"/>	Office Supplies	\$ <input type="text"/>
Private Donations	\$ <input type="text"/>	Insurance	\$ <input type="text"/>
Online Donations	\$ <input type="text"/>	Program Fees	\$ <input type="text"/>
Auctions	\$ <input type="text"/>	Memberships	\$ <input type="text"/>
Parking	\$ <input type="text"/>	Online Expenses	\$ <input type="text"/>
Miscellaneous Revenues	\$ <input type="text"/>	Secretary of State Tax	\$ <input type="text"/>
Total Revenues	\$ <input type="text"/>	Miscellaneous Expenses	\$ <input type="text"/>
		Total Expenses	\$ <input type="text"/>

This use case prototype shows how a HOOF accountant would add revenues/expenses to the database. He/she would enter in the values into the boxes and then once they are entered, they would click submit to add it to the database.

HORSES OFFERING OPPORTUNITIES FOR FAMILIES, INC.

Edit Financial Information

Revenues		Expenses	
Foundation	\$100 <input type="text"/>	Office Supplies	(\$30) <input type="text"/>
Private Donations	\$1222 <input type="text"/>	Insurance	(\$100) <input type="text"/>
Online Donations	\$4000 <input type="text"/>	Program Fees	(\$300) <input type="text"/>
Auctions	\$999 <input type="text"/>	Memberships	(\$500) <input type="text"/>
Parking	\$400 <input type="text"/>	Online Expenses	(\$100) <input type="text"/>
Miscellaneous Revenues	\$800 <input type="text"/>	Secretary of State Tax	(\$30) <input type="text"/>
Total Revenues	\$7521 <input type="text"/>	Miscellaneous Expenses	(\$50) <input type="text"/>
		Total Expenses	(\$1110) <input type="text"/>

This use case prototype shows how a HOOF accountant would edit revenues/expenses to the database. Values are populated already and the members would click on modify and values are updated in the database.

HORSES OFFERING OPPORTUNITIES FOR FAMILIES, INC.

Delete Financial Information

Revenues	Expenses
Foundation <input type="text"/>	\$ <input type="text"/> <input type="checkbox"/>
Private Donations <input type="text"/>	\$ <input type="text"/> <input type="checkbox"/>
Online Donations <input type="text"/>	\$ <input type="text"/> <input type="checkbox"/>
Auctions <input type="text"/>	\$ <input type="text"/> <input type="checkbox"/>
Parking <input type="text"/>	\$ <input type="text"/> <input type="checkbox"/>
Miscellaneous Revenues <input type="text"/>	\$ <input type="text"/> <input type="checkbox"/>
Total Revenues <input type="text"/>	\$ <input type="text"/>
Confirmation	
Are you sure you want to delete? <input checked="" type="checkbox"/> <input type="checkbox"/>	
<input type="button" value="Submit"/> <input type="button" value="Cancel"/> <input type="button" value="Modify"/>	

This use case prototype shows how a HOOF accountant would delete revenues/expenses to the database. Each value box is accompanied with a discard icon. The accountant would then click on delete and then the check mark to confirm the deletion from the database



Camps

Camp ID: *	Event ID:	Search for Record:
<input type="text" value="000"/>	<input type="text"/>	<input type="text"/> Search
Camp Name:		
<input type="text"/>		
Start Date	End Date	
<input type="text"/> <input type="button" value="Calendar"/>	<input type="text"/> <input type="button" value="Calendar"/>	
Location:		
<input type="text"/>		
Delete Record		
<input type="button" value="No"/>		
<input type="button" value="Submit Record"/>		

The screenshot shows a web application interface for adding a silent auction item. At the top left is the logo 'HOOF KY' with a horse silhouette. The main title 'HORSES OFFERING OPPORTUNITIES FOR FAMILIES, INC.' is centered above the form. The form is titled 'Add Silent Auction Item'. It contains several input fields: 'Inventory item name', 'Inventory item Description', 'Inventory item quantity', 'Inventory sale status' (all in a row), 'Donor Name', 'Donor Address', 'Item Value', and 'Final Sale Value' (also in a row). A button 'Export to Spreadsheet' is located below these. A large red rectangular box highlights the 'Submit' button on the left. A modal dialog box titled 'Confirm Adding?' is centered over the form, containing 'Confirm' and 'Cancel' buttons. The entire interface is set against a light gray background.

This use case prototype depicts the final confirmation step of the adding in the HOOF, Inc. system. After Adding a Silent Auction Item to the Silent Auction Item table the user will click 'Submit' button which will then prompt the user to confirm the adding. Once the adding have been confirmed by the user, the silent auction item will be added in the HOOF, Inc. Azure database.

HORSES OFFERING OPPORTUNITIES FOR FAMILIES, INC.

Edit Silent Auction Item

Inventory item name	Donor Name
Inventory item Description	Donor Address
Inventory item quantity	Item Value
Inventory sale status	Final Sale Value
Export to Spreadsheet	

Submit Cancel Delete

This use case prototype depicts the final confirmation step of the adding in the HOOF, Inc. system. After Adding a Silent Auction Item to the Silent Auction Item table the user will click 'Submit' button which will then prompt the user to confirm the adding. Once the adding have been confirmed by the user, the silent auction item will be added in the HOOF, Inc. Azure database.

HORSES OFFERING OPPORTUNITIES FOR FAMILIES, INC.

Delete Silent Auction Item

Inventory item name	Donor Name
Inventory item Description	Donor Address
Inventory item quantity	Item Value
Inventory sale status	Final Sale Value
Export to Spreadsheet	

Confirm Delete?
Confirm Cancel

Submit Cancel Delete

This use case prototype depicts the final confirmation step of the adding in the HOOF, Inc. system. After Adding a Silent Auction Item to the Silent Auction Item table the user will click 'Submit' button which will then prompt the user to confirm the adding. Once the adding have been confirmed by the user, the silent auction item will be added in the HOOF, Inc. Azure database.