FULL USE CASE DESCRIPTIONS

Use Case Specification: ADD GENERAL VOLUNTEER

1. Use-Case Name – Add General Volunteer

1.1 Brief Description

This use case explains how a user will fill out a general volunteer form

2. Flow of Events

2.1 Basic Flow

1. User selects general volunteer form.
2. System loads general volunteer form.
3. User types in first name.
4. User types in last name.
5. User types in city.
6. User types in state.
7. User types in zip code.
8. User types in phone number.
9. User types in date of birth.
10. User types in email.
11. User uploads copy of driver's license.
12. User uploads copy of immunization records.
13. User uploads verification of TB Test.
14. User selects general volunteer type – (medical student, undergraduate student, or general volunteer).
15. User selects interest area.
16. User selects submit general volunteer form.
    7.1. System submits completed general volunteer form.
    7.2. System asks for all required fields to be filled out.
    7.3. System asks for valid email address.
17. System stores all input to database to await approval of SOS Staff.
18. System sends confirmation that general volunteer form has been received.
19. Use case ends.

2.2 Alternative Flows

None

3. Special Requirements

None.

4. Pre-conditions

None.

5. Post-conditions

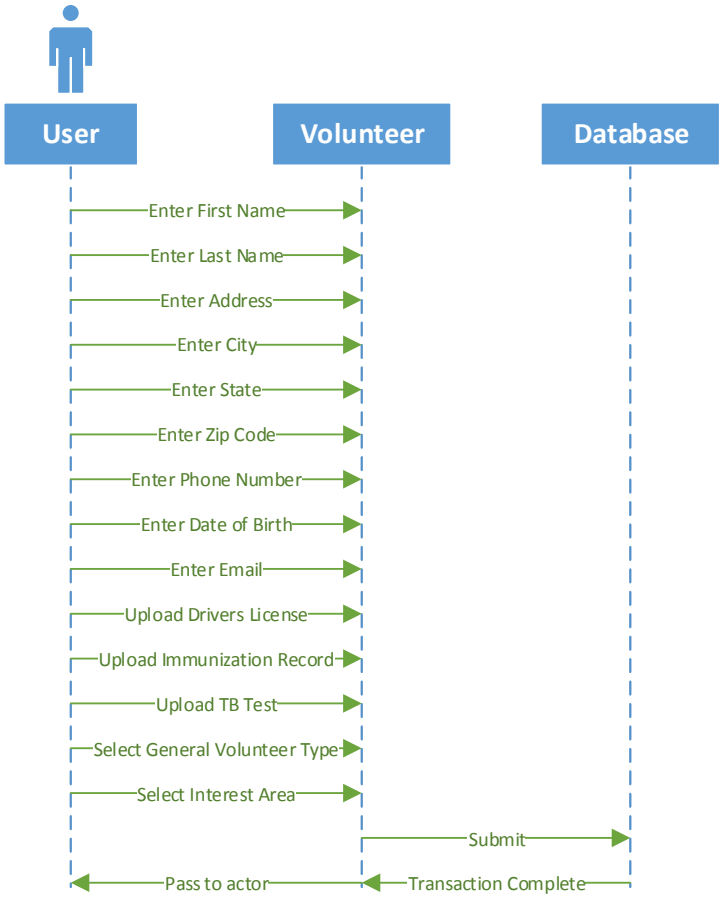
5.1 Approval Pending

System will be pending for approval by staff for general volunteer form.

6. Extension Points

None.

General Volunteer – Sequence Diagram



Use Case Specification: ADD PHYSICIAN VOLUNTEER

1. Use-Case Name – Add Physician Volunteer

1.1 Brief Description

This use case explains how a user will fill out a physician volunteer form

2. Flow of Events

2.1 Basic Flow

1. User selects physician volunteer form.
2. System loads physician volunteer form.
3. User types in first name.
4. User types in last name.
5. User types in city.
6. User types in state.
7. User types in zip code.
8. User types in phone number.
9. User types in date of birth.
10. User types in email.
11. User uploads copy of driver's license.
12. User uploads copy of immunization records.
13. User uploads verification of TB Test.
14. User enters medical license number.
15. User enters practice/medical specialty.
16. User enters current employer/practice.
17. User enters medical centers with privileges to practice.
18. User enters states licensed to practice medicine in.
19. User selects submit physician volunteer form.
    7.1. System submits completed physician volunteer form.
    7.2. System asks for all required fields to be filled out.
    7.3. System asks for valid email address.
20. System stores information to database to await approval of SOS Staff.
21. System sends confirmation that physician volunteer form has been received.
22. Use case ends.

2.2 Alternative Flows

None.

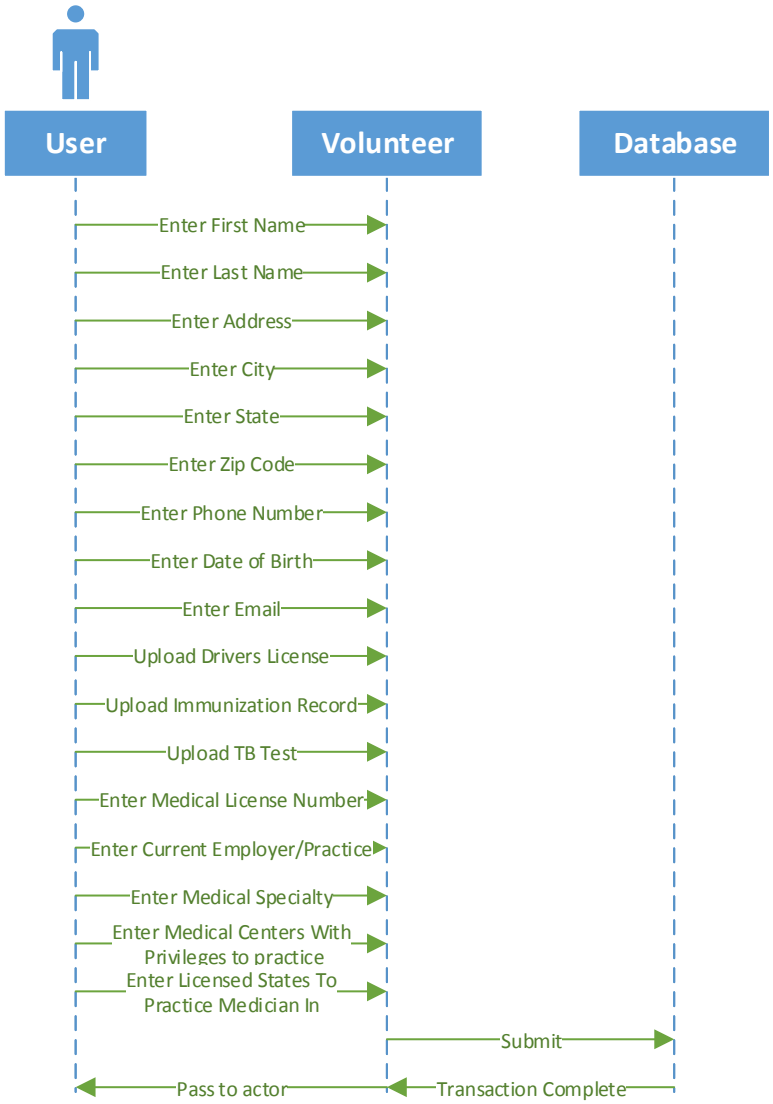
3. Special Requirements

Volunteer must have a medical license number, practice specialty, employer, licensed in one or more states, and privileges to practice in one or more medical centers.

4. Pre-conditions

None.

Physician Volunteer – Sequence Diagram



Use Case Specification: ADD MEDICAL INTERPRETER VOLUNTEER

1. Use-Case Name – Add Medical Interpreter Volunteer

1.1 Brief Description

This use case explains how a user will fill out a medical interpreter volunteer form

2. Flow of Events

2.1 Basic Flow

1. User selects medical interpreter volunteer form.
2. System loads medical interpreter volunteer form.
3. User types in first name.
4. User types in last name.
5. User types in city.
6. User types in state.
7. User types in zip code.
8. User types in phone number.
9. User types in date of birth.
10. User types in email.
11. User uploads copy of driver's license.
12. User uploads copy of immunization records.
13. User uploads verification of TB Test
14. User selects fluent languages
15. User uploads interpreter credentials
16. User selects submit medical interpreter volunteer form.
    7.1. System submits completed medical interpreter volunteer form.
    7.2. System asks for all required fields to be filled out.
    7.3. System asks for valid email address.
17. System stores information to database to await approval of SOS Staff.
18. System sends confirmation that medical interpreter volunteer form has been received.
19. Use case ends.

2.2 Alternative Flows

None.

3. Special Requirements

Volunteer must be fluent in foreign language and have credentials.

4. Pre-conditions

None.

5. Post-conditions

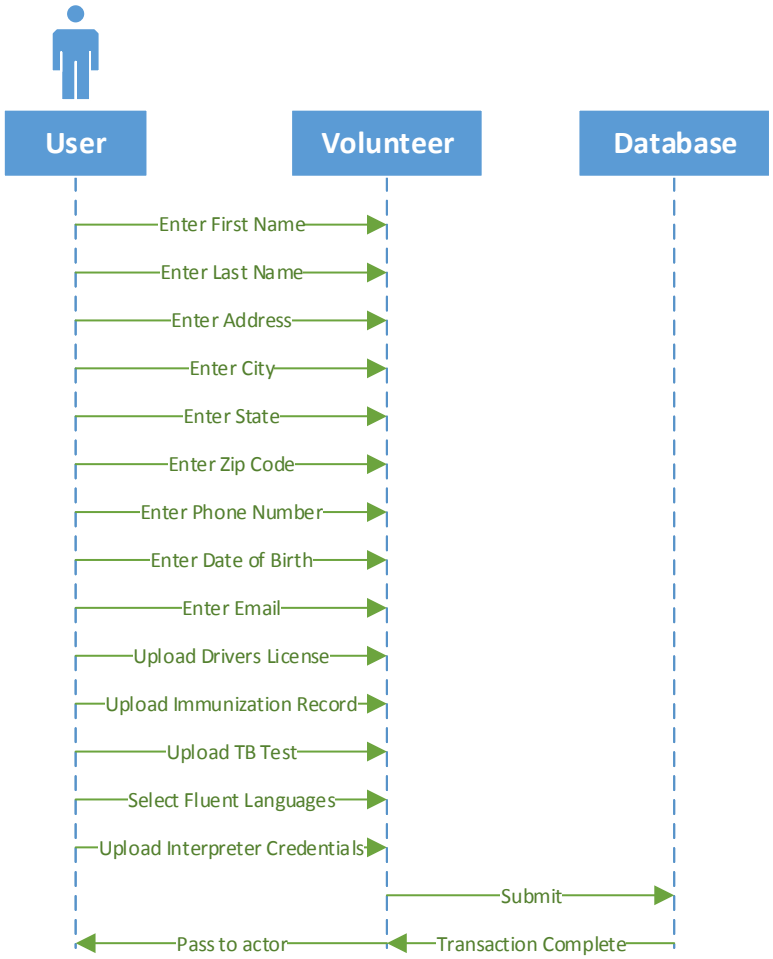
5.1 Approval Pending

System will be pending for approval by staff for medical interpreter volunteer form.

6. Extension Points

None.

Medical Interpreter Volunteer – Sequence Diagram



Use Case Specification: ADD NON-PHYSICIAN CLINICAL VOLUNTEER

1. Use-Case Name – Add Non-Physician Clinical Volunteer

1.1 Brief Description

This use case explains how a user will fill out a non-physician clinical volunteer form

2. Flow of Events

2.1 Basic Flow

1. User selects non-physician clinical volunteer form.
2. System loads non-physician clinical volunteer form.
3. User types in first name.
4. User types in last name.
5. User types in city.
6. User types in state.
7. User types in zip code.
8. User types in phone number.
9. User types in date of birth.
10. User types in email.
11. User uploads copy of driver's license.
12. User uploads copy of immunization records.
13. User uploads verification of TB Test.
14. User selects professional title
15. User states current place of employment
16. User selects submit non-physician clinical volunteer form.
    7.1. System submits completed non-physician clinical volunteer form.
    7.2. System asks for all required fields to be filled out.
    7.3. System asks for valid email address.
17. System stores information to database to await approval of SOS Staff.
18. System sends confirmation that non-physician clinical volunteer form has been received.
19. Use case ends.

2.2 Alternative Flows

None.

3. Special Requirements

Volunteer must have a professional title and current place of employment.

4. Pre-conditions

None.

5. Post-conditions

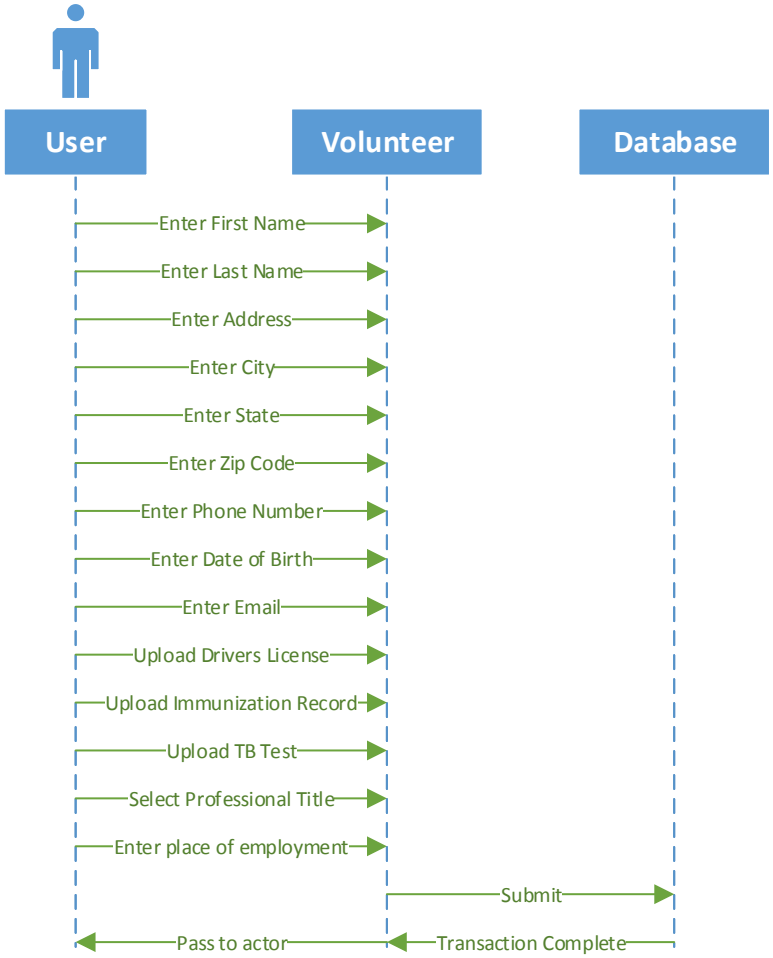
5.1 Approval Pending

System will be pending for approval by staff for non-physician clinical volunteer form.

6. Extension Points

None.

Non-Physician Clinical Volunteer – Sequence Diagram



Use Case Specification: MODIFY VOLUNTEER

1. Use-Case Name – Modify Volunteer

1.1 Brief Description

This use case explains how staff member will modify a volunteer.

2. Flow of Events

2.1 Basic Flow

1. Staff selects Modify Volunteer.
2. System loads page with all of specified volunteer's submitted information and credentials (Contact Info, Credentials, volunteer type – physician, non-physician clinical, medical interpreter, general).
3. Staff makes desired changes to contact info, credentials, and/or volunteer type.
4. Staff selects saves changes.
5. System updates volunteer with specified modifications.
6. Use case ends.

2.2 Alternative Flows

3. Special Requirements

3.1 Modify Privileges

Staff must have delete privilege.

4. Pre-conditions

4.1 Volunteer must exist

A volunteer record must have been created in order for it to be modified.

5. Post-conditions

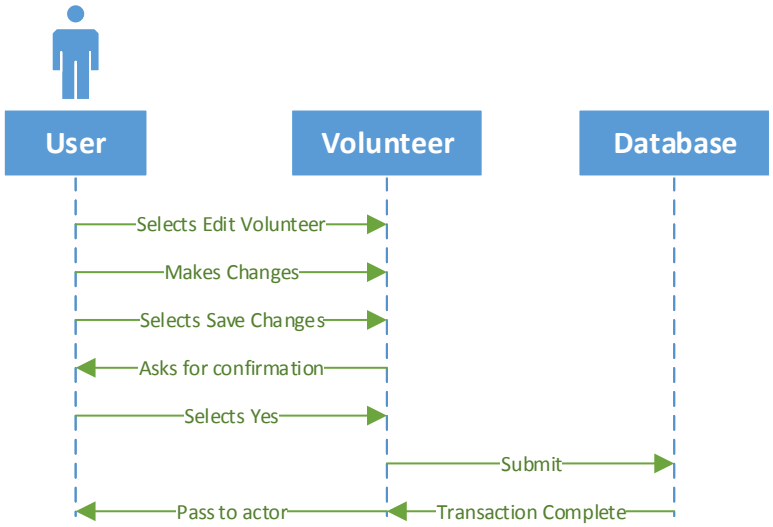
5.1 Volunteer Modified

Volunteer information has been changed/updated.

6. Extension Points

None.

Modify Volunteer – Sequence Diagram



Use Case Specification: DELETE VOLUNTEER

1. Use-Case Name – Delete Volunteer

1.1 Brief Description

This use case explains how staff member will delete a volunteer.

2. Flow of Events

2.1 Basic Flow

1. Staff selects Delete Volunteer
2. System prompts message “Are you sure you want to delete volunteer? All volunteer information will be permanently deleted.”
3. User selects Yes or No
   3.1. User selects Yes and system permanently deletes volunteer and all associated information from the database.
   3.2. User selects No and the system cancels the request to delete volunteer
4. UseCase Ends

2.2 Alternative Flows

3. Special Requirements

3.1 Delete Privileges

Staff must have delete privilege.

4. Pre-conditions

4.1 Volunteer must exist

A volunteer record must have been created in order for it to be deleted.

5. Post-conditions

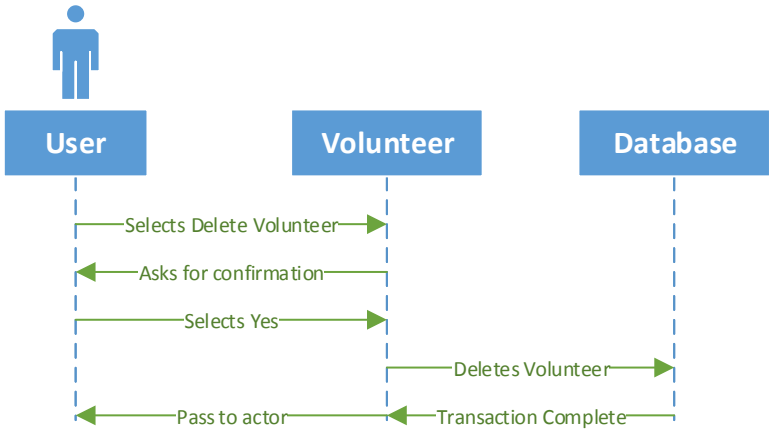
5.1 Volunteer Deleted

Volunteer has been removed from database

6. Extension Points

None.

Delete Volunteer – Sequence Diagram



Use Case Specification: SEND CONTACT FORM

1. Use-Case Name – Send Contact Form

1.1 Brief Description

This use case explains how a user will fill out a contact form.

2. Flow of Events

2.1 Basic Flow

1. User selects contact form on website.
2. System loads contact form.
3. User types in first name.
4. User types in last name.
5. User types in city.
6. User types in state.
7. User types in zip code.
8. User types in phone number.
9. User types in email.
10. User types in comment.
11. User submits contact form.
12. System sends contact form to staff email.
13. Use case ends.

2.2 Alternative Flows

3. Special Requirements

None.

4. Pre-conditions

None.

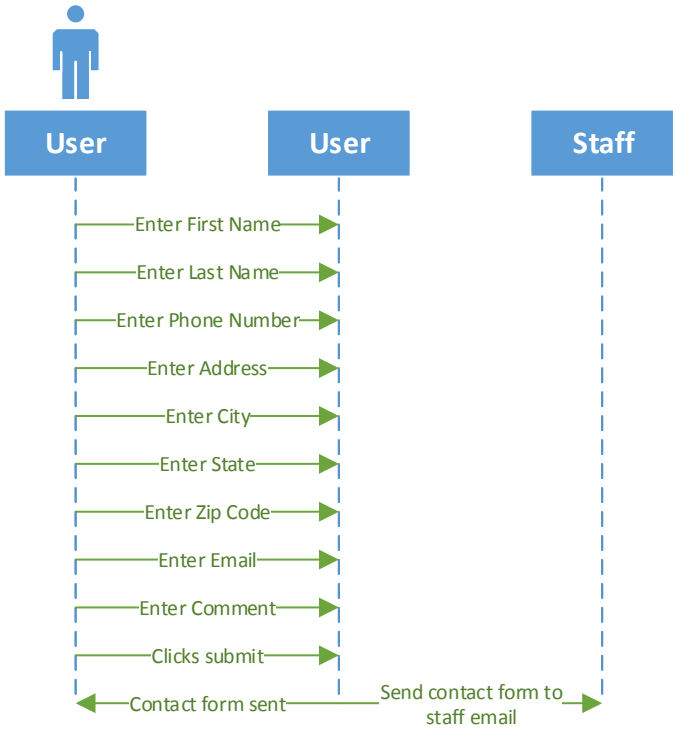
5. Post-conditions

None

6. Extension Points

None.

Send Contact Form – Sequence Diagram



Use Case Specification: Add Donor

1. Use-Case Name: Add Donor

1.1 Brief Description

This use case will allow the actor or user to donate to Surgery on Sunday (SOS) Louisville. Once the user has accessed the SOS Louisville webpage they will click the “DONATE” link. This link will then take the user to a secured PayPal site. Once in PayPal the user will enter in the amount they wish to donate. Next, the user will enter their information. Lastly, once they user has finished entering in their information they will click “Donate Now” to process their donation.

2. Flow of Events

2.1 Basic Flow

1. User clicks the “DONATE” link on homepage
2. User enters in a USD amount
3. Users selects a method to donate
4. Users enters in the required information based on the method they selected
5. User has the option to add special instructions to the seller
6. User clicks the “Donate Now” button

2.2 Alternative Flows

2.2.1 Donate with PayPal

1. User clicks the “DONATE” link on homepage
2. User enters in a USD amount
3. User clicks the “Donate with PayPal” button
4. User enters their email for PayPal
5. User enters their password for PayPal
6. User clicks the “Log In” button
7. User has the option to enter special instructions to the seller
8. User clicks the “Donate Now” button

2.2.2 Onetime PayPal Payment

1. User clicks the “DONATE” link on homepage
2. User enters in a USD amount
3. User does not check the “Make this a monthly donation” box
4. User clicks the “Donate with PayPal” button
5. User enters their email for PayPal
6. User enters their password for PayPal
7. User clicks the “Log In” button
8. User has the option to add special instructions to the seller
9. User clicks the “Donate Now” button

2.2.3 Monthly PayPal Payment

1. User clicks the “DONATE” link on homepage
2. User enters in a USD amount
3. User checks the “Make this a monthly donation” box
4. User clicks the “Donate with PayPal” button
5. User enters their email for PayPal
6. User enters their password for PayPal
7. User clicks the “Log In” button
8. User has the option to enter special instructions to the seller
9. User clicks the “Donate Now” button

2.2.4 Donate with Debit or Credit Card

1. User clicks the “DONATE” link on homepage
2. User enters in a USD amount
3. User does not check the “Make this a monthly donation” box
3. User clicks the “Donate with Debit or Credit Card” button
4. User selects their Country
5. User enters their card Number
6. User enters their card’s Expiration Date
7. User enters their card’s Security Code
8. User enters their First Name
9. User enters their Last Name
10. User enters their Street Address
11. User has the option to add a second line to their Street Address
12. User enters their City
13. User enters their State
14. User enters their ZIP code
15. User selects which type of phone number they are going to provide
16. User enters in their Phone Number
17. User enters their Email
18. User has the option to check a “Save this information for next time.” box
19. User has the option to enter special instructions to the seller
20. User clicks the “Donate Now” button

3. Special Requirements

3.1 USD Amount

The entered USD amount must be greater than or equal to $1.00.

3.2 Valid Debit or Credit Card

The entered Debit or Debit Card must not be expired.

3.3 Valid Email

The entered email must be an active account.

4. Pre-conditions

None

5. Post-conditions

5.1 Added Donor

The donor’s information has been automatically added into the Donor Table in the database.

5.2 Successful Donation

The donation has been made to SOS Louisville.

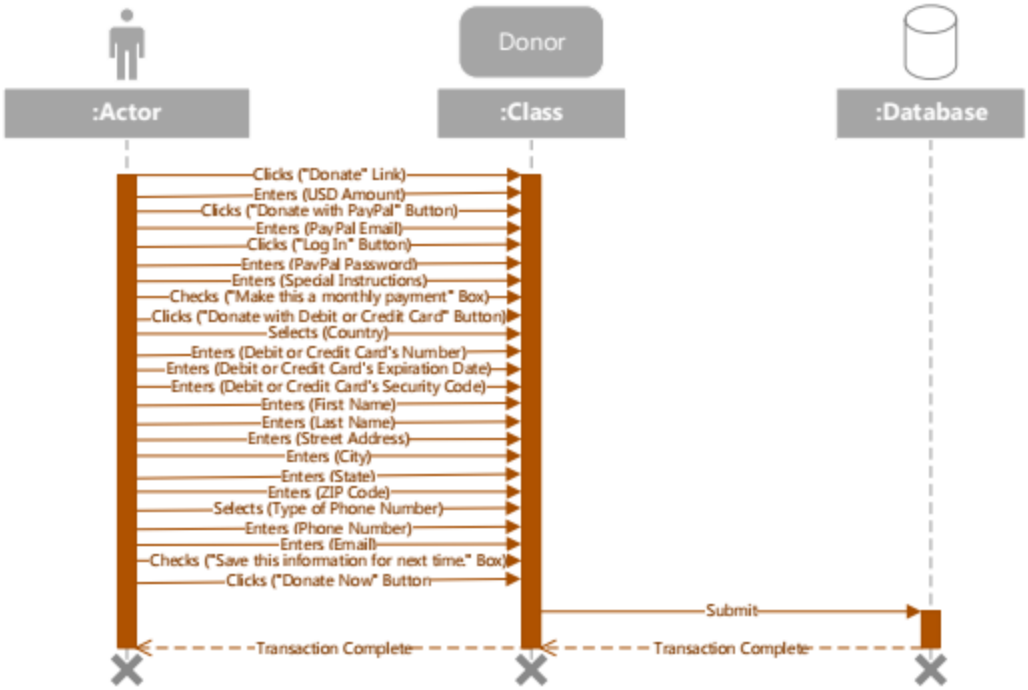
5.3 Confirmation

After a donation has been made a confirmation email will be sent to the user.

6. Extension Points

None

Add Donor – Sequence Diagram



Use Case Specification: Modify Donor

1. Use-Case Name: Modify Donor

1.1 Brief Description

This use case will allow the actor or user to modify a previous donor of Surgery on Sunday (SOS) Louisville. Once the user has accessed the Azure webpage they will click the “Login” link. This link will then prompt the user to enter in their account information. Once the user has successfully logged in they will have access to the database storing the donor information. Next, the user will find the donor they wish to modify. Lastly, once the user finds the donor they will be able to modify the selected donor’s information.

2. Flow of Events

2.1 Basic Flow

1. User will click login once they have access the Azure webpage
2. User enters user name for Azure webpage
3. User enters password for Azure webpage
4. User click the “Login” Button
5. User enters the Email of the Donor they wish to modify
6. User click the “Generate Report” button
7. User selects the Donor they wish to Modify from the report
8. User clicks the “Edit” button
9. User can modify the Donor’s First Name
10. User can modify the Donor’s Last Name
11. User can modify the Donor’s Email
12. User click the “Save” button
13. User selects “Logout”

2.2 Alternative Flows

2.2.1 Search with First Name

1. User will click login once they have access the Azure webpage
2. User enters user name for Azure webpage
3. User enters password for Azure webpage
4. User click the “Login” Button
5. User enters the First Name of the Donor they wish to modify
6. User click the “Generate Report” button
7. User selects the Donor they wish to Modify from the report
8. User clicks the “Edit” button
9. User can modify the Donor’s First Name
10. User can modify the Donor’s Last Name
11. User can modify the Donor’s Email
12. User click the “Save” button
13. User selects “Logout”

2.2.2 Search with Last Name

1. User will click login once they have access the Azure webpage
2. User enters user name for Azure webpage
3. User enters password for Azure webpage
4. User click the “Login” Button
5. User enters the Last Name of the Donor they wish to modify
6. User click the “Generate Report” button

7. User selects the Donor they wish to Modify from the report
8. User clicks the “Edit” button
9. User can modify the Donor’s First Name
10. User can modify the Donor’s Last Name
11. User can modify the Donor’s Email
12. User click the “Save” button
13. User selects “Logout”

3. Special Requirements

3.1 Simultaneous Login

User cannot login if already logged in.

4. Pre-conditions

4.1 Existing Donor

The donor being modified must have donated in the past.

5. Post-conditions

5.1 Modified Donor

The donor’s information has been modified.

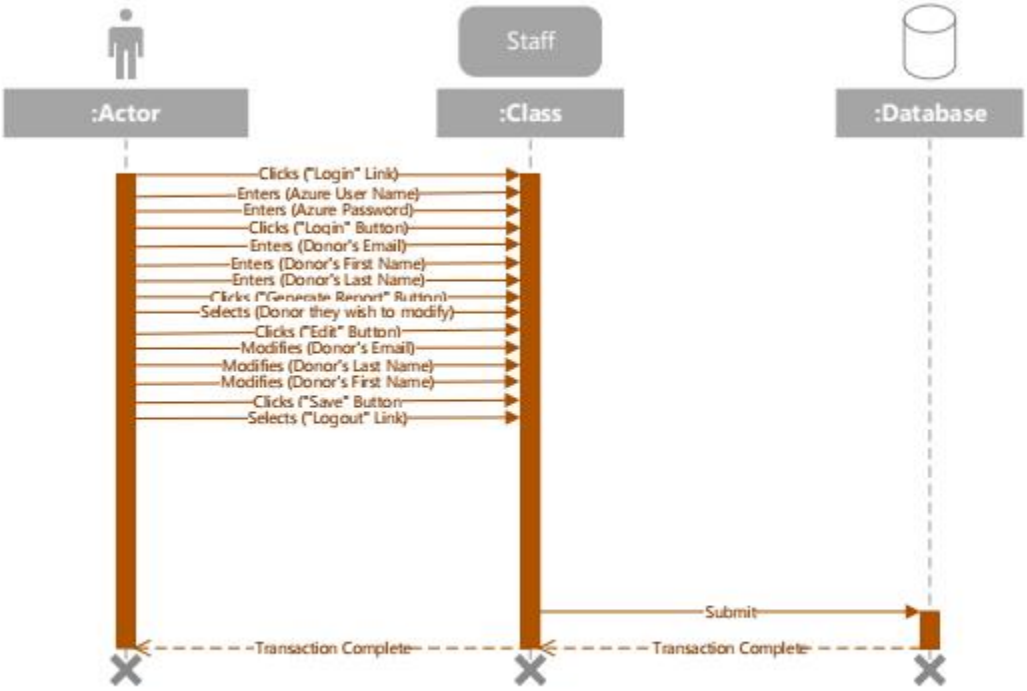
5.2 Confirmation

After the donor’s information has been modified a confirmation email will be sent to the user.

6 Extension Points

None

Modify Donor – Sequence Diagram



Use Case Specification: Delete Donor

1. Use-Case Name: Delete Donor

1.1 Brief Description

This use case will allow the actor or user to delete a previous donor of Surgery on Sunday (SOS) Louisville. Once the user has accessed the Azure webpage they will click the “Login” link. This link will then prompt the user to enter in their account information. Once the user has successfully logged in they will have access to the database storing the donor information. Next, the user will find the donor they wish to delete. Lastly, once the user finds the donor they will be able to delete the donor.

2. Flow of Events

2.1 Basic Flow

1. User will click login once they have access the Azure webpage
2. User enters user name for Azure webpage
3. User enters password for Azure webpage
4. User click the “Login” Button
5. User enters the Email of the Donor they wish to delete
6. User click the “Generate Report” button
7. User selects the Donor they wish to delete from the report
8. User clicks the “Delete” button
9. User click the “Save” button
10. User selects “Logout”

2.2 Alternative Flows

2.2.1 First Name

1. User will click login once they have access the Azure webpage
2. User enters user name for Azure webpage
3. User enters password for Azure webpage
4. User click the “Login” Button
5. User enters the Last Name of the Donor they wish to delete
6. User click the “Generate Report” button
7. User selects the Donor they wish to delete from the report
8. User clicks the “Delete” button
9. User click the “Save” button
10. User selects “Logout”

2.2.2 Last Name

1. User will click login once they have access the Azure webpage
2. User enters user name for Azure webpage
3. User enters password for Azure webpage
4. User click the “Login” Button
5. User enters the Last Name of the Donor they wish to delete
6. User click the “Generate Report” button
7. User selects the Donor they wish to delete from the report
8. User clicks the “Delete” button
9. User click the “Save” button
10. User selects “Logout”

3. Special Requirements

3.1 Simultaneous Login

User cannot login if already logged in.

4. Pre-conditions

4.1 Existing Donor

The donor being deleted must have donated in the past.

5. Post-conditions

5.1 Deleted Donor

The donor has been deleted.

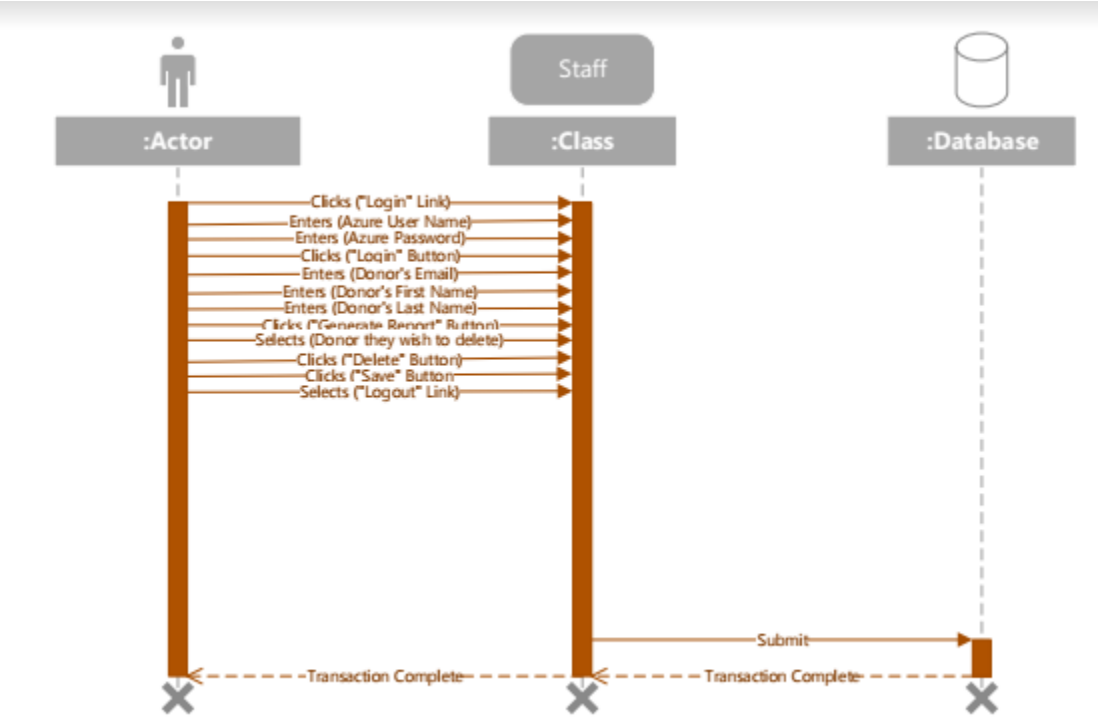
5.2 Confirmation

After the donor has been deleted a confirmation email will be sent to the user.

6. Extension Points

None

Delete Donor – Sequence Diagram



Use Case Specification: View Donation

1. Use-Case Name: View Donation

1.1 Brief Description

This use case will allow the actor or user to view a previous donation to Surgery on Sunday (SOS) Louisville. Once the user has accessed the Azure webpage they will click the “Login” link. This link will then prompt the user to enter in their account information. Once the user has successfully logged in they will have access to the database storing the donor information. Next, the user will find the donation they wish to view. Lastly, the user will view the donation.

2. Flow of Events

2.1 Basic Flow

1. User will click login once they have access the Azure webpage
2. User enters user name for Azure webpage
3. User enters password for Azure webpage
4. User click the “Login” Button
5. User enters the Email of the Donor who donated the donation they wish to view
6. User click the “Generate Report” button
7. User selects the Donor who donated the donation they wish to view
8. User views the donation
9. User selects “Logout”

2.2 Alternative Flows

2.2.1 First Name

1. User will click login once they have access the Azure webpage
2. User enters user name for Azure webpage
3. User enters password for Azure webpage
4. User click the “Login” Button
5. User enters the First Name of the Donor who donated the donation they wish to view
6. User click the “Generate Report” button
7. User selects the Donor who donated the donation they wish to view
8. User views the donation
9. User selects “Logout”

2.2.2 Email

1. User will click login once they have access the Azure webpage
2. User enters user name for Azure webpage
3. User enters password for Azure webpage
4. User click the “Login” Button
5. User enters the Last Name of the Donor who donated the donation they wish to view
6. User click the “Generate Report” button
7. User selects the Donor who donated the donation they wish to view
8. User views the donation
9. User selects “Logout”

3. Special Requirements

3.1 Simultaneous Login

User cannot login if already logged in.

4. Pre-conditions

4.1 Existing Donor

The donation being viewed must have been donated in the past.

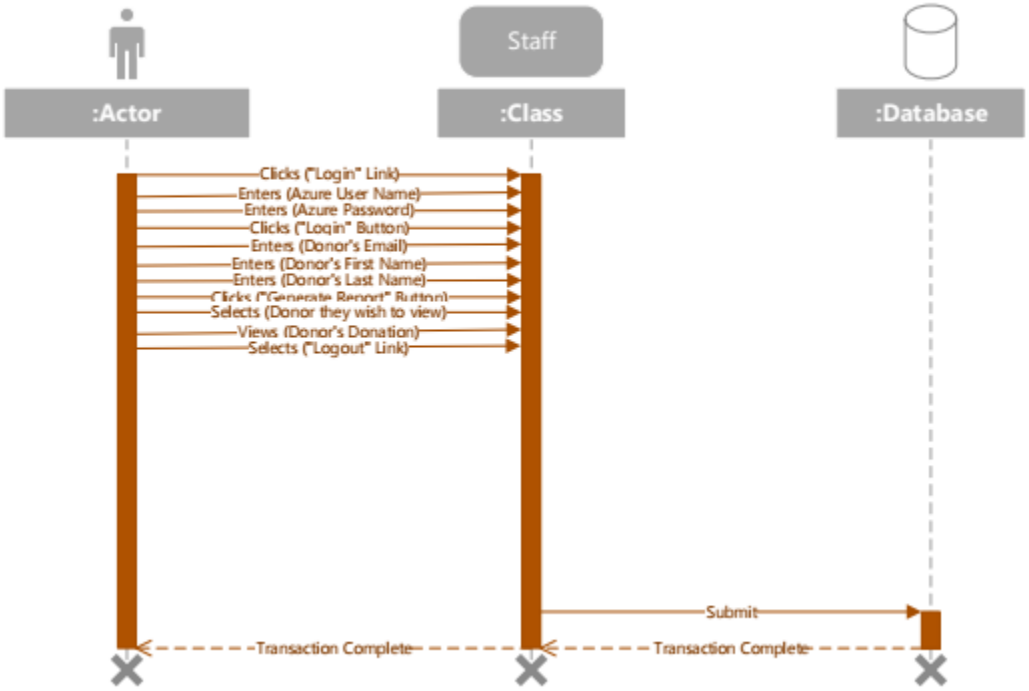
5. Post-conditions

None

6. Extension Points

None

View Donation – Sequence Diagram



Use Case Specification: Add Event

1. Use-Case Name: Add Event

1.1 Brief Description

This use case describes how a staff would add an event on the website.

2. Flow of Events

2.1 Basic Flow

1. Staff goes to www.soslouisville.org
2. Staff clicks on “Event” tab
3. Staff is on the event list and calendar page
4. Staff clicks on “Add event” button
5. Staff enters event time
6. Staff enters event date
7. Staff enters event location
8. Staff enters event name
9. Staff enters event descriptions
10. Staff clicks “Submit” to calendar and event list when finished
11. The system prints the event onto the calendar and event list
12. The system stores the information into database
13. Use case ends

2.2 Alternative Flows

1. Staff goes to www.soslouisville.org
2. Staff clicks “Event” tab
3. Staff is on the event list and calendar page
4. Staff clicks on “Add event” button
5. Staff enters event time
6. Staff enters event date
7. Staff enters event location
8. Staff enters event name
9. Staff enters event descriptions
10. Staff clicks “Cancel”
11. The system does not print the event onto the calendar and event list
12. The system does not store the information into database
13. Use case ends

3. Special Requirements

Event must be in the future.

4. Pre-conditions

The staff needs to have the permission to add event.

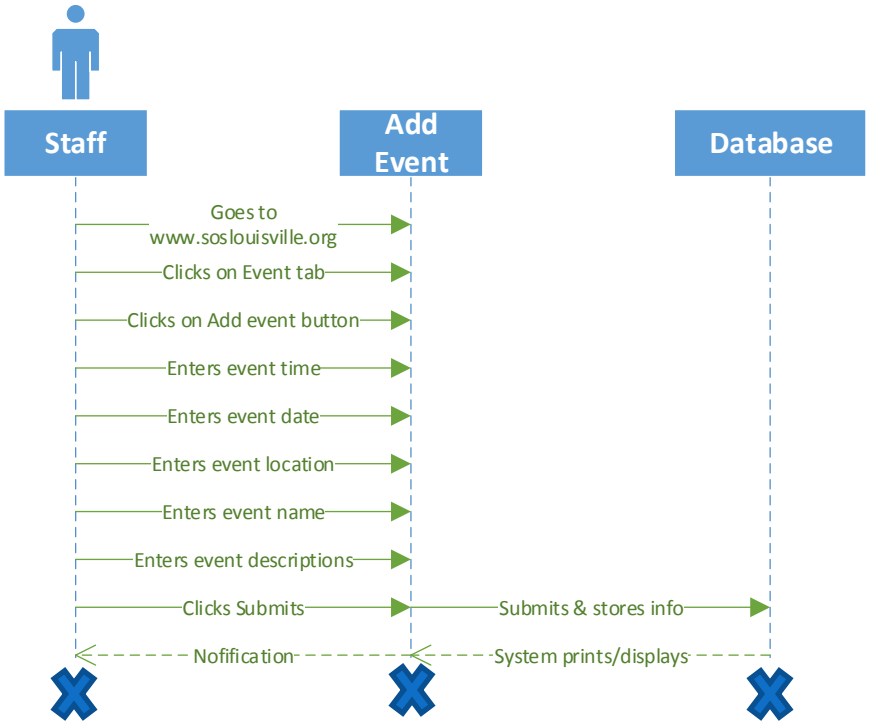
5. Post-conditions

The system posts the event onto the calendar and event list.
Issue notification that an event has been added.

6. Extension Points

None

Add Event – Sequence Diagram



Use Case Specification: Modify Event

1. Use-Case Name: Modify Event

1.1 Brief Description

This use case describes how a staff would modify an event on the website.

2. Flow of Events

2.1 Basic Flow

1. Staff goes to www.soslouisville.org
2. Staff clicks “Event” tab
3. Staff is on the event list and calendar page
4. Staff clicks on “Edit event” button
5. Staff edits event time
6. Staff edits event date
7. Staff edits event location
8. Staff edits event name
9. Staff edits event descriptions
10. Staff clicks “Submit” to calendar and event list when finished
11. The system prints the updated event onto the calendar and event list
12. The system stores the information into database
13. Use cases ends

2.2 Alternative Flows

1. Staff goes to www.soslouisville.org
2. Staff clicks “Event” tab
3. Staff is on the event list and calendar page
4. Staff clicks on “Edit event” button
5. Staff edits event time
6. Staff edits event date
7. Staff edits event location
8. Staff edits event name
9. Staff edits event descriptions
10. Staff clicks “Cancel”
11. The system does not print the updated event onto the calendar and event list
12. The system does not store the information into database
13. Use case ends

3. Special Requirements

Must have an existing event.

4. Pre-conditions

The staff needs to have the permission to edit event.

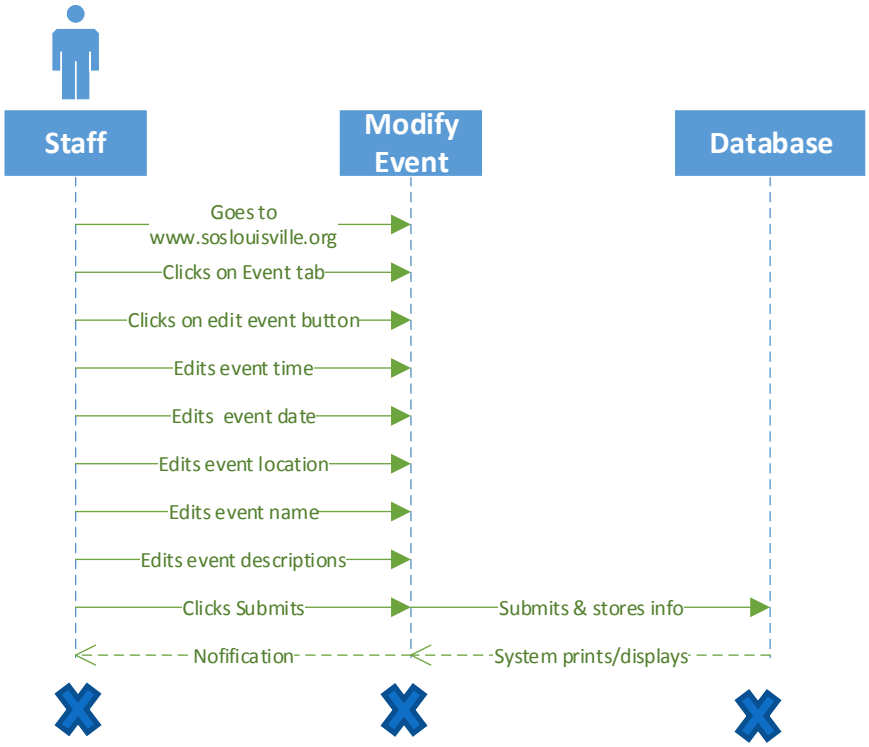
5. Post-conditions

The system posts the updated even onto the calendar and event list.
Issue notification that an event has been edited.

6. Extension Points

None

Modify Event – Sequence Diagram



Use Case Specification: Delete Event

1. Use-Case Name: Delete Event

1.1 Brief Description

This use case describes how a staff would delete an event on the website.

2. Flow of Events

2.1 Basic Flow

1. Staff goes to www.soslouisville.org
2. Staff clicks “Event” tab
3. Staff is on the event list and calendar page
4. Staff clicks on “Delete event” button
5. Staff clicks “Submit” to confirm
6. The system deletes the event off the calendar and event list
7. The system removes the information into database
8. Use cases ends

2.2 Alternative Flows

1. Staff goes to www.soslouisville.org
2. Staff clicks “Event” tab
3. Staff is on the event list and calendar page
4. Staff clicks on “Delete event” button
5. Staff clicks “Cancel” instead of “Submit”
6. The system does not delete the event
7. The system does not remove the information into database
8. Use case ends

3. Special Requirements

Must have an existing event.

4. Pre-conditions

The staff needs to have the permission to delete event.

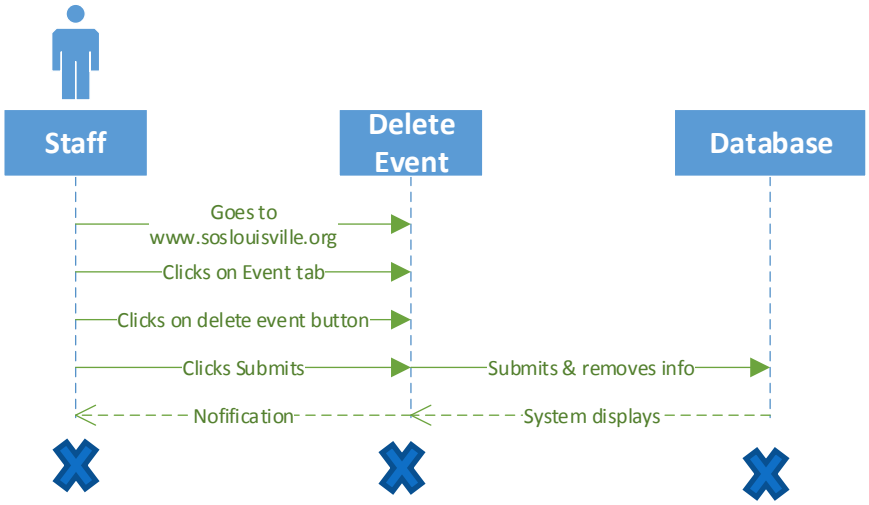
5. Post-conditions

The system removes the event off the calendar and event list.
Issue notification that an event has been deleted.

6. Extension Points

None

Delete Event – Sequence Diagram



Use Case Specification: Register for Event

1. Use-Case Name: Register for Event

1.1 Brief Description

This use case describes how a user would register for event.

2. Flow of Events

2.1 Basic Flow

1. User goes to www.soslouisville.org
2. User clicks on “Event” tab
3. User is on the event list and calendar page
4. User clicks “Register for event”
5. User select the event
6. User fills out a form
7. User enters name
8. User enters type of volunteer
9. User enters email
10. User enters phone number
11. User clicks “Submit” to confirm
12. The system displays a confirmation pop up that an event has been registered
13. The system stores the information into database
14. Use cases ends

2.2 Alternative Flows

1. User goes to www.soslouisville.org
2. User clicks on “Event” tab
3. User is on the event list and calendar page
4. User clicks “Register for event”
5. User select the event
6. User fills out a form
7. User enters name
8. User enters type of volunteer
9. User enters email
10. User enters phone number
11. User clicks “Cancel”
12. The system does not display a confirmation pop up that an event has been registered
13. The system does not store the information into database
14. Use cases ends

3. Special Requirements

Must have an existing event.

4. Pre-conditions

The user needs to be on the event page.

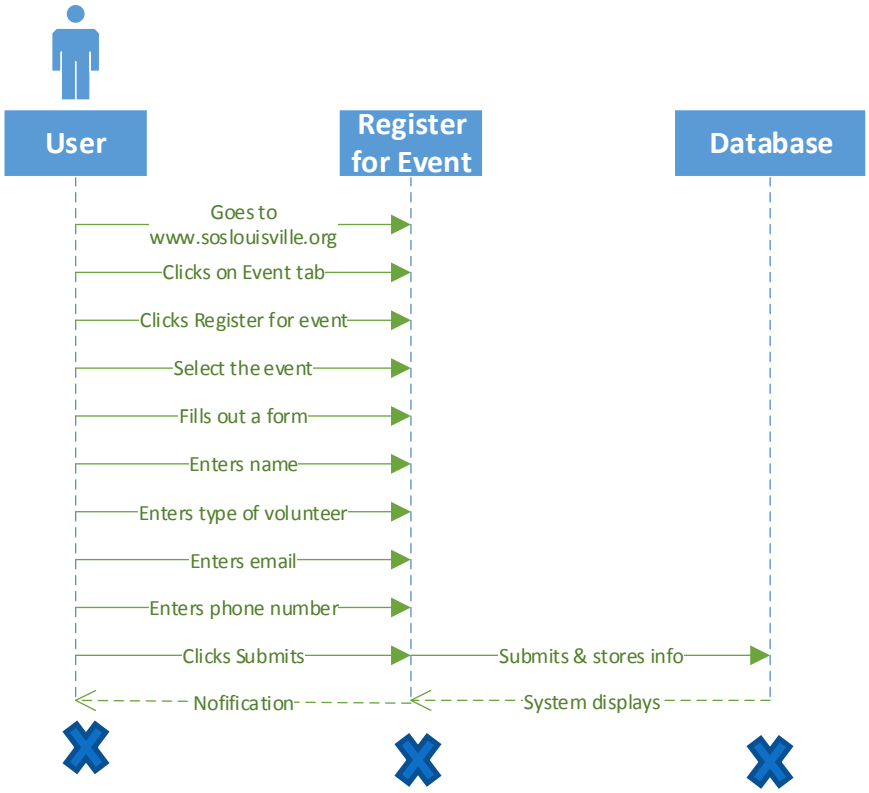
5. Post-conditions

The system displays user have registered for event.
Issue notification for the volunteers.

6. Extension Points

None

Register for Event – Sequence Diagram



Use Case Specification: View Event

1. Use-Case Name: View Event

1.1 Brief Description

This use case describes how a staff and user would view an event on the website.

2. Flow of Events

2.1 Basic Flow

1. Staff/user goes to www.soslouisville.org
2. Staff/user clicks on “Event” tab
3. Staff/user is on the event list and calendar page
4. Staff/user clicks “Event list” for more details
5. The system displays the event list in detail
6. Use cases ends

2.2 Alternative Flows

1. Staff/user goes to www.soslouisville.org
2. Staff/user clicks on “Event” tab
3. Staff/user is on the event list and calendar page
4. Staff/user does not click on the event list
5. The system does not display the event list in detail
6. Use case ends

3. Special Requirements

Must have an existing event.

4. Pre-conditions

The staff/user needs to be on the event page.

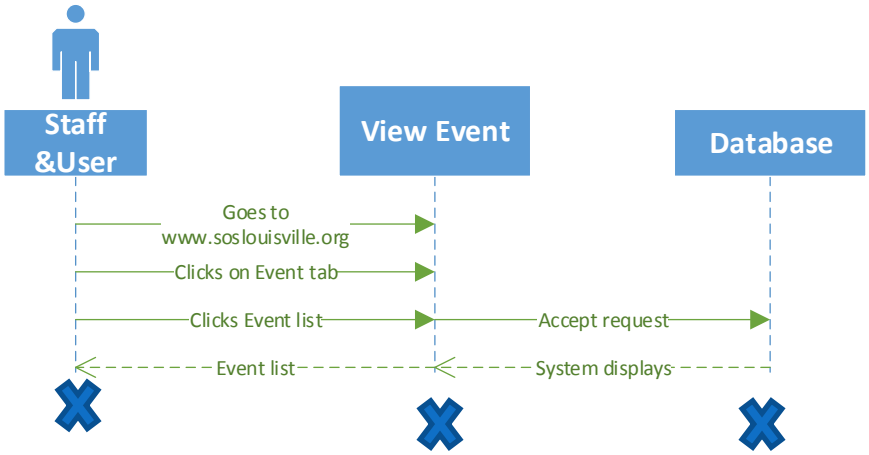
5. Post-conditions

The system displays the event list in detail.
Setup notification for an event.

6. Extension Points

None

View Event – Sequence Diagram



Use Case Specification: View Event Calendar

1. Use-Case Name: View Event Calendar

1.1 Brief Description

This use case describes how a staff and user would view an event calendar on the website.

2. Flow of Events

2.1 Basic Flow

1. Staff/user goes to www.soslouisville.org
2. Staff/user clicks on “Event” tab
3. Staff/user is on the event list and calendar page
4. Staff/user clicks “Event calendar” for more details
5. The system displays the event calendar in detail
6. Use cases ends

2.2 Alternative Flows

1. Staff/user goes to www.soslouisville.org
2. Staff/user clicks on “Event” tab
3. Staff/user is on the event list and calendar page
4. Staff/user does not click on the event calendar
5. The system does not display the event calendar in detail
6. Use case ends

3. Special Requirements

Must have an existing event.

4. Pre-conditions

The staff/user needs to be on the event page.

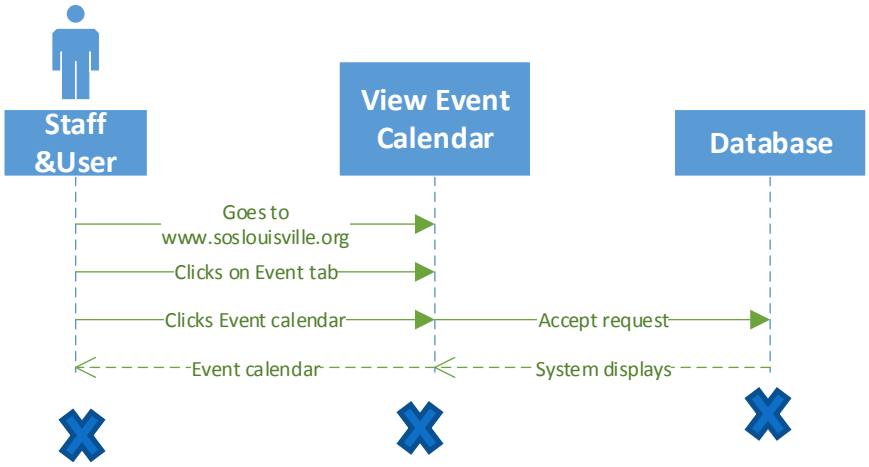
5. Post-conditions

The system displays the event calendar in detail.
Setup notification for an event.

6. Extension Points

None

View Event Calendar – Sequence Diagram



Use Case Specification: Schedule Volunteer for Event

1. Use-Case Name: Schedule Volunteer for Event

1.1 Brief Description

This use case describes how a staff would schedule volunteer for event.

2. Flow of Events

2.1 Basic Flow

1. Staff goes to www.soslouisville.org
2. Staff clicks on “Event” tab
3. Staff is on the event list and calendar page
4. Staff clicks “Schedule event”
5. Staff enters volunteer’s email
6. Staff clicks “Submit” to confirm
7. The system displays the number of volunteers for an event
8. The system stores the information into database
9. Use cases ends

2.2 Alternative Flows

1. Staff goes to www.soslouisville.org
2. Staff clicks on “Event” tab
3. Staff is on the event list and calendar page
4. Staff clicks “Schedule event”
5. Staff enters volunteer’s email
6. Staff clicks “Cancel”
7. The system does not display the number of volunteers for an event
8. The system does not store the information into database
9. Use case ends

3. Special Requirements

Must have an existing event.

4. Pre-conditions

The staff needs to be on the event page and have permission to schedule.

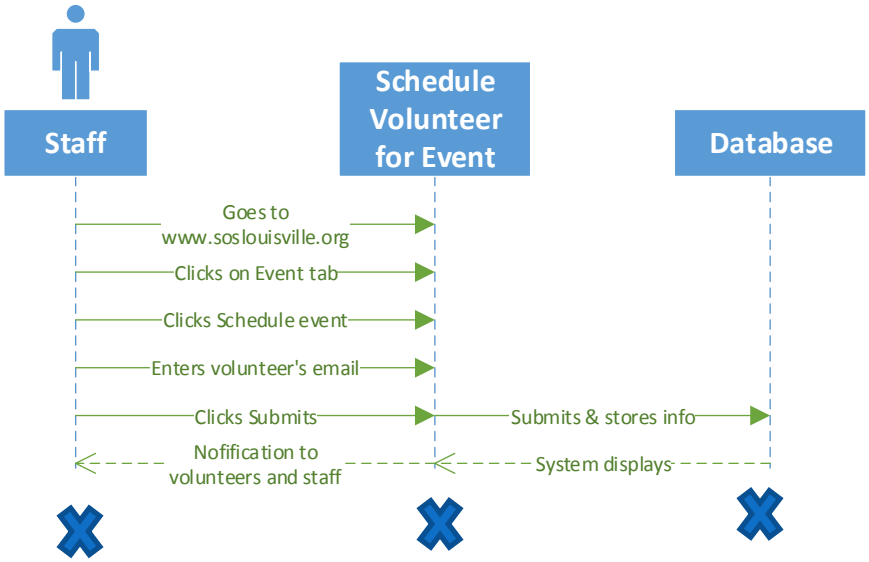
5. Post-conditions

The system displays the number of volunteers for an event.
Issue notification for the volunteers and they will have to fill out information

6. Extension Points

None

Schedule Volunteer for Event – Sequence Diagram



Use Case Specification: View Social Media

1. Use-Case Name: View Social Media

1.1 Brief Description

This use case will describe how a visitor would view Surgery on Sunday Louisville's social media platforms.

2. Flow of Events

2.1 Basic Flow

1. User goes to www.soslouisville.org
2. User clicks on one of three social media icon links: "Facebook, "Twitter", or "Google Plus"
3. Users user is redirected to the selected social media page
4. Users enters log in credentials for selected social media platform
5. User is directed to SOS Louisville's social media page
6. User views SOS Louisville's social media content
7. Use case ends

2.2 Alternative Flows

2.2.1 View Facebook Page

1. User goes to www.soslouisville.org
2. User clicks on "Facebook" icon on homepage
3. User is redirected to Facebook.
4. User enters email
5. User enters password
6. User selects "Enter"
7. User is directed to SOS Louisville's Facebook page
8. User views SOS Louisville's content
9. Use case ends

2.2.2 View Twitter Page

1. User goes to www.soslouisville.org
2. User clicks on "Twitter" icon on homepage
3. User is redirected to Twitter.
4. User enters username
5. User enters password
6. User selects "Enter"
7. User is directed to SOS Louisville's Twitter page
8. User views SOS Louisville's content
9. Use case ends

2.2.3 View Google Plus Page

1. User goes to www.soslouisville.org
2. User clicks on "Google Plus" icon on homepage
3. User is redirected to Google Plus.
4. User enters email
5. User enters password
6. User selects "Enter"
7. User is directed to SOS Louisville's page
8. User views SOS Louisville's content
9. Use case ends

3. Special Requirements

None

4. Pre-conditions

4.1 Facebook Account

The user must have a Facebook account to post or share SOS Louisville's content.

4.2 Twitter Account

The user must have a Twitter account to post or share SOS Louisville's content.

4.3 Google Plus Account

The user must have a Google Plus account to post or share SOS Louisville's content.

5. Post-conditions

5.1 View Social Media Pages

User is directed to SOS Louisville's social media pages.

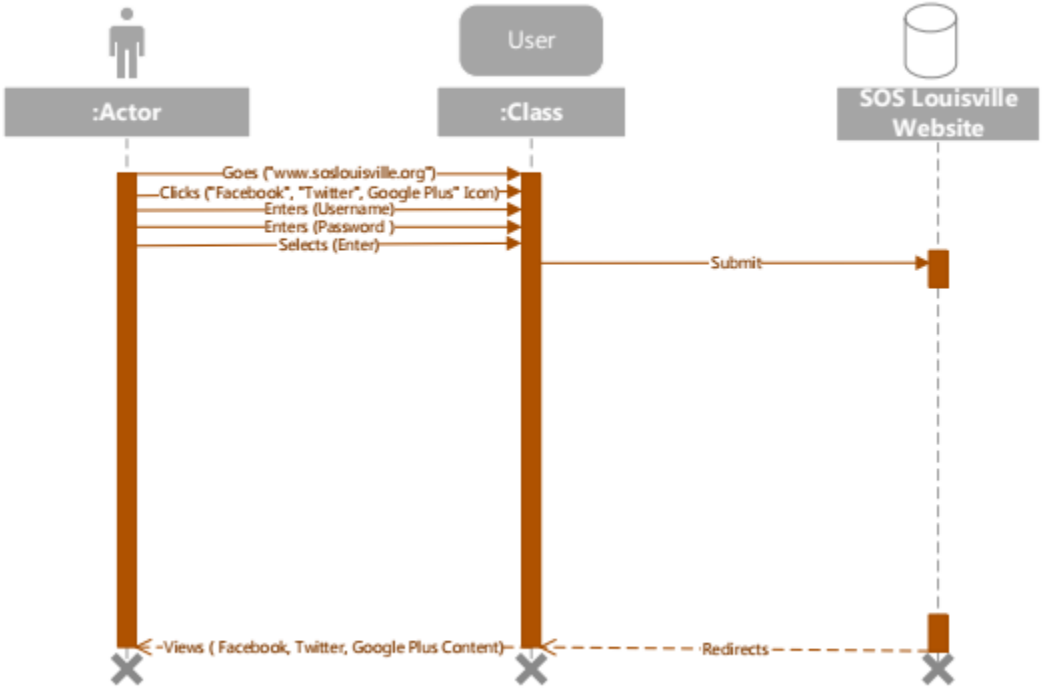
5.2 Notification

SOS Louisville will receive a notification when a user likes their social media page

6 Extension Points

None

View Social Media – Sequence Diagram



Use Case Specification: Translate Web Page to English

1. Use-Case Name: Translate Web Page to English

1.1 Brief Description

This use case will describe how a user will translate Surgery on Sunday's page from Spanish to English.

2. Flow of Events

2.1 Basic Flow

1. User goes to www.soslouisville.org
2. User locates language dropdown bar on the homepage
3. User clicks "Select Language" button
4. User selects "English" option
5. User views page in English
6. Use case ends

3. Special Requirements

3.1 User is an English speaker.

4. Pre-conditions

4.1 View website in Spanish

Website language is in Spanish

5. Post-conditions

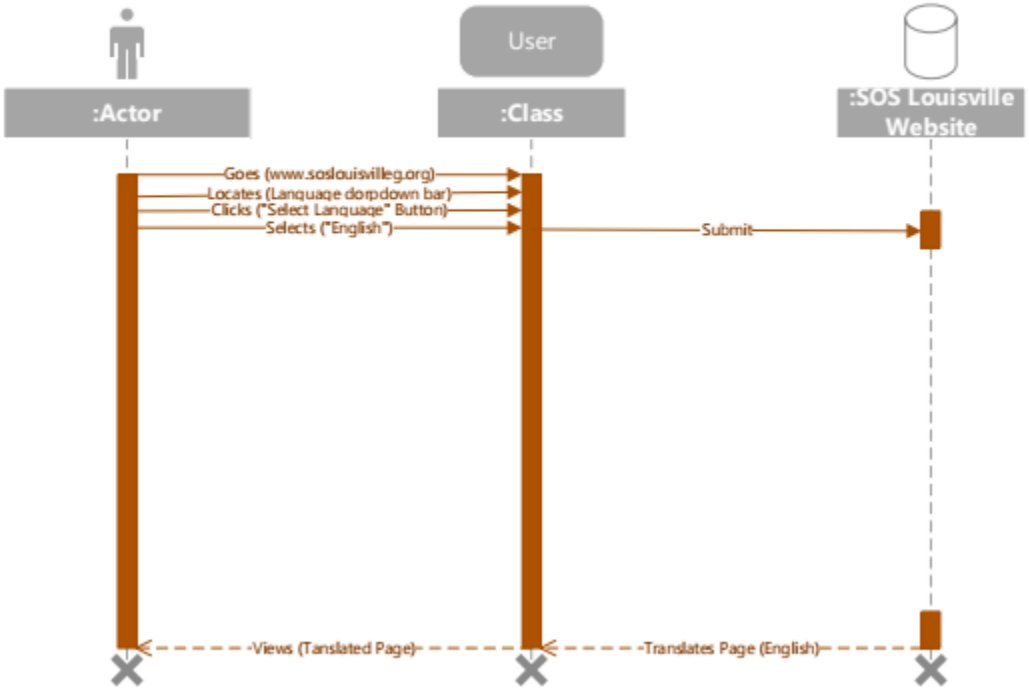
5.1 View website in English

Website is translated to English

6. Extension Points

None

Translate Web Page to English – Sequence Diagram



Use Case Specification: Translate Web Page to Spanish

1. Use-Case Name: Translate Web Page to Spanish

1.1 Brief Description

This use case will describe how a user will translate Surgery on Sunday's page from English to Spanish.

2. Flow of Events

2.1 Basic Flow

1. User goes to www.soslouisville.org
2. User locates language dropdown bar on the homepage
3. User clicks "Select Language" button
4. User selects "Spanish" option
5. User views page in Spanish.
6. Use case ends

3. Special Requirements

3.1 User is a Spanish speaker.

4. Pre-conditions

4.1 View website in English

Website language is in English

5. Post-conditions

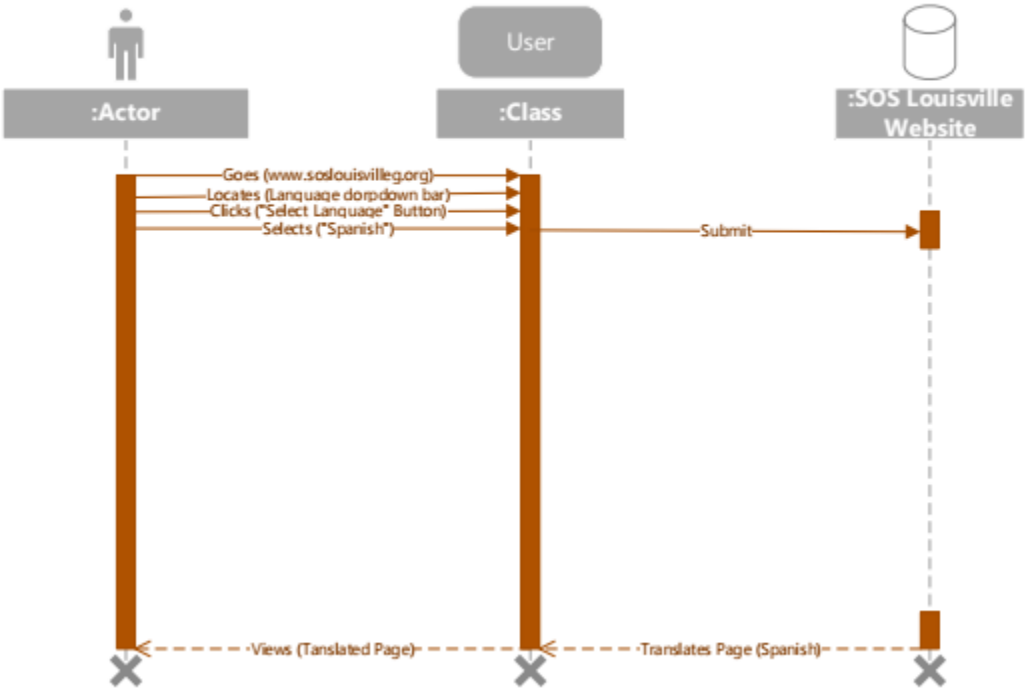
5.1 View website in Spanish

Website is translated to Spanish

6. Extension Points

None

Translate Web Page to Spanish – Sequence Diagram



Use Case Specification: Subscribe to Email Newsletter

1. Use-Case Name: Subscribe to Email Newsletter

1.1 Brief Description

This use case will describe how a user will subscribe to SOS Louisville's newsletter.

2. Flow of Events

2.1 Basic Flow

1. User goes to www.soslouisville.org
2. User locates "Newsletter" form on homepage
3. User enters email into form
4. User enters name into form
5. User views option to "Subscribe" or "Unsubscribe"
6. User selects "Subscribe"
7. User receives subscription confirmation email.
8. Use case ends

2.2 Alternative Flows

2.2.1 Unsubscribe to Newsletter

1. User goes to www.soslouisville.org
2. User locates "Newsletter" form on homepage
3. User enters email into form
4. User enters name into form
5. User views option to "Subscribe" or "Unsubscribe"
6. User selects "Unsubscribe"
7. User receives un-subscription confirmation email.
8. Use case ends

3. Special Requirements

3.1 Valid Email

The email entered must be an active account.

3.2 Subscribe or Unsubscribe

The user must select "Subscribe" or "Unsubscribe"

3.3 No information left blank

No information in the form can be left blank

4. Pre-conditions

4.1 Not Subscribed

The user does not have a subscription to the newsletter.

4.2 Subscribed

The user is subscribed to the email newsletter.

5. Post-conditions

5.1 User is Subscribed

User receives subscription confirmation email from SOS Louisville

5.2 Newsletter

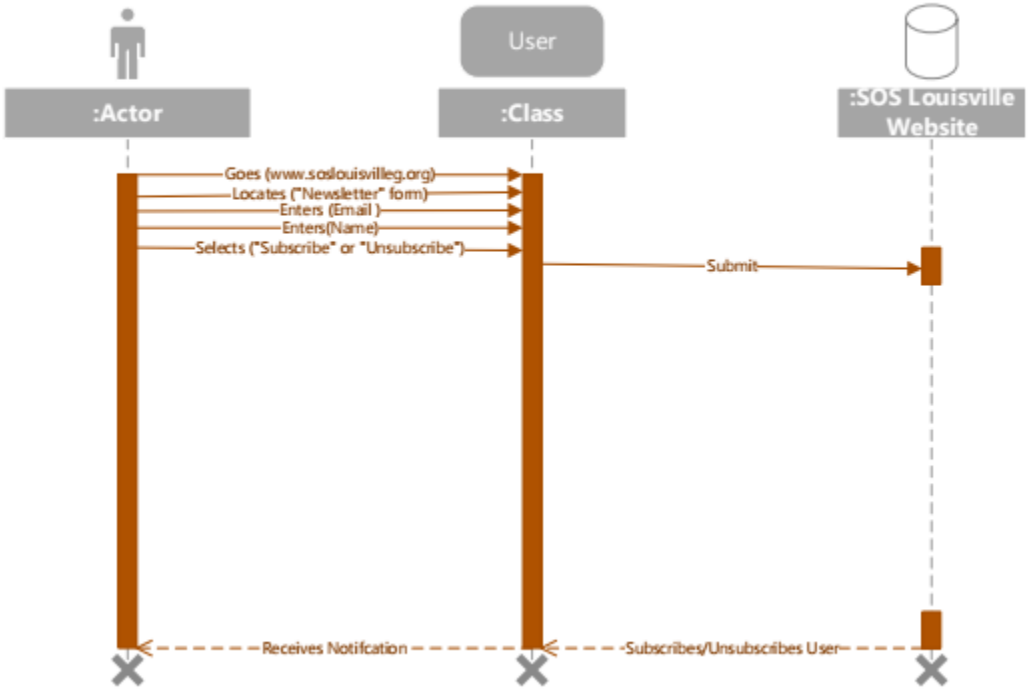
User will receive week/monthly newsletter from SOS Louisville

5.3 User is Unsubscribed
User receives un-subscription confirmation email from SOS Louisville

6. Extension Points

None

Subscribe to Email Newsletter – Sequence Diagram



Use Case Specification: Modify Subscriber

1. Use-Case Name: Modify Subscriber

1.1 Brief Description

This use case will describe how a staff member will modify a pre-existing subscriber in SOS Louisville's database.

2. Flow of Events

2.1 Basic Flow

1. Staff enters subscriber database
2. Staff locates subscriber
3. Staff modifies subscriber's email
4. Staff selects "save"
5. End of use case

2.2 Alternative Flows

2.2.1 Modify Subscriber's name

1. Staff enters subscriber database
2. Staff locates subscriber
3. Staff modifies subscriber's name
4. Staff selects "save"
5. End of use case

3. Special Requirements

3.1 Valid Email

The email entered must be an active account.

3.2 Subscriber's Name

Subscriber's name must be correctly associated with the email

4. Pre-conditions

4.1 Subscribed

The user is subscribed to the email newsletter.

5. Post-conditions

5.1 Modified Subscriber's Email

The new email address will receive the weekly/monthly subscription

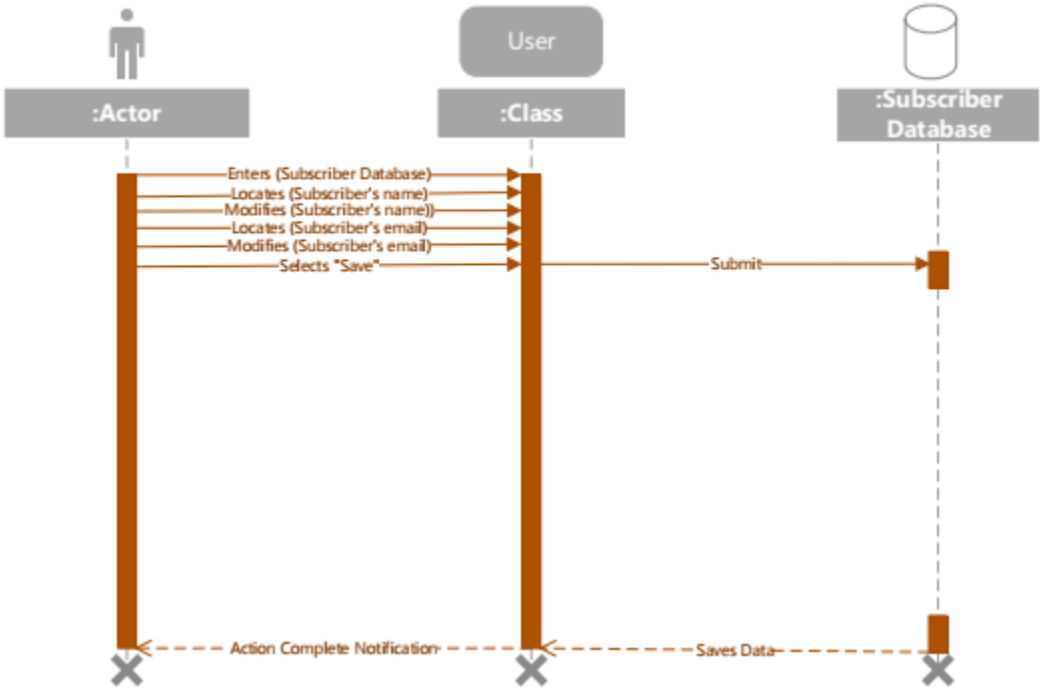
5.2 Modified Subscriber's Name

The weekly/monthly newsletter will be addressed to the modified subscriber name

6. Extension Points

None

Modify Subscriber – Sequence Diagram



Use Case Specification: Delete Subscriber

1. Use-Case Name: Delete Subscriber

1.1 Brief Description

This use case will describe how a staff member will delete a pre-existing subscriber in SOS Louisville's database.

2. Flow of Events

2.1 Basic Flow

1. Staff enters subscriber database
2. Staff locates subscriber
3. Staff unchecks subscriber box next to name
4. Staff selects "save"
5. End of use case

3. Special Requirements

3.1 Subscribed

The name and email must of an active subscriber.

4. Pre-conditions

4.1 Subscribed

The user is subscribed to the email newsletter.

5. Post-conditions

5.1 Deleted Subscriber

Subscriber name and email will no longer be available in the database.

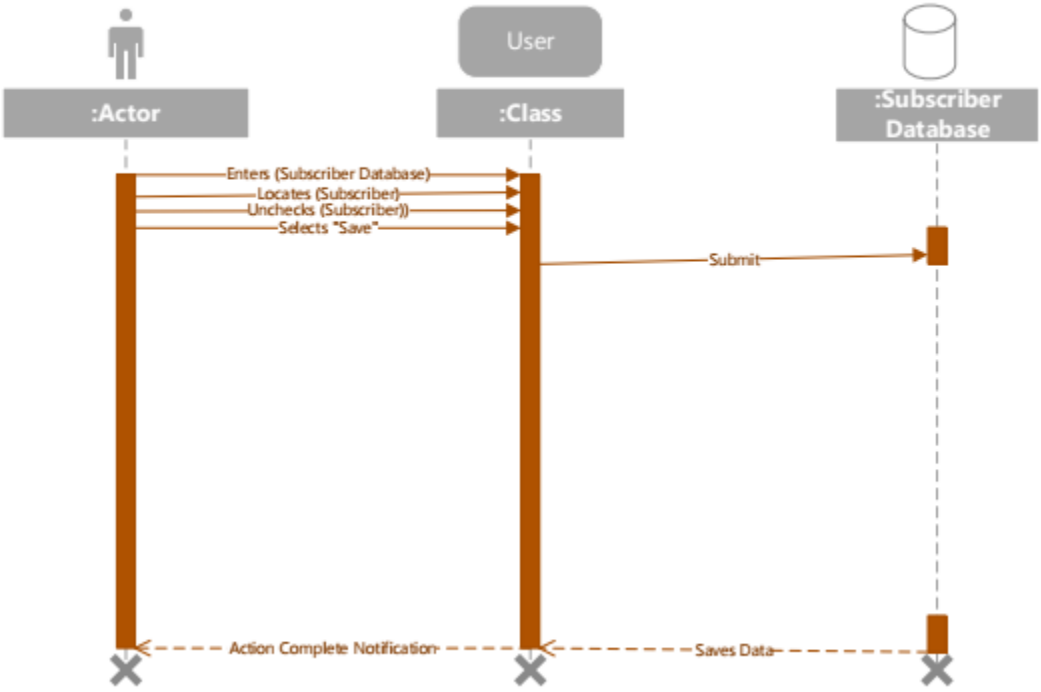
5.2 Unsubscribed to Newsletter

Subscriber will not receive weekly/monthly newsletter

6. Extension Points

None

Delete Subscriber – Sequence Diagram



Use Case Specification: Send Newsletter

1. Use-Case Name: Send Newsletter

1.1 Brief Description

This use case will describe how a staff member will send a weekly/monthly newsletter to subscribers.

2. Flow of Events

2.1 Basic Flow

1. Staff creates content for weekly/monthly newsletter.
2. Staff enters subscriber database
3. Staff extrapolates subscribers' emails from database
4. Staff navigates to organization's email page
5. Staff selects compose email
6. Staff enters subscriber's emails into "to" textbox
7. Staff enters newsletter subject into "subject" textbox
8. Staff enters newsletter content into "body" textbox
9. Staff reviews information entered
10. Staff clicks "send" button
11. Use case ends

3. Special Requirements

3.1 Valid Email

The email entered must be an active account.

3.2 Subscribed User

The email entered must be of a subscribed user

4. Pre-conditions

4.1 Subscribed

The user is subscribed to the email newsletter.

5. Post-conditions

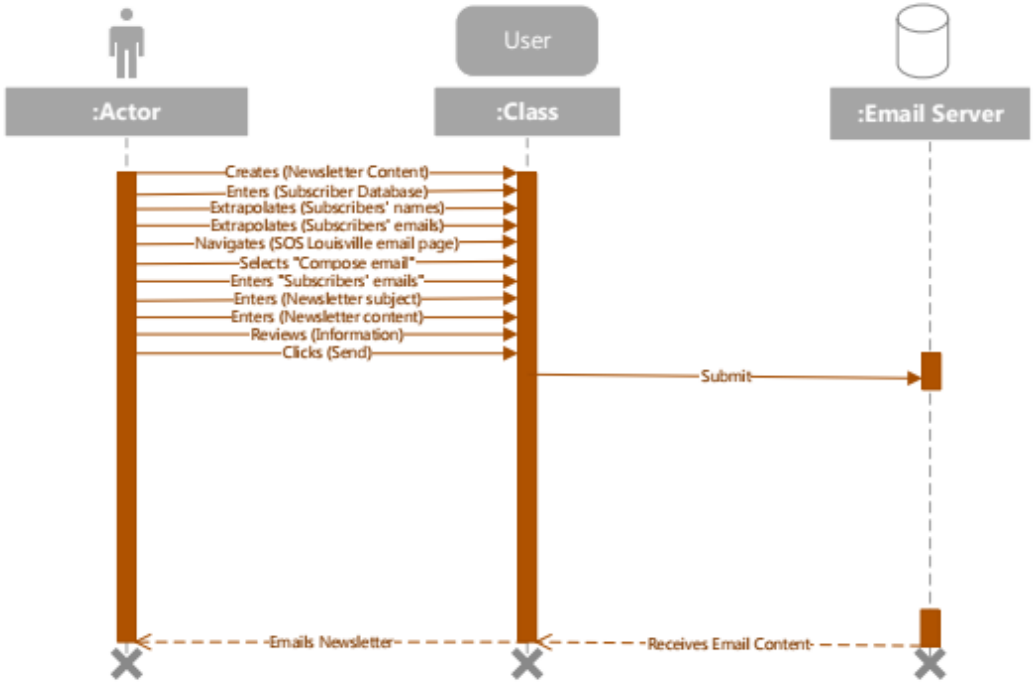
5.1 Newsletter

User receives weekly/monthly newsletter

6. Extension Points

None

Send Newsletter – Sequence Diagram



Use Case Specification: Create Admin Account

1. Create Admin Account

1.1 Brief Description

This use case will allow the actor to successfully create an administrator account for Microsoft Azure and WordPress. Initial creation of administrator accounts will be done by an SOS Administrator. Once initial admin account is created, the admin will then login into the Azure portal, select a subscription you want the admin to access, and add them as an owner. Once complete, you will then be able to add them as a co-administrator to the Azure account.

2. Flow of Events

2.1 Basic Flow

1. Admin logs into Azure Portal
2. Admin logs into the Subscription portal
3. Admin clicks Add Control (IAM) button
4. Admin selects Add
5. Admin selects Role
6. Admin selects Owner
7. Admin types email address of new owner.
8. Admin selects Save

2.2 Alternative Flows

2.2.1 Co-Administrator Created from Owner

1. Admin logs into Azure Portal
2. Admin logs into the Subscription portal
3. Admin clicks Add Control (IAM) button
4. Admin selects Add
5. Admin selects Role
6. Admin selects Owner
7. Admin types email address of new owner.
8. Admin selects Save
9. Admin right-clicks the new Owner
10. Admin selects Add as co-administrator

3. Special Requirements

3.1 Active Microsoft Azure Subscription

SOS Louisville administrators and staff will purchase a subscription for Microsoft Azure.

3.2 Valid Email Address

A valid email address is required to be an administrator. This email address will typically be on with domain of SOS Louisville.

4. Pre-conditions

4.1 Purchased a Microsoft Azure Subscription

SOS Louisville has an active account with Microsoft Azure.

5. Post-conditions

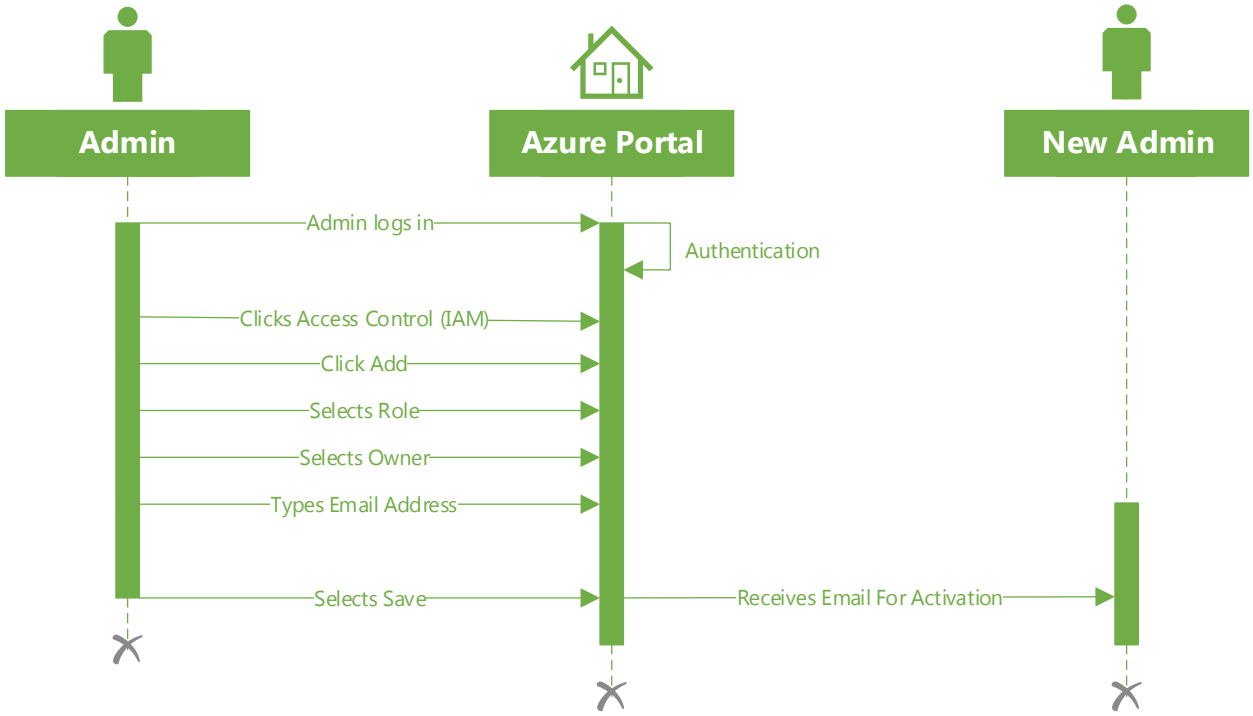
5.1 Administrator Created

Administrator has been created and thus can manipulate the entire system.

6. Extension Points

None

Create Admin Account – Sequence Diagram



Use Case Specification: Modify Admin Account

1. Modify Admin Account

1.1 Brief Description

The administrator will be able to modify an existing admin account. They will have to login to either a top tier admin account above all others, or login to the admin account being modified.

2. Flow of Events

2.1 Basic Flow

1. Admin logs into their account
2. Admin selects the subscription the admin to be modified is under
3. Admin selects Access Control (IAM)
4. Admin may modify access levels of the account.

2.2 Alternative Flows

2.2.1 Modify Admin from Root Account

1. Admin logs into their account
2. Admin selects the subscription the admin to be modified is under
3. Admin selects Access Control (IAM)
4. Admin may modify access levels of the account

2.2.2 Modify Admin Account from User Perspective

1. Admin logs into their account
2. Admin goes to Azure Account Center
3. Admin selects Edit
4. Admin selects Profile
5. Admin select Edit Details
6. Admin may change Name
7. Admin may change Email Address
8. Admin may change Password
9. Admin may change Home Address
10. Admin will click Save Changes

3. Special Requirements

3.1 Admin Account Previously Created

An active admin account to be modified must already be created

4. Pre-conditions

4.1 Admin Account Previously Created

An active admin account to be modified must already be created

5. Post-conditions

5.1 Admin Account Access Level Modified

The access level of the admin account has been changed

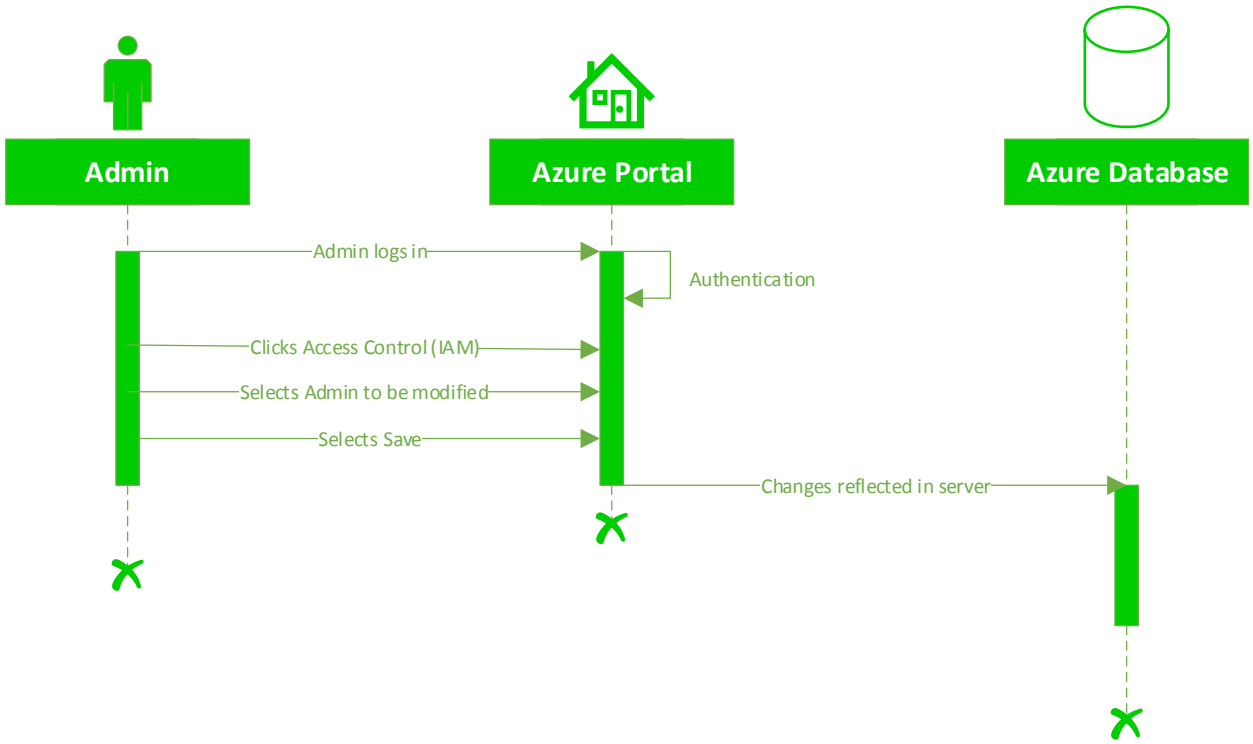
5.2 Admin Account Information Modified

The information for the admin account has changed (i.e., Name, email address, home address, password, etc.)

6. Extension Points

None

Modify Admin Account – Sequence Diagram



Use Case Specification: Delete Admin Account

1. Delete Admin Account

1.1 Brief Description

Administrators will be able to log in to Microsoft Azure and delete an admin account that was previously created.

2. Flow of Events

2.1 Basic Flow

1. Admin logs into Azure Portal
2. Admin logs into the Subscription portal
3. Admin clicks Add Control (IAM) button
4. Admin right-click the user
5. Admin selects Remove Co-Administrator
6. Admin unchecks Owner box

3. Special Requirements

3.1 Valid Administrators

Subscription must have more than one administrator to delete an account.

4. Pre-conditions

None

5. Post-conditions

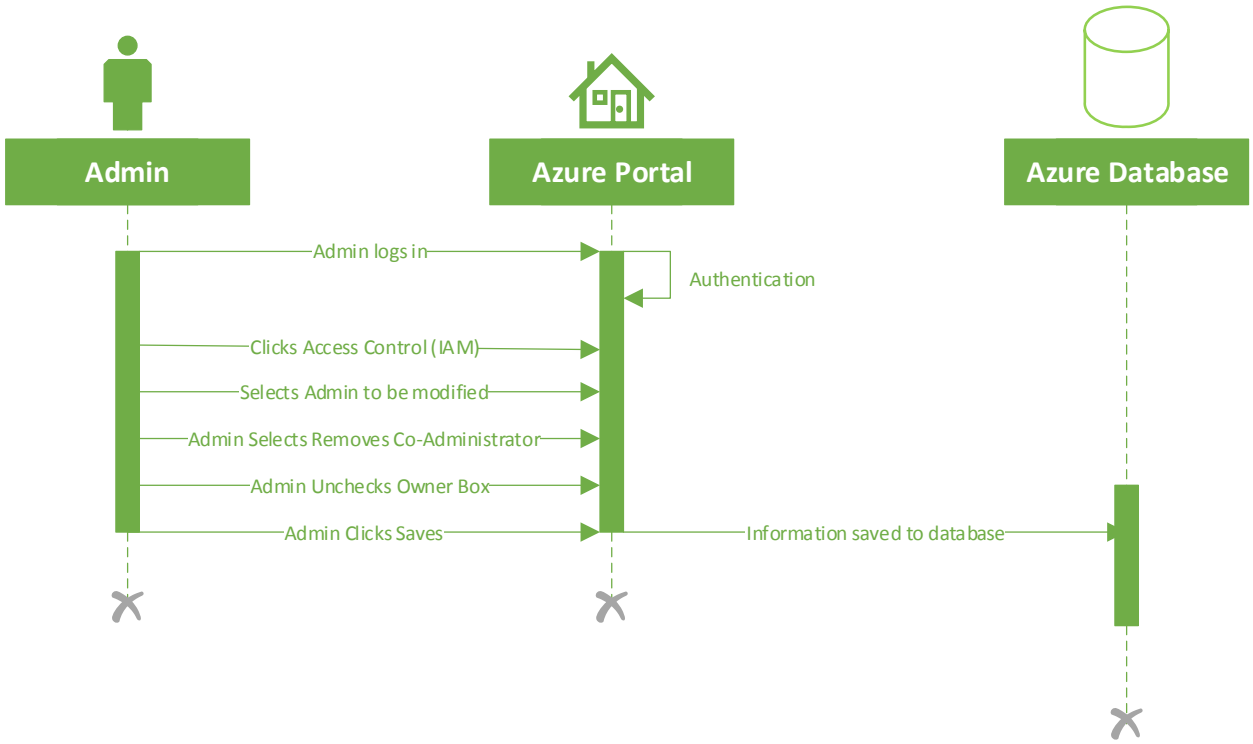
5.1 Admin Account deleted

Once the administrator has unchecked the Owner box, the user will no longer be an admin.

6. Extension Points

None

Delete Admin Account – Sequence Diagram



Use Case Specification: Login Admin Account

1. Login Admin Account

1.1 Brief Description

Administrators are taken through the process of logging into Microsoft Azure.

2. Flow of Events

2.1 Basic Flow

1 Admin goes to https://portal.azure.com
2 Admin will type in the username
3 Admin will type in the password
4 Admin presses Enter

4.1 Alternative Flows

4.1.1 Admin Logs in For First Time

1. Admin receives an email regarding their new account
2. Admin will be given their username
3. Admin will be given their temporary password
4. Admin goes to https://portal.azure.com
5. Admin will type in the username
6. Admin will type in the temporary password
7. Admin presses Enter
8. Admin is instructed to change their temporary password
9. Admin can access the full dashboard

3. Special Requirements

None

4. Pre-conditions

4.1 Valid Email

New admin must have a valid email addresses

5. Post-conditions

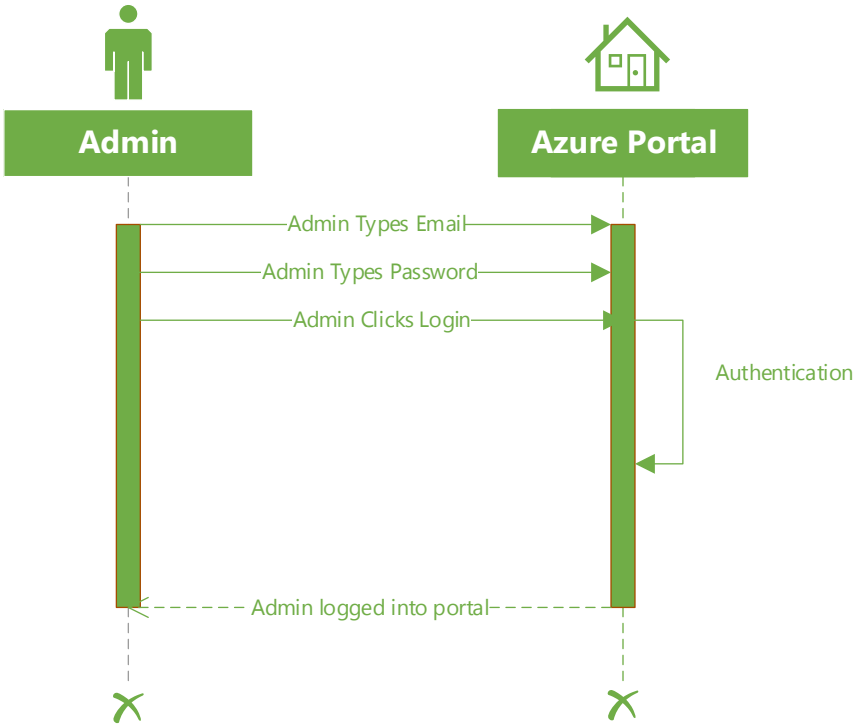
5.1 Admin Is Logged In

Admin can log in to the organizational dashboard

6. Extension Points

None

Login Admin Account – Sequence Diagram



Use Case Specification: Backup Database

1. Backup Database

1.1 Brief Description

This use case describes the steps needed to back up the relation database in Microsoft Azure. There are two ways this can easily be done. One, we back up the database using Azure's geo-redundant functions. This is already set up once the database is created. Second, we can export all the data to a local PC.

2. Flow of Events

2.1 Basic Flow

1. Admin logs into Microsoft Azure
2. Admin clicks on their database
3. Admin clicks Export
4. Admin clicks Storage
5. Admin enters username
6. Admin enters password
7. Admin clicks Create to export database

2.2 Alternative Flows

2.2.1 Automatic Data Retention Via Azure

1. Admin logs into Microsoft Azure
2. Admin clicks Recovery Services Vaults
3. Admin creates a new Recovery Services Vault
4. Admin clicks on their database
5. Admin clicks on their server name
6. Admin clicks Long-term backup retention
7. Admin clicks on their database
8. Admin clicks Configure
9. Admin selects their Recovery Services Vault
10. Admin selects their retention policy.
11. Admin selects Save Changes.

3. Special Requirements

None

4. Pre-conditions

4.1 Server Present

Server is up and running with Microsoft Azure

4.2 Database is Present

Database is up and running with Microsoft Azure and data is populated to it.

5. Post-conditions

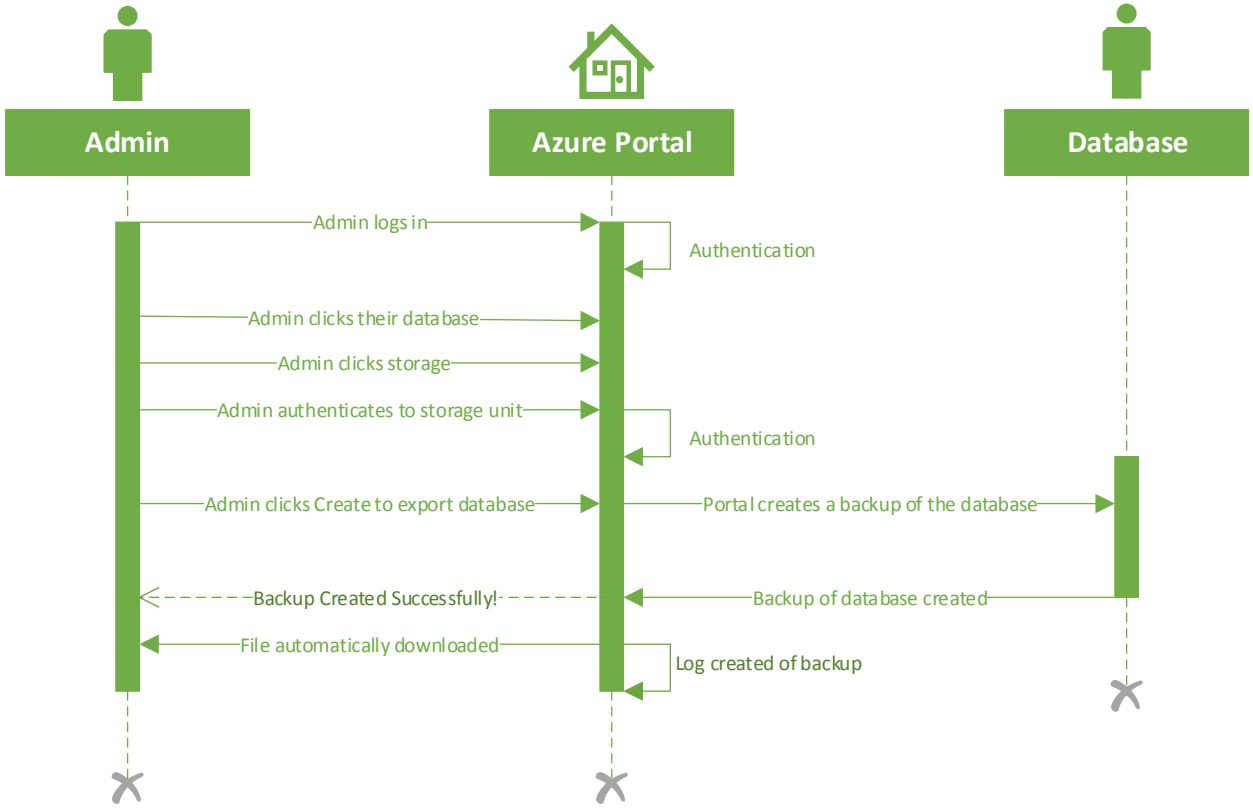
5.1 Data is Backed Up

Data is successfully backed up for their retention period.

6. Extension Points

None

Backup Database – Sequence Diagram



Use Case Specification: Recover Database

1. Recover Database

1.1 Brief Description

This use case defines how the administrators will be able to recover their data in the event there is data loss. If the administrators regularly export their database, they can import their database back in. If they have set up long-term data retention, they will be able to restore their data from the last backup in Microsoft Azure.

2. Flow of Events

2.1 Basic Flow

1. Admin logs into Azure Portal
2. Admin selects their SQL Server
3. Admin clicks Import Database
4. Admin selects their storage
5. Admin enters their username
6. Admin enters their password
7. Admin selects the latest backup from file
8. Admin clicks OK

2.2 Alternative Flows

2.2.1 Data Backup via Azure

1. Admin logs into Azure portal
2. Admin clicks on their database
3. Admin clicks on their server name
4. Admin clicks Long-term backup retention
5. Admin clicks on their database
6. Admin clicks on Overview
7. Admin clicks Restore on the top bar
8. Admin selects the backup they want to restore from
9. Admin clicks OK.

3. Special Requirements

3.1 Manual Backups Created

The admins have already backed up their database manually

3.2 Backups Created via Azures Retention

Backups have been previously set up in Long-term data retention

4. Pre-conditions

4.1 Database Experiences Emergency Loss

Database and/or server has quit working and there has been a loss of data

5. Post-conditions

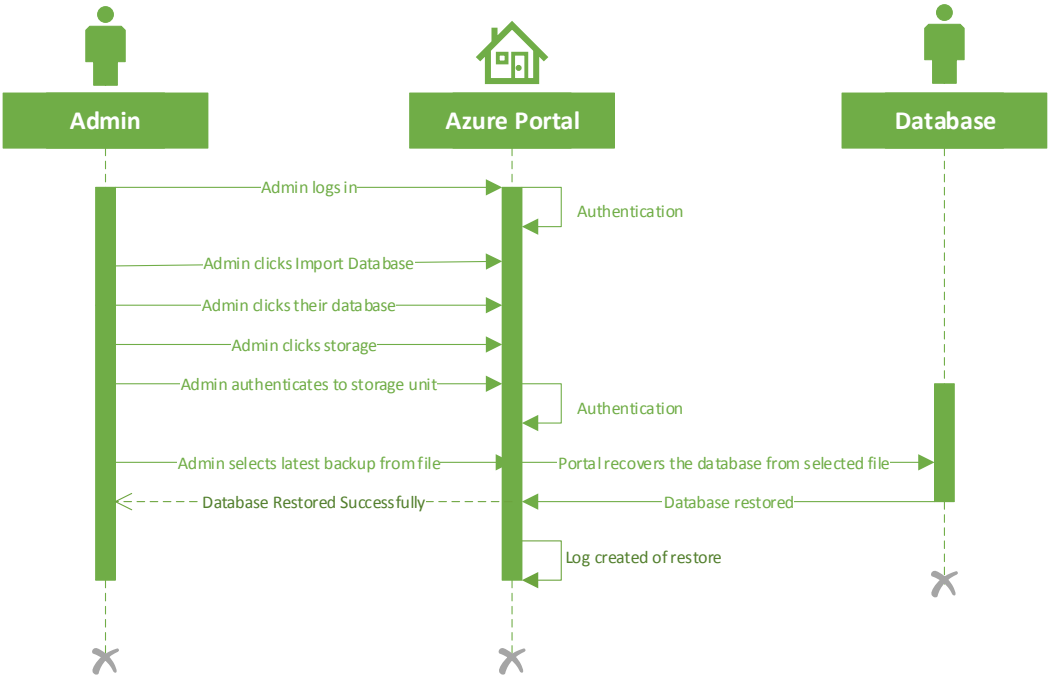
5.1 Database Restored

Database is restored from a manual backup or the retention policy

6. Extension Points

None

Recover Database – Sequence Diagram



Use Case Specification: Create Reports

1. Create Reports

1.1 Brief Description

This use case explains how the administrators of SOS Louisville can successfully and efficiently build reports based off the needs of an event or to build the distribution list for current subscribers. They will be able to select a Volunteer's qualifications, staff needed, and other entities.

2. Flow of Events

2.1 Basic Flow

1. Admin logs into Microsoft Azure
2. Admin clicks SQL Database on the left side.
3. Admin clicks Object-Relational Mapping (ORM) extension
4. Admin will be able to report contents of volunteer
5. Admin will be able to report contents of staff
6. Admin will be able to report contents of donor
7. Admin will be able to report contents of donations
8. Admin will be able to report the newsletter list
9. Admin will click Save Changes

2.2 Alternative Flows

2.2.1 Report Volunteers

1. Admin logs into Microsoft Azure
2. Admin clicks SQL Database on the left side
3. Admin clicks Object-Relational Mapping (ORM) extension
4. Admin can report the name
5. Admin can report the home address
6. Admin can report the email address
7. Admin can report the phone number
8. Admin can report the qualifications
9. Admin can report their specialty
10. Admin can report their availability
11. Admin will click Save Changes

2.2.2 Report Staff

1. Admin logs into Microsoft Azure
2. Admin clicks SQL Database on the left side
3. Admin clicks Object-Relational Mapping (ORM) extension
4. Admin can report the name
5. Admin can report the home address
6. Admin can report the email address
7. Admin can report the phone number
8. Admin can report the specialty
9. Admin can report the role
10. Admin will click Save Changes

2.2.3 Report Donor

1. Admin logs into Microsoft Azure
2. Admin clicks SQL Database on the left side
3. Admin clicks Object-Relational Mapping (ORM) extension

4. Admin can report the name
5. Admin can report the email address
6. Admin will click Save Changes

2.2.4 Report Donation

1. Admin logs into Microsoft Azure
2. Admin clicks SQL Database on the left side
3. Admin clicks Object-Relational Mapping (ORM) extension
4. Admin can report the donation amount
5. Admin will click Save Changes

2.2.5 Report Subscriber

1. Admin logs into Microsoft Azure
2. Admin clicks SQL Database on the left side
3. Admin clicks Object-Relational Mapping (ORM) extension
4. Admin can report the subscriber's name
5. Admin can report the subscriber's email address
6. Admin will click Save Changes

3. Special Requirements

None

4. Pre-conditions

4.1 Object-Relational Mapping Set Up

SOS Consultants must provide you with an ORM that is functional so that reporting from the database is easy and efficient.

5. Post-conditions

5.1 Reports Created

Reports are created based on the admin specifications.

6. Extension Points

None

Create Reports – Sequence Diagram