

**Elaboration Specification**

**CIS 320**

**Team Hungry for Opportunity**

**Carter Rowe, Ben Rome, Mike Bergamini**

## System Requirements

Functional Requirements: (States what the system should be able to do)

### **Donations:**

- The system will accept food donations
- The system will allow donors to edit their donations
- The system will allow donors to delete their donations
- The system will accept monetary donations
- The system will allow donors to edit their monetary donations
- The system will allow donors to delete their monetary donations
- The system will include an item list for donations
- The system will allow the item list to be edited
- The system will allow the items on the list to be deleted

### **Staff/Volunteer:**

- The system will allow volunteers to sign up
- The system will allow volunteers to edit their account
- The system will allow an admin to delete volunteer accounts
- The system will allow staff to sign up
- The system will allow staff to edit their account
- The system will allow an admin to delete staff accounts
- The system will allow staff to login
- The system will allow volunteers to login
- The system will allow staff to log their hours
- The system will allow staff to edit their hours
- The system will allow staff to delete their hours
- The system will allow a volunteer to log their hours
- The system will allow a volunteer to edit their hours
- The system will allow a volunteer to delete their hours
- The system will allow staff to add inventory
- The system will allow staff to edit inventory
- The system will allow staff to delete inventory

### **Misc. Features:**

- The system will have an event calendar
- The system will allow the event calendar to be edited
- The system will allow the event calendar to be deleted
- The system will allow events to be added
- The system will allow events to be edited
- The system will allow events to be deleted

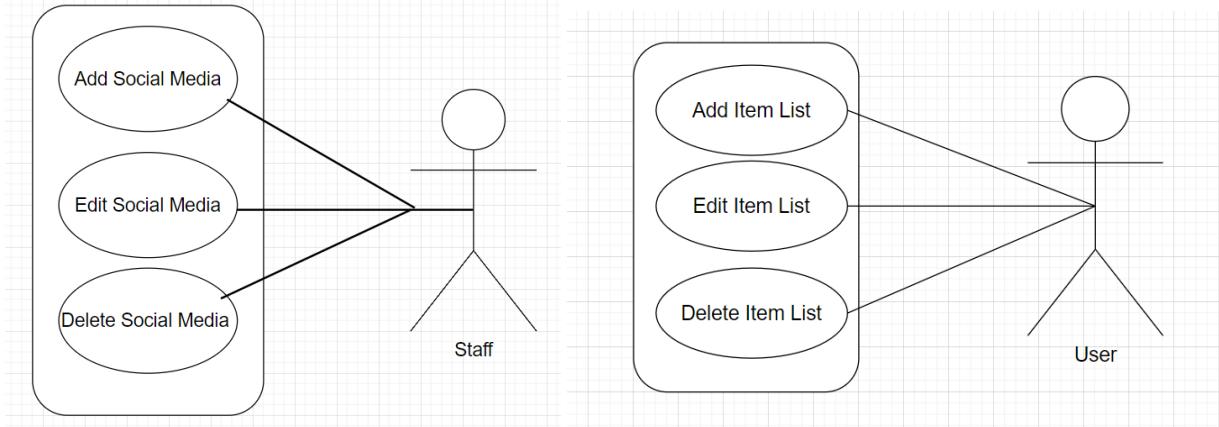
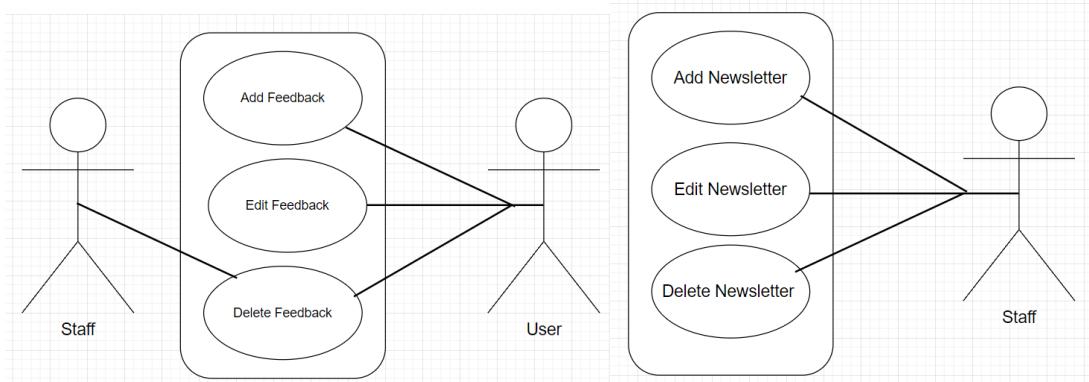
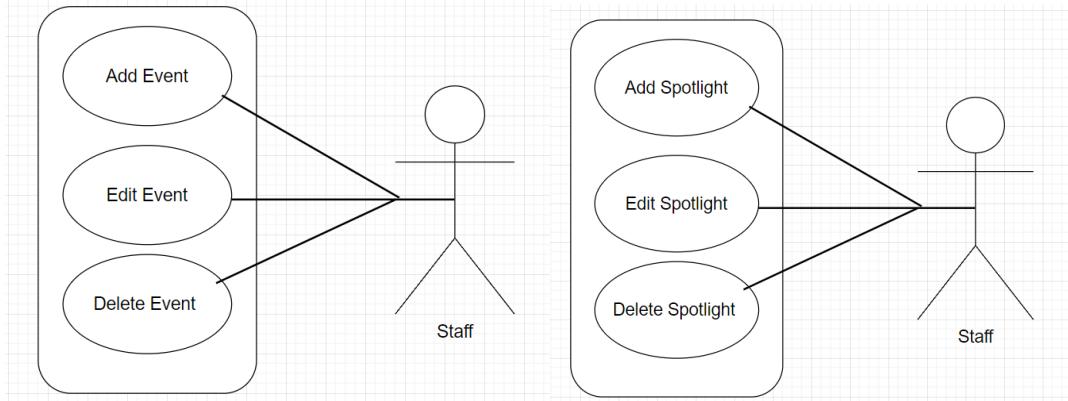
- The system will allow users to give feedback through a survey
- The system will allow users to edit their feedback
- The system will allow feedback to be deleted
- The system will include a spotlight on staff
- The system will allow the spotlight to be edited
- The system will allow the spotlight to be deleted
- The system will have an integrated social media feed
- The system will allow the integrated social media feed to be edited
- The system will allow the integrated social media feed to be deleted
- The system will have a newsletter
- The system will allow the newsletter to be edited
- The system will allow the newsletter to be deleted
- The system will accept recipes suggestions
- The system will allow recipe suggestions to be edited
- The system will allow recipe suggestions to be deleted
- The system will include helpful links to other UofL resources
- The system will allow these links to be edited
- The system will allow these links to be deleted

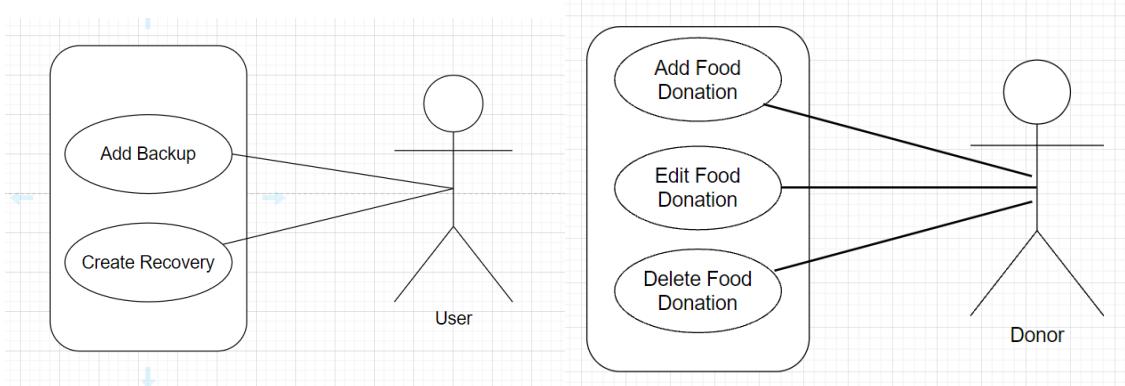
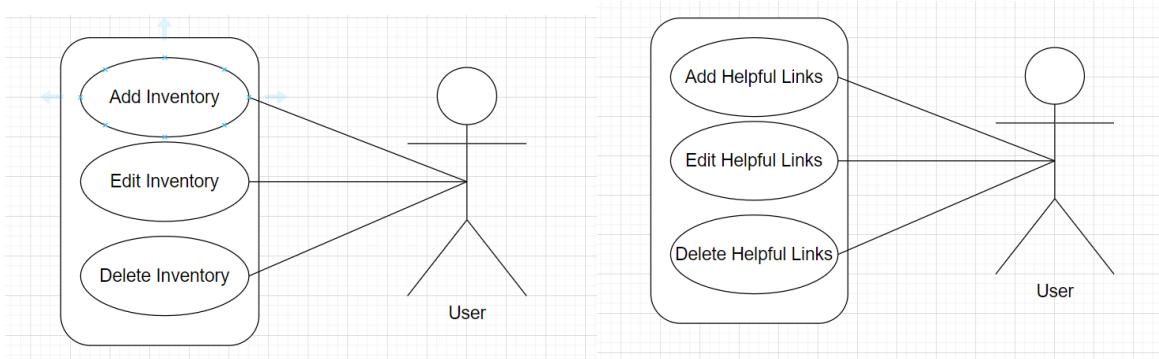
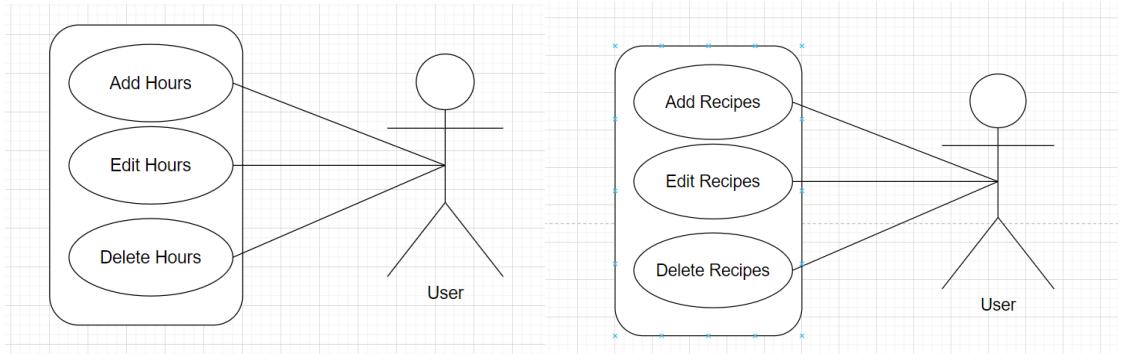
Non-Functional Requirements: (Constraints placed on the system)

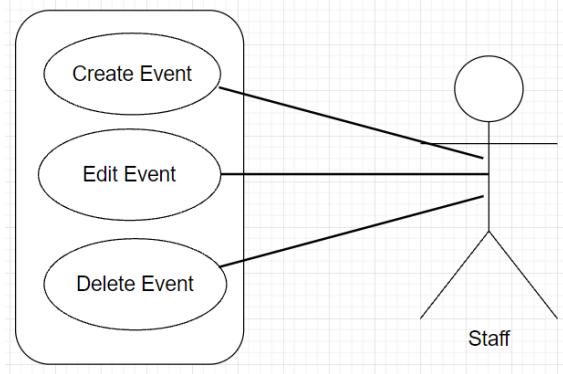
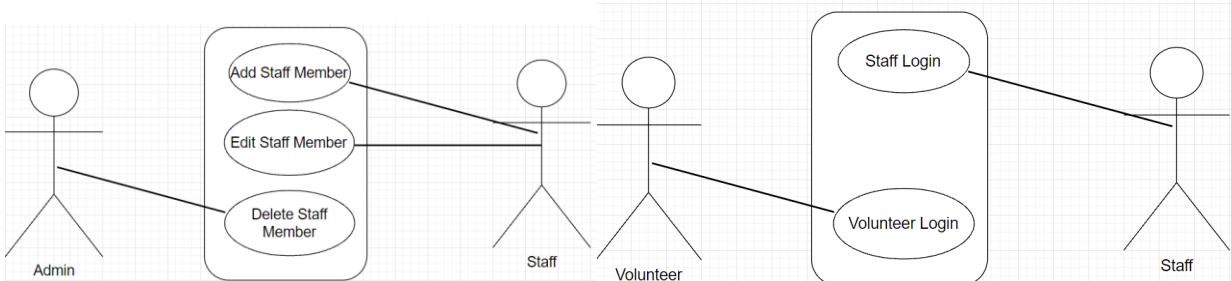
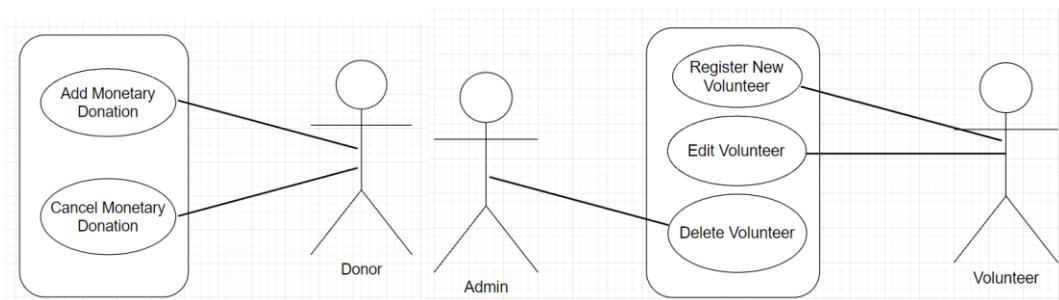
- The system will use WordPress
- The system will use a SQL server database management system
- The system will use Clover for payments
- The system will have some form of security

## Use Case Diagrams

A use case is a description of how a person will perform an action within a system. Use case diagrams visualize who performs what action in the system. The text inside the oval depicts a use case, what the system can do, and the stick figure represents a user, who performs the use case(s).







## Trace Matrix

Use cases may be traced to top-level requirements because they show who performs what action and makes sure that all the important requirements are fully defined. A trace matrix makes sure that all requirements have a use case associated with it.

		Requirements	
		Use Cases	Requirements
		Add donation (food)	The system will accept food donations.
		Edit donation (food)	The system will edit our food donations.
		Delete donation (food)	The system will delete our food donations.
		Add donation (monetary)	The system will accept monetary donations.
		Edit donation (monetary)	The system will edit our monetary donations.
		Delete donation (monetary)	The system will delete our monetary donations.
		Add volunteer	The system will include an item in our inventory.
		Edit volunteer	The system will edit an item in our inventory.
		Delete volunteer	The system will delete an item in our inventory.
		Add staff	The system will add a staff member.
		Edit staff	The system will edit a staff member.
		Delete staff	The system will delete a staff member.
		Add staff login	The system will allow our staff to log in.
		Add volunteer login	The system will allow our volunteers to log in.
		Add event calendar	The system will allow our events to be listed.
		Edit event calendar	The system will allow our events to be edited.
		Delete event calendar	The system will allow our events to be deleted.
		Add feedback	The system will allow our inventory to be reviewed.
		Edit feedback	The system will allow our inventory to be edited.
		Delete feedback	The system will allow our inventory to be deleted.
		Add spotlight	The system will allow our events to be highlighted.
		Edit spotlight	The system will allow our events to be edited.
		Delete spotlight	The system will allow our events to be deleted.
		Add social media	The system will add our social media.
		Edit social media	The system will edit our social media.
		Delete social media	The system will delete our social media.
		Add event	The system will add a new event.
		Edit event	The system will edit an event.
		Delete event	The system will delete an event.
		Add newsletter	The system will add a new newsletter.
		Edit newsletter	The system will edit a newsletter.
		Delete newsletter	The system will delete a newsletter.
		Add item list	The system will add a new item list.
		Edit item list	The system will edit an item list.
		Delete item list	The system will delete an item list.
		Add hours	The system will add a new hour.
		Edit hours	The system will edit an hour.
		Delete hours	The system will delete an hour.
		Add recipes	The system will add a new recipe.
		Edit recipes	The system will edit a recipe.
		Delete recipes	The system will delete a recipe.
		Add inventory	The system will add a new inventory item.
		Edit inventory	The system will edit an inventory item.
		Delete inventory	The system will delete an inventory item.
		Add helpful links	The system will add a new helpful link.
		Edit helpful links	The system will edit a helpful link.
		Delete helpful links	The system will delete a helpful link.
		Create Backup	The system will have a backup.
		Create Recovery	The system will have a recovery plan.
		Create Backup	The system will have a backup.

# Use Case Specification: <Delete Event Calendar>

## 1. Use-Case Name

### 1.1 Brief Description

Once an event calendar is added and/or edited it can be removed if it is no longer needed. If events cease, then there will be no events for the event calendar to contain so it will have no use. Alternatively, if a separate event calendar is created for seasonal events it will need to be deleted after the season ends.

## 2. Flow of Events

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member finds the WordPress plugin for the calendar
- Staff member deletes the plugin
- Staff member presses save

## 3. Pre-conditions

< An event calendar is added, events are created >

## 4. Post-conditions

< The system no longer has an event calendar; the system no longer has events>

# Use Case Specification: <Add Event>

## 5. Use-Case Name

### 5.1 Brief Description

Events are organized functions that support the Cardinal Cupboard and their side projects. Adding an event to the event calendar is important so that people can find upcoming events and their dates. This use case will allow events to be added to the event calendar.

## 6. Flow of Events

### 6.1 Basic Flow

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member accesses event calendar extension
- Staff member adds event date
- Staff member adds event description
- Staff member adds event title
- Staff member presses save

## 7. Pre-conditions

### 7.1 < Event calendar must already exist >

## 8. Post-conditions

### 8.1 < Event added to event calendar on website >

# Use Case Specification: <Edit Event>

## 9. Use-Case Name

### 9.1 Brief Description

Events are organized functions that support the Cardinal Cupboard and their side projects.

Adding an event to the event calendar is important so that people can find upcoming events and their dates. This use case will allow events added to the event calendar to be edited.

## 10. Flow of Events

### 10.1 Basic Flow

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member accesses event calendar extension
- Staff member clicks on existing event
- Staff member edits event date
- Staff member edits event title
- Staff member edits event description
- Staff member presses save

Note: any combination of factors can be edited (ex. Staff member edits the event time but not date and description etc.)

## 11. Pre-conditions

### 11.1 < An event on the calendar must already exist >

## 12. Post-conditions

### 12.1 < A new event added to event calendar on website in place of the old one >

# Use Case Specification: <Delete Event>

## 13. Use-Case Name

### 13.1 Brief Description

Events are organized functions that support the Cardinal Cupboard and their side projects. Adding an event to the event calendar is important so that people can find upcoming events and their dates. This use case will allow events added to the event calendar to be removed.

## 14. Flow of Events

### 14.1 Basic Flow

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member accesses event calendar extension
- Staff member clicks on existing event
- Staff member deletes event date
- Staff member deletes event title
- Staff member deletes event description
- Staff member presses save

## 15. Pre-conditions

### 15.1 < An event on the calendar must already exist >

## 16. Post-conditions

### 16.1 < An event is removed from the event calendar>

# Use Case Specification: <Add Feedback>

## 17. Use-Case Name

### 17.1 Brief Description

Feedback is essential to any business that interacts with customers because it is important to know how they feel about the service(s) provided to them. This use case allows users to write their feedback through a form that can be analyzed by the Cardinal Cupboard to better understand their customers. Customers will type their response in a text box provided and/or leave a star rating.

## 18. Flow of Events

### 18.1 Basic Flow

- User visits website
- User clicks on feedback link
- User is brought to feedback form
- User enters feedback in text box provided
- User clicks on one of five stars to leave a rating
- User presses submit

### 18.2 Alternative Flows

#### 18.2.1 < User only leaves comment >

- User visits website
- User clicks on feedback link
- User is brought to feedback form
- User enters feedback in text box provided
- User presses submit

#### 18.2.2 < User only leaves rating >

- User visits website
- User clicks on feedback link
- User is brought to feedback form
- User clicks on one of five stars to leave a rating
- User presses submit

## 19. Pre-conditions

### 19.1 < Link to feedback form available in website >

**20. Post-conditions**

- 20.1 < System displays a “thank you” message, User is directed back to homepage via another link >

# Use Case Specification: <Edit Feedback>

## 21. Use-Case Name

### 21.1 Brief Description

Feedback is essential to any business that interacts with customers because it is important to know how they feel about the service(s) provided to them. This use case allows users to edit their feedback through a form that can be analyzed by the Cardinal Cupboard to better understand their customers. Customers will edit their response in a text box provided or change the number of stars given.

## 22. Flow of Events

### 22.1 Basic Flow

- User visits website
- User clicks on feedback link
- User is brought to feedback form
- User enters feedback in textbox provided
- User edits feedback in text box
- User clicks on one of five stars to edit their rating
- User presses submit

### 22.2 Alternative Flows

#### 22.2.1 < User only edits comment >

- User visits website
- User clicks on feedback link
- User is brought to feedback form
- User enters feedback in text box provided
- User edits feedback in text box
- User presses submit

#### 22.2.2 < User only edits rating >

- User visits website
- User clicks on feedback link
- User is brought to feedback form
- User clicks on one of five stars to enter their rating
- User clicks on a different star to edit their rating

- User presses submit

## **23. Pre-conditions**

**23.1 < User clicks on link to feedback form, User enters feedback >**

## **24. Post-conditions**

**24.1 < Feedback is changed, system displays a “thank you” message, User is directed back to homepage via another link >**

# Use Case Specification: <Delete Feedback>

## 25. Use-Case Name

### 25.1 Brief Description

Feedback is essential to any business that interacts with customers because it is important to know how they feel about the service(s) provided to them. This use case allows users to delete their feedback through a form that can be analyzed by the Cardinal Cupboard to better understand their customers. Customers will delete their response in a text box provided and/or remove a star rating.

## 26. Flow of Events

### 26.1 Basic Flow

- User visits website
- User clicks on feedback link
- User is brought to feedback form
- User enters feedback in text box provided
- User deletes feedback in text box
- User clicks on one of five stars to leave a rating
- User clicks on the same star again to remove a rating
- User closes the form

### 26.2 Alternative Flows

#### 26.2.1 < User only deletes comment >

- User visits website
- User clicks on feedback link
- User is brought to feedback form
- User enters feedback in text box provided
- User deletes feedback in the textbox
- User clicks on a star to leave a rating
- User presses submit

#### 26.2.2 < User only deletes rating >

- User visits website
- User clicks on feedback link
- User is brought to feedback form
- User enters feedback in the textbox

- User clicks on one of five stars to leave a rating
- User clicks on the same star to remove it
- User presses submit

## **27. Pre-conditions**

**27.1 < User clicks on link to feedback form, User enters feedback >**

## **28. Post-conditions**

**28.1 < Part of, or all, of the feedback is deleted, System displays a “thank you message” if form is submitted, User is directed back to homepage via another link >**

# **Use Case Specification: <Add Newsletter>**

## **29. Use-Case Name**

### **29.1 Brief Description**

A newsletter will allow recipients to be informed of any upcoming events, changes, or other information related to the Cardinal Cupboard. It is a useful and easy way to engage with the community around UofL. Users will be able sign up on the website to receive the newsletter. This use case will add a newsletter to the website.

## **30. Flow of Events**

### **30.1 Basic Flow**

- Staff member visits the website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member pastes code from newsletter provider
- Staff member presses save

## **31. Pre-conditions**

**31.1 <Must have WordPress linked to the website>**

## **32. Post-conditions**

**32.1 < Newsletter added to website >**

# **Use Case Specification: <Edit Newsletter>**

## **33. Use-Case Name**

### **33.1 Brief Description**

A newsletter will allow recipients to be informed of any upcoming events, changes, or other information related to the Cardinal Cupboard. It is a useful and easy way to engage with the community around UofL. Users will be able sign up on the website to receive the newsletter. This use case will edit the newsletter that will go on the website.

## **34. Flow of Events**

### **34.1 Basic Flow**

- Staff member visits the website
- Staff member logs in
- Staff member clicks link to newsletter provider
- Staff member edits newsletter
- Staff member returns to the website
- Staff member clicks link to WordPress
- Staff member pastes code from newsletter provider
- Staff member presses save

## **35. Pre-conditions**

**35.1 <Must have WordPress linked to the website, must have a pre-existing newsletter>**

## **36. Post-conditions**

**36.1 < New Newsletter added to website >**

# **Use Case Specification: <Delete Newsletter>**

## **37. Use-Case Name**

### **37.1 Brief Description**

A newsletter will allow recipients to be informed of any upcoming events, changes, or other information related to the Cardinal Cupboard. It is a useful and easy way to engage with the community around UofL. Users will be able sign up on the website to receive the newsletter. However, if it is deemed unnecessary due to the lack of people signing up etc., it can be removed. This use case will remove the newsletter from the website.

## **38. Flow of Events**

### **38.1 Basic Flow**

- Staff member visits the website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member deletes code from newsletter provider
- Staff member presses save

## **39. Pre-conditions**

**39.1 <Must have WordPress linked to the website, must have a pre-existing newsletter>**

## **40. Post-conditions**

**40.1 < Newsletter removed from the website >**

# **Use Case Specification: <Add social media>**

## **41. Use-Case Name**

### **41.1 Brief Description**

Social media is a great way to expand a business and reach more customers. By adding an integrated social media feed to the website, visitors will be able to see what people are saying about the Cardinal Cupboard on other social media platforms in real time. This use case will add the integrated social media feed to the homepage of the website.

## **42. Flow of Events**

### **42.1 Basic Flow**

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member adds code for the integrated social media feed
- Staff member presses save

## **43. Pre-conditions**

### **43.1 <Website must be linked to WordPress >**

## **44. Post-conditions**

### **44.1 < Social media feed appears on homepage >**

# **Use Case Specification: <Edit social media>**

## **45. Use-Case Name**

### **45.1 Brief Description**

Social media is a great way to expand a business and reach more customers. By adding an integrated social media feed to the website, visitors will be able to see what people are saying about the Cardinal Cupboard on other social media platforms in real time. This use case will allow the integrated social media feed to be edited to display a different social media platform.

## **46. Flow of Events**

### **46.1 Basic Flow**

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member edits code for the integrated social media feed
- Staff member presses save

## **47. Pre-conditions**

### **47.1 < Code must already exist on WordPress >**

## **48. Post-conditions**

### **48.1 < New social media feed appears on homepage >**

# **Use Case Specification: <Delete social media>**

## **49. Use-Case Name**

### **49.1 Brief Description**

Social media is a great way to expand a business and reach more customers. By adding an integrated social media feed to the website, visitors will be able to see what people are saying about the Cardinal Cupboard on other social media platforms in real time. However, if a business does not have a strong social media presence, this may not be necessary. This use case will allow the integrated social media feed to be removed from the website.

## **50. Flow of Events**

### **50.1 Basic Flow**

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member deletes code for the integrated social media feed
- Staff member presses save

## **51. Pre-conditions**

### **51.1 < Code must already exist on WordPress >**

## **52. Post-conditions**

### **52.1 < Social media feed removed from homepage >**

# **Use Case Specification: <Add Spotlight>**

## **53. Use-Case Name**

### **53.1 Brief Description**

The spotlight is going to be implemented for the purpose of commending staff members and volunteers for their hard work. It will include a picture and a brief description of who they are and what they have done to support the community and cardinal cupboard. The spotlight will be on the homepage of the Cardinal Cupboard website.

## **54. Flow of Events**

### **54.1 Basic Flow**

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member adds code for the spotlight
- Staff member inserts picture
- Staff member adds description
- Staff member presses save

## **55. Pre-conditions**

### **55.1 <Website must be linked to WordPress >**

## **56. Post-conditions**

### **56.1 < Spotlight appears on homepage >**

# Use Case Specification: <Edit Spotlight>

## 57. Use-Case Name

### 57.1 Brief Description

The spotlight is going to be implemented for the purpose of commending staff members and volunteers for their hard work. It will include a picture and a brief description of who they are and what they have done to support the community and cardinal cupboard. The spotlight will be on the homepage of the Cardinal Cupboard website. This use case allows the picture and description of the spotlight to be edited.

## 58. Flow of Events

### 58.1 Basic Flow

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member inserts new picture
- Staff member edits description
- Staff member presses save

## 59. Pre-conditions

### 59.1 < Spotlight must already exist>

## 60. Post-conditions

### 60.1 < A new spotlight will be added to the homepage >

# **Use Case Specification: <Delete Spotlight>**

## **61. Use-Case Name**

### **61.1 Brief Description**

The spotlight is going to be implemented for the purpose of commanding staff members and volunteers for their hard work. It will include a picture and a brief description of who they are and what they have done to support the community and cardinal cupboard. The spotlight will be on the homepage of the Cardinal Cupboard website. This use case will allow the spotlight to be removed from the website.

## **62. Flow of Events**

### **62.1 Basic Flow**

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member deletes code for the spotlight
- Staff member presses save

## **63. Pre-conditions**

### **63.1 <Website must be linked to WordPress >**

## **64. Post-conditions**

### **64.1 < Spotlight removed from homepage >**

# Use Case Specification: Add Food Donation

## 64.2 Add Food Donation

### 64.3 Brief Description

This use case allows a donor to give food to the pantry. This will allow the donor to enter their name, email address, donation description, and select a donation date. This information is then stored in the database. They will then be sent a confirmation email with a secure link to access the donation information.

## 65. Flow of Events

### 65.1 Basic Flow

- Donor enters Name
- Donor enters Email Address
- Donor enters Donation Description
- Donor selects Donation Date
- Donor presses 'Submit'
- Donor's Name, Email, Description, and Date are stored in the SQL Database
- System sends Donor a confirmation email with donation details and secure link

### 65.2 Alternative Flows

None

## 66. Special Requirements

None

## 67. Pre-conditions

- All fields must not be empty/null
- Donor must enter valid email address
- Donation time can only be selected from the available date/time slots

## 68. Post-conditions

- A record of the food donation has been created and stored in the SQL database.

## 69. Extension Points

None

# Use Case Specification: Edit Food Donation

## 70. Edit Food Donation

### 70.1 Brief Description

This use case allows a donor to edit the details of their food donation using the secure link they were provided when they created it.

## 71. Flow of Events

### 71.1 Basic Flow

- Donor accesses donation record using secure link from email
- Donor makes changes to Donation Description or Donation Date
- Donor presses 'Save Changes'
- Record is updated in SQL Database
- System sends Donor a confirmation email with the updated information and secure link

### 71.2 Alternative Flows

None

## 72. Special Requirements

None

## 73. Pre-conditions

- **Donor must have created a food donation and obtained a secure link**
- Description and Date must not be empty/null
- Donation time can only be selected from the available date/time slots

## 74. Post-conditions

- The record of the food donation has been updated in the SQL database.

## 75. Extension Points

None

# Use Case Specification: Delete Food Donation

## 76. Delete Food Donation

### 76.1 Brief Description

This use case allows a donor to cancel their food donation using the secure link they were provided when they created it.

## 77. Flow of Events

### 77.1 Basic Flow

- Donor accesses donation record using secure link from email
- Donor presses 'Cancel Donation'
- Record is removed from SQL Database
- System sends Donor a confirmation email

### 77.2 Alternative Flows

None

## 78. Special Requirements

None

## 79. Pre-conditions

- Donor must have created a food donation and obtained a secure link

## 80. Post-conditions

- The record of the food donation has been removed from the SQL database.

## 81. Extension Points

None

# Use Case Specification: Add Monetary Donation

## 81.1 Add Monetary Donation

### 81.2 Brief Description

This use case allows a donor to give funds to the pantry. This will allow the donor to enter their name, email address, payment information, and amount. This information is then stored in the database. They will then be sent a confirmation email with a secure link to access the donation information.

## 82. Flow of Events

### 82.1 Basic Flow

- Donor enters Name
- Donor enters Email Address
- Donor enters Card Number
- Donor enters Card Expiration Date
- Donor enters Security Code
- Donor enters Donation Amount
- Donor presses 'Submit'
- Donor's Name, Email, Payment Information, and Amount are stored in the SQL Database
- System sends Donor a confirmation email with donation details and a secure link

### 82.2 Alternative Flows

None

## 83. Special Requirements

None

## 84. Pre-conditions

- All fields must not be empty/null
- Donor must enter valid email address
- Donor must enter valid payment information
- Donation Amount must be positive

## 85. Post-conditions

- A record of the monetary donation has been created and stored in the SQL database
  - One week after form is submitted the transaction will be processed

## 86. Extension Points

None

# Use Case Specification: Cancel Monetary Donation

## 86.1 Refund Monetary Donation

### 86.2 Brief Description

This use case allows a donor to cancel their recent donation. This can only be done within one week of the initial transaction.

## 87. Flow of Events

### 87.1 Basic Flow

- Donor access donation information from secure link in email
- Donor presses 'Cancel Donation'
- Donor can optionally select why they are cancelling their donation from a list of options
- Donor presses 'Confirm'

### 87.2 Alternative Flows

None

## 88. Special Requirements

None

## 89. Pre-conditions

- Donor must have created a monetary donation and obtained a secure link
- Donor must cancel within one week of donation

## 90. Post-conditions

- The record of the monetary donation has been removed from the SQL database.
  - The transaction is cancelled before it is processed

## 91. Extension Points

None

# Use Case Specification: Register New Volunteer

## 91.1 **Register New Volunteer**

### 91.2 **Brief Description**

This use case allows new volunteers to create an account.

## 92. **Flow of Events**

### 92.1 **Basic Flow**

- Volunteer enters Email Address
- Volunteer enters Password
- Volunteer reenters Password to confirm
- Volunteer enters Name
- Volunteer enters Date of Birth
- Volunteer presses 'Register'
- Volunteer's Information is stored in the SQL Database
- System sends a confirmation email

### 92.2 **Alternative Flows**

Volunteer is a UofL Student

- Volunteer enters ULink ID
- Volunteer enters ULink Password
- Volunteer presses 'Register with ULink ID'

## 93. **Special Requirements**

None

## 94. **Pre-conditions**

- All fields must not be empty/null
- Volunteer must enter valid email address
- Date of Birth must be less recent current date (cannot be born in the future)

## 95. **Post-conditions**

- The volunteer's account is created and stored in the SQL database.

## 96. **Extension Points**

None

# Use Case Specification: Edit Volunteer

## 96.1 Edit Volunteer

### 96.2 Brief Description

This use case allows volunteers to edit their account information.

## 97. Flow of Events

### 97.1 Basic Flow

- Volunteer selects 'Edit Account Information'
- Volunteer makes changes to password, name, or email address
- Volunteer selects 'Apply Changes'
- Volunteer's record is updated accordingly in the SQL Database
- System sends Volunteer a confirmation email of account changes

### 97.2 Alternative Flows

Password Change if Registered with ULink

- Volunteer selects 'Edit Account Information'
- Volunteer clicks 'Change Password' Link
- Volunteer is directed to the ULink website where they can change the shared password for all their UofL accounts

## 98. Special Requirements

None

## 99. Pre-conditions

- **Volunteer must have a registered account and be logged in**
- **All fields must not be set as empty/null**
- Volunteer must enter valid email address

## 100. Post-conditions

- **The volunteer's account information is edited and updated in the SQL database.**

## 101. Extension Points

None

# Use Case Specification: Delete Volunteer

## 101.1 Delete Volunteer

### 101.2 Brief Description

This use case allows system administrators to delete volunteer accounts.

## 102. Flow of Events

### 102.1 Basic Flow

- Admin selects 'Manage Users'
- Admin selects 'Volunteers' Tab
- Admin selects Volunteer account from list
- Admin presses 'Delete User'
- Volunteer record is removed from the SQL Database

### 102.2 Alternative Flows

None

## 103. Special Requirements

None

## 104. Pre-conditions

- **There must be existing volunteer accounts**
- Volunteers are inactive or have requested their accounts be deleted

## 105. Post-conditions

- **The volunteer is no longer able to login to the secure area of the website.**
- **The volunteer's account information is edited and updated in the SQL database.**

## 106. Extension Points

None

# Use Case Specification: Add New Staff Member

## 106.1 Add New Staff Member

### 106.2 Brief Description

This use case allows new staff members to create an account.

## 107. Flow of Events

### 107.1 Basic Flow

- Staff navigates to 'Create Staff Account'
- Staff enters lousiville.edu email address
- Staff enters lousiville.edu email account password
- Staff enters Name
- Staff presses 'Create/Connect Account'
- Staff Information is stored in the SQL Database
- System sends a confirmation email

### 107.2 Alternative Flows

None

## 108. Special Requirements

None

## 109. Pre-conditions

- All fields must not be empty/null
- All staff must have a valid louisville.edu email account

## 110. Post-conditions

- The staff member's account is created and stored in the SQL database.

## 111. Extension Points

None

# Use Case Specification: Edit Staff Member

## 111.1 Edit Staff Member

### 111.2 Brief Description

This use case allows staff members to edit their account information.

## 112. Flow of Events

### 112.1 Basic Flow

- Staff selects 'Edit Account Information'
- Staff makes changes to name, or email address
- Staff selects 'Apply Changes'
- Staff's record is updated accordingly in the SQL Database
- System sends Staff a confirmation email of account changes

### 112.2 Alternative Flows

Change Password

- Staff selects 'Edit Account Information'
- Staff clicks 'Change Password' Link
- Staff is directed to the ULink website where they can change the shared password for all their UofL accounts

## 113. Special Requirements

None

## 114. Pre-conditions

- Staff must have a registered account and be logged in
- All fields must not be set as empty/null
- Staff must enter valid louisville.edu email address

## 115. Post-conditions

- The staff member's account information is edited and updated in the SQL database.

## 116. Extension Points

None

# Use Case Specification: Delete Staff Member

## 116.1 Delete Staff Member

### 116.2 Brief Description

This use case allows system administrators to delete staff accounts.

## 117. Flow of Events

### 117.1 Basic Flow

- Admin selects 'Manage Users'
- Admin selects 'Staff' Tab
- Admin selects Staff account from list
- Admin presses 'Delete User'
- Staff record is removed from the SQL Database

### 117.2 Alternative Flows

None

## 118. Special Requirements

None

## 119. Pre-conditions

- **There must be existing staff accounts**
- Staff member in question no longer works at the Cardinal Cupboard

## 120. Post-conditions

- **The staff member is no longer able to login to the secure area of the website.**
- **The staff member's account information is edited and updated in the SQL database.**

## 121. Extension Points

None

# Use Case Specification: Staff Member Login

## 121.1 Staff Member Login

### 121.2 Brief Description

This use case allows staff members to login to the secure area of the account.

## 122. Flow of Events

### 122.1 Basic Flow

- Staff presses 'Login' Tab
- Staff enters email address
- Staff enters password
- Staff presses 'Login' button
- System verifies email and password match a staff record in the SQL Database

### 122.2 Alternative Flows

Forgot Password

- Staff selects 'Forgot Password' link
- Staff is directed to the ULink website where they can reset their password

## 123. Special Requirements

None

## 124. Pre-conditions

- Staff must have created/connected a staff account on the website
- Stored Account Credentials must match the email and password the Staff enters

## 125. Post-conditions

- The staff member is able to access the secure area of the website.

## 126. Extension Points

None

# Use Case Specification: Volunteer Login

126.1

## 126.2 Volunteer Login

### 126.3 Brief Description

This use case allows volunteers to login to the secure area of the account.

## 127. Flow of Events

### 127.1 Basic Flow

- Volunteer presses 'Login' Tab
- Volunteer enters email address
- Volunteer enters password
- Volunteer presses 'Login' button
- System verifies email and password match a volunteer record in the SQL Database

### 127.2 Alternative Flows

UofL Student Volunteer Forgot Password

- Volunteer selects 'Forgot Password' link
- Volunteer is directed to the ULink website where they can reset their password

Non-UofL Student Volunteer Forgot Password

- Volunteer selects 'Forgot Password' link
- System emails Volunteer a secure link to reset password
- Volunteer clicks the secure link
- Volunteer enters new password
- Volunteer confirms new password
- Volunteer presses 'Reset Password'
- Volunteer's record is updated in the SQL Database

## 128. Special Requirements

None

## 129. Pre-conditions

- Volunteer must have created/connected an account on the website
- Stored Account Credentials must match the email and password the Volunteer enters

## 130. Post-conditions

- The volunteer is able to access the secure area of the website.

## 131. Extension Points

None

# Use Case Specification: Add Calendar Event

## 132. Add Calendar Event

### 132.1 Brief Description

This use case allows a staff member to add an event to the events calendar.

## 133. Flow of Events

### 133.1 Basic Flow

- Staff logs in
- Staff presses 'Manage Events' tab
- Staff presses 'Add Event'
- Staff enters Event Name
- Staff enters Event Description
- Staff enters Event Date/Time
- Staff enters Event Location
- Staff presses 'Submit Event'

### 133.2 Alternative Flows

None

## 134. Special Requirements

None

## 135. Pre-conditions

- Staff is logged in
- Date/Time cannot be earlier than current date (can't schedule in the past)
- All fields must not be empty/null

## 136. Post-conditions

- Event is created and appears on the Events Calendar section of the website
- The record of the event has been added to the SQL database.

## 137. Extension Points

None

# Use Case Specification: Edit Calendar Event

## 138. Edit Calendar Event

### 138.1 Brief Description

This use case allows a staff member to edit an event on the events calendar.

## 139. Flow of Events

### 139.1 Basic Flow

- Staff logs in
- Staff presses 'Manage Events' tab
- Staff presses 'Edit Event'
- Staff Selects an upcoming event from list
- Staff changes Event Name, Event Description, Event Date/Time, and/or Event Location
- Staff presses 'Submit Event'

### 139.2 Alternative Flows

None

## 140. Special Requirements

None

## 141. Pre-conditions

- Staff is logged in
- Upcoming events have been created
- Event must be upcoming, can't make changes to past events
- Date/Time cannot be earlier than current date (can't schedule in the past)
- All fields must not be empty/null

## 142. Post-conditions

- Event changes are saved and reflected on the Events Calendar section of the website
- The record of the event has been updated in the SQL database.

## 143. Extension Points

None

# Use Case Specification: Add item list

## 143.1 Brief Description

The system will allow donors to see what items are needed

## 144. Flow of Events

### 144.1 Basic Flow

- User logs in
- User opens item list
- User clicks add
- User adds item name
- User adds item description
- User adds item quantity
- User saves item list
- User exits item list

### 144.2 Alternative Flows

- User logs in
- User does not have clearance to edit item list

## 145. Pre-conditions

none

## 146. Post-conditions

Items are added

# Use Case Specification: Edit item list

## 146.1 Brief Description

The system will allow staff to add items to the list

## 147. Flow of Events

### 147.1 Basic Flow

- User logs in
- User opens item list
- User clicks edit
- User edits item list
- User saves item list
- User exits item list

## 147.2 Alternative Flows

- User logs in
- User does not have clearance to edit item list

## 148. Pre-conditions

Item list exists

## 149. Post-conditions

Item list is edited

# Use Case Specification: Delete item list

150.

## 150.1 Brief Description

The system will allow staff to delete items from the list

## 150.2 Flow of Events

## 150.3 Basic Flow

- User logs in
- User opens item list
- User clicks delete
- User deletes item list
- User exits item list

## 150.4 Alternative Flows

- User logs in
- User does not have authority to delete item list

## 151. Pre-conditions

151.1 Item list exists

## 152. Post-conditions

Item list is deleted

# Use Case Specification: Add hours

## 152.1 **Brief Description**

The system will allow staff and volunteers to log their hours

## 153. **Flow of Events**

### 153.1 **Basic Flow**

- User logs in
- User accesses time keeping page
- User logs hours
- User logs date
- User logs place
- User logs time

## 154. **Pre-conditions**

### 154.1 User is staff or volunteer

## 155. **Post-conditions**

Hours are added

# Use Case Specification: Edit hours

## 155.1 **Brief Description**

The system will allow staff and volunteers to edit their hours

## 156. **Flow of Events**

### 156.1 **Basic Flow**

- User logs in
- User accesses time keeping page
- User edits hours
- User edits date
- User edits place
- User edits time

## 157. **Pre-conditions**

### 157.1 User is staff or volunteer and has entered hours

## 158. **Post-conditions**

Hours are edited

# Use Case Specification: Delete hours

## 158.1 **Brief Description**

The system will allow staff and volunteers to delete their hours

## 159. **Flow of Events**

### 159.1 **Basic Flow**

- User logs in
- User accesses time keeping page
- User deletes hours

## 160. **Pre-conditions**

160.1 User is staff or volunteer and has entered hours

## 161. **Post-conditions**

Hours are deleted

# Use Case Specification: Add Recipes

## 161.1 **Brief Description**

161.2 The system will allow users to add recipes

## 162. **Flow of Events**

### 162.1 **Basic Flow**

- User logs in
- User accesses recipes page
- User enters date
- User enters recipes

## 163. **Pre-conditions**

163.1 none

## 164. **Post-conditions**

Recipes are added

# Use Case Specification: Edit Recipes

## 164.1 Brief Description

The system will allow users to edit recipes

## 165. Flow of Events

### 165.1 Basic Flow

- User logs in
- User accesses recipes page
- User enters date
- User edits recipes

## 166. Pre-conditions

### 166.1 Recipe has been entered

## 167. Post-conditions

Recipes are edited

# Use Case Specification: Delete Recipes

## 167.1 Brief Description

The system will allow users to delete recipes

## 168. Flow of Events

### 168.1 Basic Flow

- User logs in
- User accesses recipes page
- User deletes recipes

## 169. Pre-conditions

### 169.1 Recipe has been entered

## 170. Post-conditions

Recipes are deleted

# Use Case Specification: Add Inventory

## 170.1 Brief Description

The system will allow staff to add inventory

## 171. Flow of Events

### 171.1 Basic Flow

- User logs in
- User accesses inventory page
- User adds name
- User adds description
- User add quantity

## 172. Pre-conditions

### 172.1 none

## 173. Post-conditions

Inventory is added

# Use Case Specification: Edit Inventory

## 173.1 Brief Description

The system will allow staff to edit inventory

## 174. Flow of Events

### 174.1 Basic Flow

- User logs in
- User accesses inventory page
- User adds name
- User adds description
- User add quantity

## 175. Pre-conditions

### 175.1 Inventory exists

## 176. Post-conditions

Inventory is edited

# Use Case Specification: Delete Inventory

## 176.1 Brief Description

The system will allow staff to delete inventory

## 177. Flow of Events

### 177.1 Basic Flow

- User logs in
- User accesses inventory page
- User deletes inventory

## 178. Pre-conditions

### 178.1 Inventory exists

## 179. Post-conditions

Inventory is deleted

# Use Case Specification: Add helpful links

## 179.1 Brief Description

The system will allow an admin to add links to additional resources

## 180. Flow of Events

### 180.1 Basic Flow

- Admin logs in
- Admin adds link to REACH Center
- Admin adds link to Campus Housing
- Admin adds link to Campus Health Services

## 181. **Pre-conditions**

181.1 none

## 182. **Post-conditions**

Links are added

# Use Case Specification: Edit helpful links

## 182.1 **Brief Description**

The system will allow an admin to edit links to additional resources

## 183. **Flow of Events**

### 183.1 **Basic Flow**

- Admin logs in
- Admin edits links on website

## 184. **Pre-conditions**

184.1 Links have been added

## 185. **Post-conditions**

Links are edited

# Use Case Specification: Delete helpful links

## 185.1 **Brief Description**

The system will allow an admin to delete links to additional resources

## 186. **Flow of Events**

### 186.1 **Basic Flow**

- Admin logs in
- Admin deletes links on website

## 187. **Pre-conditions**

187.1 Links have been added

## 188. **Post-conditions**

Links are deleted

# Use Case Specification: Create backup

## 188.1 **Brief Description**

The system will have a backup

## 189. **Flow of Events**

### 189.1 **Basic Flow**

- Admin logs in
- Admin creates a backup of all data

## 190. **Pre-conditions**

190.1 none

## 191. **Post-conditions**

Data is backed up

# Use Case Specification: Create Recovery

## 191.1 **Brief Description**

The system will have a method of recovery

## 192. **Flow of Events**

### 192.1 **Basic Flow**

- Admin logs in
- Admin creates a method of recovery

## 193. **Pre-conditions**

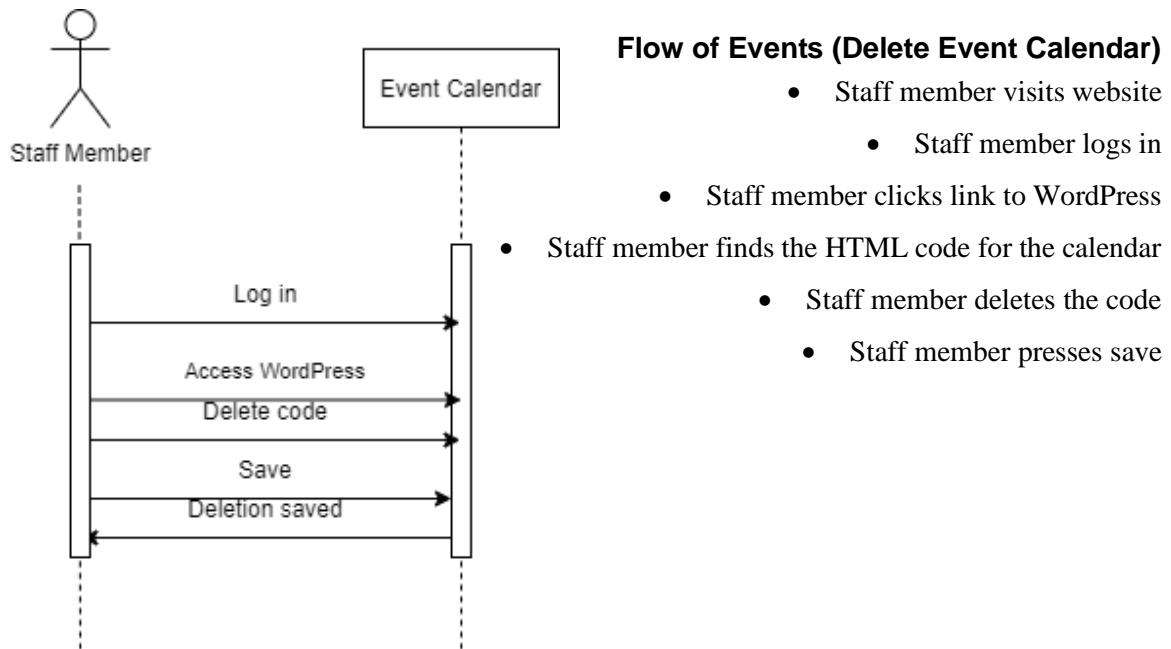
193.1 Data is backed up

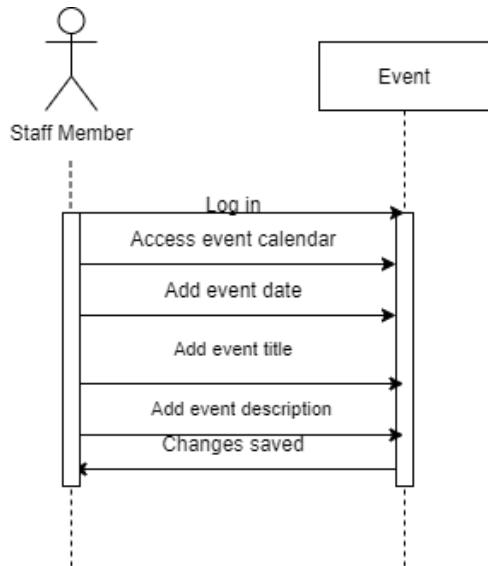
## 194. **Post-conditions**

Data is recovered

## Sequence Diagrams

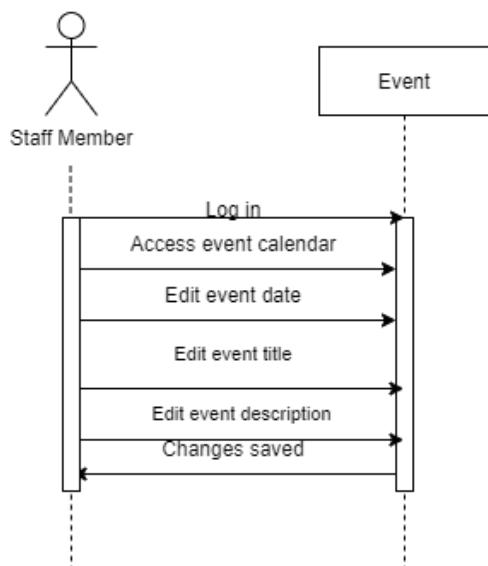
A sequence diagram shows the interactions between objects in a use case. between an actor, or user, and a class, or function. It shows how and in what order objects in a class work together.





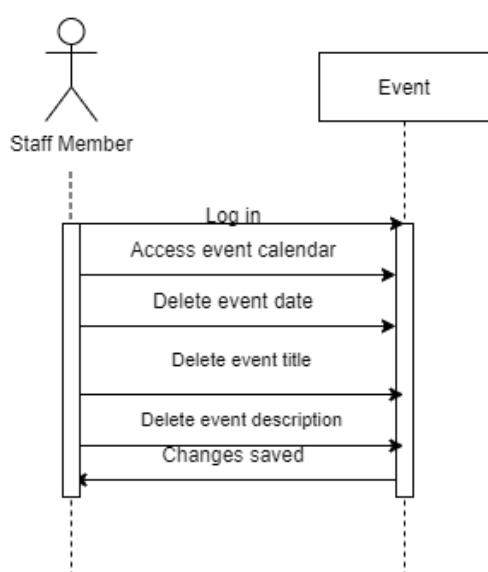
### Basic Flow (Add Event)

- Staff member visits website
  - Staff member logs in
- Staff member clicks link to WordPress
- Staff member accesses event calendar extension
  - Staff member adds event date
  - Staff member adds event description
  - Staff member adds event title
  - Staff member presses save



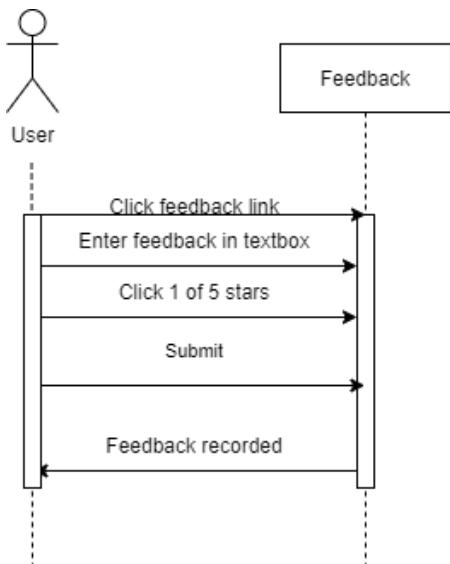
### Basic Flow (Edit Event)

- Staff member visits website
  - Staff member logs in
- Staff member clicks link to WordPress
- Staff member accesses event calendar extension
  - Staff member clicks on existing event
  - Staff member edits event date
  - Staff member edits event title
- Staff member edits event description
  - Staff member presses save



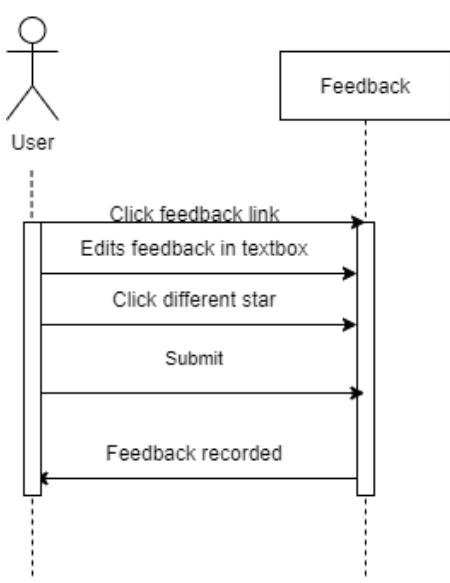
### Basic Flow (Delete Event)

- Staff member visits website
  - Staff member logs in
- Staff member clicks link to WordPress
- Staff member accesses event calendar extension
  - Staff member clicks on existing event
  - Staff member deletes event date
  - Staff member deletes event title
- Staff member deletes event description
  - Staff member presses save



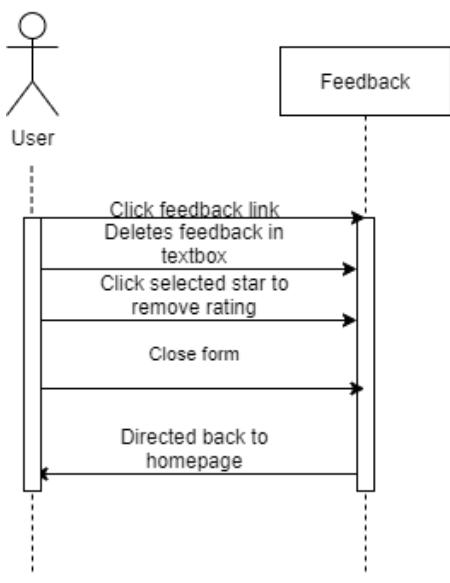
#### Basic Flow (Add Feedback)

- User visits website
- User clicks on feedback link
- User is brought to feedback form
- User enters feedback in text box provided
- User clicks on one of five stars to leave a rating
  - User presses submit



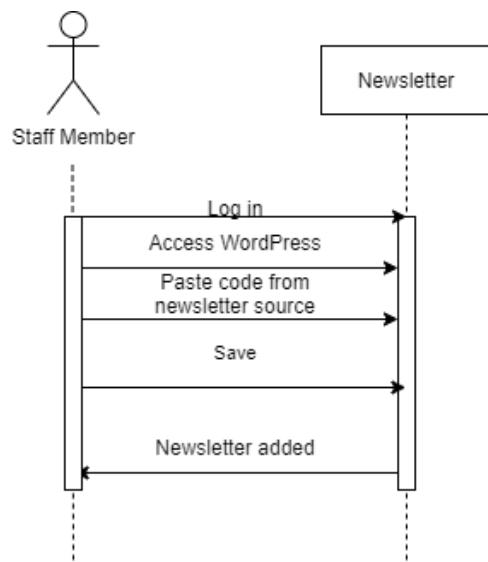
#### Basic Flow (Edit Feedback)

- User visits website
- User clicks on feedback link
- User is brought to feedback form
- User enters feedback in textbox provided
  - User edits feedback in text box
- User clicks on one of five stars to edit their rating
  - User presses submit



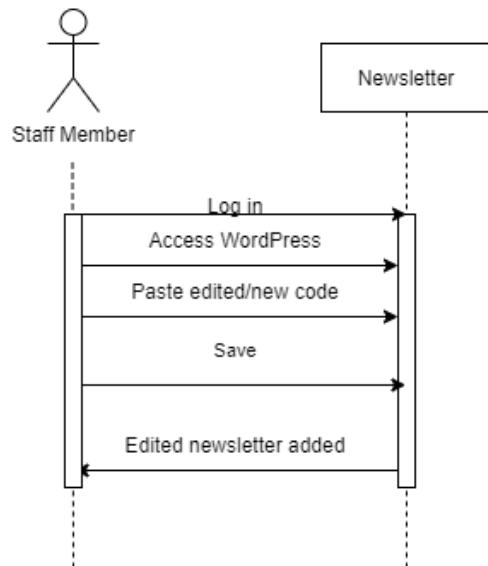
#### Basic Flow (Delete Feedback)

- User visits website
- User clicks on feedback link
- User is brought to feedback form
- User enters feedback in text box provided
  - User deletes feedback in text box
- User clicks on one of five stars to leave a rating
- User clicks on the same star again to remove a rating
  - User closes the form



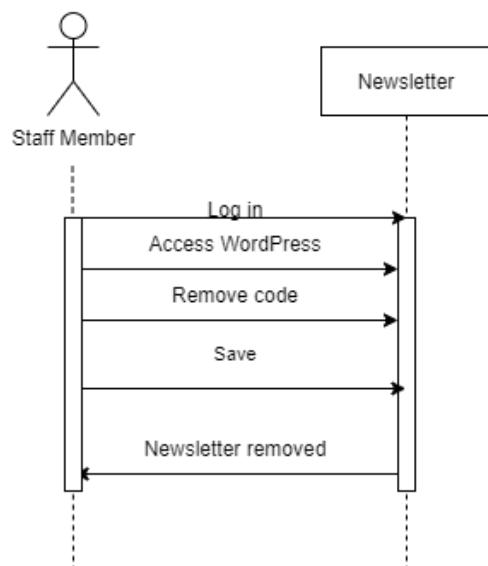
#### Basic Flow (Add Newsletter)

- Staff member visits the website
  - Staff member logs in
- Staff member clicks link to WordPress
- Staff member pastes code from newsletter provider
  - Staff member presses save



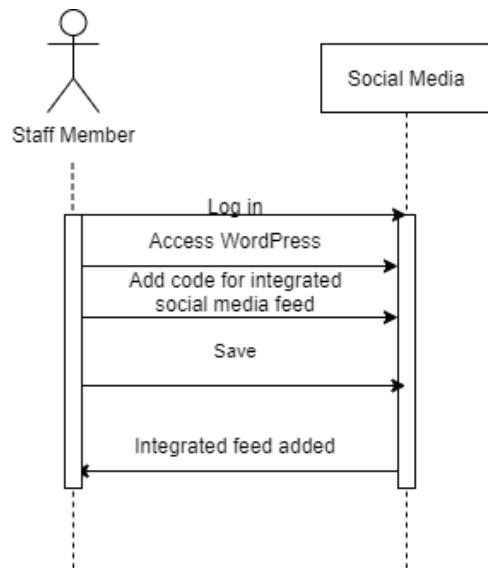
#### Basic Flow (Edit Newsletter)

- Staff member visits the website
  - Staff member logs in
- Staff member clicks link to newsletter provider
  - Staff member edits newsletter
  - Staff member returns to the website
  - Staff member clicks link to WordPress
- Staff member pastes code from newsletter provider
  - Staff member presses save



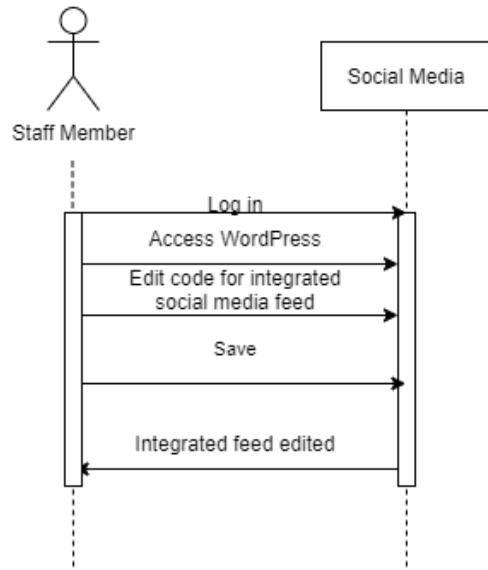
#### Basic Flow (Delete Newsletter)

- Staff member visits the website
  - Staff member logs in
- Staff member clicks link to WordPress
- Staff member deletes code from newsletter provider
  - Staff member presses save



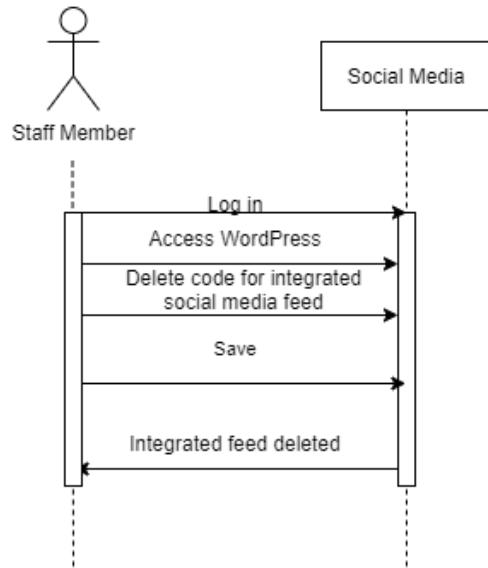
#### Basic Flow (Add social media)

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member adds code for the integrated social media feed
- Staff member presses save



#### Basic Flow (Edit social media)

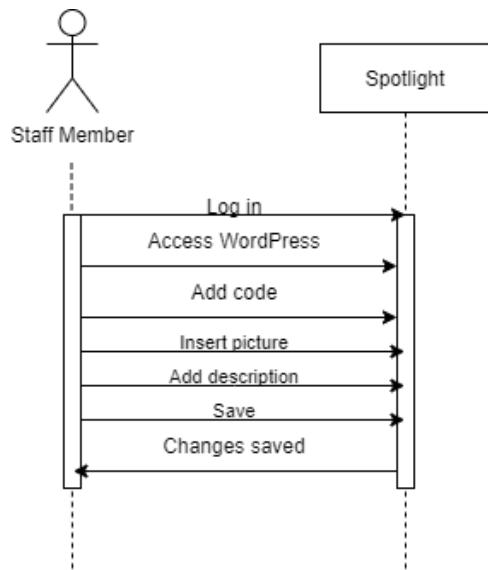
- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member edits code for the integrated social media feed
- Staff member presses save



#### Basic Flow (Delete social media)

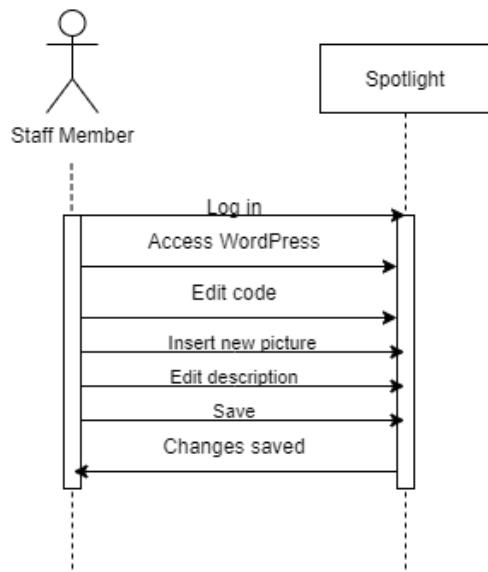
- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member deletes code for the integrated social media feed
- Staff member presses save





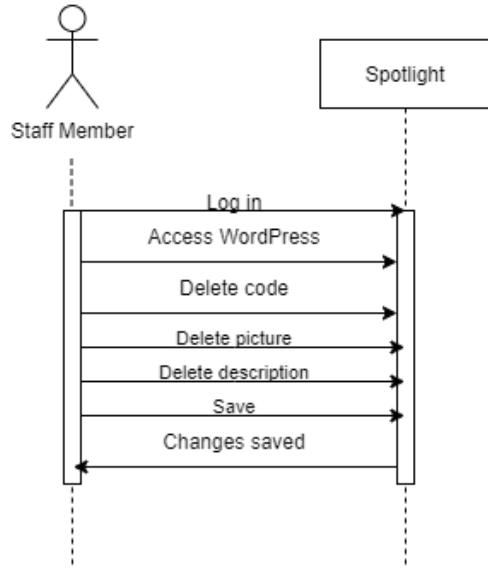
#### Basic Flow (Add Spotlight)

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member adds code for the spotlight
  - Staff member inserts picture
  - Staff member adds description
  - Staff member presses save



#### Basic Flow (Edit Spotlight)

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member inserts new picture
- Staff member edits description
- Staff member presses save

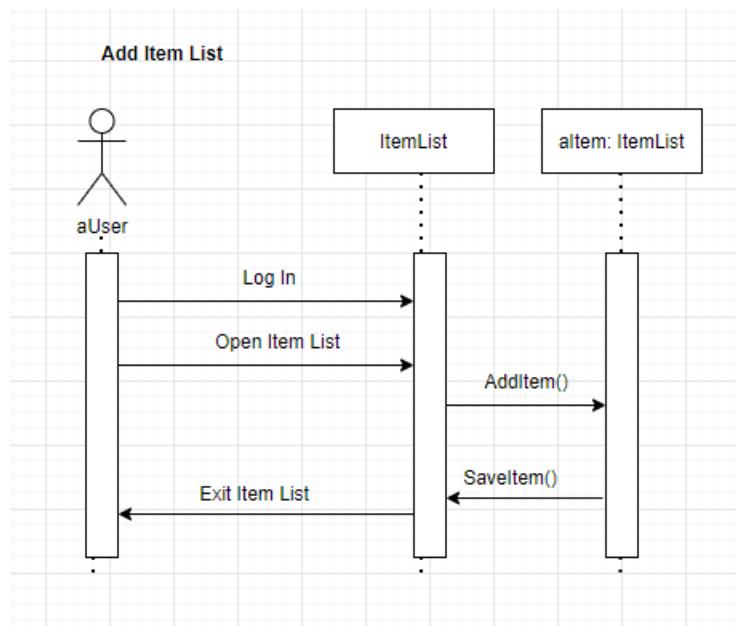


#### Basic Flow (Delete Spotlight)

- Staff member visits website
- Staff member logs in
- Staff member clicks link to WordPress
- Staff member deletes code for the spotlight
- Staff member presses save

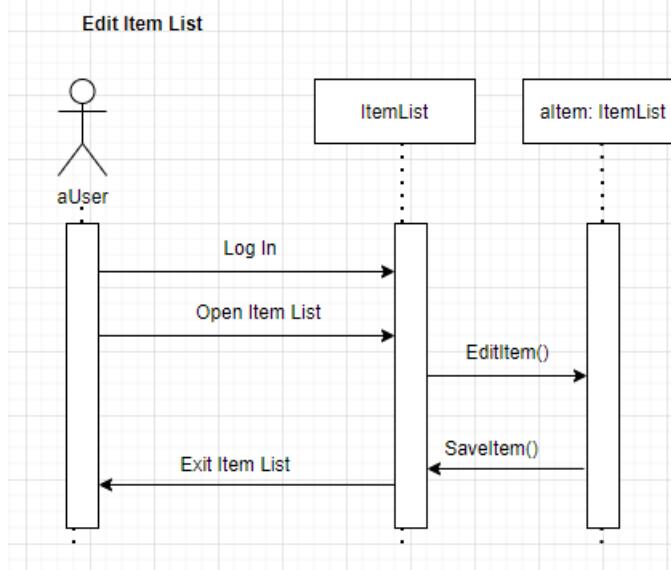
#### 194.1 Add Item List Basic Flow

- User logs in
- User opens item list
- User clicks add
- User adds item name
- User adds item description
- User adds item quantity
- User saves item list
- User exits item list



#### 194.2 Edit Item List Basic Flow

- User logs in
- User opens item list
- User clicks edit
- User edits item list
- User saves item list
- User exits item list

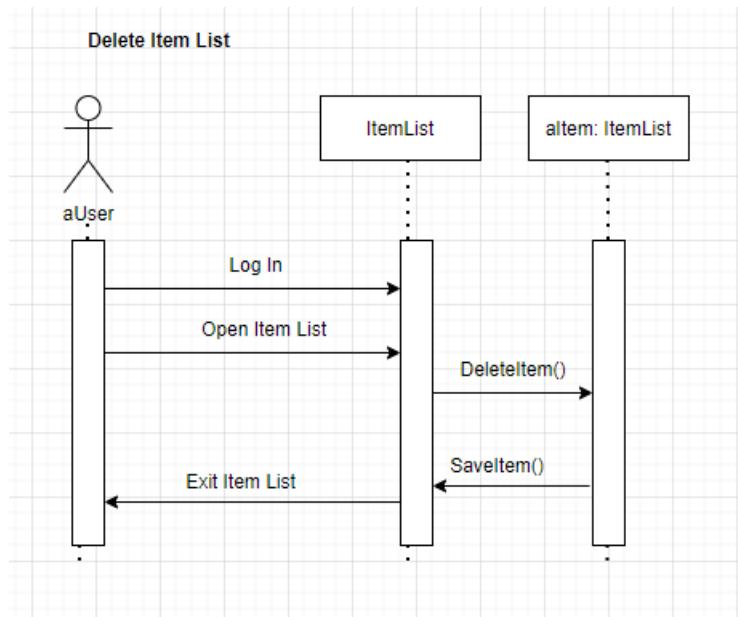


194.3

194.4

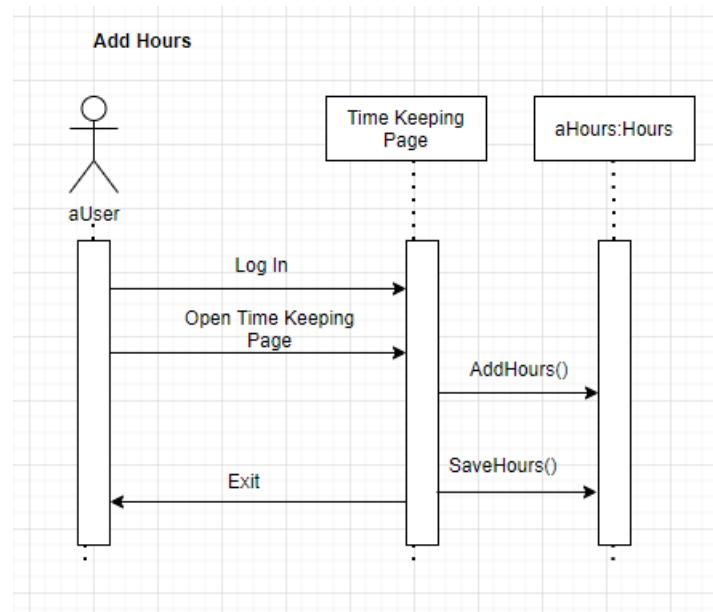
194.5 Delete Item List Basic Flow

- User logs in
- User opens item list
- User clicks delete
- User deletes item list
- User exits item list



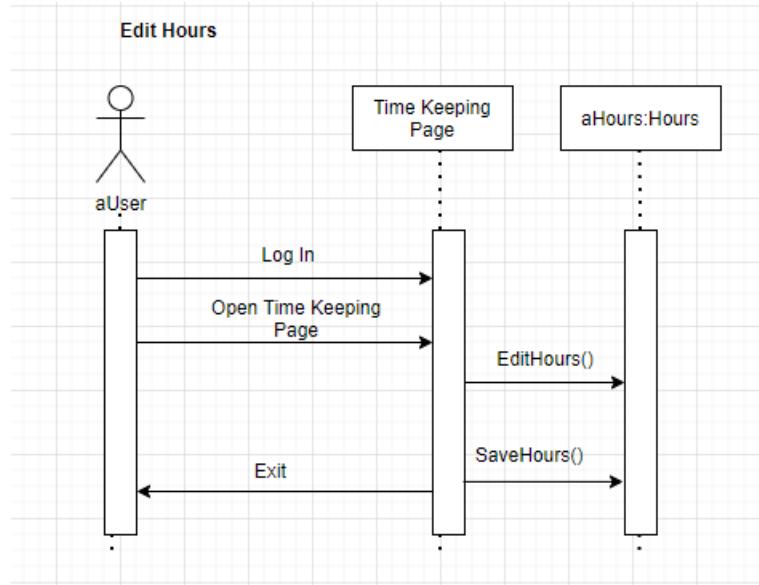
194.6 Add Hours Basic Flow

- User logs in
- User accesses time keeping page
- User logs date
- User logs place
- User logs time



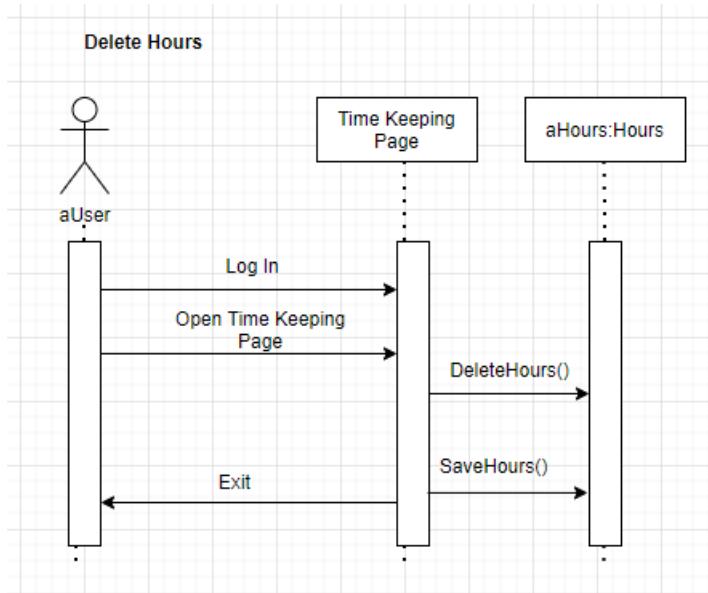
#### 194.7 Edit Hours Basic Flow

- User logs in
- User accesses time keeping page
- User edits hours
- User edits date
- User edits place
- User edits time



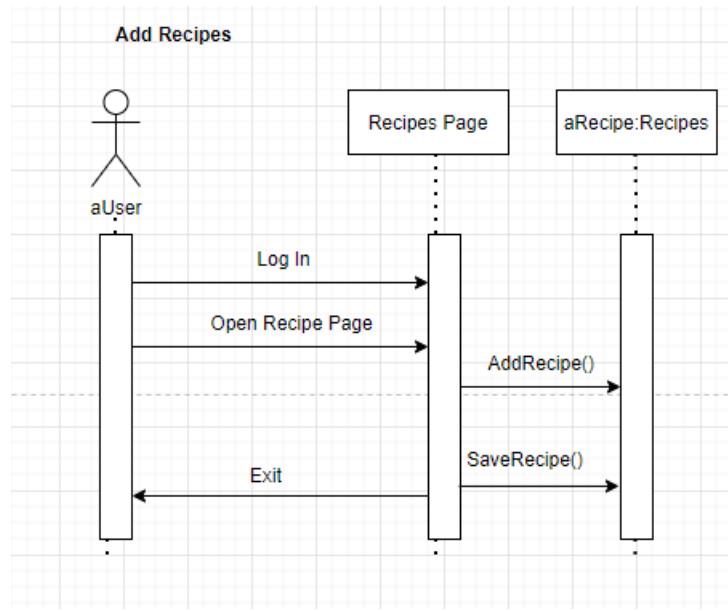
#### 194.8 Delete Hours Basic Flow

- User logs in
- User accesses time keeping page
- User deletes hours
- User SaveHours



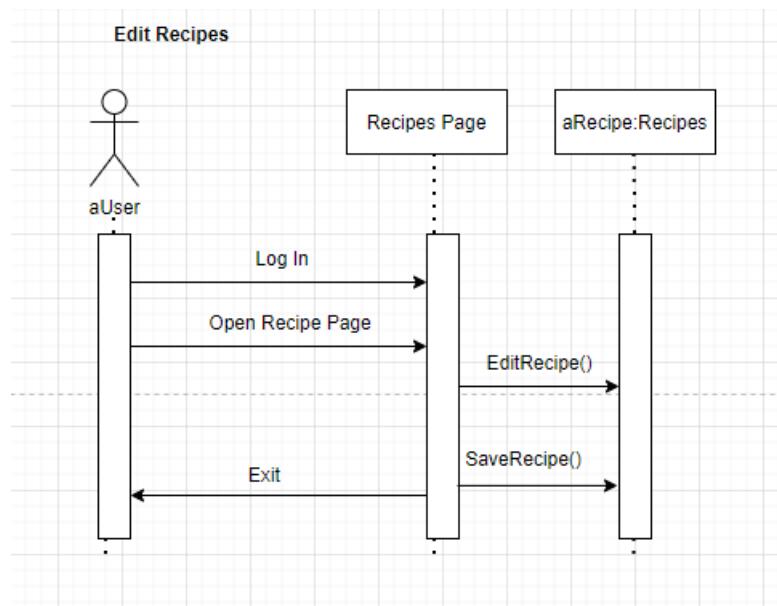
#### 194.9 Add Recipes Basic Flow

- User logs in
- User accesses recipes page
- User enters date
- User adds recipes
- User saves recipes
- User exits



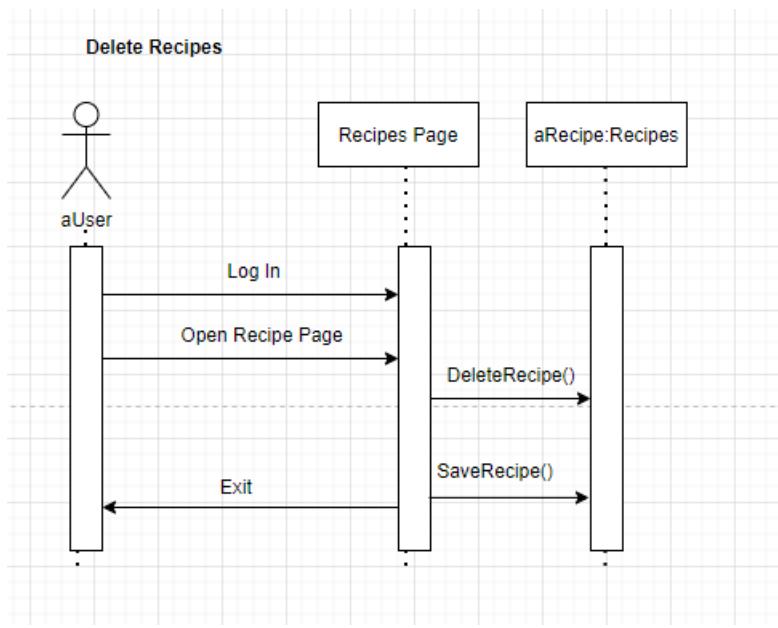
#### 194.10 Edit Recipes Basic Flow

- User logs in
- User accesses recipes page
- User enters date
- User edits recipes
- User saves recipe
- User exits



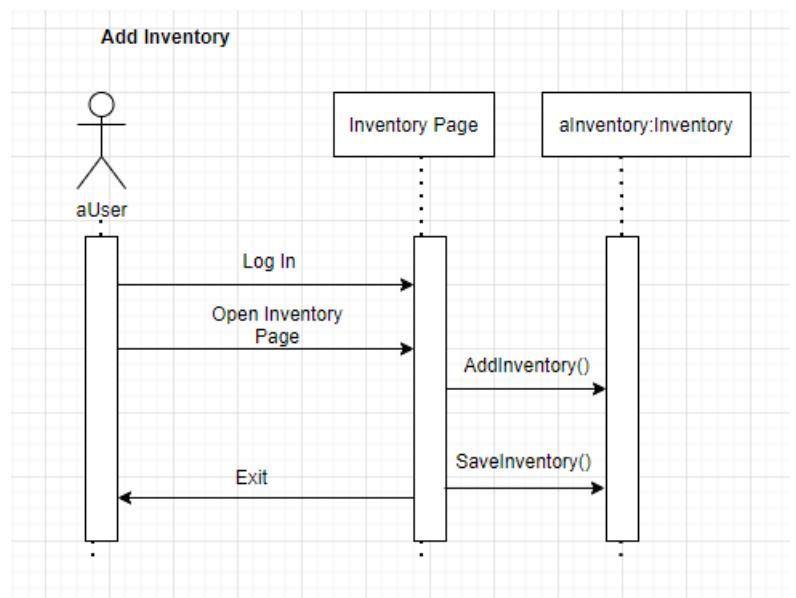
#### 194.11 Delete Recipe Basic Flow

- User logs in
- User accesses recipes page
- User deletes recipes
- User saves recipes
- User exits



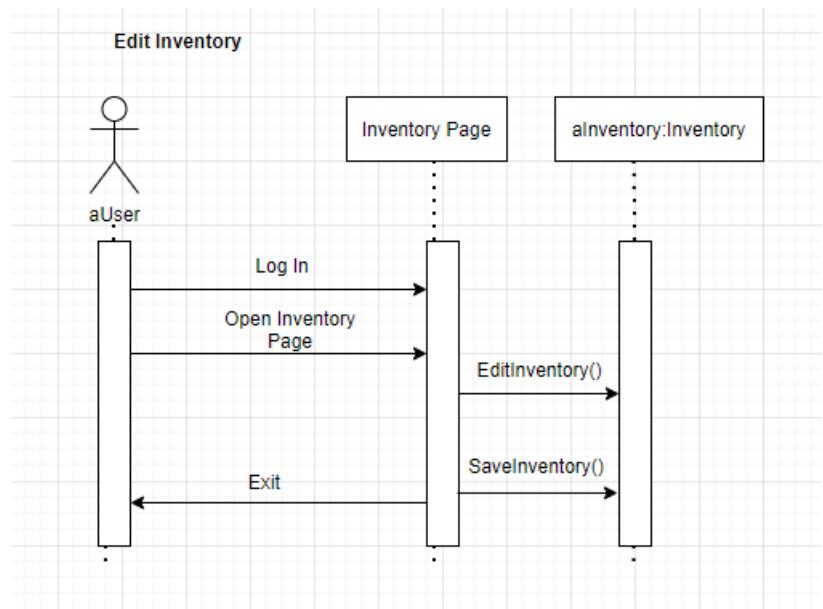
#### 194.12 Add Inventory Basic Flow

- User logs in
- User accesses inventory page
- User adds inventory name
- User adds inventory description
- User adds inventory quantity
- User saves inventory
- User exits



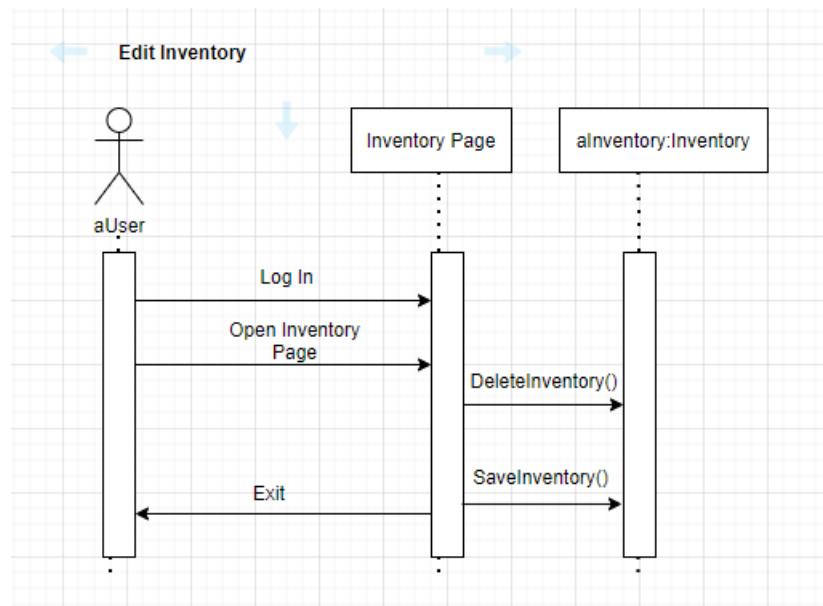
#### 194.13 Edit Inventory Basic Flow

- User logs in
- User accesses inventory page
- User adds name
- User adds description
- User add quantity
- User saves inventory
- User exits



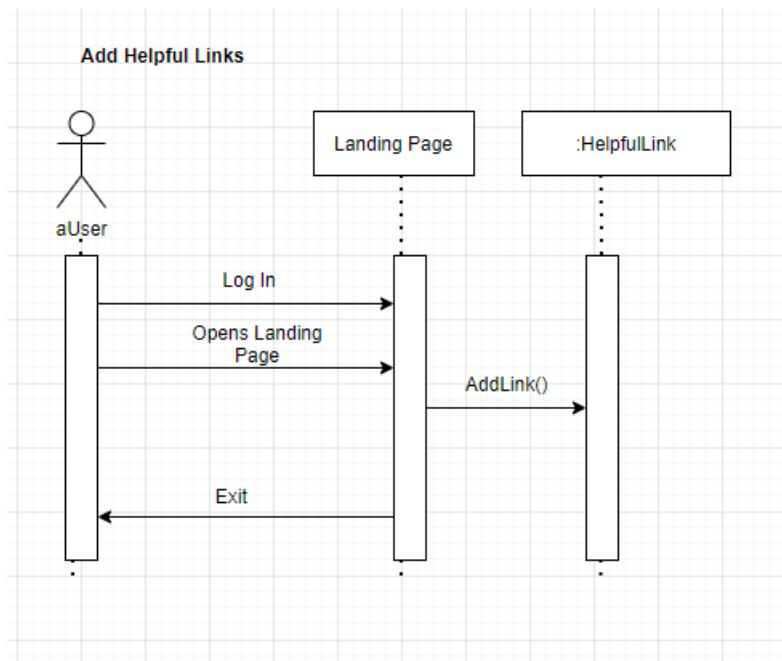
#### 194.14 Delete Inventory Basic Flow

- User logs in
- User accesses inventory page
- User deletes inventory
- User saves inventory
- User exits



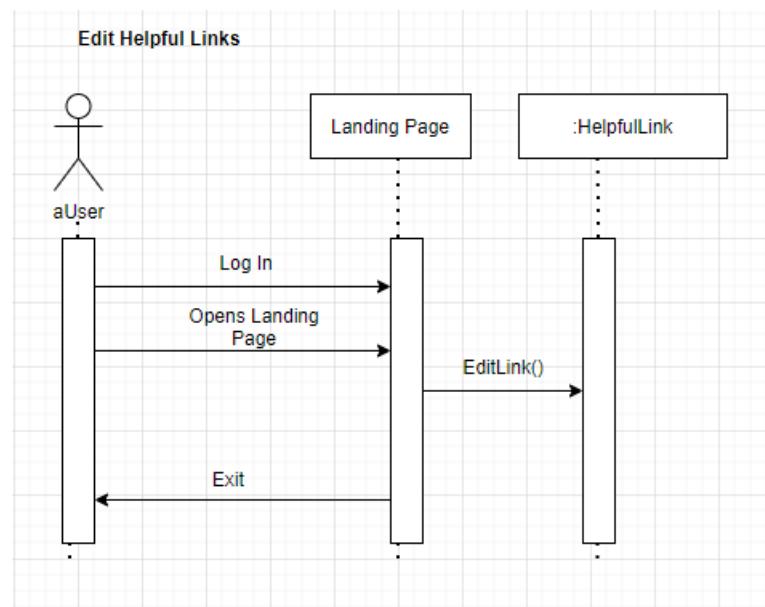
#### 194.15 Add Helpful Links Basic Flow

- Admin logs in
- Admin opens landing page
- Admin adds link to REACH Center
- Admin exits



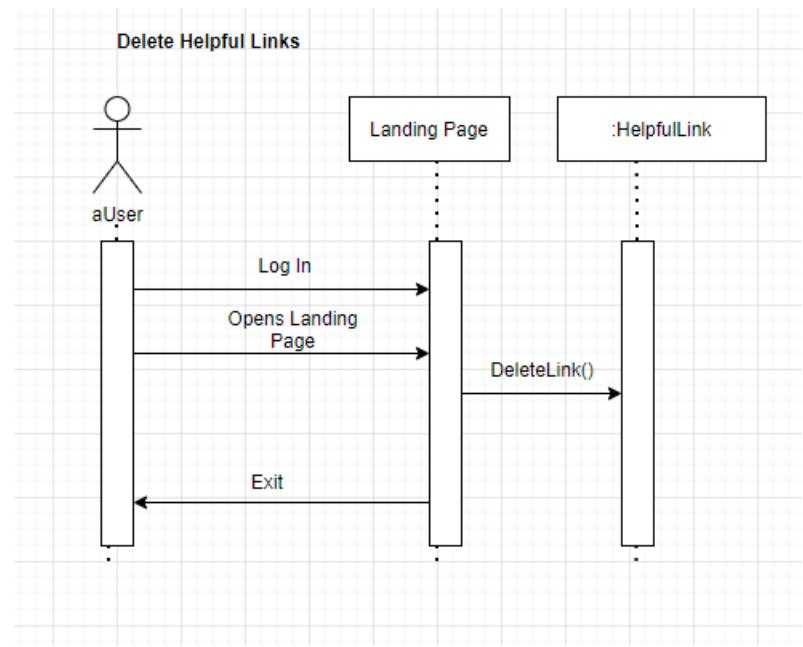
#### 194.16 Edit Helpful Links Basic Flow

- Admin logs in
- Admin opens landing page
- Admin edits links
- Admin exits



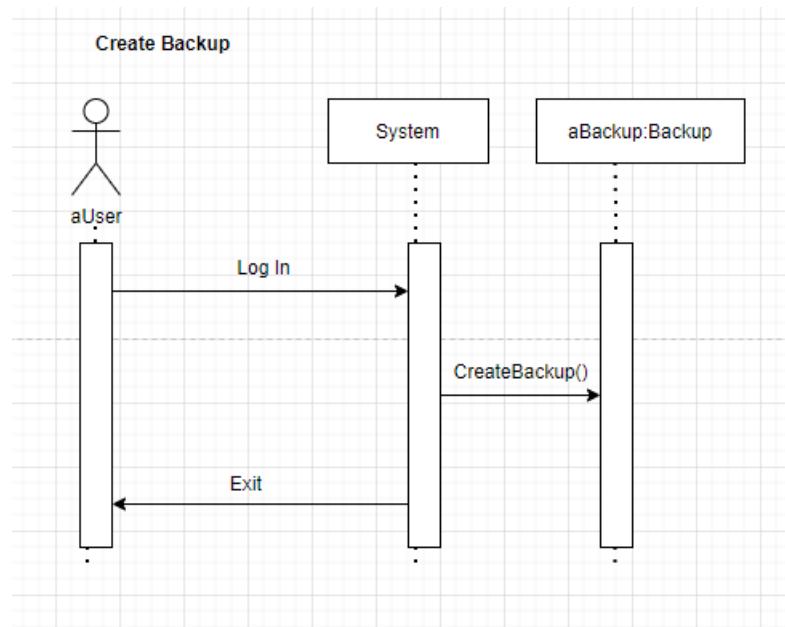
194.17 Admin Deletes Helpful Link Basic Flow

- Admin logs in
- Admin opens landing page
- Admin deletes helpful link
- Admin exits



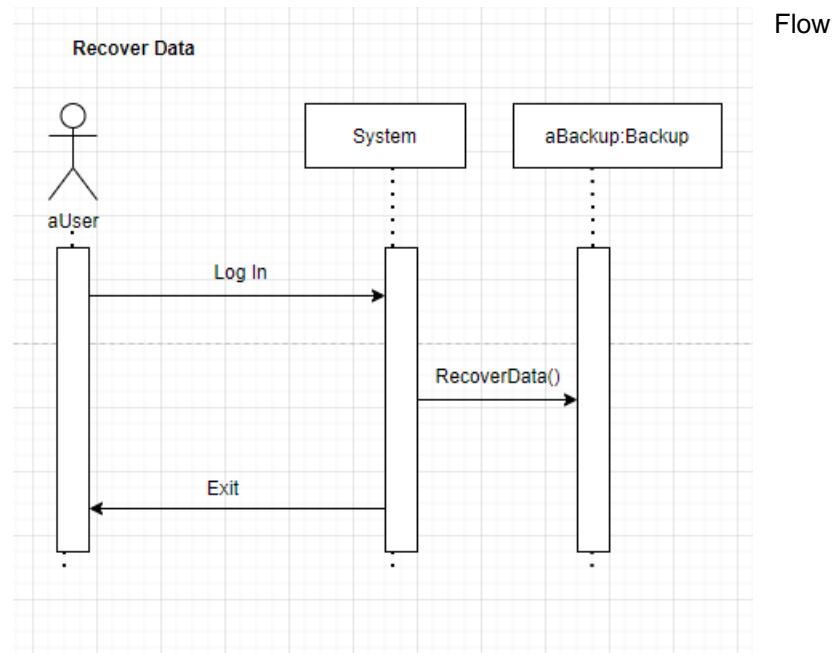
194.18 Create Backup Basic Flow

- Admin logs in
- Admin creates a backup of all data
- Admin exits

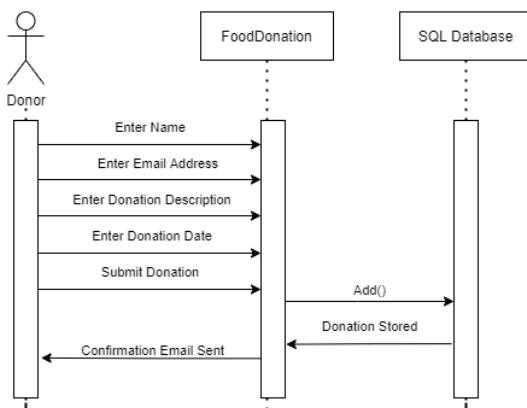


### 194.19 Recover Data Basic

- Admin logs in
- Admin recovers data
- Admin exits



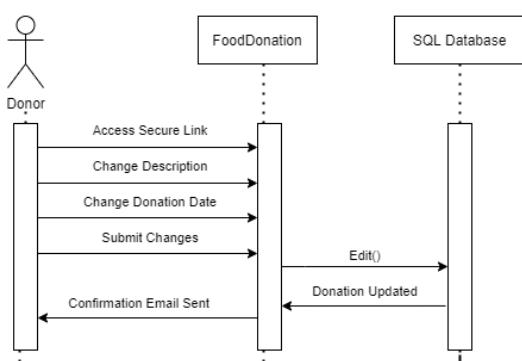
### Add Food Donation



### Basic Flow (Add Food Donation)

- Donor enters Name
- Donor enters Email Address
- Donor enters Donation Description
- Donor selects Donation Date
- Donor submits Donation Information
- Donor's Name, Email, Description, and Date are stored in the SQL Database
- System sends Donor a confirmation email with donation details and secure link

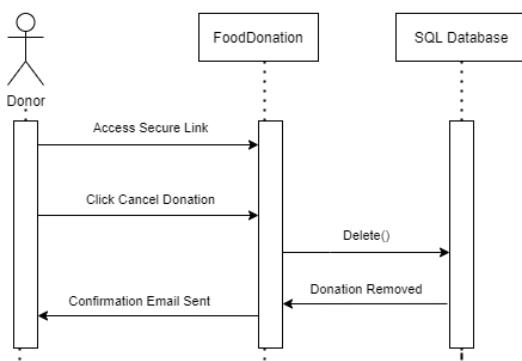
#### Edit Food Donation



#### Basic Flow (Edit Food Donation)

- Donor accesses donation record using secure link
- Donor makes changes to Donation Description
- Donor makes changes to Donation Date
- Donor presses 'Save Changes'
- Record is updated in SQL Database
- System sends Donor a confirmation email with the updated information and secure link

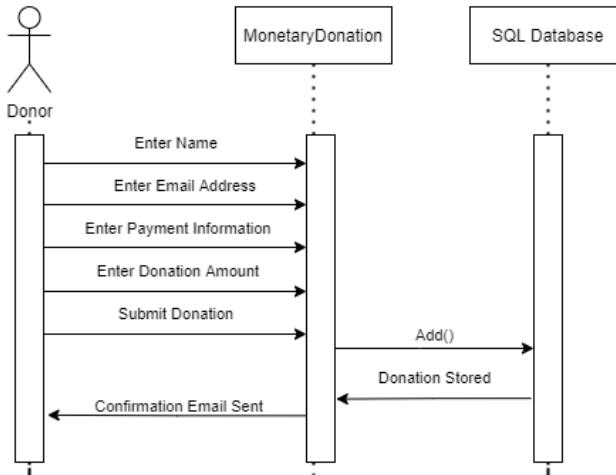
#### Delete Food Donation



#### Basic Flow (Delete Food Donation)

- Donor accesses donation record using secure link
- Donor presses 'Cancel Donation'
- Record is removed from storage in SQL Database
- System sends Donor a confirmation email

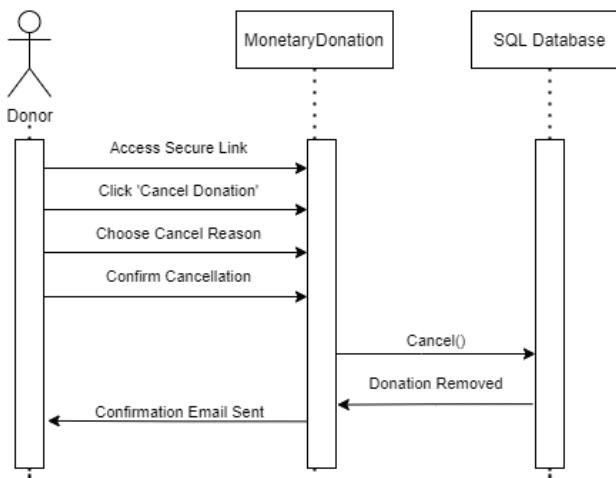
### Add Monetary Donation



### Basic Flow (Add Monetary Donation)

- Donor enters Name
- Donor enters Email Address
- Donor enters Card Number
- Donor enters Card Expiration Date
- Donor enters Security Code
- Donor enters Donation Amount
- Donor presses 'Submit'
- Donor's Name, Email, Payment Information, and Amount are stored in the SQL Database
- System sends Donor a confirmation email with donation details and a secure link

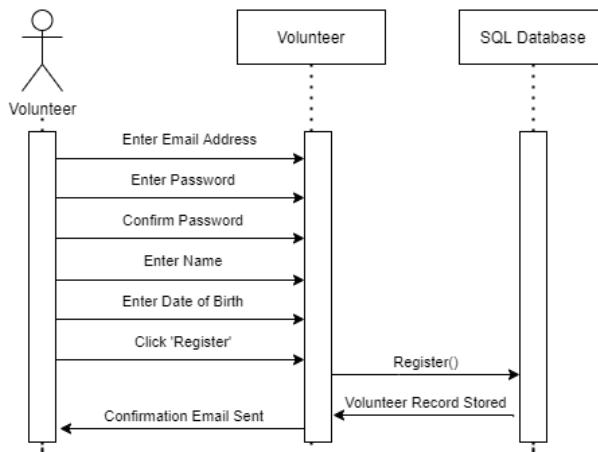
### Cancel Monetary Donation



### Basic Flow (Cancel Monetary Donation)

- Donor access donation information from secure link in email
- Donor presses 'Cancel Donation'
- Donor can optionally explain why they are cancelling their donation
- Donor presses 'Confirm'
- System sends confirmation email to Donor

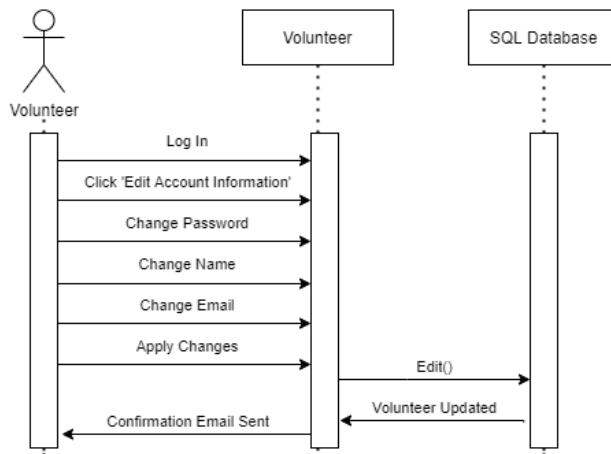
### Register New Volunteer



### Basic Flow (Register New Volunteer)

- Volunteer enters Email Address
- Volunteer enters Password
- Volunteer reenters Password to confirm
- Volunteer enters Name
- Volunteer enters Date of Birth
- Volunteer presses 'Register'
- Volunteer's Information is stored in the SQL Database
- System sends a confirmation email

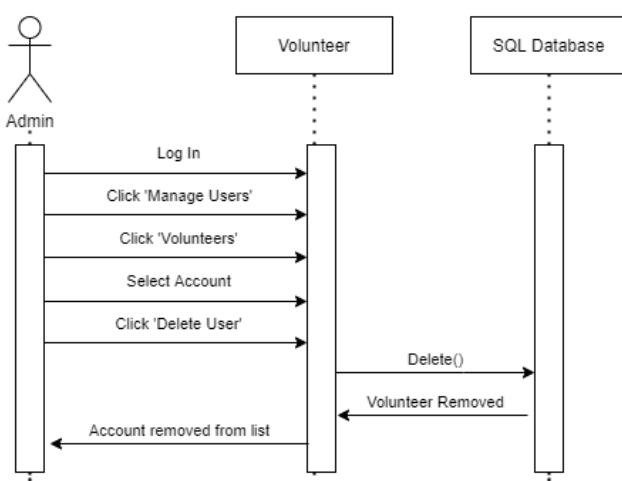
### Edit Volunteer



### Basic Flow (Edit Volunteer)

- Volunteer selects 'Edit Account Information'
- Volunteer makes changes to password
- Volunteer makes changes to name
- Volunteer makes changes to email address
- Volunteer selects 'Apply Changes'
- Volunteer's record is updated accordingly in the SQL Database
- System sends Volunteer a confirmation email of account changes

### Delete Volunteer

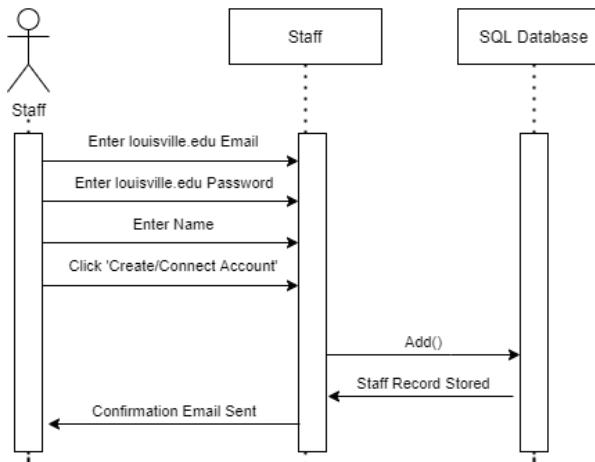


### Basic Flow (Delete Volunteer)

- Admin selects 'Manage Users'
- Admin selects 'Volunteers' Tab
- Admin selects Volunteer account from list
- Admin presses 'Delete User'
- Volunteer record is removed from the SQL Database



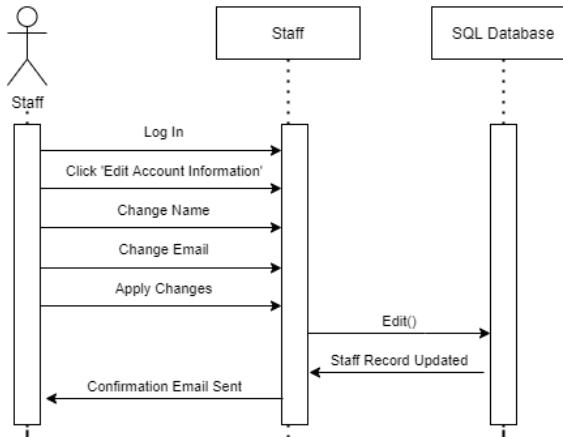
#### Add New Staff Member



#### Basic Flow (Add New Staff Member)

- Staff enters louisville.edu email address
- Staff enters louisville.edu email account password
- Staff enters Name
- Staff presses 'Create/Connect Account'
- Staff Information is stored in the SQL Database
- System sends a confirmation email

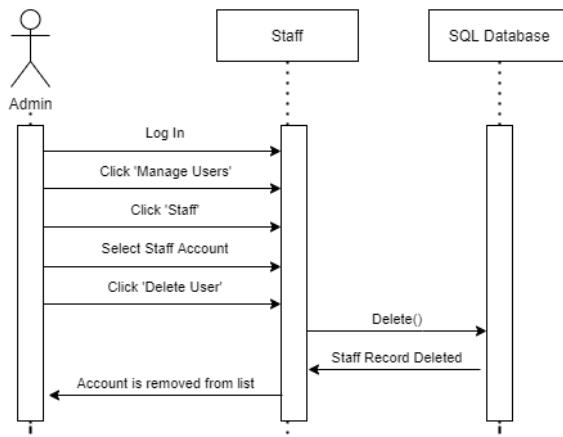
#### Edit Staff Member



#### Basic Flow (Edit Staff Member)

- Staff selects 'Edit Account Information'
- Staff makes changes to name
- Staff makes changes to email address
- Staff selects 'Apply Changes'
- Staff's record is updated accordingly in the SQL Database
- System sends Staff a confirmation email of account changes

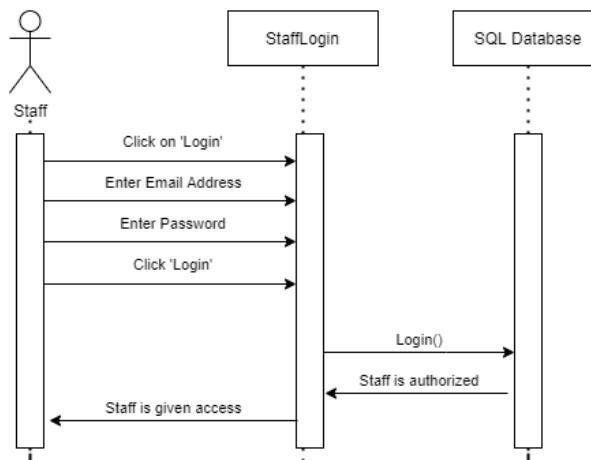
#### Delete Staff Member



#### Basic Flow (Delete Staff Member)

- Admin selects 'Manage Users'
- Admin selects 'Staff' Tab
- Admin selects Staff account from list
- Admin presses 'Delete User'
- Staff record is removed from the SQL Database

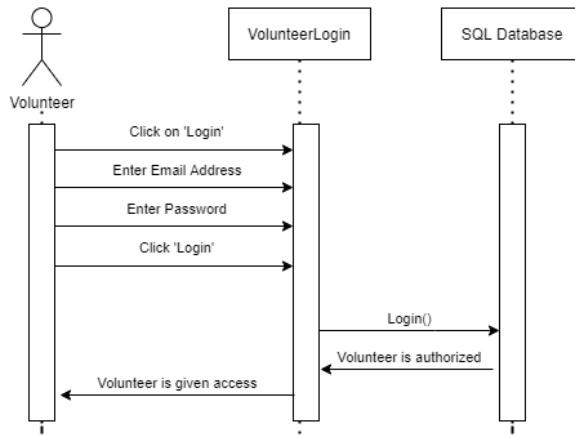
### Staff Login



### Basic Flow (Staff Member Login)

- Staff presses 'Login' Tab
- Staff enters email address
- Staff enters password
- Staff presses 'Login' button
- System verifies email and password match a staff record in the SQL Database

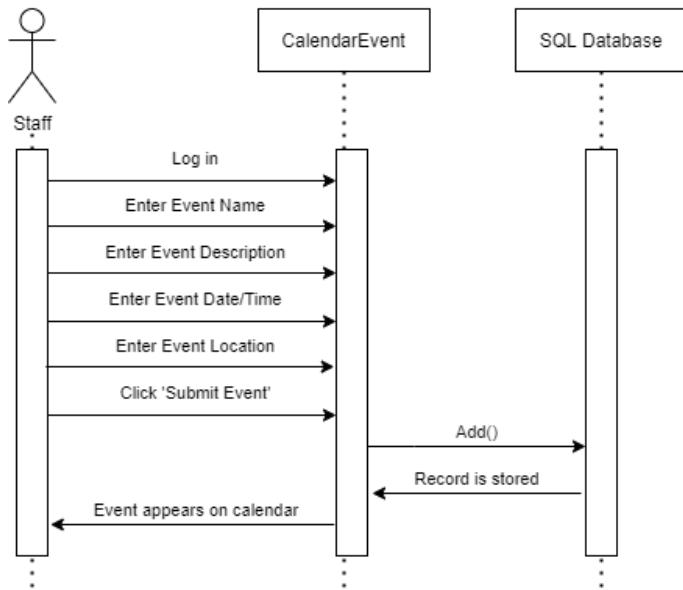
### Volunteer Login



### Basic Flow (Volunteer Login)

- Volunteer presses 'Login' Tab
- Volunteer enters email address
- Volunteer enters password
- Volunteer presses 'Login' button
- System verifies email and password match a volunteer record in the SQL Database

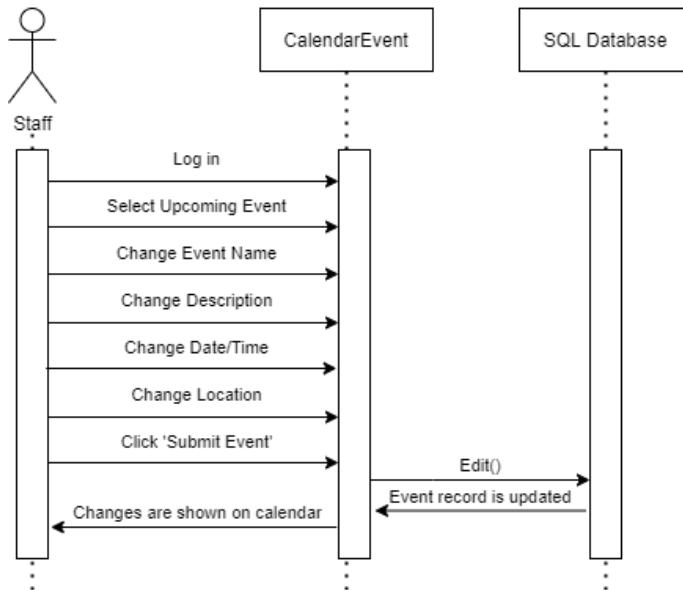
### Add Calendar Event



### Basic Flow (Add Calendar Event)

- Staff logs in
- Staff presses 'Manage Events' tab
- Staff presses 'Add Event'
- Staff enters Event Name
- Staff enters Event Description
- Staff enters Event Date/Time
- Staff enters Event Location
- Staff presses 'Submit Event'

### Edit Calendar Event



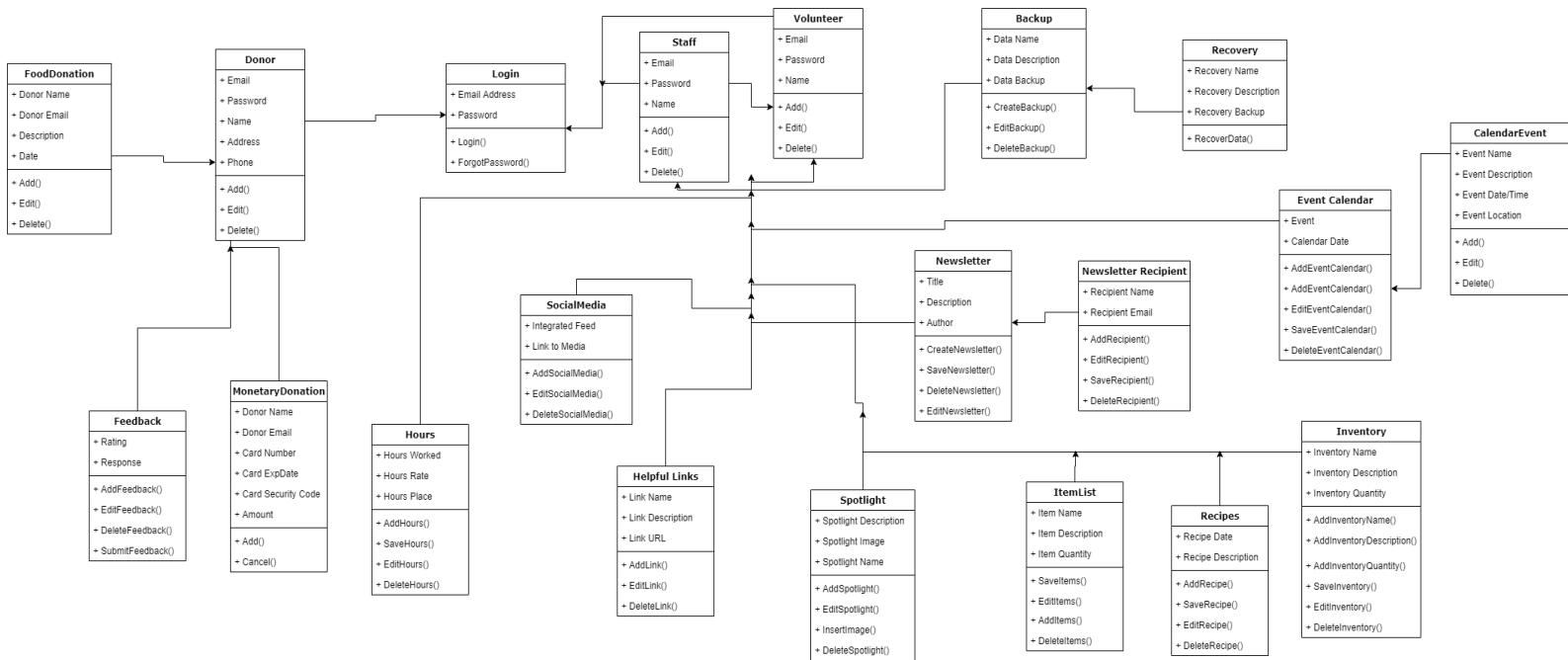
### Basic Flow (Edit Calendar Event)

- Staff logs in
- Staff presses 'Manage Events' tab
- Staff presses 'Edit Event'
- Staff Selects an upcoming event
- Staff changes Event Name
- Staff changes Event Description
- Staff changes Event Date/Time
- Staff changes Event Location
- Staff presses 'Submit Event'

## Class Diagram

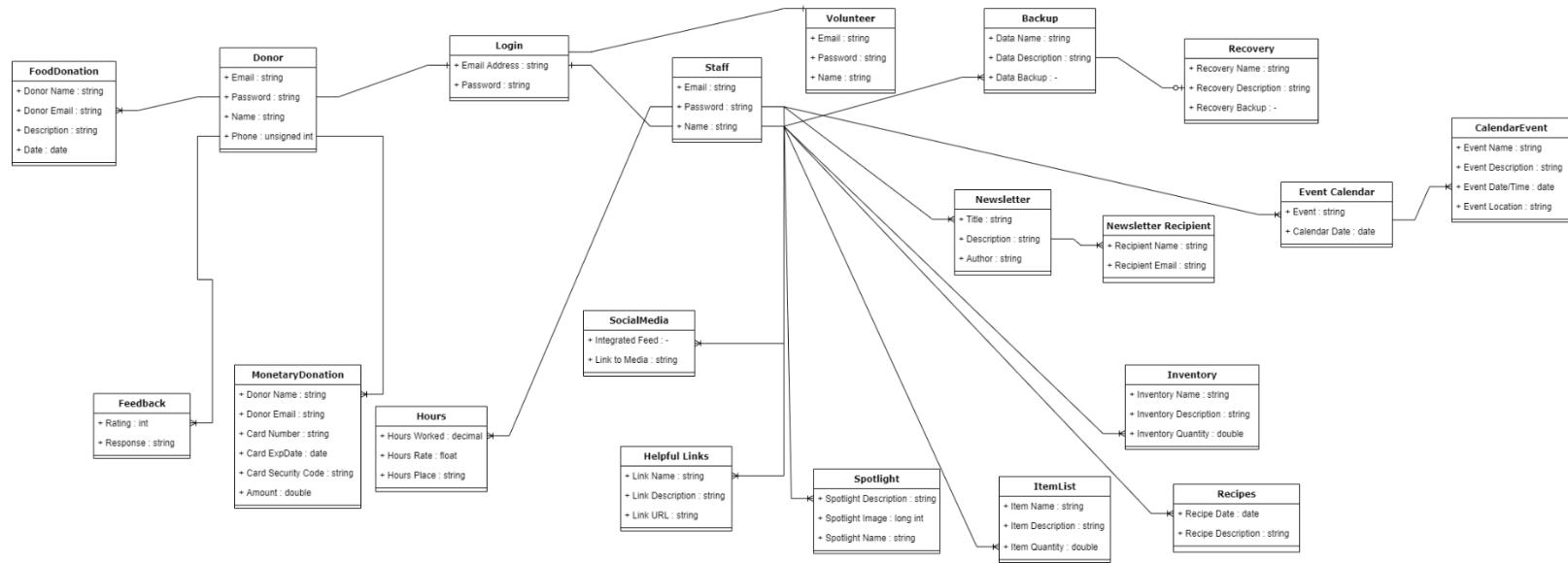
A class diagram is a model that shows the classes and relationships among classes that remain constant over time. A class is something that is used to create an instance or object. The methods (in the lower section of each class) are actions the user performs inside the class. The attributes (in the upper section of each class) lists what information makes up each class. For example, a staff member would add to the item list by pressing “add items” (method) and entering the name, description, and quantity (attributes) then pressing “save items.”

Verb-noun analysis was used to create this diagram. Nouns make up the class name and attributes while verbs make up the methods for each class. The verbs and nouns come from the use case descriptions.



## Database Design & Data Definitions

To achieve 3NF (third normal form), all duplicates must be removed, and all attributes must be dependent on the primary key alone. This diagram shows how all the different classes interact with each other and being in 3NF ensures that the diagrams integrity is intact.



**Food Donation table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Donor Name	Donor's Name	25 Bytes	string	Primary Key
Donor Email	Donor's Email	32 bytes	string	N/A
Description	Donation Description	100 bytes	string	N/A
Date	Date of Donation	4 bytes	date	N/A

**Donor table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Email	Donor's Email	32 bytes	string	Primary Key
Password	Donor's Password	16 bytes	string	N/A
Name	Name of Donor	25 Bytes	string	N/A
Phone	Phone of Donor	10 Bytes	Uint	N/A

**Login table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Email Address	User Email	32 Bytes	string	Primary Key
Password	User Password	16 bytes	string	N/A

**Staff/Volunteer table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Email	User Email	32 Bytes	string	Primary Key
Password	User Password	16 bytes	string	N/A
Name	User Name	25 bytes	string	N/A

**Feed Back table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Rating	User Rating	4 bytes	uint	Primary Key
Response	User Response	100 bytes	string	N/A

**Monetary Donation table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Donor Name	Donor's Name	25 Bytes	string	Primary Key
Donor Email	Donor's Email	32 bytes	string	N/A
Card Number	Donor Credit Card Number	19 bytes	string	N/A
Card ExpDate	Card ExpDate	4 bytes	date	N/A
Card Security Code	Card security code	3 bytes	string	N/A
Amount	Amount of donation	4 bytes	Long Unsigned int	N/A

**Hours table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Hours Worked	Amount of hours worked	16 bytes	decimal	Primary Key
Hour Rate	Pay per hours	4 bytes	float	N/A
Hours Place	Name of place worked	32 bytes	string	N/A

**Social Media table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Integrated Feed	Social media feeds			N/A
Links to Social Media	Social Media Links	100 bytes	string	N/A

**Helpful Links table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Link Name	Link's Name	25 Bytes	string	Primary Key
Link Description	Link's description	100 Bytes	string	N/A
Link URL	URL	100 bytes	string	N/A

**Backup table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
<b>Data Name</b>	Data's Name	<b>25 Bytes</b>	string	<b>Primary Key</b>
<b>Data Description</b>	Description of the Data Base	<b>100 Bytes</b>	string	N/A
<b>Data Backup</b>	Physical backup of Data		Data Base	N/A

**Recovery table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
<b>Recovery Name</b>	Recovery Data Base Name	<b>25 Bytes</b>	string	<b>Primary Key</b>
<b>Recovery Description</b>	Description of the Recovery Data Base	<b>100 Bytes</b>	string	N/A
<b>Recovery Backup</b>	The physical Data Base to be Recovered		Data Base	N/A

**Event Calendar table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
<b>Event</b>	Social media feeds	<b>25 Bytes</b>	string	<b>Primary Key</b>
<b>Calendar Date</b>	Date	<b>4 Bytes</b>	date	N/A

**Calendar Event table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
<b>Event Name</b>	Event Name	<b>25 Bytes</b>	string	<b>Primary Key</b>
<b>Event Description</b>	Event Description	<b>100 bytes</b>	string	N/A
<b>Event Date/Time</b>	Event Date and Time	<b>4 Bytes</b>	date	N/A
<b>Event Location</b>	Location of event	<b>32 Bytes</b>	string	N/A

**Newsletter table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Title	Newsletter Title	25 Bytes	string	Primary Key
Newsletter Description	Description of the Newsletter	100 Bytes	string	N/A
Author	Author of Newsletter	25 Bytes	string	N/A

**Newsletter Recipient table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Recipient Name	Recipient Name	25 Bytes	string	Primary Key
Recipient Email	Recipient Email	32 Bytes	string	N/A

**Spotlight table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Spotlight Name	Spotlight Name	25 Bytes	string	Primary Key
Spotlight image	Image to go along with spotlight		png	N/A
Spotlight Description	Spotlight description	25 Bytes	string	N/A

**Item List table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Item Name	Item Name	25 Bytes	string	Primary Key
Item Description	Description of the item	25 Bytes	string	N/A
Item Quantity	Quantity of item	8 Bytes	Double	N/A

**Recipes table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
Recipe Name	Recipe Name	25 Bytes	string	Primary Key
Recipient Description	Recipe Description	100 Bytes	string	N/A

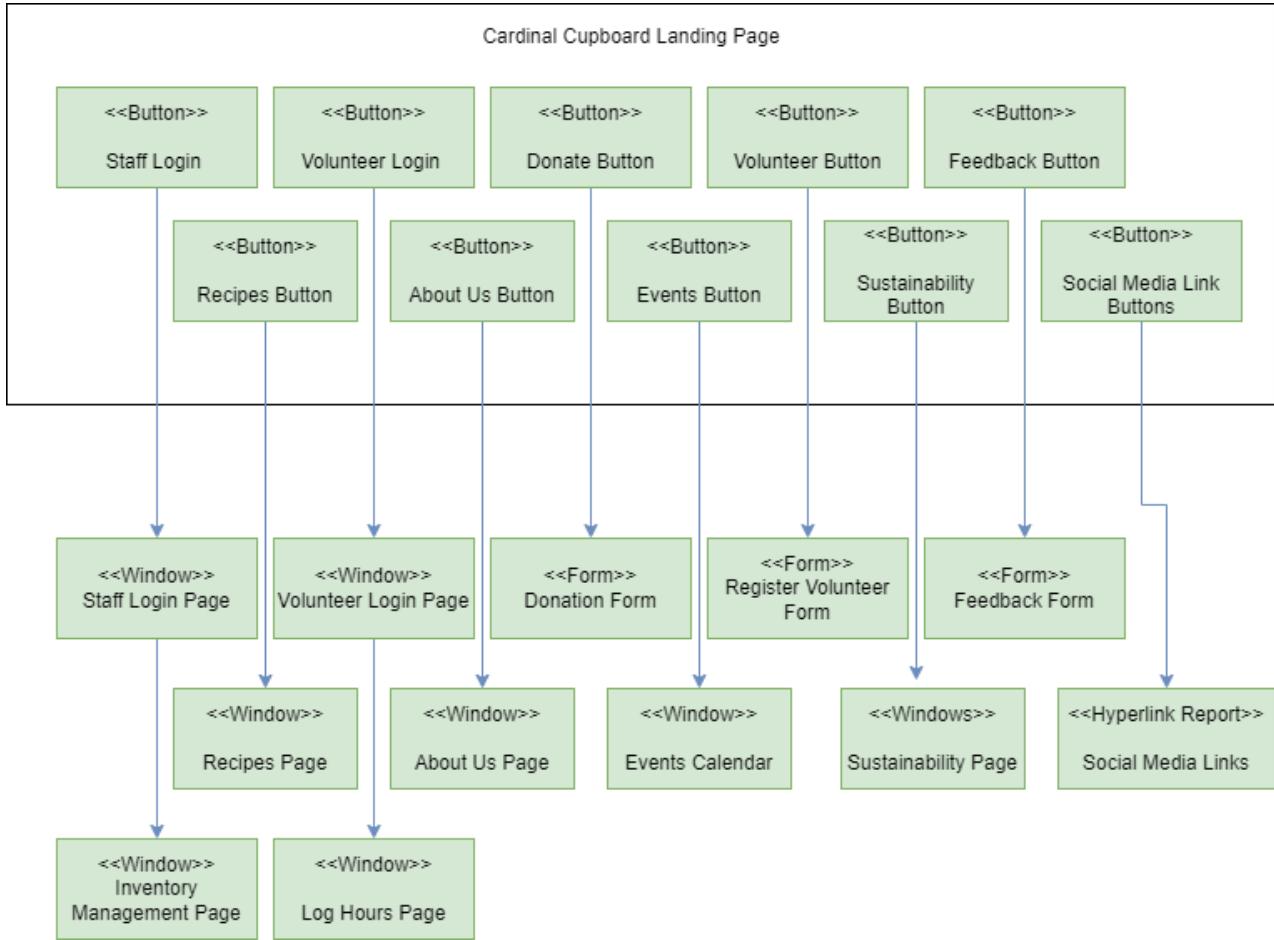
<b>Recipe Date</b>	Date recipe was submitted	<b>4 Bytes</b>	Date	N/A
--------------------	---------------------------	----------------	------	-----

**Inventory table:**

Attribute	Attribute Definition	Attribute Size	Attribute Type	Primary Key or Foreign Key?
<b>Inventory Name</b>	Item Name	<b>25 Bytes</b>	string	<b>Primary Key</b>
<b>Inventory Description</b>	Description of the inventory	<b>25 Bytes</b>	string	N/A
<b>Inventory Quantity</b>	Quantity of item	<b>8 Bytes</b>	Double	N/A

# Window Navigation Diagram

The navigation diagram outlines the basic structure of our system. This form shows the interfaces within our system and how they interact.



## Screen Layouts – Data Capture Forms

These layouts will be sections of the HTML prototypes for data capture or input. Each one of these screen layouts will be a visual representation of realizing the use cases. These forms will be used to take information into the system.

### Backup and Recover Data

This form implements the use case by creating a backup database with attributes of date, size and number of files and by allowing the user to recover data, as well as listing the last date, size and files recovered.

Backup Data

Date Last Backed up: 9/9/2004  
Backup Size: 50MB  
Files: 359

Recover Data

Date Last Recovered: 8/22/2002  
Recovery Size: 39MB  
Files: 299

### Add/Edit/Delete Recipes

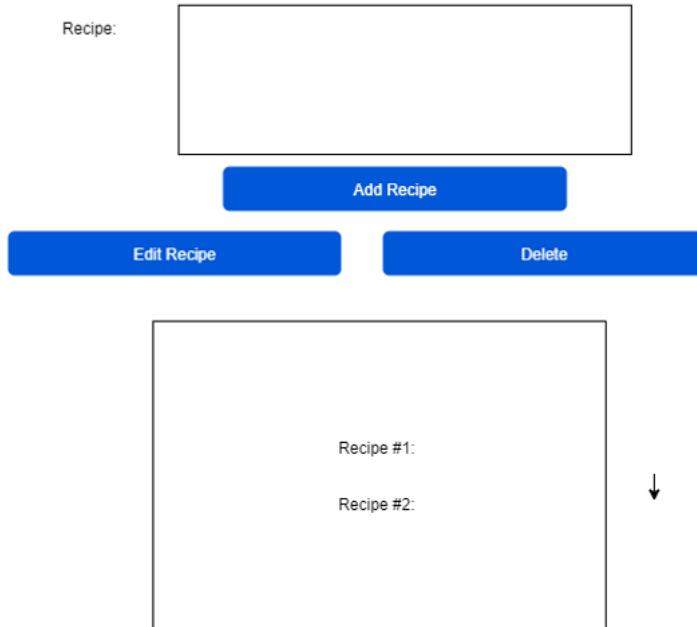
This form allows users to add a recipe by typing it into the text box and selecting the 'Add Recipe' button. To edit an existing recipe, select the recipe from the lower box and select the 'Edit Recipe' button. To delete a recipe, select the existing recipe from the lower box and select 'Delete'.

Recipe:

Add Recipe

Edit Recipe      Delete

Recipe #1:  
Recipe #2:



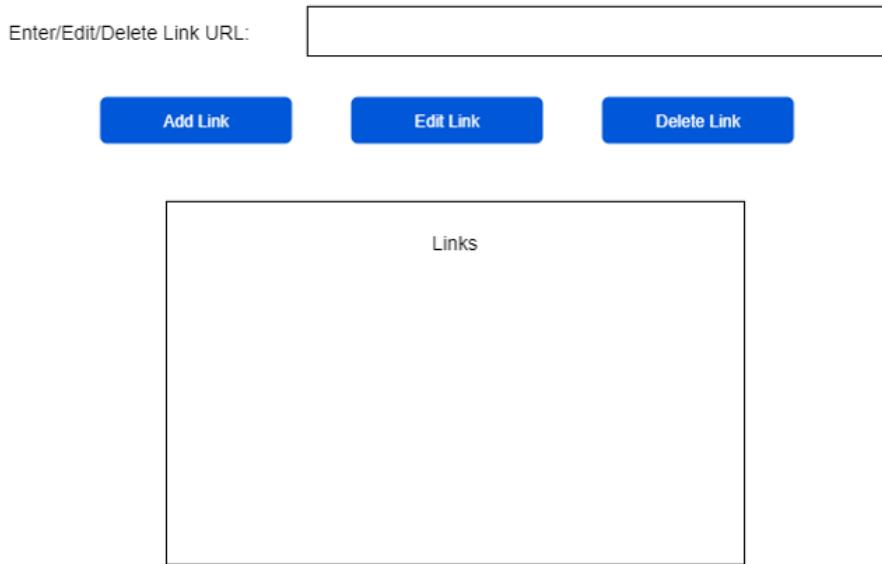
### Add/Edit/Delete Links

This form allows staff members to add, edit, and delete external links that would direct users to other helpful UofL resources. Staff members type the link into the text box and press one of the three buttons.

Enter/Edit/Delete Link URL:

Add Link      Edit Link      Delete Link

Links



### Add/Edit/Delete Items

This form allows the user to add, edit and delete items by using the attributes item name, item description and item quantity described in the use case.

Item Name	<input type="text"/>
Item Description	<input type="text"/>
Item Quantity	<input type="text"/>
<input type="button" value="Add Item"/>	
<input type="button" value="Edit Item"/> <input type="button" value="Delete"/>	

Items:	Quantity:	
1. Corn	1	
2. Sweet Peas	5	
3. Potatos	6	
4. Carrots	2	
5. Bacon	9	
6. Condensed Milk	5	
7. Bananas	6	
8. Corn Mix	4	
9. Baking Soda	2	
10. Ketchup	4	
11. Onions	5	
12. Cheerios	6	

### Add/Edit/Delete Inventory

This form allows the user to add, edit and delete inventory by using the attributes inventory name, inventory description and inventory quantity described in the use case.

Inventory Name	<input type="text"/>
Inventory Description	<input type="text"/>
Inventory Quantity	<input type="text"/>
<b>Add Inventory</b>	
<b>Edit Inventory</b>	
<b>Delete</b>	

Inventory:	Quantity:	
1. Corn	1	
2. Sweet Peas	5	
3. Potatos	6	
4. Carrots	2	
5. Bacon	9	
6. Condensed Milk	5	
7. Bananas	6	
8. Corn Mix	4	
9. Baking Soda	2	
10. Ketchup	4	
11. Onions	5	
12. Cheerios	6	

### Employee Spotlight

This section highlights an employee or volunteer that is to be featured according to the use case.

### Employee Spotlight



John Doe

*Volunteer*

## About John Doe

John Doe, a community member at UOFL, has provided exceptional services to The Cardinal Cupboard and the UOFL community alike. We would like to thank him for all his time and hard work and dedication to make the cardinal community great.

## Want To Be Featured?

Sign up to volunteer today! Attend and help out at various events, help stock the pantry, help with inventory, take in donations, and more.

### Add/Edit/Delete Newsletter

This form allows the user to join the newsletter, edit an existing email, or unsubscribe from the newsletter altogether.

## Join Our Newsletter

Join to receive updates about events, locations, volunteer opportunities, and more!

Email:

[Join](#)

[Edit Email](#)

[Unsubscribe](#)

## Feedback

This section allows users to leave their feedback by typing in the text box and/or clicking on one of five stars to leave a rating. After the user hits submit the use case is complete.

Leave a Review:

250 word limit

Rate Us:



Thank you for your feedback!

## Add/Edit/Delete Events

This section allows a staff member to add, edit, or delete the event calendar. The add button will prompt them with a new event template for a calendar. The edit button will allow them to change an existing calendar. The delete button will allow an existing calendar to be removed.

[<>](#)
October 2014
[<>](#)

Mo	Tu	We	Th	Fr	Sa	Su
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	<b>24</b>	25	26	27	28
29	30	31	1	2	3	4
5	6	7	8	9	10	11

October 24

**Donation Drive**
Info:

Time: 12-4PM
>Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Location: SAC

Add
Edit
Delete

## Social Media

This use case implements an integrated feed onto the website. Staff members can add a new integrated feed, edit the existing one, or delete it altogether should it not be needed.

## About Us

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.



## Edit Calendar event

This use case allows for an existing event to be changed. This is useful because some unexpected situations could occur such as weather or other hazards.

## Edit Calendar Event

Name:	<input type="text" value="Fall Food Drive"/>
Description:	<input type="text" value="mass donation event to support the Cardinal Cupboard"/>
Date/Time	<input type="text" value="02/24/2022 5:00PM"/>
	Form: mm/dd/yyyy x:xx PM/AM
Location:	<input type="text" value="Red Barn"/>

## Add Calendar Event

This use case allows a staff member to add a new event. This event will have a date, time, location, name, and description. This allows the UofL community to join in on the event.

### Add Calendar Event

Name:	<input type="text"/>
Description:	<input type="text"/>
Date/Time	<input type="text"/>
	Form: mm/dd/yyyy x:xx PM/AM
Location:	<input type="text"/>

## **Login**

This use case allows staff members and volunteers alike to login and make changes to the website. This is important because it allows for a layer of security on the website and grants privileges.

Email

Password

[Forgot Password](#)

## **Manage Users Volunteers**

This form allows the admin to delete users meeting the requirements of the use case.

## Manage Users

Staff

Volunteers

Name      Email Address

John Smith    jsmith01@louisville.edu

Jane Doe    jdoe01@louisville.edu

[Delete User\(s\)](#)

### Edit Staff Member

This form allows the admin to edit a staff member using the attributes name and email meeting the requirements of the use case.

#### Edit Staff Member

Name:

Email Address:

[Apply Changes](#)

### Add Staff Member

This form allows the admin to add a staff member using the attributes of name, email and ulink password meeting the requirements of the use case.

### Add Staff Member

Name:

Email Address:

ULink Password:

### Manage Users

This form allows an admin to delete users meeting the requirements of the use case.

#### Manage Users

Staff

Volunteers

	<u>Name</u>	<u>Email Address</u>
<input checked="" type="checkbox"/>	John Smith	jsmith01@louisville.edu
<input type="checkbox"/>	Jane Doe	jdoe01@louisville.edu

### Edit Volunteer

This form allows the user to edit the volunteer attributes of name, email, password and date of birth meeting the requirements of the use case.

## Edit Volunteer

Name:	John Smith
Email Address:	jsmith01@louisville.edu
Password:	P@ssw0rd!
Confirm Password:	P@ssw0rd!
Date of Birth	11/11/2001
Form: mm/dd/yyyy	

Apply Changes

## Register New Volunteer

This form allows the user to register a new volunteer using the attributes name, email, password, confirm password and date of birth, meeting the requirements of the use case.

### Register New Volunteer

Name:	<input type="text"/>
Email Address:	<input type="text"/>
Password:	<input type="text"/>
Confirm Password:	<input type="text"/>
Date of Birth	<input type="text"/>
Form: mm/dd/yyyy	

Register

## Your Monetary Donation

This form allows the user to see the monetary donation the user made with the option of cancelling the donation after making the donation meeting the requirements of the use case.

### Your Monetary Donation

Name:

Email Address:

Card Number:

Exp. Date:  CVV:   
Form: mm/yyyy

Amount:

Optionally, if you would like to tell us your reason for cancelling please enter it below

### New Monetary Donation

This allows the user to donate money using the attributes name, email, card number, expiration date, cvv and amount meeting the requirements of the use case.

#### New Monetary Donation

Name:

Email Address:

Card Number:

Exp. Date:  CVV:   
Form: mm/yyyy

Amount:

## Your Food Donation

This form allows the user to donate food using the attributes name, email, description, and donation date meeting the requirements of the use case.

Your Food Donation

Name:	<input type="text" value="John Smith"/>
Email Address:	<input type="text" value="jsmith01@louisville.edu"/>
Description:	<input type="text" value="Canned goods and whole grain bread"/>
Donation Date:	<input type="text" value="02/15/2022"/>
Form: mm/dd/yyyy	
<input type="button" value="Cancel Donation"/>	

## Edit Food Donation

This form allows the user to edit food donations with the attributes of name, email, description and donation date meeting the requirements of the use case

### Edit Food Donation

Name:

Email Address:

Description:

Donation Date:

Form: mm/dd/yyyy

### New Food Donation

This form allows the user to donate food using the attributes of name, email address, description and donation date meeting the requirements of the use case.

### New Food Donation

Name:

Email Address:

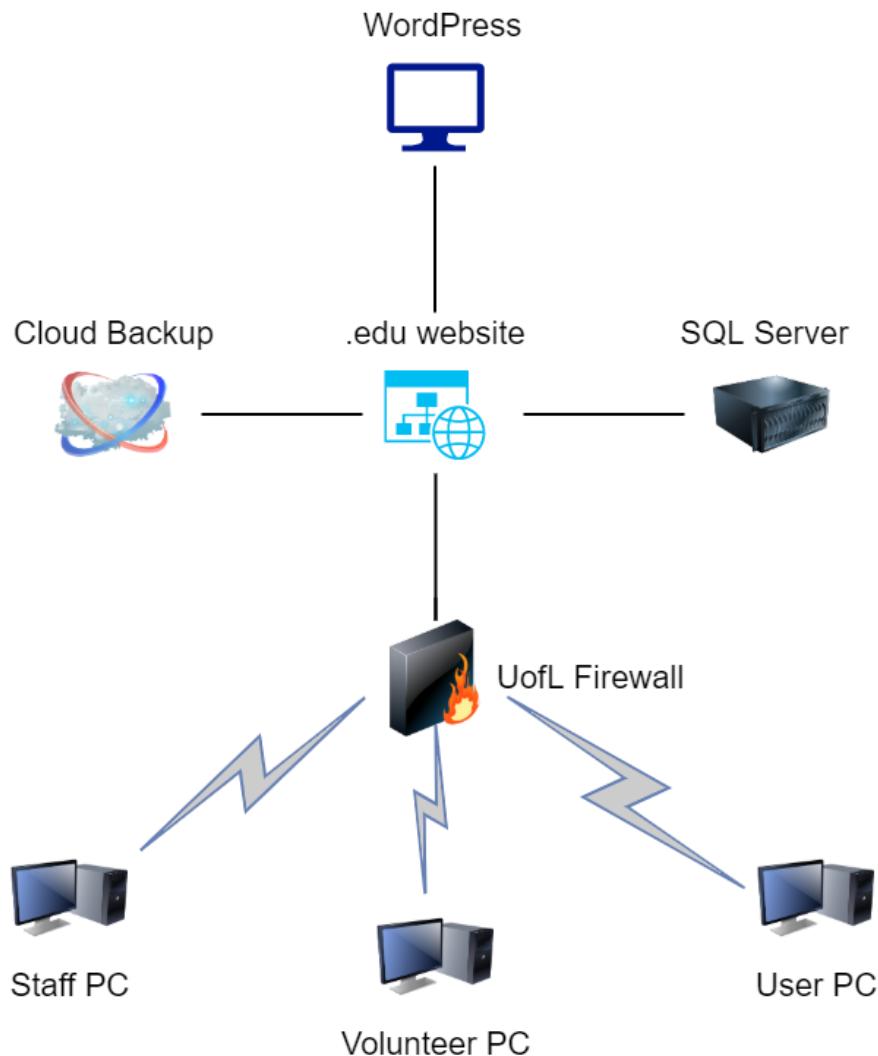
Description:

Donation Date:

Form: mm/dd/yyyy

## Physical Architecture Design

This deployment diagram represents the relationship between the hardware components used in the physical infrastructure of the information system. It shows how to access the website starting from the PC, firewall, then website which is connected to a cloud server and database server (SQL) as well as a content management system (WordPress). Since The Cardinal Cupboard is in the UofL community, it is safe to assume that those accessing the website are also a part of the cardinal community. Therefore, whenever a volunteer, staff member, or other users access the website they go through the universities firewall which protects the website from any harm or attacks. Furthermore, the website is linked to a SQL server which stores all data. It is also linked to a cloud server which holds the backup and recovery for the website should anything occur. Lastly, it is linked to WordPress which is where the website can be designed from.



## Design Procedures for Security Concerns and Non-functional Requirements

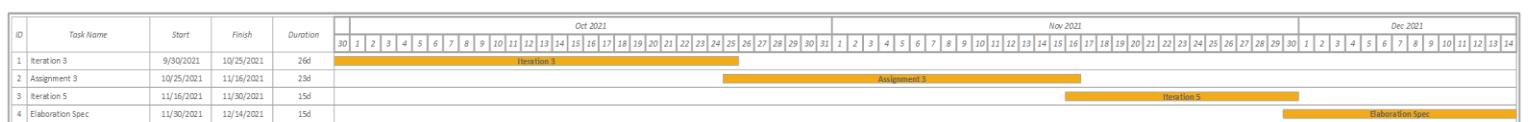
### Procedures needed to address non-functional requirements:

- Create the website using WordPress
  - The new website will be created using WordPress. WordPress is free to use and easy to learn so any staff member or volunteer should be able to use it to its full potential. A login will be required to use WordPress.
- Connect to the UofL firewall
  - Security is important for any organization, especially one dealing with money from donations. Connecting to the UofL firewall will help form a layer of security and help prevent attacks from happening.
- Use an SQL database
  - Having a place to store data and keep it safe and confidential is of upmost importance. Connecting to a SQL server database will provide a place to store information about staff, volunteers, donors, inventory, etc.
- Integrate Clover for payments
  - All donations will be handled by Clover through a virtual terminal. This will make keeping track of donation records easy and efficient.

### How each requirement may be addressed during detailed design:

Detailed design describes how all the main components of the overall design fit together. The first step to address in the design will be creating the website using WordPress. Next will be connecting the website to a SQL server and UofL firewall. Last, would be implementing the payment system Clover.

Gantt Chart



# Prototypes

## Cardinal CupBoard

### What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

#### New Food Donation

Name:

Email Address:

Description:

Donation Date:  Form: mm/dd/yyyy

### Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below.

Cardinal Cupboard  
(502) 852-6691

To subscribe for email updates.

### Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

### Events

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

[+ Google Calendar](#)

## Cardinal CupBoard

### What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

#### New Monetary Donation

Name:

Email Address:

Card Number:

Exp. Date:  Form: mm/yyyy

CVV:

Amount:

### Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below.

Cardinal Cupboard  
(502) 852-6691

To subscribe for email updates.

### Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

### Events

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

[+ Google Calendar](#)

# Cardinal CupBoard

## What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

### Register New Volunteer

Name:

Email Address:

Password:

Confirm Password:

Date of Birth:  Form: mm/dd/yyyy

### Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu) or at the phone numbers below

Cardinal Cupboard (502) 852-6691

to subscribe for email updates

### Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

### Events

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

[+ Google Calendar](#)

# Cardinal CupBoard

## What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

### Your Food Donation

Name:  John Smith

Email Address:  jsmith01@louisville.edu

Description:  Canned goods and whole grain bread

Donation Date:  02/15/2022  
Form: mm/dd/yyyy

### Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu) or at the phone numbers below

Cardinal Cupboard (502) 852-6691

to subscribe for email updates

### Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

### Events

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

[+ Google Calendar](#)


Cardinal CupBoard

### What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

### Contact Info

Available by email at  
[involved@louisville.edu](mailto:involved@louisville.edu)  
 or at the phone numbers below:

**Cardinal Cupboard**  
 (502) 852-6691

[Login](#) to subscribe for email updates.

### Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

### Events

Showing events after 12/12  
[Look for earlier events](#)

Showing events until 1/15 [Look for more](#)

[Google Calendar](#)

### Your Monetary Donation

Name:

Email Address:

Card Number:

Exp. Date:  CVV:   
 Form: mm/yyyy

Amount:

[Cancel Donation](#)

Optional, if you would like to tell us your reason for cancelling please enter it below

[Confirm](#)


Cardinal CupBoard

### What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

### Contact Info

Available by email at  
[involved@louisville.edu](mailto:involved@louisville.edu)  
 or at the phone numbers below:

**Cardinal Cupboard**  
 (502) 852-6691

[Login](#) to subscribe for email updates.

### Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

### Events

Showing events after 12/12  
[Look for earlier events](#)

Showing events until 1/15 [Look for more](#)

[Google Calendar](#)

### Edit Food Donation

Name:

Email Address:

Description:

Donation Date:   
 Form: mm/dd/yyyy

[Save Changes](#)

# Cardinal CupBoard

## What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

### Edit Volunteer

Name:

Email Address:

Password:

Confirm Password:

Date of Birth:   
Format: mm/dd/yyyy

### Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

to subscribe for email updates.

### Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

### Events

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

[+ Google Calendar](#)

# Cardinal CupBoard

## What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

## How do I get involved?

To get involved, please fill out this form. For more information, contact the program directors at [communityaction@louisville.edu](mailto:communityaction@louisville.edu).

Regular hours for the food bank are between Monday-Friday, 9am – 5pm.

Please Contact Us!

Date Last Backed up: 9/9/2004  
Backup Size: 50MB  
Files: 359

Date Last Recovered: 8/22/2002  
Recovery Size: 39MB  
Files: 299

### Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

to subscribe for email updates.

### Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

### Events

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

[+ Google Calendar](#)

# Cardinal CupBoard

## What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

Recipe:

[Add Recipe](#)

[Edit Recipe](#) [Delete](#)

↓

Recipe #1:

Recipe #2:

## Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

[Login](#) to subscribe for email updates.

## Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)

[TikTok](#) [SnapChat](#)

## Events

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

+ Google Calendar

# Cardinal CupBoard

## What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

Enter/Edit/Delete Link URL:

[Add Link](#) [Edit Link](#) [Delete Link](#)

Links

↓

## Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

[Login](#) to subscribe for email updates.

## Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)

[TikTok](#) [SnapChat](#)

## Events

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

+ Google Calendar

## Cardinal CupBoard

Item Name:

Item Description:

Item Quantity:

Items:      Quantity:

1. Corn	1
2. Sweet Peas	5
3. Potatos	6
4. Carrots	2
5. Bacon	9
6. Condensed Milk	5
7. Bananas	6
8. Corn Mix	4
9. Baking Soda	2
10. Ketchup	4
11. Onions	5
12. Cheerios	6

### Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

[Login](#) to subscribe for email updates.

### Social Media Links

### Events

Showing events after 12/12.  
[Look for earlier events](#)

Showing events until 1/15. [Look for more](#)

## Cardinal CupBoard

Inventory Name:

Inventory Description:

Inventory Quantity:

Inventory:      Quantity:

1. Corn	1
2. Sweet Peas	5
3. Potatos	6
4. Carrots	2
5. Bacon	9
6. Condensed Milk	5
7. Bananas	6
8. Corn Mix	4
9. Baking Soda	2
10. Ketchup	4
11. Onions	5
12. Cheerios	6

### Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

[Login](#) to subscribe for email updates.

### Social Media Links

### Events

Showing events after 12/12.  
[Look for earlier events](#)

Showing events until 1/15. [Look for more](#)

**Employee Spotlight**

**About John Doe**

John Doe, a community member at UOFL, has provided exceptional services to The Cardinal Cupboard and the UOFL community alike. We would like to thank him for all his time and hard work and dedication to make the cardinal community great.

John Doe  
Volunteer

**Want To Be Featured?**

Sign up to volunteer today! Attend and help out at various events, help stock the pantry, help with inventory, take in donations, and more.

**Contact Info**

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu) or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

**Login** to subscribe for email updates.

**Social Media Links**

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

**Events**

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

[+ Google Calendar](#)

**Cardinal CupBoard**

**What is Cardinal Cupboard?**

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

**Join Our Newsletter**

Join to receive updates about events, locations, volunteer opportunities, and more!

Email:

[Join](#) [Edit Email](#) [Unsubscribe](#)

**Contact Info**

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu) or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

**Login** to subscribe for email updates.

**Social Media Links**

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

**Events**

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

[+ Google Calendar](#)

# Cardinal CupBoard

## What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

**Leave a Review:**

250 word limit

**Rate Us:**

★
★
★
★
★

**Submit**

Thank you for your feedback!

## Contact Info

Available by email at  
[involved@louisville.edu](mailto:involved@louisville.edu)  
 or at the phone numbers below:

**Cardinal Cupboard**  
 (502) 852-6691

**Login** to subscribe for email updates.

## Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

## Events

Showing events after 12/12.  
[Look for earlier events](#)  
 Showing events until 1/15. [Look for more](#)

[+ Google Calendar](#)

# Cardinal CupBoard

## What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

## How do I get involved?

To get involved, please fill out this form. For more information, contact the program directors at [communityaction@louisville.edu](mailto:communityaction@louisville.edu).

Regular hours for the food bank are between Monday-Friday, 9am – 5pm.

Please Contact Us!

**Cardinal Cupboard Events**

<	October 2014	>				
Mo	Tu	We	Th	Fr	Sa	Su
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4
5	6	7	8	9	10	11

**October 24**

**Donation Drive**

Info: Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Time: 12-4PM  
 Location: SAC

**Add** **Edit** **Delete**

## Contact Info

Available by email at  
[involved@louisville.edu](mailto:involved@louisville.edu)  
 or at the phone numbers below:

**Cardinal Cupboard**  
 (502) 852-6691

**Login** to subscribe for email updates.

## Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

## Events

Showing events after 12/12.  
[Look for earlier events](#)  
 Showing events until 1/15. [Look for more](#)

[+ Google Calendar](#)

# Cardinal CupBoard

## What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

### Edit Calendar Event

Name: Fall Food Drive

Description: mass donation event to support the Cardinal Cupboard

Date/Time: 02/24/2022 5:00PM  
Form: mm/dd/yyyy xox PM/AM

Location: Red Barn

[Submit Event](#)

### Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below:

Cardinal Cupboard  
(502) 852-6691

[Login](#) to subscribe for email updates.

### Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

### Events

Showing events after 12/12. [Look for earlier events](#)

Showing events until 1/15. [Look for more](#)

[+ Google Calendar](#)

# Cardinal CupBoard

## What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

### Add Calendar Event

Name:

Description:

Date/Time:   
Form: mm/dd/yyyy x:xx PM/AM

Location:

### Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

to subscribe for email updates.

### Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

### Events

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

[Google Calendar](#)

# Cardinal CupBoard

Email

Password

[Forgot Password](#)

### Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

to subscribe for email updates.

### Social Media Links

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

### Events

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

[Google Calendar](#)

## Cardinal CupBoard

### What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

#### Manage Users

Staff	Volunteers						
<table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 20%;">Name</th> <th style="width: 80%;">Email Address</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/> John Smith</td> <td>jsmith01@louisville.edu</td> </tr> <tr> <td><input type="checkbox"/> Jane Doe</td> <td>jdoe01@louisville.edu</td> </tr> </tbody> </table>	Name	Email Address	<input checked="" type="checkbox"/> John Smith	jsmith01@louisville.edu	<input type="checkbox"/> Jane Doe	jdoe01@louisville.edu	
Name	Email Address						
<input checked="" type="checkbox"/> John Smith	jsmith01@louisville.edu						
<input type="checkbox"/> Jane Doe	jdoe01@louisville.edu						
<input type="button" value="Delete User(s)"/>							

#### Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

[Login](#) to subscribe for email updates.

#### Social Media Links

[FaceBook](#)
[Instagram](#)
[Twitter](#)

[TikTok](#)
[SnapChat](#)

#### Events

Showing events after 12/12. [Look for earlier events](#)

Showing events until 1/15. [Look for more](#)

+ Google Calendar

## Cardinal CupBoard

### What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

### How do I get involved?

To get involved, please fill out this form. For more information, contact the program directors at [communityaction@louisville.edu](mailto:communityaction@louisville.edu).

Regular hours for the food bank are between Monday-Friday, 9am – 5pm.

Please Contact Us!

#### Edit Staff Member

Name:	<input type="text"/>
Email Address:	<input type="text"/>

#### Contact Info

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

[Login](#) to subscribe for email updates.

#### Social Media Links

[FaceBook](#)
[Instagram](#)
[Twitter](#)

[TikTok](#)
[SnapChat](#)

#### Events

Showing events after 12/12. [Look for earlier events](#)

Showing events until 1/15. [Look for more](#)

+ Google Calendar

# Cardinal CupBoard

## What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

## How do I get involved?

To get involved, please fill out this form. For more information, contact the program directors at [communityaction@louisville.edu](mailto:communityaction@louisville.edu).

Regular hours for the food bank are between Monday-Friday, 9am – 5pm.

Please Contact Us!

**Add Staff Member**

Name:

Email Address:

ULink Password:

[Create/Connect Account](#)

**Contact Info**

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

[Login](#) to subscribe for email updates.

**Social Media Links**

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

**Events**

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

[+ Google Calendar](#)

# Cardinal CupBoard

## What is Cardinal Cupboard?

Cardinal Cupboard is a program designed to provide families with a temporary food supply when needed and to assist them in becoming more self-reliant. Volunteers are needed to sort canned goods, assemble food packages, help clients through the food pantry, and help the food bank staff prepare care packages.

**Manage Users**

Staff	Volunteers
<input type="checkbox"/> John Smith <a href="#">jsmith01@louisville.edu</a>	
<input type="checkbox"/> Jane Doe <a href="#">jdoe01@louisville.edu</a>	

[Delete User\(s\)](#)

**Contact Info**

Available by email at [involved@louisville.edu](mailto:involved@louisville.edu)  
or at the phone numbers below:

**Cardinal Cupboard**  
(502) 852-6691

[Login](#) to subscribe for email updates.

**Social Media Links**

[FaceBook](#) [Instagram](#) [Twitter](#)  
[TikTok](#) [SnapChat](#)

**Events**

Showing events after 12/12. [Look for earlier events](#)  
Showing events until 1/15. [Look for more](#)

[+ Google Calendar](#)