



Elaboration Phase

Quality Control

Garrett Diebold

Jessica Hoffman

Lizbeth Juarez-Robles

Damon Quire

Mariah Skeeters

Kaylee Thomas

TABLE OF CONTENTS

System Requirements	7
Functional	7
Donors.....	7
Surgeons.....	8
Nurses	8
Clinical language interpreters	9
Non-clinical volunteers	10
Events.....	10
Referrals.....	11
Non-functional	11
Availability.....	11
Storage.....	11
Maintenance	12
Constraints	12
Use Case Diagrams.....	13
Requirements Traceability Matrix	21
Functional System Requirements Traceability Matrix	21
System Requirements 01 -24.....	21
System requirements 25-56.....	22
Nonfunctional System Requirements Traceability Matrix.....	22
System requirements 57-71.....	22
Functional Use Cases	23
Nonfunctional Use Cases	25
Use Cases and Sequence Diagrams.....	26
Use Case ID: 1.....	26
Use Case ID: 2.....	28
Use Case ID: 3.....	30
Use Case ID: 4.....	32

Use Case ID: 5.....	34
Use Case ID: 6.....	36
Use Case ID: 7.....	37
Use Case ID: 8.....	38
Use Case ID: 9.....	40
Use Case ID: 10	42
Use Case ID: 11	45
Use Case ID: 12	47
Use Case ID: 13	49
Use Case ID: 14	50
Use Case ID: 15	52
Use Case ID: 16	54
Use Case ID: 17	56
Use Case ID: 18	57
Use Case ID: 19	60
Use Case ID: 20	62
Use Case ID: 21	64
Use Case ID: 22	66
Use Case ID: 23	68
Use Case ID: 24	70
Use Case ID: 25	72
Use Case ID: 26	74
Use Case ID: 27	77
Use Case ID: 28	79
Use Case ID: 29	81
Use Case ID: 30	83
Use Case ID: 31	85
Use Case ID: 32	87
Use Case ID: 33	89
Use Case ID: 34	91

Use Case ID: 35	93
Use Case ID: 36	95
Use Case ID: 37	97
Use Case ID: 38	99
Use Case ID: 39	101
Use Case ID: 40	103
Use Case ID: 41	106
Use Case ID: 42	109
Use Case ID: 43	111
Use Case ID: 44	113
Use Case ID: 45	115
Use Case ID: 46	117
Use Case ID: 47	119
Use Case ID: 48	122
Use Case ID: 49	124
Use Case ID: 50	126
Use Case ID: 53	130
Use Case ID: 54	132
Use Case ID: 55	134
Use Case ID: 56	135
Use Case ID: 57	136
How to Read the Class Diagram.....	138
Class Diagram.....	139
SOS Database Design	140
Entity Relationship Diagram	141
Data Definitions	142
Patient.....	142
Event	143
SurgicalEvent.....	144
FundraisingEvent.....	145

VolunteerTeam	146
Volunteer	147
Category.....	148
Specialty.....	149
License.....	150
User	151
UserType	152
Donor	153
Donation	154
User Interface Navigation Diagram and Screen Layouts	155
Site Layout - Home.....	156
Site Layout – Admin Portal.....	156
Deployment Diagram	157
Non-Functional Requirements Design Procedures.....	158
Availability.....	158
Storage	158
Maintenance	159
Constraints	159
Gantt Chart	160
Iterations 3 – 5 and Elaboration Phase.....	161
Prototypes.....	162
Update Delete Database Record.....	163
Surgery Scheduling.....	164
Donation Submission	165
About Us	166
Send Emails	167
Events.....	168
Get Involved/Volunteer Form.....	169
Donation	170
Admin Portal	171

Admin Portal - Reports.....	172
Admin Portal – Login.....	173
Dynamic Volunteer Form	174
Donation Form	175
Maintain Nurse Records	176
Schedule Surgery	178
Thank You Email.....	179
Email Nurses.....	180
Add Interpreter.....	181
Reports.....	182
Maintain Interpreter	183
Maintain Nurse Records	184
Thank You Emails - Nurses	185
Maintain Interpreter Records	186
Referrals Reports	187
New Referrals.....	188
Referral Form	189
Referral Login.....	190
Event Report	191
Add Attendance	192
Thank You Email.....	193
Refer a Patient	194
Fundraising.....	195
Referral Notification	196
Volunteer Is Updated or Deleted:.....	197
Volunteer is Added:	198
Send Thank You Emails:	199
Schedule Event with Interpreter and Volunteers Included:	200
Send Emails to Interpreters and Volunteers:.....	203

SYSTEM REQUIREMENTS

System requirements are the functional and non-functional deliverables of a system. These requirements are necessary for the system to perform desired business processes effectively. Functional requirements are defined as the physical components within the system. Nonfunctional requirements are defined as the virtual components within the system.

Functional

Donors

ID	Requirement	Risk
01	The new system shall allow users to insert donor records into the database	High
02	The new system shall allow users to update donor records	High
03	The new system shall allow users to delete donor records	Low
04	The new system shall automatically insert donor records when donations are submitted online	Medium
05	The new system shall utilize a user-friendly payment system for donations	Medium
06	The new system shall allow for donations by card	Low
07	The new system shall automatically deposit new donations into SOS bank account	High
08	The new system shall provide reports about donors	Medium
09	The new system shall provide reports about donations	Medium
10	The new system shall send automated emails thanking donors	Low

11	The new system shall send newsletter emails to donors to keep them informed	Low
-----------	---	------------

Surgeons

12	The new system shall allow users to insert surgeon records into the database	High
13	The new system shall allow users to update surgeon records	High
14	The new system shall allow users to delete surgeon records	Low
15	The new system shall use dynamic forms to capture additional information specific to surgeons	Medium
16	The new system shall store information about surgeon licenses	Low
17	The new system shall provide reports about surgeons	Medium
18	The new system shall send automated emails to a surgeon scheduled for an event	Low
19	The new system shall allow users to easily match surgeons to surgical events	High
20	The new system shall send emails to thank surgeons	Low

Nurses

21	The new system shall allow users to insert nurse records into the database	High
22	The new system shall allow users to update nurse records	High
23	The new system shall allow users to delete nurse records	Low
24	The new system shall use dynamic forms to capture additional information specific to nurses	Medium

25	The new system shall store information about nurse licenses (if applicable)	Low
26	The new system shall provide reports about nurses	Medium
27	The new system shall send automated emails to a nurse scheduled for an event	Low
28	The new system shall allow users to easily match nurses to surgical events	High
29	The new system shall send emails to thank nurses	Low

Clinical language interpreters

30	The new system shall allow users to insert interpreter records into the database	High
31	The new system shall allow users to update interpreter records	High
32	The new system shall allow users to delete nurse records	Low
33	The new system shall use dynamic forms to capture additional information specific to interpreters	Medium
34	The new system shall store information about interpreter licenses (if applicable)	Low
35	The new system shall provide reports about interpreters	Medium
36	The new system shall send automated emails to a interpreter scheduled for an event	Low
37	The new system shall allow users to easily match interpreters to surgical events	High
38	The new system shall send emails to thank clinical language interpreters	Low

Non-clinical volunteers

39	The new system shall allow users to insert non-clinical volunteer records into the database	High
40	The new system shall allow users to update volunteer records	High
41	The new system shall allow users to delete volunteer records	Low
42	The new system shall use dynamic forms to capture additional information specific to non-clinical volunteers	Medium
43	The new system shall provide reports about volunteers	Medium
44	The new system shall send automated emails to a volunteer scheduled for an event	Low
45	The new system shall allow users to easily match volunteers to surgical events	High
46	The new system shall send emails to thank volunteers	Low

Events

47	The new system shall allow users to insert event records into the database	High
48	The new system shall allow users to update event records	High
49	The new system shall allow users to delete event records	Low
50	The new system shall provide reports about events	Medium
51	The new system shall manage volunteer attendance for events	Medium

Referrals

52	The new system shall capture additional data about patients on the referral form	Medium
53	The new system shall include instructions for getting access to referral form	Low
54	The new system shall use dynamic forms to capture additional information	Low
55	The new system shall provide reports about referrals (if applicable)	Medium
56	The new system shall email SOS coordinators when a new referral has been submitted	Low

Non-functional

Availability

57	The new system shall be available every day	High
58	The new system shall not crash if website traffic increases	High
59	The new system's website shall be available to the public	High
60	The new system shall be in place by the end of Spring 2018, unless otherwise stated	Medium

Storage

61	The new system's data storage shall be available to SOS coordinators only	High
62	The new system shall provide enough data storage	High
63	The new system shall use efficient database design and relationships	Medium
64	The new system's data shall be backed up	High

Maintenance

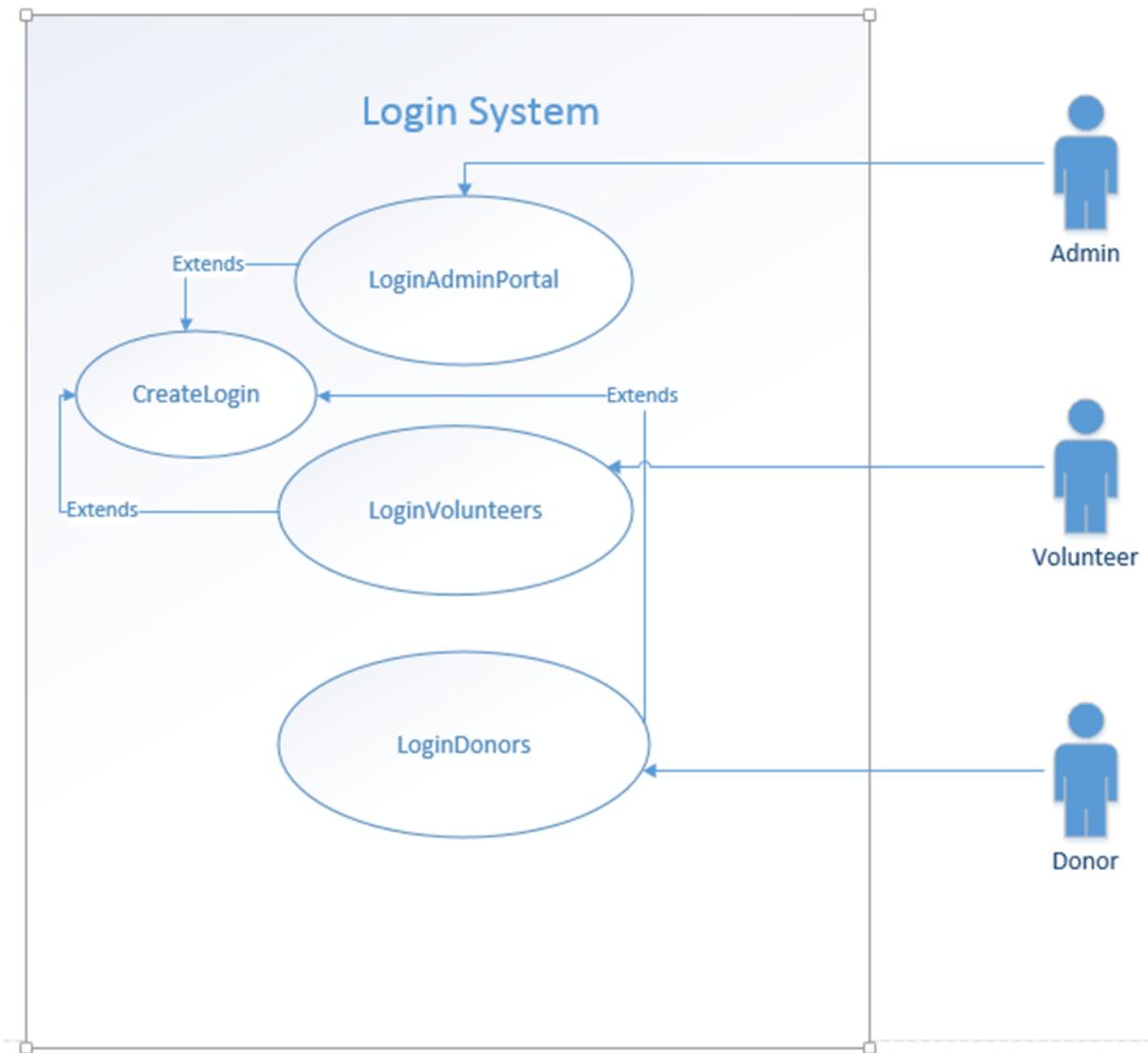
65	The new system shall be easy to maintain	High
66	The new system shall be thoroughly tested and not break or have bugs	High
67	The new system maintenance shall be available to the IT coordinator	Medium

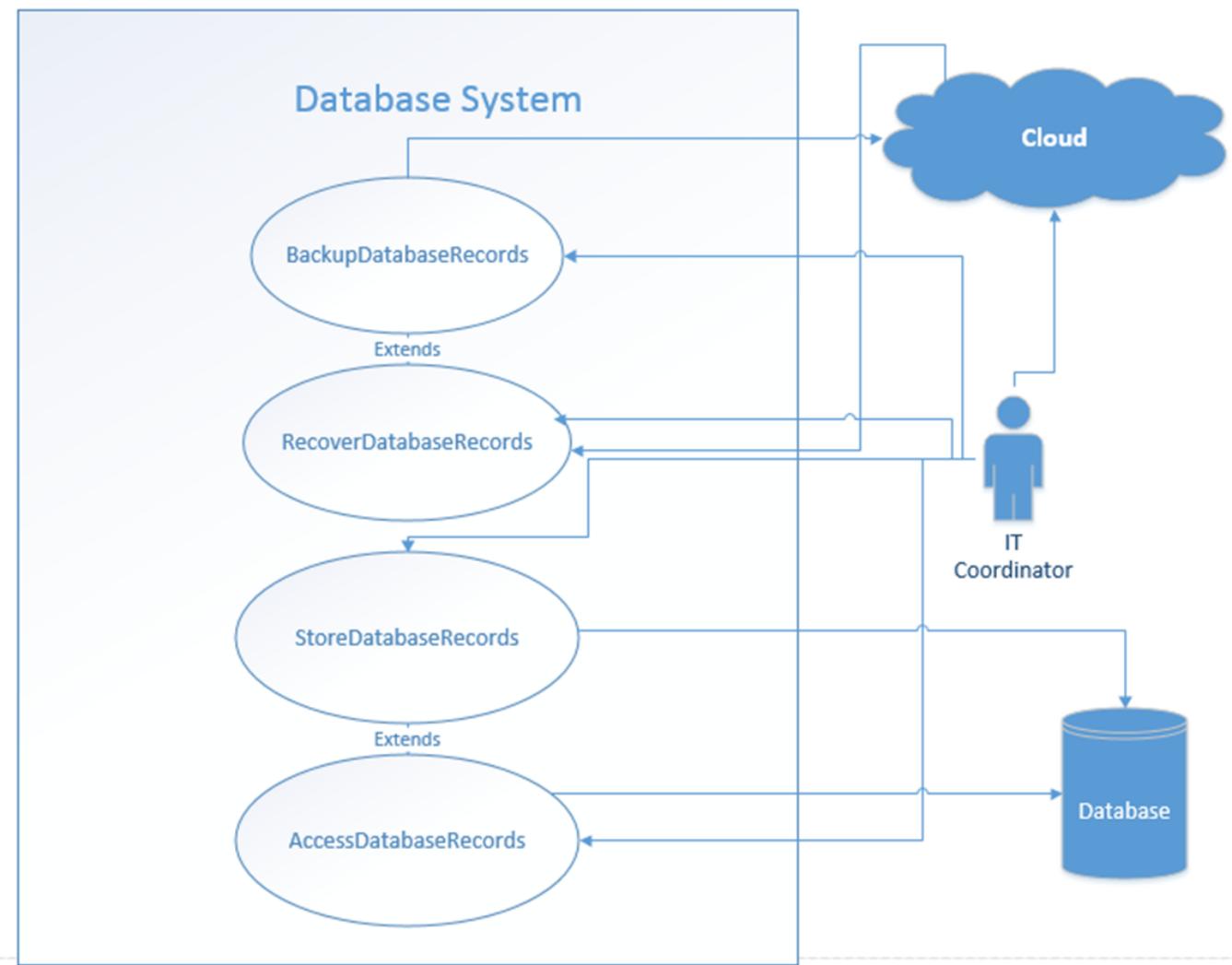
Constraints

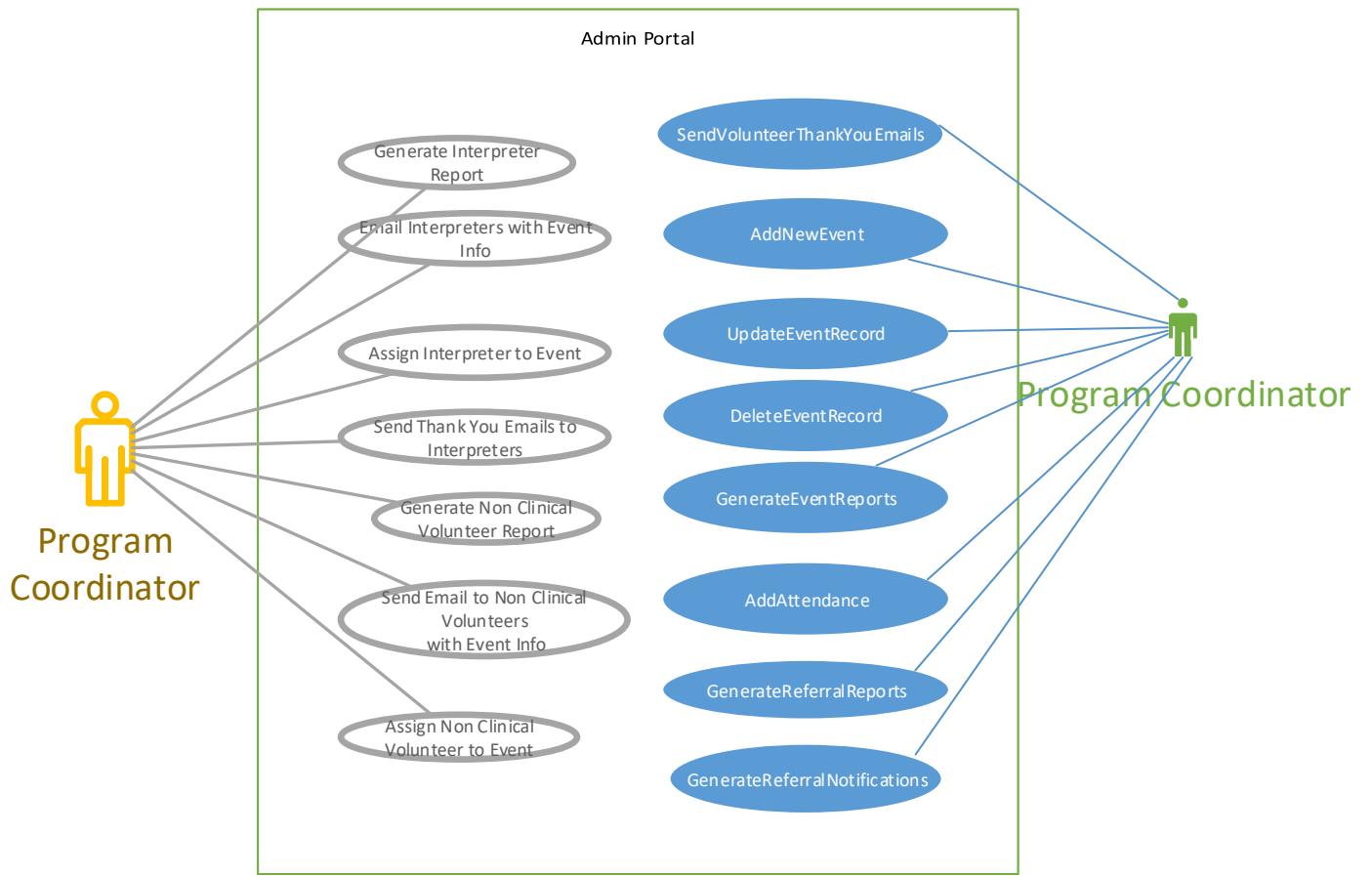
68	The new system shall not handle sensitive data about patients	High
69	The new system shall not violate HIPAA	High
70	The new system shall not interface with the RedCap or Epic Connect patient database	High
71	The new system shall allow for disclosure of all volunteers, per the Federal Tort Claims Act	High

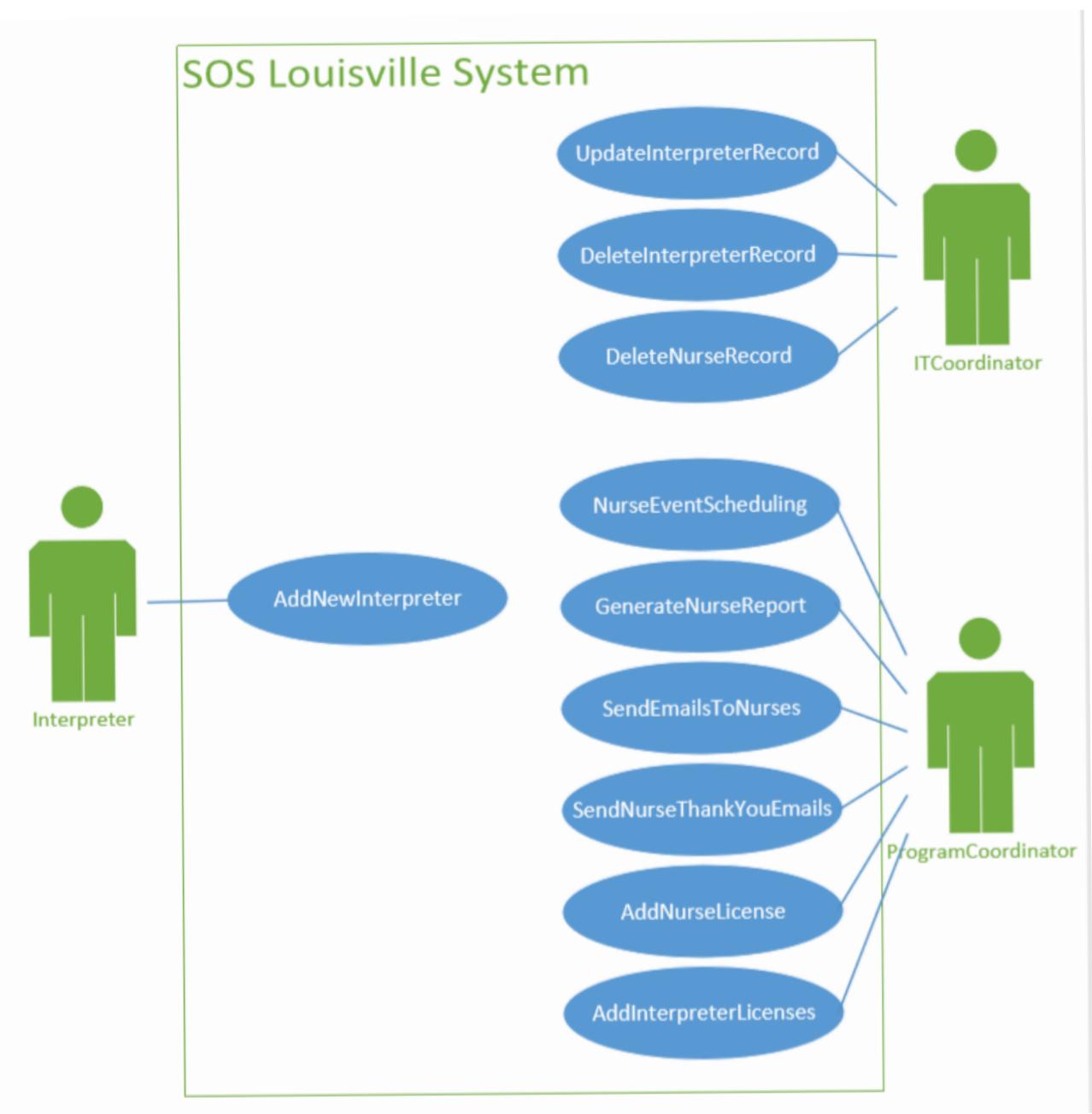
USE CASE DIAGRAMS

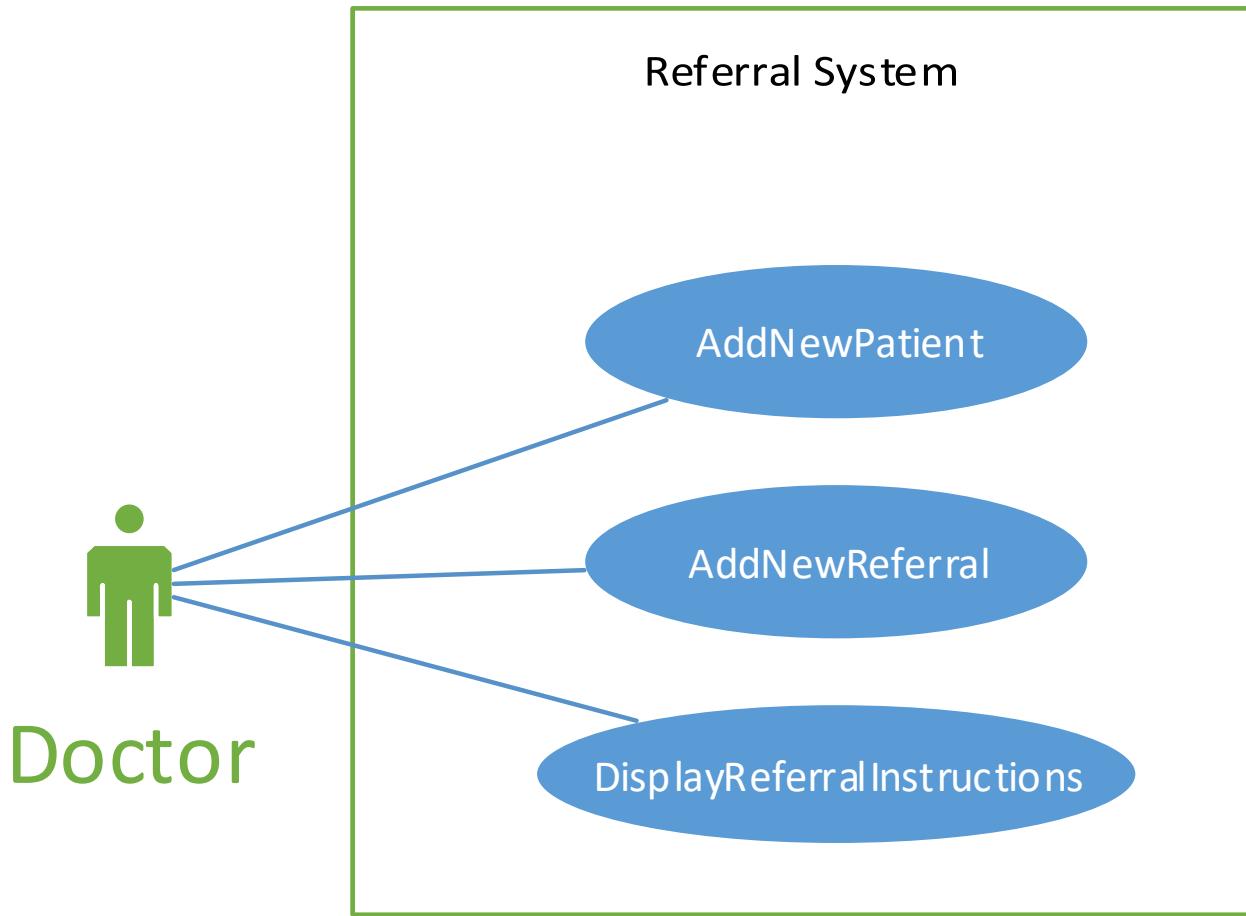
The following diagrams depict how the types of users interact with parts of the system. These users, called actors, are associated with the aforementioned use cases with arrows. Arrows with the 'Extends' description indicate a use case that extends the functionality of another. Below, Admins, Volunteers, and Donors interact with the Login System by initiating the use cases they benefit from. The Database System is mostly used by the IT Coordinator, who performs actions that initiates those use cases. Backing up and restoring data involves the Cloud, while storing and accessing data involves the database.

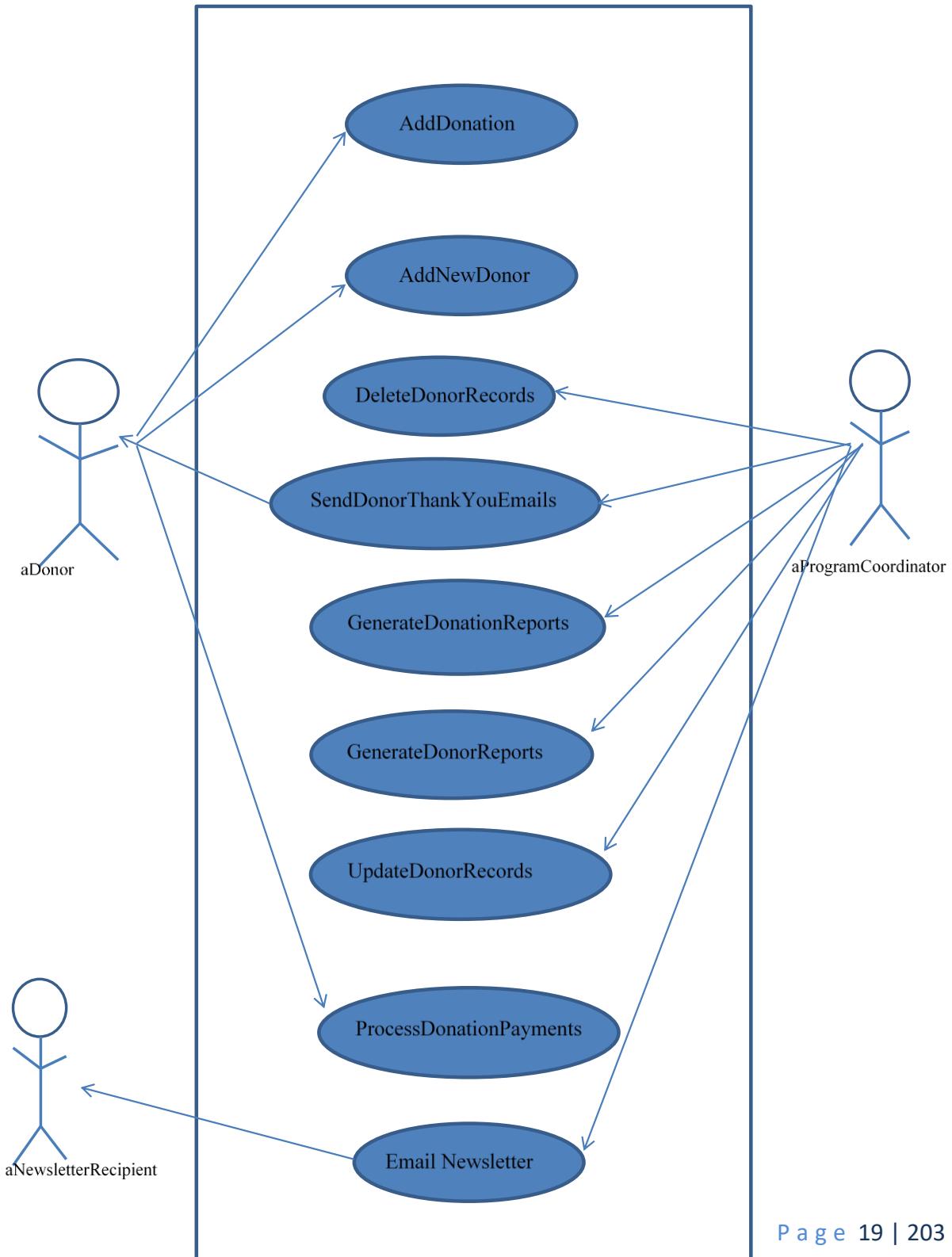


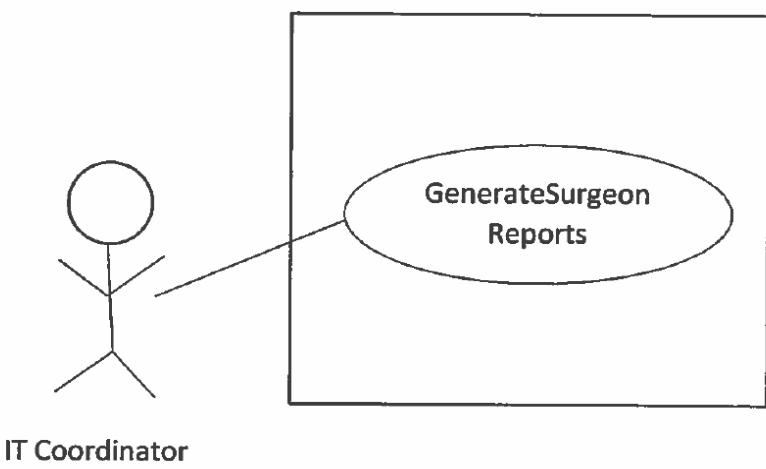
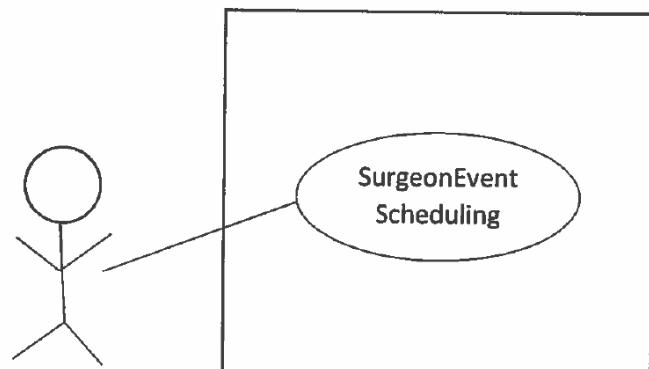
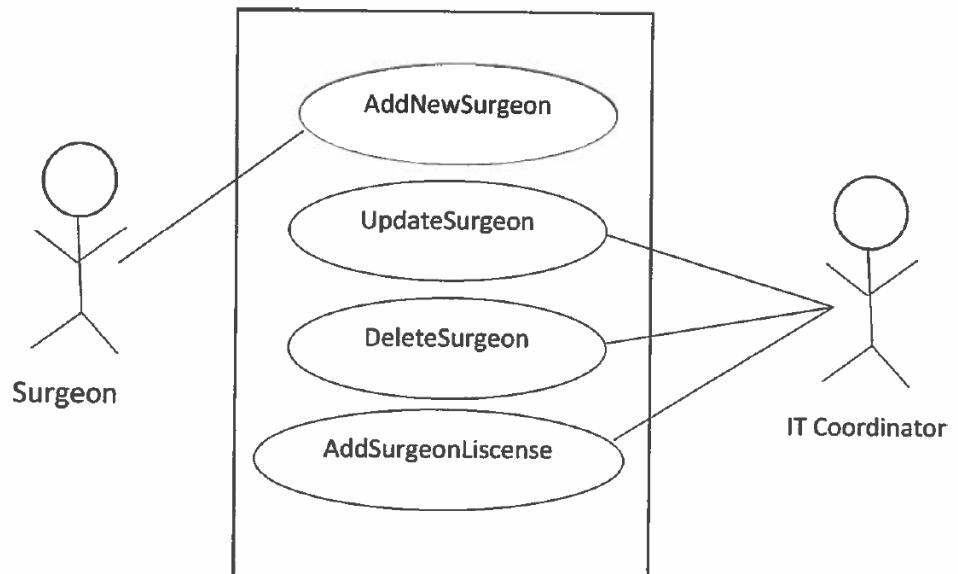












REQUIREMENTS TRACEABILITY MATRIX

The Requirements Traceability Matrix is a document that traces and maps user requirements with use cases. The purpose of the Requirements Traceability Matrix is to ensure that all requirements defined for a system are tested for. In the Requirements Traceability Matrix presented, the use cases are listed in the top row and listed by their use case number. All functional requirements are listed in the far-left column. Each requirement is matched to one or more use cases. A relationship between a functional requirement and a use case can be found by searching for the intersecting cell marked with an "X."

Functional System Requirements Traceability Matrix

System Requirements 01 -24

ID	Functional System Requirements	Risk	Use Cases																	
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
01	The new system shall allow users to insert donor records into the database	High	X																	
02	The new system shall allow users to update donor records	High		X																
03	The new system shall allow users to delete donor records	Low			X															
04	The new system shall automatically insert donor records when donations are submitted online	Medium	X																	
05	The new system shall utilize a user-friendly payment system for donations	Medium			X															
06	The new system shall allow for donations by card or bank account number	Low			X															
07	The new system shall automatically deposit new donations into SOS bank account	High				X														
08	The new system shall provide reports about donors	Medium					X													
09	The new system shall provide reports about donations	Medium						X												
10	The new system shall send automated emails thanking donors	Low							X											
11	The new system shall send newsletter emails to donors to keep them informed	Low								X										
12	The new system shall allow users to insert surgeon records into the database	High									X									
13	The new system shall allow users to update surgeon records	High										X								
14	The new system shall allow users to delete surgeon records	Low											X							
15	The new system shall use dynamic forms to capture additional information specific to surgeons	Medium										X								
16	The new system shall store information about surgeon licenses	Low											X							
17	The new system shall provide reports about surgeons	Medium												X						
18	The new system shall send automated emails to a surgeon scheduled for an event	Low													X					
19	The new system shall allow users to easily match surgeons to surgical events	High														X				
20	The new system shall send emails to thank surgeons	Low															X			
21	The new system shall allow users to insert nurse records into the database	High																X		
22	The new system shall allow users to update nurse records	High																	X	
23	The new system shall allow users to delete nurse records	Low																		X
24	The new system shall use dynamic forms to capture additional information specific to nurses	Medium																		X

System requirements 25-56

ID	Functional System Requirements	Risk	Use Cases															
			21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
25	The new system shall store information about nurse licenses (if applicable)	Low	X															
26	The new system shall provide reports about nurses	Medium		X														
27	The new system shall send automated emails to a nurse scheduled for an event	Low			X													
28	The new system shall allow users to easily match nurses to surgical events	High				X												
29	The new system shall send emails to thank nurses	Low					X											
30	The new system shall allow users to insert interpreter records into the database	High						X										
31	The new system shall allow users to update interpreter records	High							X									
32	The new system shall allow users to delete interpreter records	Low								X								
33	The new system shall use dynamic forms to capture additional information specific to interpreters	Medium								X								
34	The new system shall store information about interpreter licenses (if applicable)	Low									X							
35	The new system shall provide reports about interpreters	Medium										X						
36	The new system shall send automated emails to a interpreter scheduled for an event	Low											X					
37	The new system shall allow users to easily match interpreters to surgical events	High												X				
38	The new system shall send emails to thank clinical language interpreters	Low													X			
39	The new system shall allow users to insert non-clinical volunteer records into the database	High														X		
40	The new system shall allow users to update volunteer records	High															X	
41	The new system shall allow users to delete volunteer records	Low																X
42	The new system shall use dynamic forms to capture additional information specific to non-clinical volunteers	Medium																
43	The new system shall provide reports about volunteers	Medium																X
44	The new system shall send automated emails to a volunteer scheduled for an event	Low																X

ID	Functional System Requirements	Risk	Use Cases															
			39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	
45	The new system shall allow users to easily match volunteers to surgical events	High	X															
46	The new system shall send emails to thank volunteers	Low		X														
47	The new system shall allow users to insert event records into the database	High			X													
48	The new system shall allow users to update event records	High				X												
49	The new system shall allow users to delete event records	Low					X											
50	The new system shall provide reports about events	Medium						X										
51	The new system shall manage volunteer attendance for events	Medium							X									
52	The new system shall capture additional data about patients on the referral form	Medium								X								
53	The new system shall include instructions for getting access to referral form	Low																X
54	The new system shall use dynamic forms to capture additional information	Low									X							
55	The new system shall provide reports about referrals (if applicable)	Medium										X						
56	The new system shall email SOS coordinators when a new referral has been submitted	Low											X					

Nonfunctional System Requirements Traceability Matrix

System requirements 57-71

ID	Nonfunctional System Requirements	Risk	Use Cases															
			50	51	52	53	54	55	56	57	58	59	59	60	61	62	63	
57	The new system shall be available every day	High																X
58	The new system shall not crash if website traffic increases	High																
59	The new system's website shall be available to the public	High		X	X													
60	The new system shall be in place by the end of Spring 2018, unless otherwise stated	High																
61	The new system's data storage shall be available to SOS coordinators only	High																
62	The new system shall provide enough data storage	High																X
63	The new system shall use efficient database design and relationships	High																
64	The new system's data shall be backed up	High															X	
65	The new system shall be easy to maintain	High																
66	The new system shall be thoroughly tested and not break or have bugs	High																
67	The new system maintenance shall be available to the IT coordinator	High																X
68	The new system shall not handle sensitive data about patients	High																
69	The new system shall not violate HIPAA	High																
70	The new system shall not interface with the RedCap or Epic Connect patient database	High																
71	The new system shall allow for disclosure of all volunteers, per the Federal Tort Claims Act	High																

FUNCTIONAL USE CASES

Use Case ID	Use Cases	Primary Actor	Brief Description
1	AddNewDonor	Donor	This use case describes how a donor adds their information when making a donation
2	UpdateDonorRecords	IT Coordinator	This use case describes how a user makes changes to donor information
3	DeleteDonorsRecords	IT Coordinator	This use case describes how a user deletes donor information
4	AddDonation	Donor	This use case describes a donor makes a donation
5	ProcessDonationPayments	Program Coordinator	This use case describes how donation payments are processed
6	GenerateDonorReports	Program Coordinator	This use case describes how a user requests reports about donors
7	GenerateDonationReports	Program Coordinator	This use case describes how a user requests reports about donations
8	SendDonorThankYouEmails	Program Coordinator	This use case describes how thank-you emails for donors are emailed
9	EmailNewsletter	Program Coordinator	This use case describes how newsletters are emailed
10	AddNewSurgeon	Surgeon	This use case describes how a surgeon's information is added to the system
11	UpdateSurgeonRecord	IT Coordinator	This use case describes how a surgeon's information is updated to the system
12	DeleteSurgeonRecord	IT Coordinator	This use case describes how a surgeon's information is deleted from the system
13	AddSurgeonLicense	Program Coordinator	This use case describes how surgeon license information is added to the system
14	GenerateSurgeonReports	Program Coordinator	This use case describes how a user requests reports about surgeons
15	SendSurgeonEmails	Program Coordinator	This use case describes how emails are sent to surgeons
16	SurgeonEventScheduling	Program Coordinator	This use case describes how events with surgeons are scheduled
17	SendSurgeonThankYouEmails	Program Coordinator	This use case describes how thank-you emails for surgeons are emailed
18	AddNewNurse	Nurse	This use case describes how a nurse's information is added to the system
19	UpdateNurseRecord	IT Coordinator	This use case describes how a nurse's information is updated to the system
20	DeleteNurseRecord	IT Coordinator	This use case describes how a nurse's information is deleted from the system
21	AddNurseLicenses	Program Coordinator	This use case describes how nurse license information is added to the system
22	GenerateNurseReport	Program Coordinator	This use case describes how a user requests reports about nurses
23	SendEmailsToNurses	Program Coordinator	This use case describes how emails are sent to nurses
24	NurseEventScheduling	Program Coordinator	This use case describes how events with nurses are scheduled
25	SendNurseThankYouEmails	Program Coordinator	This use case describes how thank-you emails for nurses are emailed

Use Case ID	Use Cases	Primary Actor	Brief Description
26	AddNewInterpreter	Interpreter	This use case describes how an interpreter's information is added to the system
27	UpdateInterpreterRecord	IT Coordinator	This use case describes how an interpreter's information is updated to the system
28	DeleteInterpreterRecord	IT Coordinator	This use case describes how an interpreter's information is deleted from the system
29	AddInterpreterLicenses	Program Coordinator	This use case describes how an interpreter's license information is added to the system
30	GenerateInterpreterReport	Program Coordinator	This use case describes how a user requests reports about interpreters
31	SendEmailsToInterpreters	Program Coordinator	This use case describes how emails are sent to interpreters
32	InterpreterScheduling	Program Coordinator	This use case describes how events with interpreters are scheduled
33	SendInterpreterThankYouEmails	Program Coordinator	This use case describes how thank-you emails for interpreters are emailed
34	AddNewNonClinicalVolunteer	Volunteer	This use case describes how a non-clinical volunteer's information is added to the system
35	UpdateNonClinicalVolunteerRecord	IT Coordinator	This use case describes how a non-clinical volunteer's information is updated to the system
36	DeleteNonClinicalVolunteerRecord	IT Coordinator	This use case describes how a non-clinical volunteer's information is deleted from the system
37	GenerateVolunteerReports	Program Coordinator	This use case describes how a user requests reports about non-clinical volunteers
38	SendEmailsToVolunteers	Program Coordinator	This use case describes how emails are sent to non-clinical volunteers
39	GenerateVolunteerEventSchedule	Program Coordinator	This use case describes how events with non-clinical volunteers are scheduled
40	SendVolunteerThankYouEmails	Program Coordinator	This use case describes how thank-you emails for non-clinical volunteers are emailed
41	AddNewEvent	Program Coordinator	This use case describes how event information is added to the system
42	UpdateEventRecord	Program Coordinator	This use case describes how event information is updated to the system
43	DeleteEventRecord	Program Coordinator	This use case describes how event information is deleted from the system
44	GenerateEventReports	Program Coordinator	This use case describes how a user requests reports about events
45	AddAttendance	Program Coordinator	This use case describes how attendance information is added
46	AddNewPatient	Doctors	This use case describes how a new patient from referrals is added to the system
47	AddNewReferral	Doctors	This use case describes how referrals are added
48	GenerateReferralReports	Program Coordinator	This use case describes how a user requests reports about referrals
49	GenerateReferralNotifications	Program Coordinator	This use case describes how referral notifications are generated after a referral is submitted

50	LoginAdminPortal	Program Coordinator	This use case describes how SOS coordinators login to their SOS accounts on the admin portal
51	LoginVolunteers	Volunteers	This use case describes how volunteers login to their SOS accounts on the website
52	LoginDonors	Donors	This use case describes how donors login to their SOS accounts on the website
53	DisplayReferralInstructions	Doctors	This use case describes how doctors obtain a referral code

NONFUNCTIONAL USE CASES

Use Case ID	Use Cases	Primary Actor	Brief Description
55	RecoverDatabaseRecords	IT Coordinator	This use case describes how SOS data is recovered in case of data loss
56	StoreDatabaseRecords	IT Coordinator	This use case describes how the SOS database will have enough memory to store everything
57	AccessDatabaseRecords	IT Coordinator	This use case describes how the IT Coordinator will access the database directly if needed

USE CASES AND SEQUENCE DIAGRAMS

SOS Louisville	Version: 1.0
Use Case Specification: AddNewDonor	Date: 10/23/2017
Use Case ID: 1	

Use Case Specification: AddNewDonor

AddNewDonor

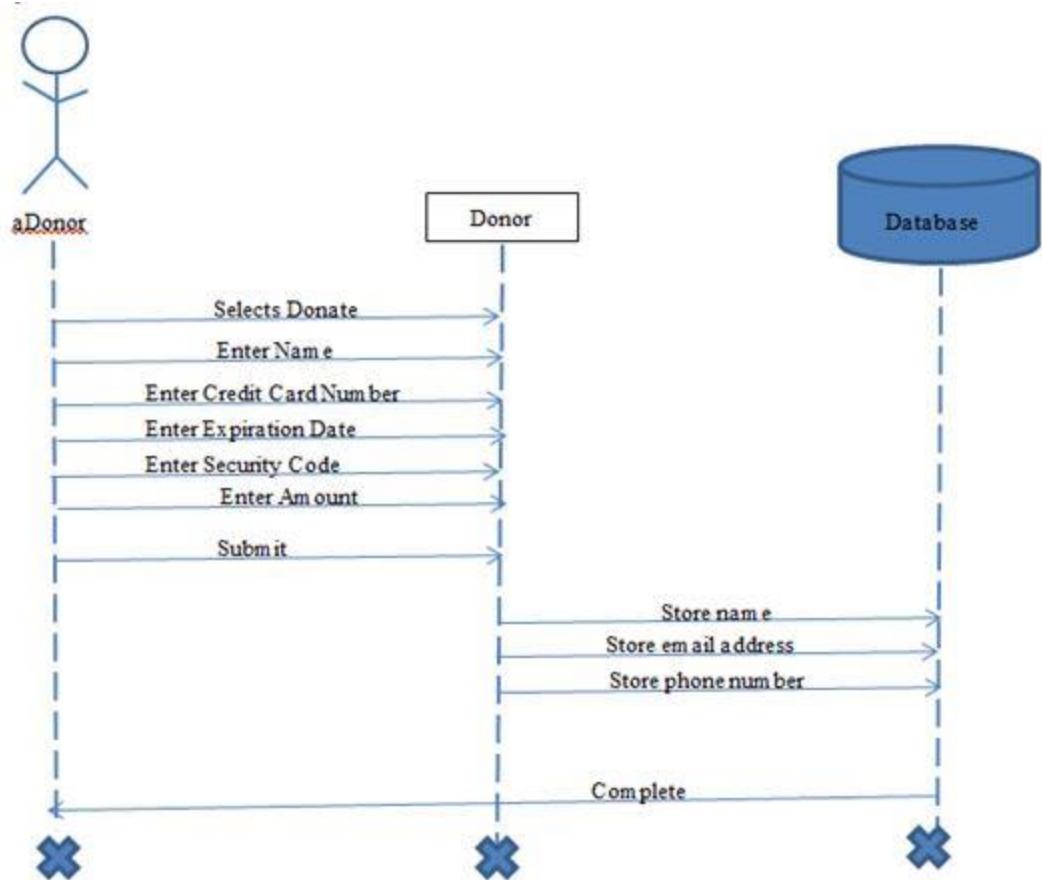
Brief Description

This use case describes how a donor adds their information when making a donation. The system will collect additional information outside of what the payment system requires to give SOS the ability to retrieve specific donor details.

Flow of Events

Basic Flow

1. The use case starts when the donor selects “Donate” on homepage.
2. The donor enters name
3. The donor enters phone number
4. The donor enters email address
5. The donor enters credit card number
6. The donor enters expiration date
7. The donor enters security code
8. The donor enters dollar amount of donation.
9. If donor does not exist in the system
 - The system creates new record with donor name, phone number, email address, and Donor ID
10. If donor does exist in system
 - Trigger AddDonation use case



SOS Louisville	Version: 1.0
Use Case Specification: UpdateDonorRecords	Date: 10/23/2017
Use Case ID: 2	

Use Case Specification: UpdateDonorRecords

UpdateDonorRecords

Brief Description

This use case describes how a user modifies donor information from the master Donor table.

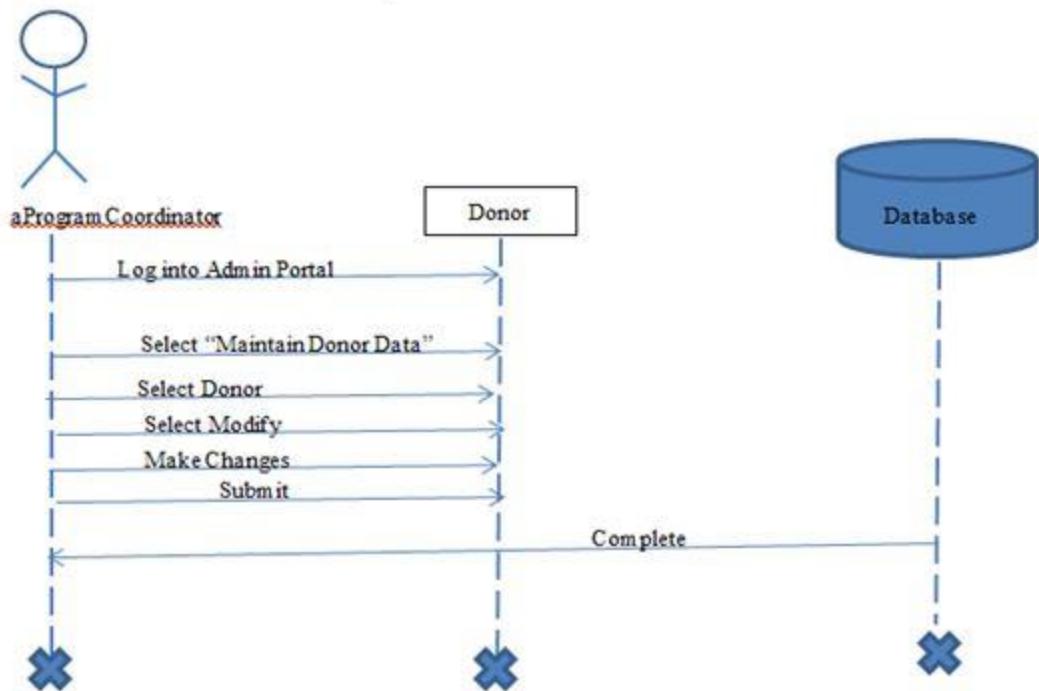
Flow of Events

Basic Flow

1. The use case when the IT Coordinator logins into the Admin portal
2. Selects “Maintain donor data”
3. Selects donor from list of current donors
4. Selects “Modify”
5. Makes changes where applicable

Pre-conditions

Donor must exist in system to be modified



SOS Louisville	Version: 1.0
Use Case Specification: DeleteDonorRecords	Date: 10/23/2017
Use Case ID: 3	

Use Case Specification: DeleteDonorRecords

DeleteDonorRecords

Brief Description

This use case describes how a user deletes donor information from the master Donor table.

Flow of Events

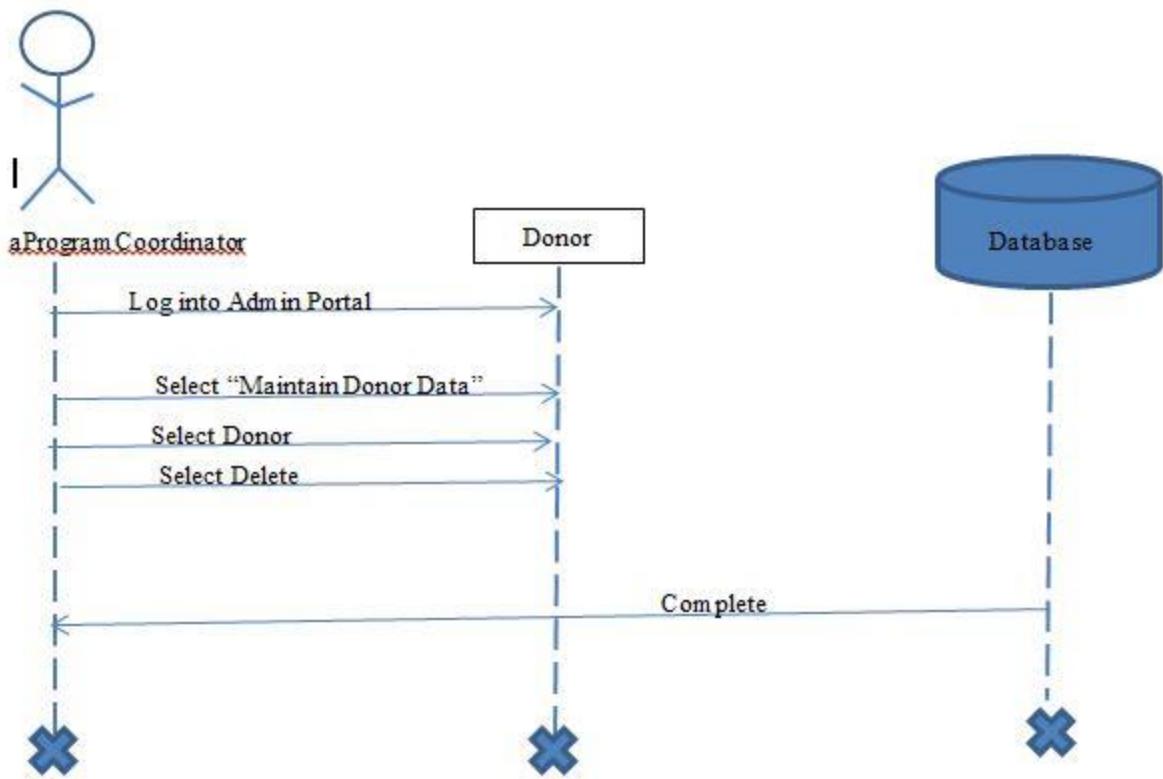
Basic Flow

1. The use case when the IT Coordinator logins into the Admin portal
2. Selects “Maintain donor data”
3. Selects donor from list of current donors
4. Selects “Delete”

Pre-conditions

Donor must exist in system to be deleted

IT Coordinator must have access to admin portal.



SOS Louisville	Version: 1.0
Use Case Specification: AddDonation	Date: 10/23/2017
Use Case ID: 4	

Use Case Specification: AddDonation

AddDonation

Brief Description

This use case describes how a donor makes a donation. The donor's credit card information will be captured securely for processing through Braintree.

Flow of Events

Basic Flow

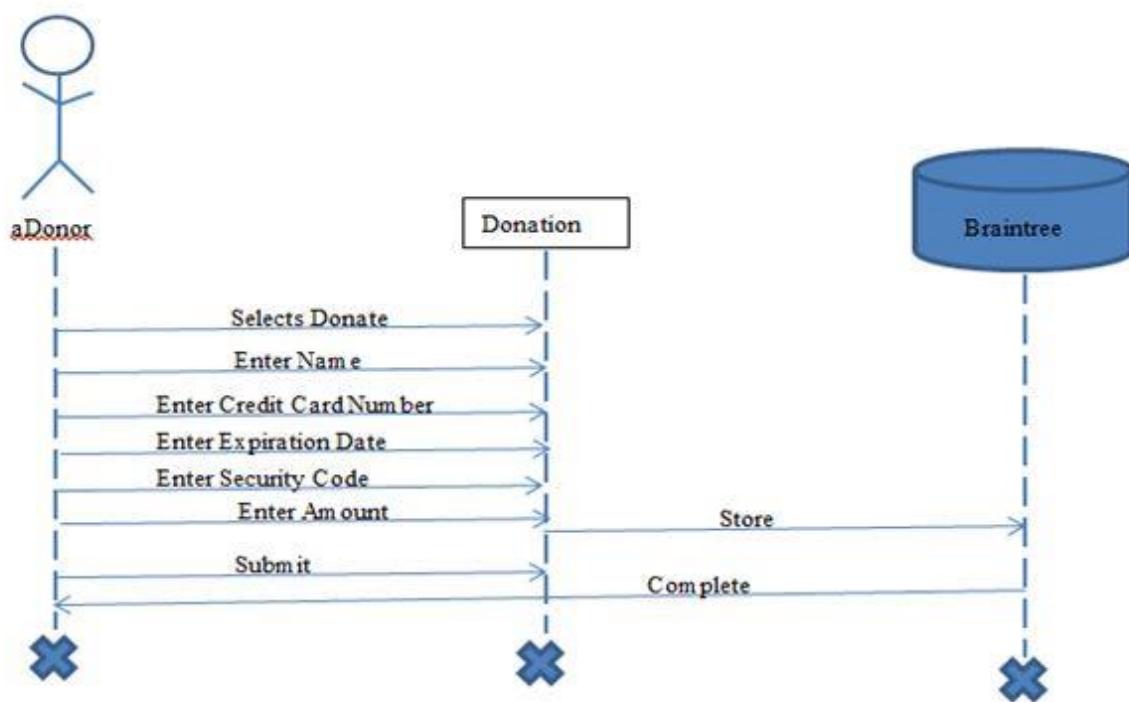
1. The use case starts when the donor selects "Donate" on homepage.
2. The donor enters name
3. The donor enters credit card number
4. The donor enters expiration date
5. The donor enters security code
6. The donor enters dollar amount of donation
7. The Braintree system stores sensitive data on servers.
8. If donor already exists in the system
 - 8.1 The system creates new record with donation amount, date, and donor ID.
9. If donor does not exist in system
 - 9.1 Trigger AddDonor use case

Special Requirements

Braintree must be synchronized for security purposes so donor has confidence that credit card information is being transmitted securely through the website

Pre-conditions

Donor must have valid credit card information.



SOS Louisville	Version: 1.0
Use Case Specification: ProcessDonationPayments	Date: 10/23/2017
Use Case ID: 5	

Use Case Specification: ProcessDonationPayments

ProcessDonationPayments

Brief Description

This use case describes how donation payments are processed by Braintree and deposited into the SOS bank account.

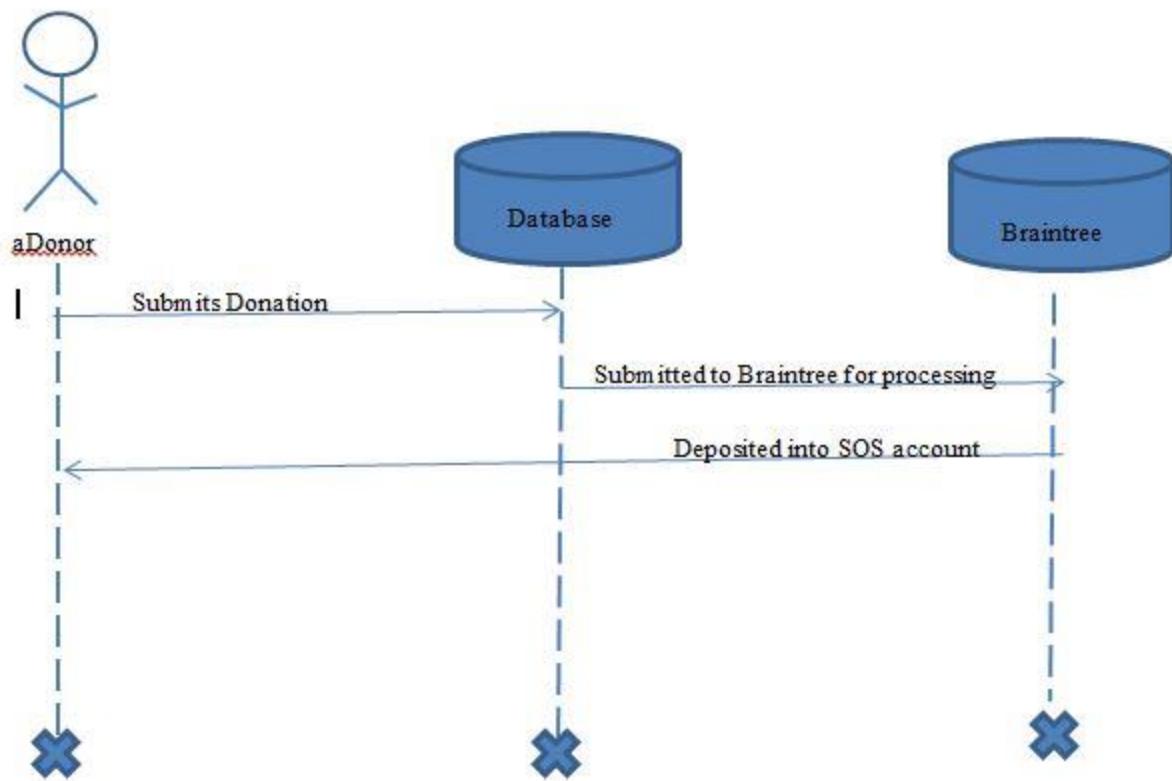
Flow of Events

Basic Flow

1. The use case starts when a donor submits a donation
2. Braintree collects payment information
3. Braintree validates payment information to determine adequate funds exist
4. If successful
 - 4.1 Braintree deposits net donation amount after fees
5. If unsuccessful
 - 5.1 The system generates email to donor indicating payment was invalid

Pre-conditions

Braintree must have the associated bank account information before donations can be submitted



SOS Louisville	Version: 1.0
Use Case Specification: GenerateDonorReports	Date: 10/23/2017
Use Case ID: 6	

Use Case Specification: GenerateDonorReports

GenerateDonorReports

Brief Description

This use case describes how the Program Coordinator requests reports about donors.

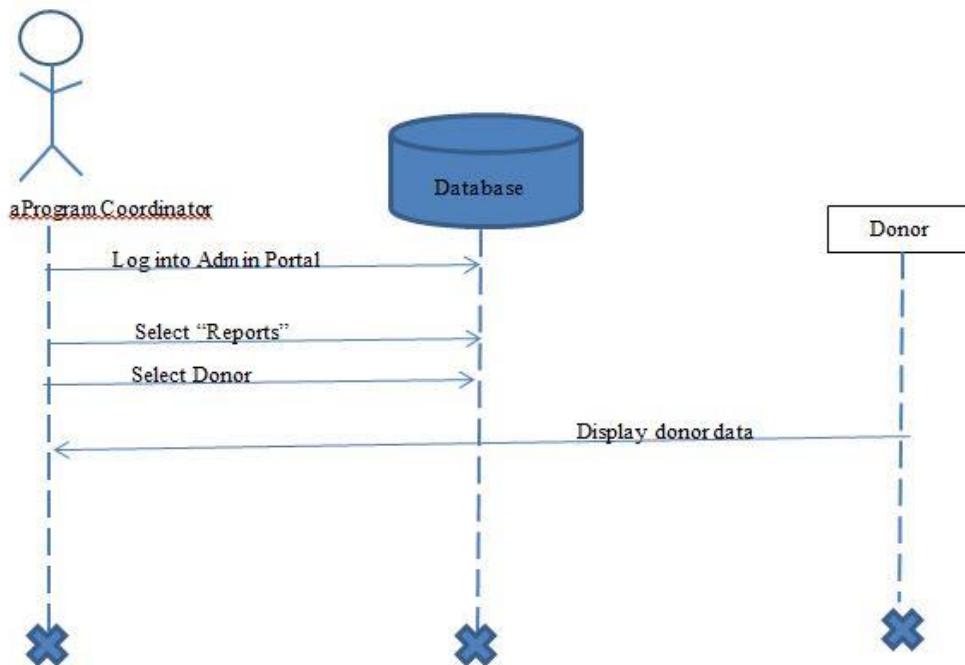
Flow of Events

Basic Flow

1. The use case starts when the Program Coordinator (PC) logs into the Admin portal
2. The PC will select “Reports” from the menu options on the homepage
3. PC selects “Donor” from drop down menu adjacent to “Show Me...” text

Pre-conditions

Program Coordinator must have access to admin portal



SOS Louisville	Version: 1.0
Use Case Specification: GenerateDonationReports	Date: 10/23/2017
Use Case ID: 7	

Use Case Specification: GenerateDonationReports

GenerateDonationReports

Brief Description

This use case describes how the Program Coordinator requests reports about donations.

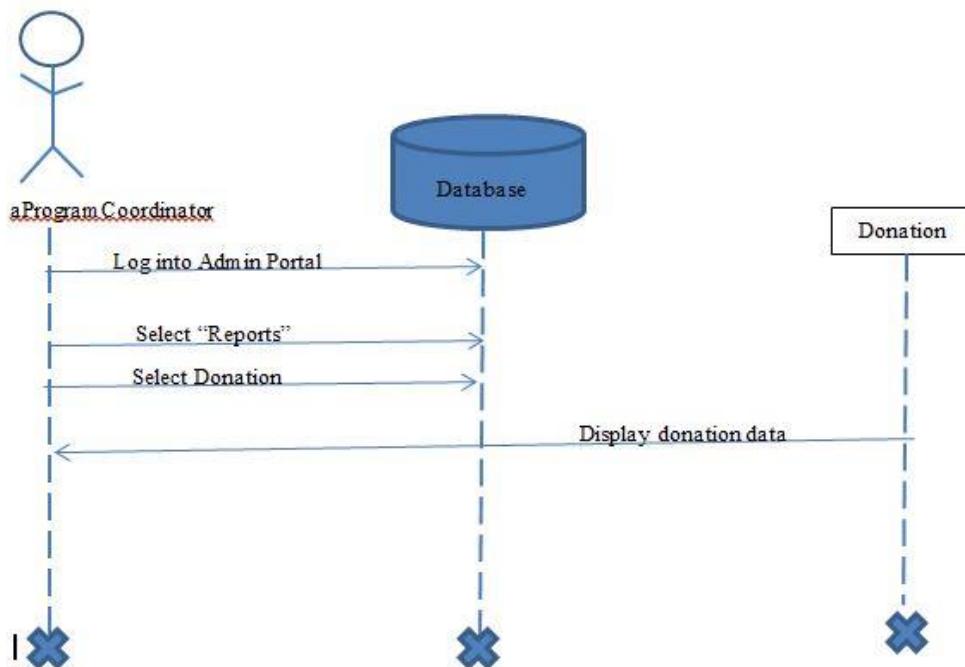
Flow of Events

Basic Flow

1. The use case starts when the Program Coordinator (PC) logins into the Admin portal
2. The PC will select “Reports” from the menu options on the homepage
3. PC selects “Donations” from drop down menu adjacent to “Show Me...” text.

Pre-conditions

Program Coordinator must have access to admin portal



SOS Louisville	Version: 1.0
Use Case Specification: SendDonorThankYouEmails	Date: 10/23/2017
Use Case ID: 8	

Use Case Specification: SendDonorThankYouEmails

SendDonorThankYouEmails

Brief Description

This use case describes how thank-you emails for donors are emailed.

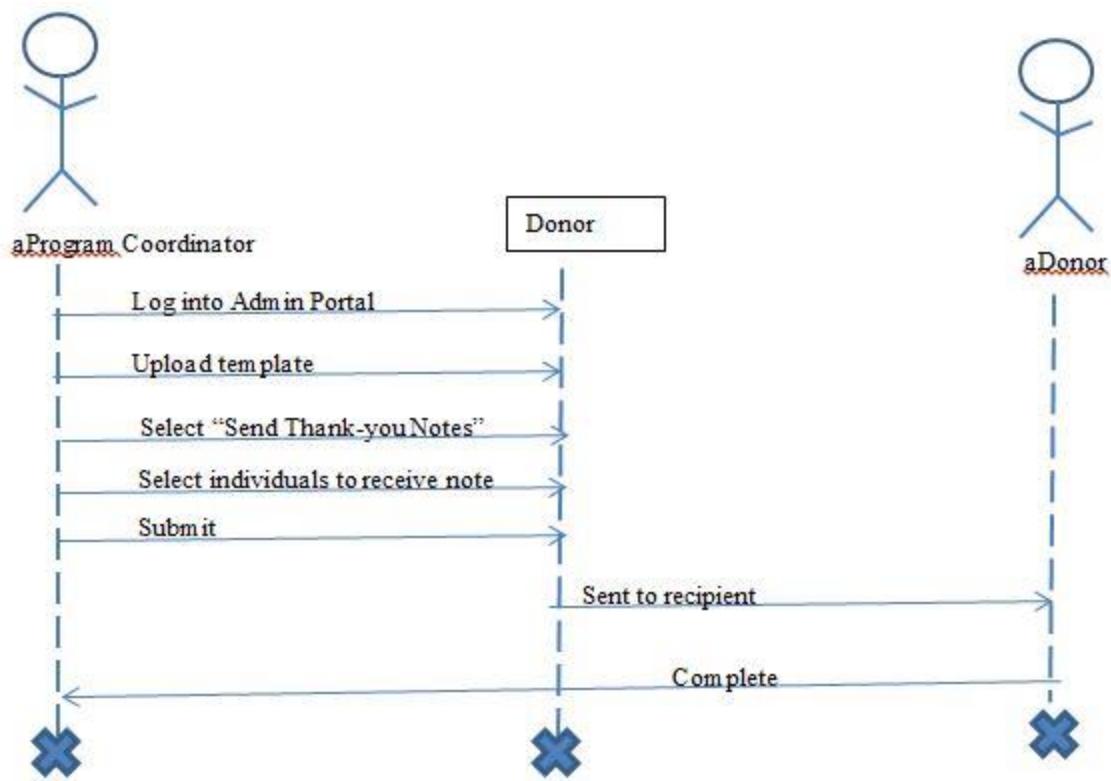
Flow of Events

Basic Flow

1. The use case starts when the Program Coordinator (PC) logs into the admin portal.
2. The thank you letter template will be uploaded to the portal to be reused.
3. The PC can hit the “Send Thank-You Notes” Button
4. A list of names and emails organized by volunteer or medical staff type will be displayed for selection
5. PC can select individual to be emailed
6. Select “Done” to send the emails formatted with a greeting by name

Pre-conditions

Program Coordinator must have access to admin portal



SOS Louisville	Version: 1.0
Use Case Specification: EmailNewsletter	Date: 10/23/2017
Use Case ID: 9	

Use Case Specification: EmailNewsletter

EmailNewsletter

Brief Description

This use case describes how newsletters are emailed. Each month SOS will send a newsletter to donors, volunteers, and medical staff to keep them up to date on the organizations efforts.

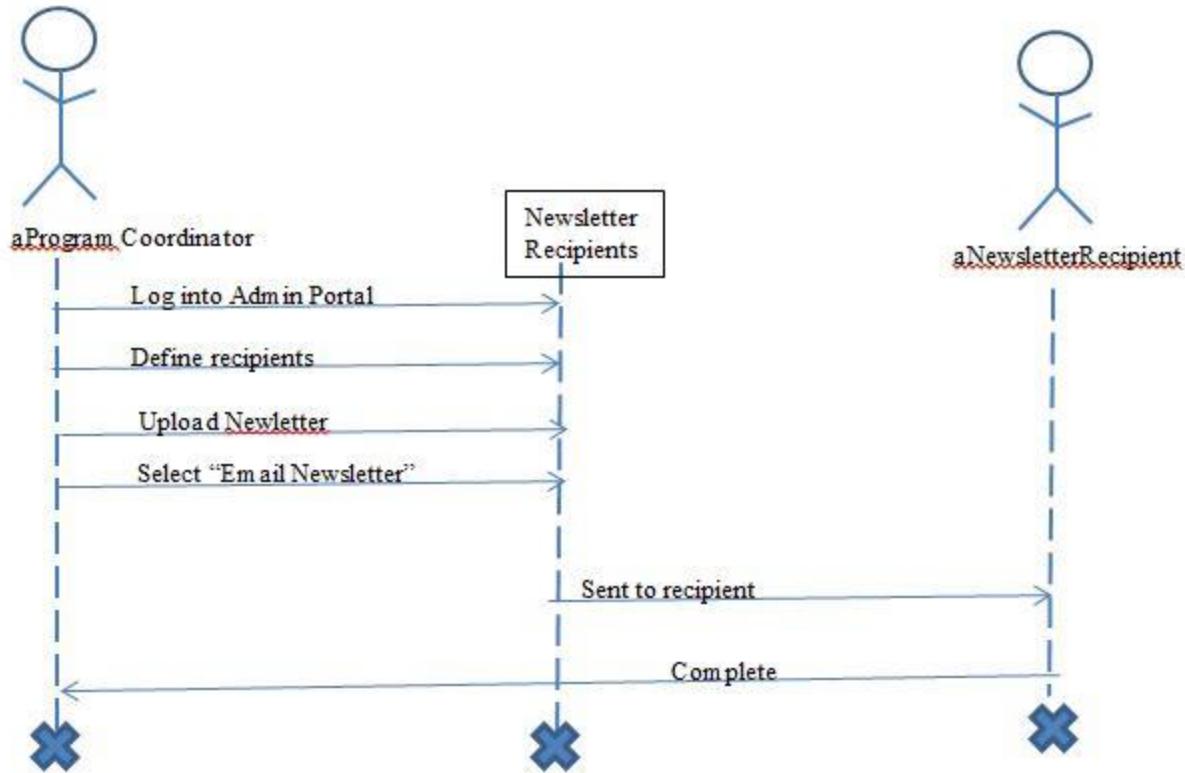
Flow of Events

Basic Flow

1. The use case starts when the Program Coordinator (PC) creates the newsletter
2. PC will upload the newsletter to the admin portal, which stores only the most recent uploaded file
3. PC will hit “Email Newsletter” button that will send the uploaded newsletter to predefined email list of recipients.

Pre-conditions

Program Coordinator must define email list of newsletter recipients prior to sending



Surgery on Sunday	Version: 1.0
Use Case Specification: AddNewSurgeon	Date: 23/Oct/17
Use Case ID: 10	

Use Case Specification: AddNewSurgeon

AddNewSurgeon

Brief Description

This use case describes how a surgeon's information is added to the system. A surgeon wishes to be added to the system, so they will go to the company website and bring up a dynamic form to fill out the required information about the doctor such as; name, address, phone, licenses, what type of surgeon, etc. They will then submit the form into the system.

Flow of Events

Basic Flow

1. The use case begins when the surgeon selects “Get Involved”
2. The surgeon will then select “physician volunteer form”
3. The surgeon will enter their first name
4. The surgeon will enter their last name
5. The surgeon will enter their street address
6. The surgeon will enter their city
7. The surgeon will enter their state
8. The surgeon will enter their zip code
9. The surgeon will enter their first contact number
10. The surgeon will enter a second contact number
11. The surgeon will enter their specialized field
12. The surgeon will enter their up-to-date licenses
13. The surgeon will enter their e-mail address
14. The surgeon will enter their hospital affiliation
15. The surgeon will enter their primary language
16. The surgeon will enter their secondary language, if applicable
17. If the surgeon is new to the system
 1. Trigger AddNewSurgeon use case and use case ends

18. If the surgeon already exists in the system

1. Trigger UpdateSurgeonRecord use case

Special Requirements

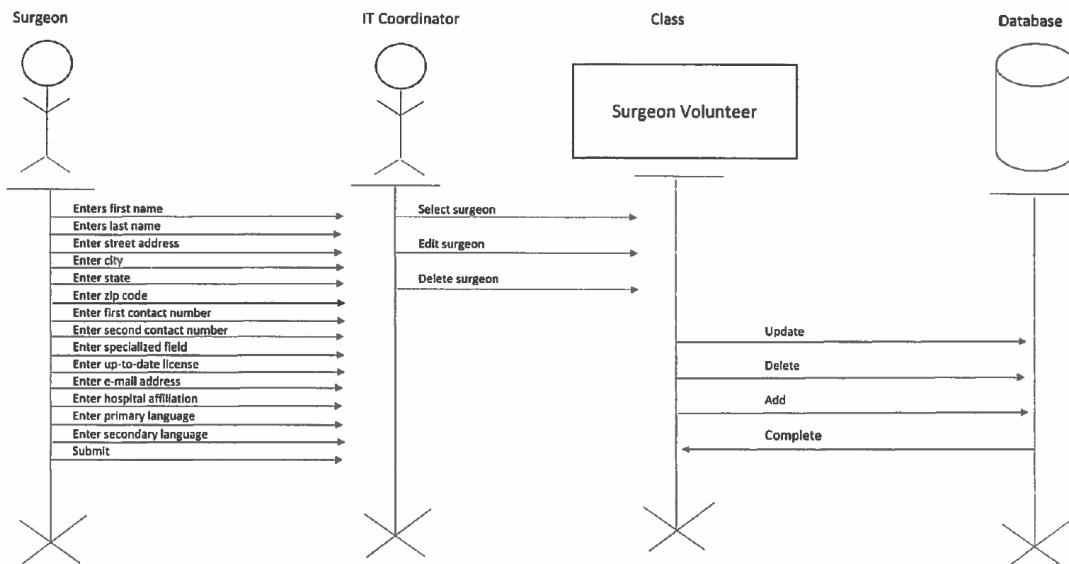
MySQL database

This must be in place in order for the information to be saved and stored for future use.

Pre-conditions

The surgeon must have applicable qualifications.

Sequence Diagram



Surgery on Sunday	Version: 1.0
Use Case Specification: UpdateSurgeonRecord	Date: 23/Oct/2017
Use Case ID: 11	

Use Case Specification: UpdateSurgeonRecord

UpdateSurgeonRecord

Brief Description

This use case describes how a surgeon's information is updated to the system.

Flow of Events

Basic Flow

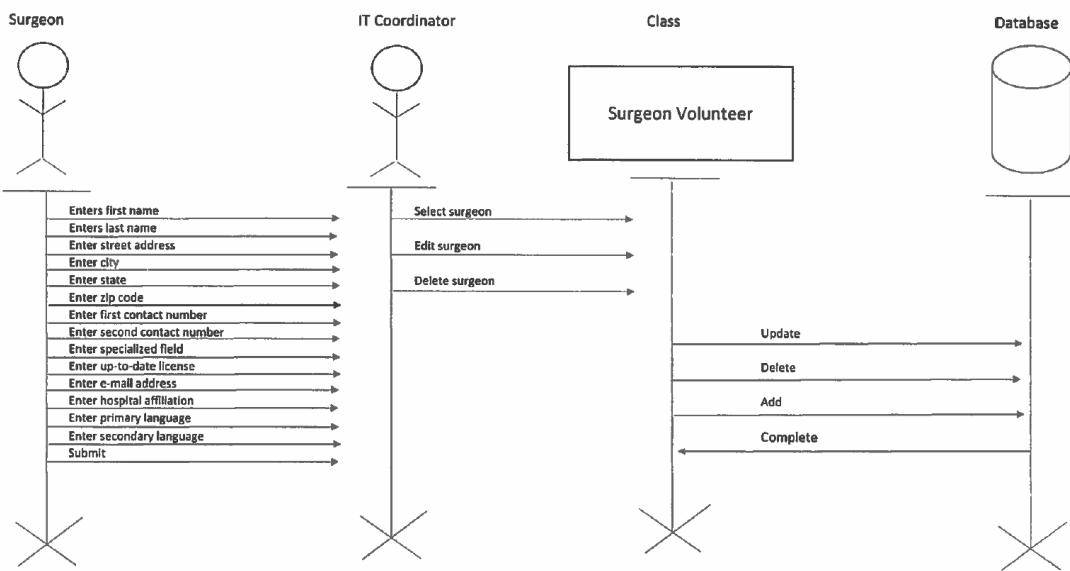
1. The use case starts when the IT Coordinator logs into the Administration portal
2. The IT Coordinator then selects “Edit Surgeon Data”
3. The IT Coordinator then selects a surgeon from the current list of surgeons
4. The IT Coordinator selects “Edit”
4. The IT Coordinator will edit all the available fields by adding, deleting, or updating information for that surgeon.

Pre-conditions

The surgeon must exist in the current system to be updated by the IT Coordinator

The IT Coordinator must be able to access the administration portal

Sequence Diagram



Surgery on Sunday	Version: 1.0
Use Case Specification: DeleteSurgeonRecord	Date: 23/Oct/2017
Use Case ID: 12	

Use Case Specification: DeleteSurgeonRecord

DeleteSurgeonRecord

Brief Description

This use case describes how a surgeon's information is removed from the system.

Flow of Events

Basic Flow

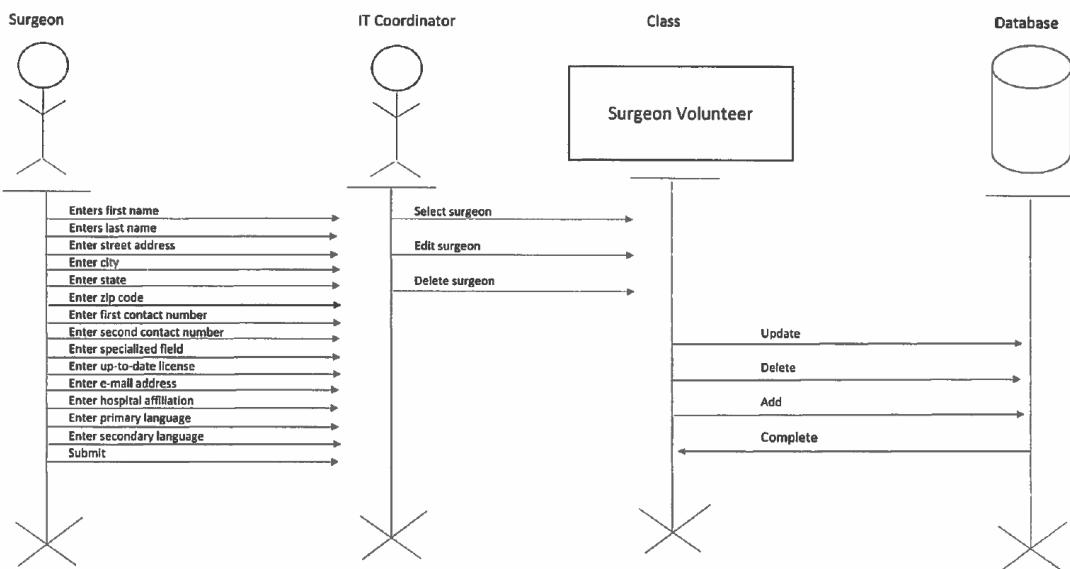
1. The use case starts when the IT Coordinator logs into the Administration portal
2. The IT Coordinator then selects “Edit Surgeon Data”
3. The IT Coordinator then selects a surgeon from the current list of surgeons
4. The IT Coordinator selects “Delete”

Pre-conditions

The surgeon must exist in the current system to be updated by the IT Coordinator

The IT Coordinator must be able to access the administration portal

Sequence Diagram



Surgery on Sunday	Version: 1.0
Use Case Specification: AddSurgeonLicense	Date: 23/Oct/2017
Use Case ID: 13	

Use Case Specification: AddSurgeonLicense

AddSurgeonLicense

Brief Description

This use case describes how surgeon license information is added to the system

Flow of Events

Basic Flow

1. The use case starts when the Program Coordinator logs into the Administration portal
2. The Program Coordinator then selects “Edit Surgeon Data”
3. The IT Coordinator then selects a surgeon from the current list of surgeons
4. The IT Coordinator selects “Edit”
5. The Program Coordinator adds the information for the new license

Pre-conditions

The surgeon must exist in the current system to be updated by the IT Coordinator

The Program Coordinator must be able to access the administration portal

The surgeon must have a valid medical license and proper documentation

Surgery on Sunday	Version: 1.0
Use Case Specification: GenerateSurgeonReports	Date: 23/Oct/2017
Use Case ID: 14	

Use Case Specification: GenerateSurgeonReports

GenerateSurgeonReports

Brief Description

This use case describes how a user requests reports about surgeons

Flow of Events

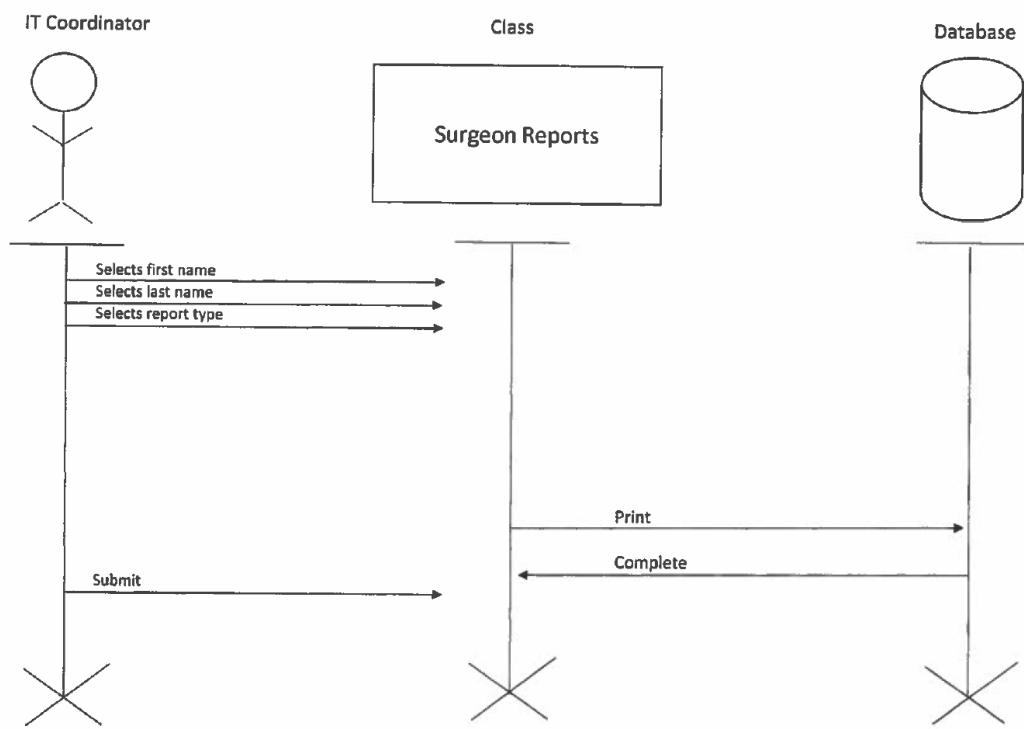
Basic Flow

1. The use case starts when the Program Coordinator logins into the administration portal
2. The Program Coordinator will select “Reports” from the menu options on the homepage
3. The drop box menu adjacent to “Show Me....” will contain additional options to determine the nature of the report where the program coordinator can select “Surgeons”.

Pre-conditions

The program coordinator must have access to administration portal

Sequence Diagram



Surgery on Sunday	Version: 1.0
Use Case Specification: SendSurgeonEmails	Date: 23/Oct/2017
Use Case ID: 15	

Use Case Specification: SendSurgeonEmails

SendSurgeonEmails

Brief Description

This use case describes how emails are sent to surgeons.

Flow of Events

Basic Flow

1. The use case starts when the program coordinator logs in to the admiration portal
2. The program coordinator will select “Email” on the homepage
3. A drop box will appear, and the program coordinator will select “Email Surgeon”
4. A list of surgeon names and corresponding emails will be displayed for selection
5. The program coordinator can then select which surgeon(s) to be emailed
6. Select “Done” to send the emails formatted with a greeting by surgeon name

Special Requirements

MailMunch must be added to the system in order to generate the emails

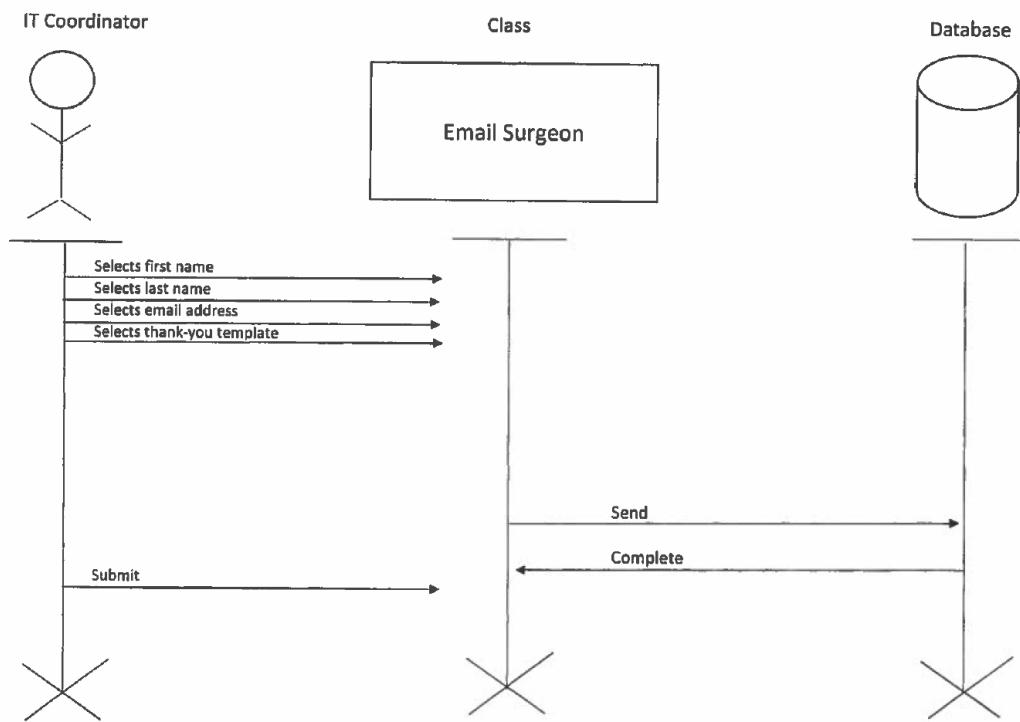
Pre-conditions

The program coordinator must have access to administration portal

The surgeon must be in the system

The surgeon must have an up-to-date email address in the system

Sequence Diagram



Surgery on Sunday	Version: 1.0
Use Case Specification: SurgeonEventScheduling	Date: 23/Oct/2017
Use Case ID: 16	

Use Case Specification: SurgeonEventScheduling

SurgeonEventScheduling

Brief Description

This use case describes how events with surgeons are scheduled.

Flow of Events

Basic Flow

1. The use case starts when the program coordinator logins to the admiration portal
2. The program coordinator will select “Scheduling” from the home page
3. A drop box will appear, and the program coordinator will select “Schedule Surgeon”
4. A dynamic form will then appear for the program coordinator to fill out
5. The program coordinator will select the type of surgeon needed
6. The program coordinator will fill out the type of surgery that is going to be performed
7. The program coordinator will fill out the date the event will be held
8. The system will then retrieve the data from the database
9. The system will provide a list of relevant surgeons available to be scheduled
10. The program coordinator will select a/or surgeon(s) to be scheduled
11. The program coordinator selects “Done”
 - a. This triggers the SendSurgeonEmails use case

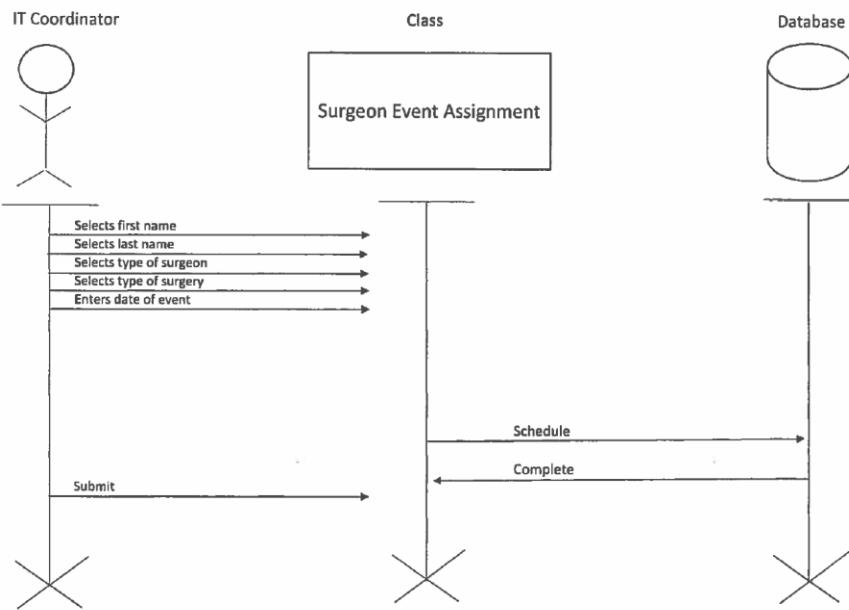
Pre-conditions

The program coordinator must have access to administration portal

The surgeon must be in the system

The surgeon must have an up-to-date email address in the system

Sequence Diagram



Surgery on Sunday	Version: 1.0
Use Case Specification: SendSurgeonThankYouEmails	Date: 23/Oct/2017
Use Case ID: 17	

Use Case Specification: SendSurgeonThankYouEmails

SendSurgeonThankYouEmails

Brief Description

This use case describes how thank-you emails for surgeons are emailed.

Flow of Events

Basic Flow

1. The use case starts when the program coordinator logs in to the admiration portal
2. The program coordinator will select "Email" from the home page
3. A drop box will appear, and the program coordinator will select "Email Surgeon"
4. The program coordinator can select from a thank-you letter template
5. A list of surgeon names and corresponding emails will be displayed for selection
6. The program coordinator can then select which surgeon(s) to be emailed
7. Select "Done" to send the emails formatted with a greeting by surgeon name

Special Requirements

MailMunch must be added to the system in order to generate the emails

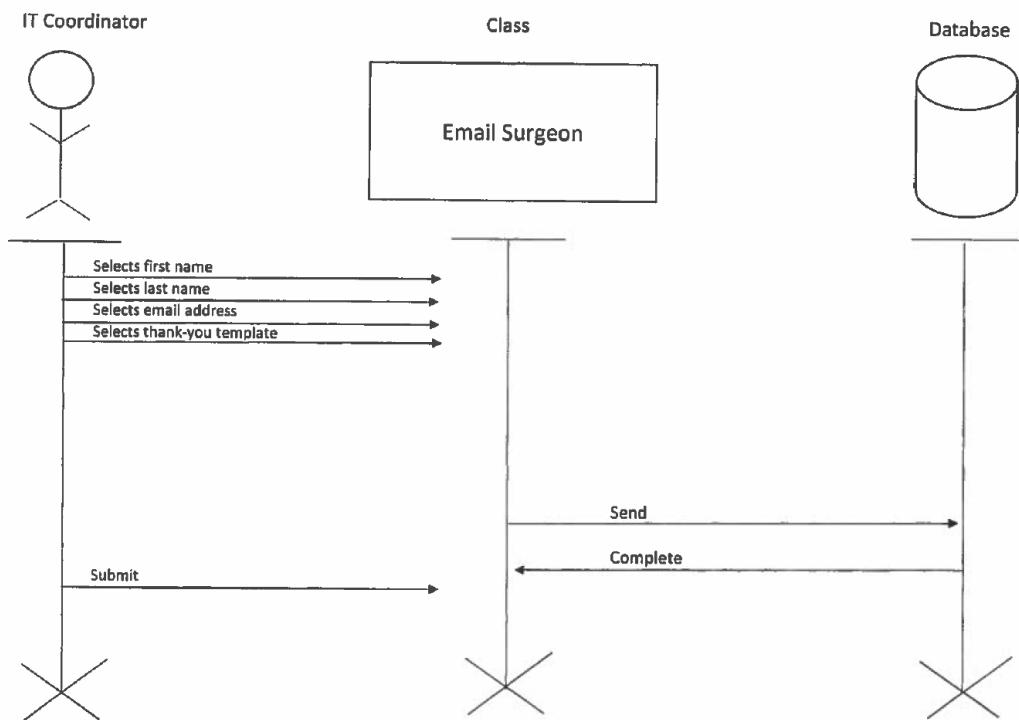
Pre-conditions

The program coordinator must have access to administration portal

The surgeon must be in the system

The surgeon must have an up-to-date email address in the system

Sequence Diagram



Surgery on Sunday	Version: 1.0
Use Case Specification: AddNewNurse	Date: 23/Oct/17
Use Case ID: 18	

Use Case Specification: AddNewNurse

AddNewNurse

Brief Description

This use case describes how a nurse's information is added to the system.

Flow of Events

Basic Flow

1. The use case begins when the Nurse selects “Get Involved”
2. The nurse will then select “physician volunteer form”
3. The nurse will enter their first name
4. The nurse will enter their last name
5. The nurse will enter their street address
6. The nurse will enter their city
7. The nurse will enter their state
8. The nurse will enter their zip code
9. The nurse will enter their first contact number
10. The nurse will enter a second contact number
11. The nurse will enter their specialized field
12. The nurse will enter their up-to-date licenses
13. The nurse will enter their e-mail address
14. The nurse will enter their hospital affiliation
15. The nurse will enter their primary language
16. The nurse will enter their secondary language, if applicable
17. If the Nurse is new to the system
 1. Trigger AddNewNurse use case and use case ends
18. If the nurse already exists in the system
 1. Trigger UpdateNurseRecord use case

Special Requirements

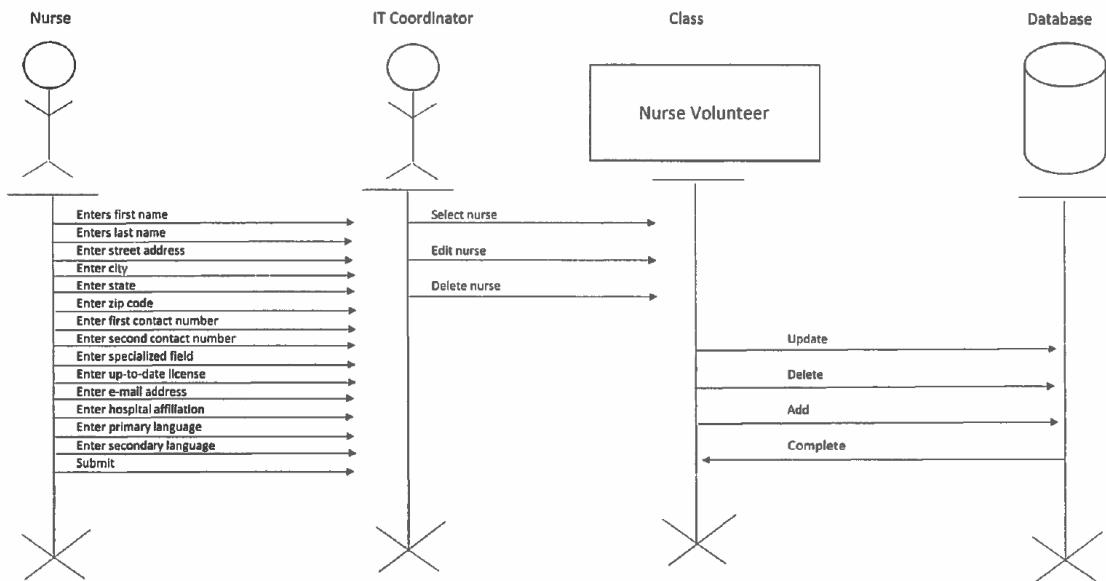
MySQL database

This must be in place in order for the information to be saved and stored for future use.

Pre-conditions

The nurse must have applicable qualifications.

Sequence Diagram



Surgery on Sunday	Version: 1.0
Use Case Specification: UpdateNurseRecord	Date: 23/Oct/2017
Use Case ID: 19	

Use Case Specification: UpdateNurseRecord

UpdateNurseRecord

Brief Description

This use case describes how a nurse's information is updated to the system.

Flow of Events

Basic Flow

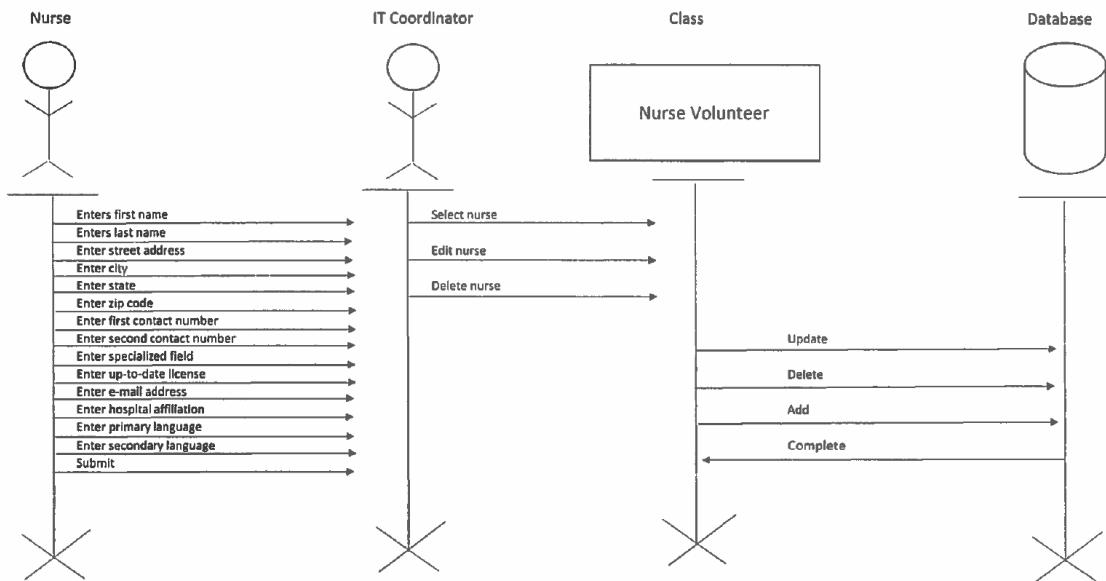
1. The use case starts when the IT Coordinator logs into the Administration portal
2. The IT Coordinator then selects “Edit Nurse Data”
3. The IT Coordinator then selects a nurse from the current list of nurses
4. The IT Coordinator selects “Edit”
4. The IT Coordinator will edit all the available fields by adding, deleting, or updating information for that nurse.

Pre-conditions

The nurse must exist in the current system to be updated by the IT Coordinator

The IT Coordinator must be able to access the administration portal

Sequence Diagram



SOS Louisville	Version: 1.0
Use Case Specification: DeleteNurseRecord	Date: 10/22/2017
Use Case ID: 20	

Use Case Specification: DeleteNurseRecord

DeleteNurseRecord

Brief Description

This use case describes how a nurse's information is deleted from the system.

Flow of Events

Basic Flow

1. The use case starts when the IT Coordinator logs into the admin portal.
2. The IT Coordinator selects Maintain Nurse Records from the menu options on the home page.
3. The IT Coordinator selects the nurse to be removed from a list of nurses.
4. The IT Coordinator selects Delete.
5. The selected nurse's record is deleted from the system.

Alternative Flows

If the process is canceled

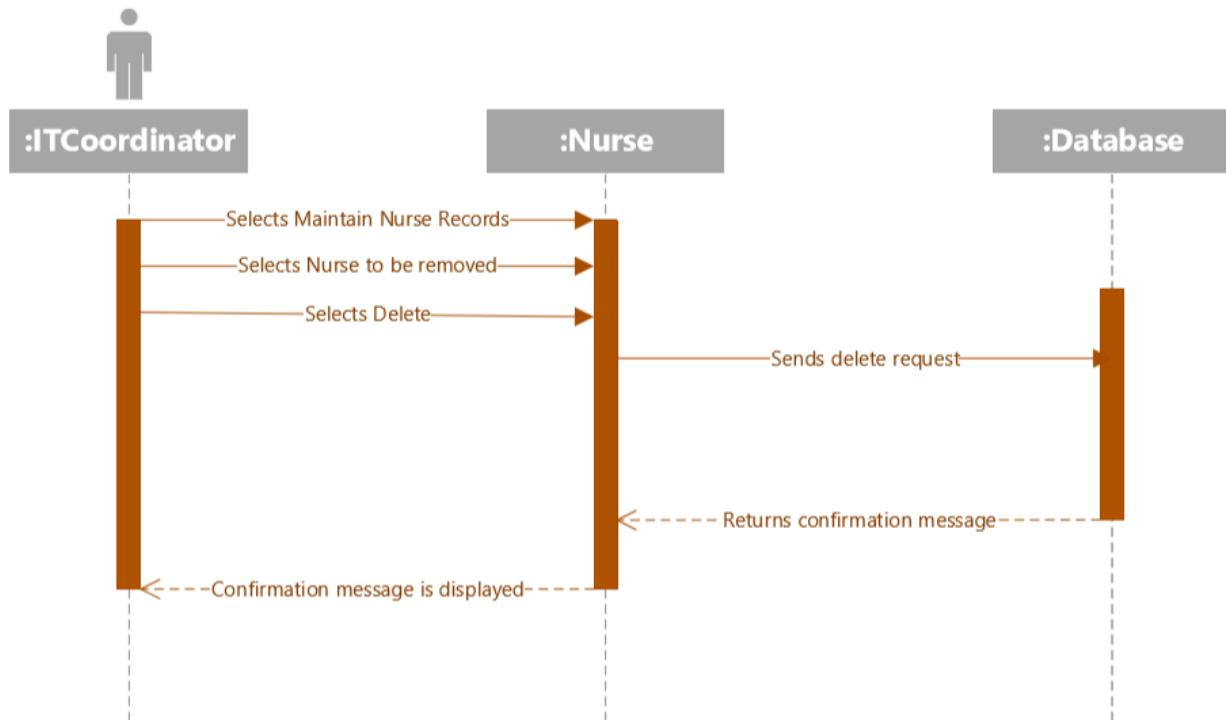
The process is terminated, and the system deletes no information.

Pre-conditions

The IT Coordinator must have access to the admin portal

The nurse being deleted must already exist in the system

1. The use case starts when the IT Coordinator logs into the admin portal.
2. The IT Coordinator selects Maintain Nurse Records from the menu options on the home page.
3. The IT Coordinator selects the nurse to be removed from a list of nurses.
4. The IT Coordinator selects Delete.
5. The selected nurse's record is deleted from the system.



SOS Louisville	Version: 1.0
Use Case Specification: AddNurseLicenses	Date: 10/22/2017
Use Case ID: 21	

Use Case Specification: AddNurseLicenses

AddNurseLicenses

Brief Description

This use case describes how a nurse's license information is added to the system.

Flow of Events

Basic Flow

1. The use case starts when the Program Coordinator (PC) logs into the admin portal.
2. The PC selects Maintain Nurse Records from the menu options on the home page.
3. The PC selects the nurse to be updated from a list of nurse records.
4. The PC selects Add License
5. A form is displayed for the PC to fill out
 - a. The PC will enter the license number.
 - b. The PC will enter the license issue date.
 - c. The PC will enter the license expiration date.
 - d. The PC will enter the licensure state.
 - e. The PC will enter the license type.
6. The PC selects submit on the license form.
7. The system responds by adding the nurse's associated license record to the system.

Alternative Flows

If the process is canceled

The process is terminated, and the system gathers no information.

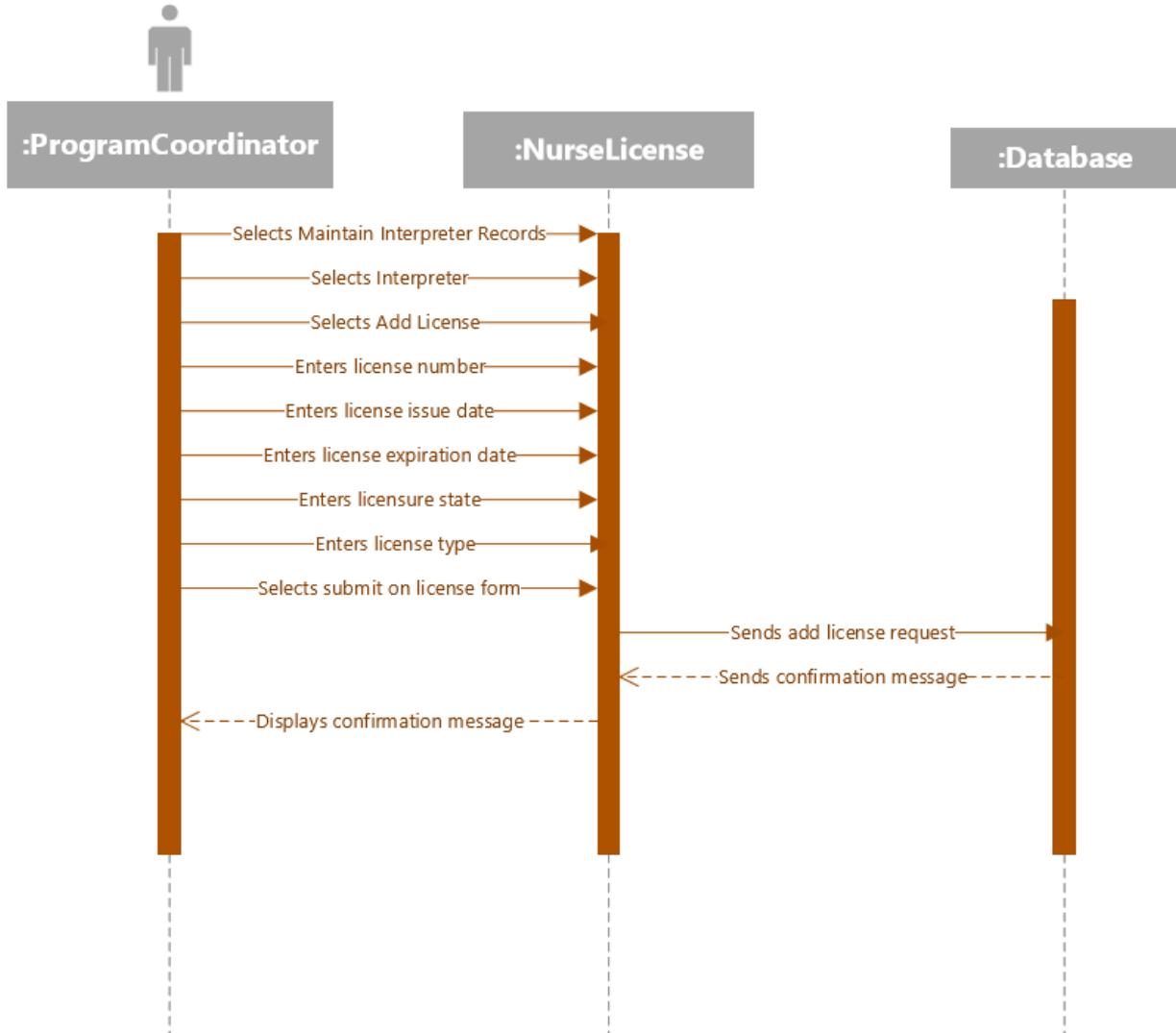
Pre-conditions

The Program Coordinator must have access to the admin portal

The nurse being updated must already exist in the system

The nurse must have a valid license and documentation

1. The use case starts when the Program Coordinator (PC) logs into the admin portal.
2. The PC selects Maintain Nurse Records from the menu options on the home page.
3. The PC selects the nurse to be updated from a list of nurse records.
4. The PC selects Add License
5. A form is displayed for the PC to fill out
 - a. The PC will enter the license number.
 - b. The PC will enter the license issue date.
 - c. The PC will enter the license expiration date.
 - d. The PC will enter the licensure state.
 - e. The PC will enter the license type.
6. The PC selects submit on the license form.
7. The system responds by adding the nurse's associated license record to the system.



SOS Louisville	Version: 1.0
Use Case Specification: GenerateNurseReport	Date: 10/23/2017
Use Case ID: 22	

Use Case Specification: GenerateNurseReport

GenerateNurseReport

Brief Description

This use case describes how a user requests reports about nurses.

Flow of Events

Basic Flow

1. The use case starts when the Program Coordinator (PC) logs into the admin portal.
2. The PC will select Reports from the menu options on the home page.
3. The PC selects Nurse Reports.
4. The desired report is displayed.
5. The report can then be exported or printed.

Alternative Flows

If the process is canceled

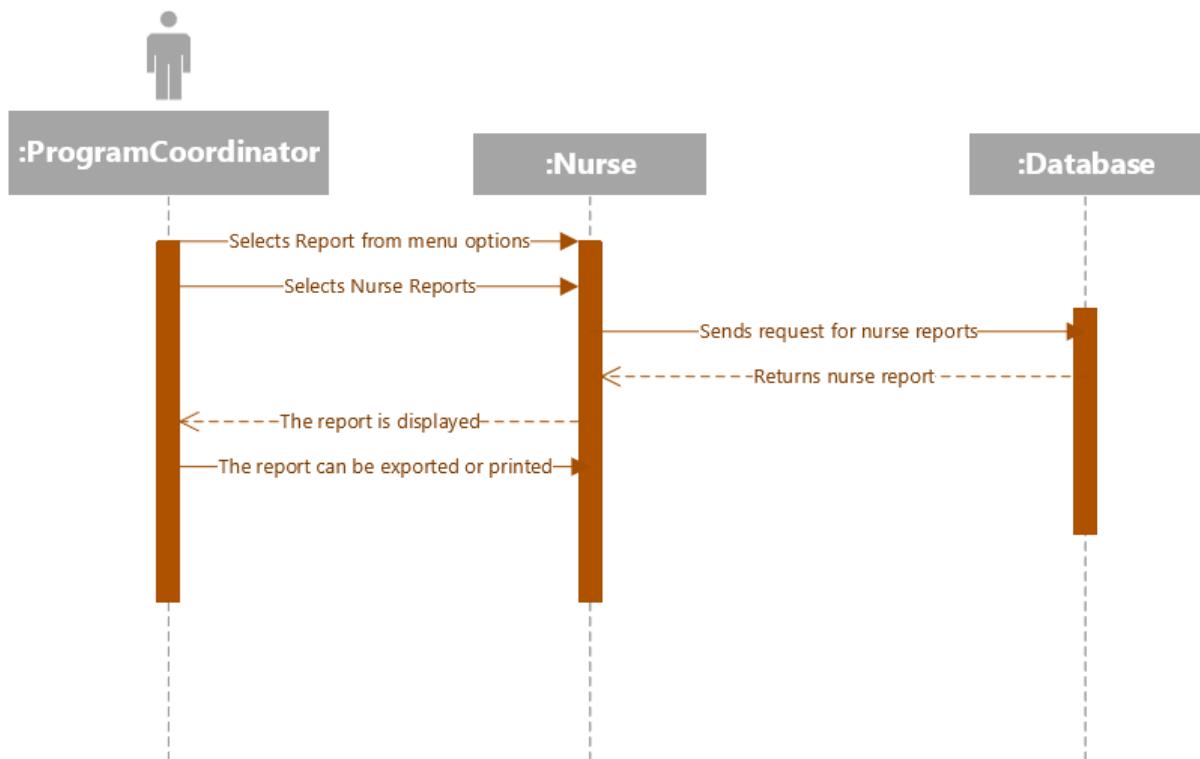
The process is terminated, and the system does not generate a report.

Pre-conditions

There must be nurse records in the system to generate a report

The Program Coordinator must have access to the admin portal

1. The use case starts when the Program Coordinator (PC) logs into the admin portal.
2. The PC will select Reports from the menu options on the home page.
3. The PC selects Nurse Reports.
4. The desired report is displayed.
5. The report can then be exported or printed.



SOS Louisville	Version: 1.0
Use Case Specification: SendEmailsToNurses	Date: 10/23/2017
Use Case ID: 23	

Use Case Specification: SendEmailsToNurses

SendEmailsToNurses

Brief Description

This use case describes how emails are sent to nurses.

Flow of Events

Basic Flow

1. The use case starts when the Program Coordinator (PC) logs in to the admin portal.
2. The PC will select Email from the menu options on the homepage.
3. A drop box will appear, and the PC will select Email Nurse.
4. A list of nurses and their emails will be displayed for selection.
5. The PC will then select which nurse(s) to be emailed.
6. The PC will select Done, to send emails to the specified nurse(s) with personalized greetings by nurse name.

Alternative Flows

If the process is canceled

The process is terminated, and the system does not email any nurses.

Special Requirements

MailMunch must be added to the system

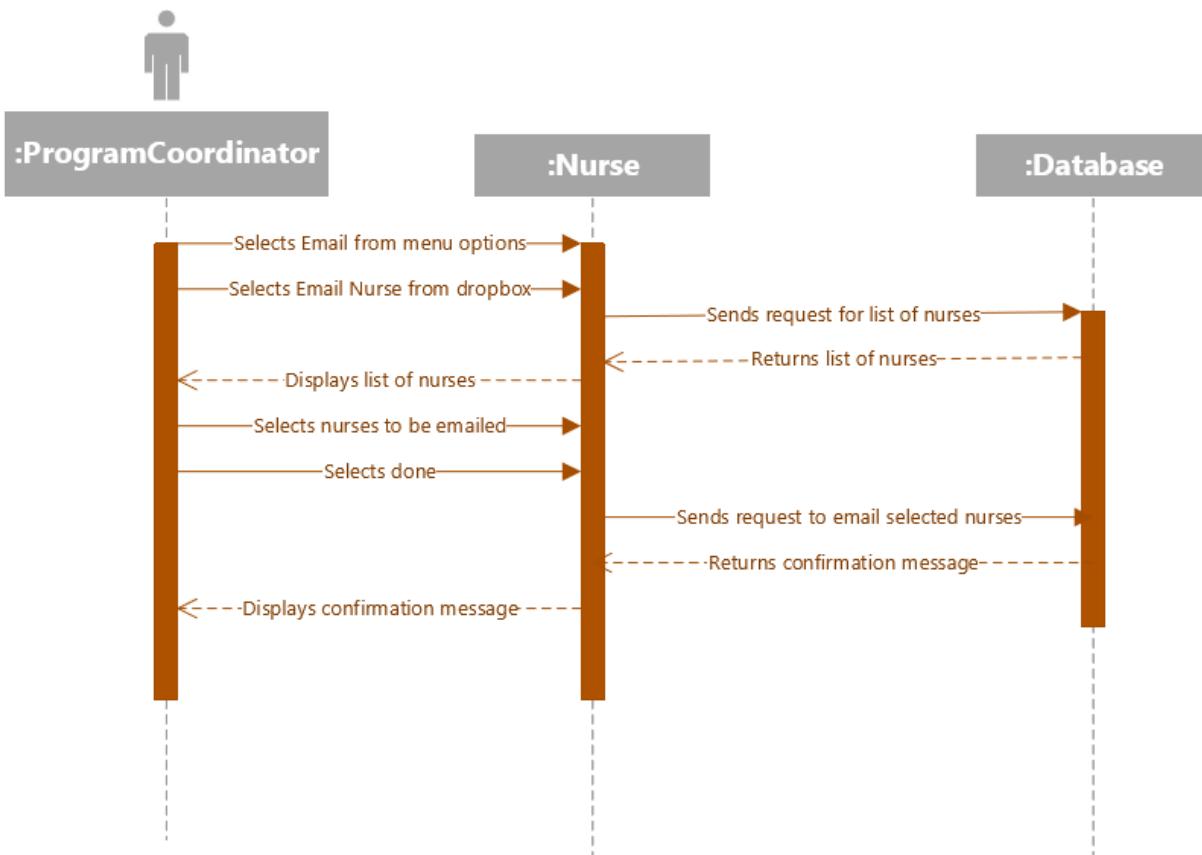
Pre-conditions

The Program Coordinator must have access to the admin portal

The nurse(s) must be in the system

The nurse(s) must have an up-to-date email address in the system

1. The use case starts when the Program Coordinator (PC) logs in to the admin portal.
2. The PC will select Email from the menu options on the homepage.
3. A drop box will appear, and the PC will select Email Nurse.
4. A list of nurses and their emails will be displayed for selection.
5. The PC will then select which nurse(s) to be emailed.
6. The PC will select Done, to send emails to the specified nurse(s) with personalized greetings by nurse name.



SOS Louisville	Version: 1.0
Use Case Specification: Nurse Event Scheduling	Date: 10/23/2017
Use Case ID: 24	

Use Case Specification: NurseEventScheduling

NurseEventScheduling

Brief Description

This use case describes how events with nurses are scheduled.

Flow of Events

Basic Flow

1. The use case starts when the Program Coordinator (PC) logs in to the admin portal.
2. The PC will select Scheduling from the menu options on the home page.
3. A drop down box will appear, and the PC will select Schedule Nurses.
4. A form will then appear, for the PC to fill out.
5. The PC will select what nurse specialties are needed for the event.
6. The PC will select what date the event is to be held.
7. The PC will select where the event will be held.
8. The system will retrieve the data from the database and display a relevant list of nurses that satisfy the above criteria.
9. The PC will select a nurse or nurses from the list.
10. The PC will select Done, triggering the Send Nurse Emails use case.

Alternative Flows

If the process is canceled

The process is terminated, and the system does not schedule any nurses.

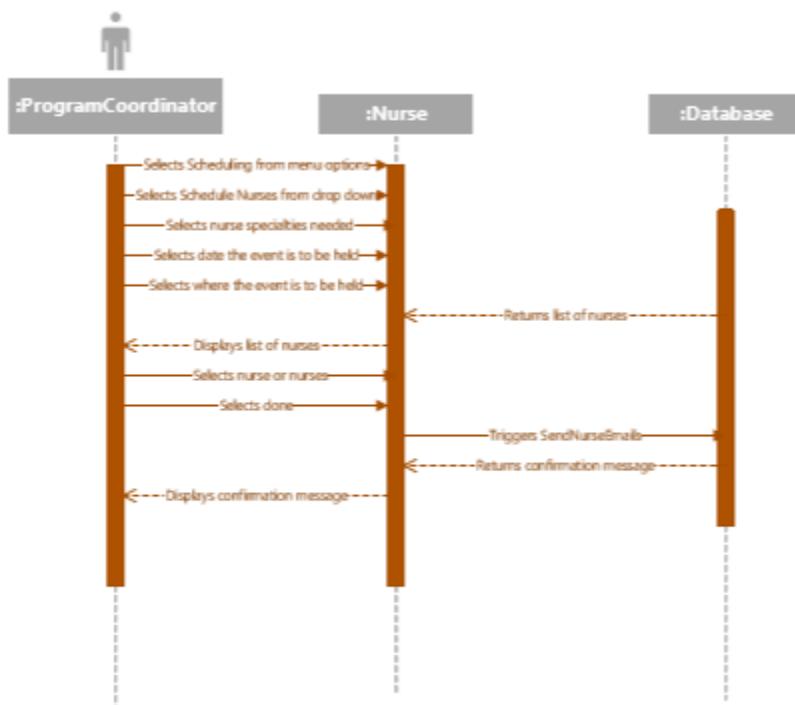
Pre-conditions

The Program Coordinator must have access to the admin portal

Nurse records must exist in the system

The nurse(s) must have an up-to-date email address in the system

1. The use case starts when the Program Coordinator (PC) logs in to the admin portal.
2. The PC will select Scheduling from the menu options on the home page.
3. A drop down box will appear, and the PC will select Schedule Nurses.
4. A form will then appear, for the PC to fill out.
5. The PC will select what nurse specialties are needed for the event.
6. The PC will select what date the event is to be held.
7. The PC will select where the event will be held.
8. The system will retrieve the data from the database and display a relevant list of nurses that satisfy the above criteria.
9. The PC will select a nurse or nurses from the list.
10. The PC will select Done, triggering the Send Nurse Emails use case.



SOS Louisville	Version: 1.0
Use Case Specification: SendNurseThankYouEmails	Date: 10/23/2017
Use Case ID: 25	

Use Case Specification: SendNurseThankYouEmails

SendNurseThankYouEmails

Brief Description

This use case describes how thank-you emails for nurses are emailed.

Flow of Events

Basic Flow

1. The use case starts when the Program Coordinator (PC) logs in to the admin portal.
2. The PC will select Email from the menu options on the homepage.
3. A drop box will appear, and the PC will select Email Nurse.
4. The PC can select a thank-you letter from a list of thank-you letter templates.
5. A list of nurses and their emails will be displayed for selection.
6. The PC will then select which nurse(s) to be emailed.
7. The PC will select Done, to send emails to the specified nurse(s) with personalized greetings by nurse name.

Alternative Flows

If the process is canceled

The process is terminated, and the system does not send any nurse thank you emails.

Special Requirements

MailMunch must be added to the system

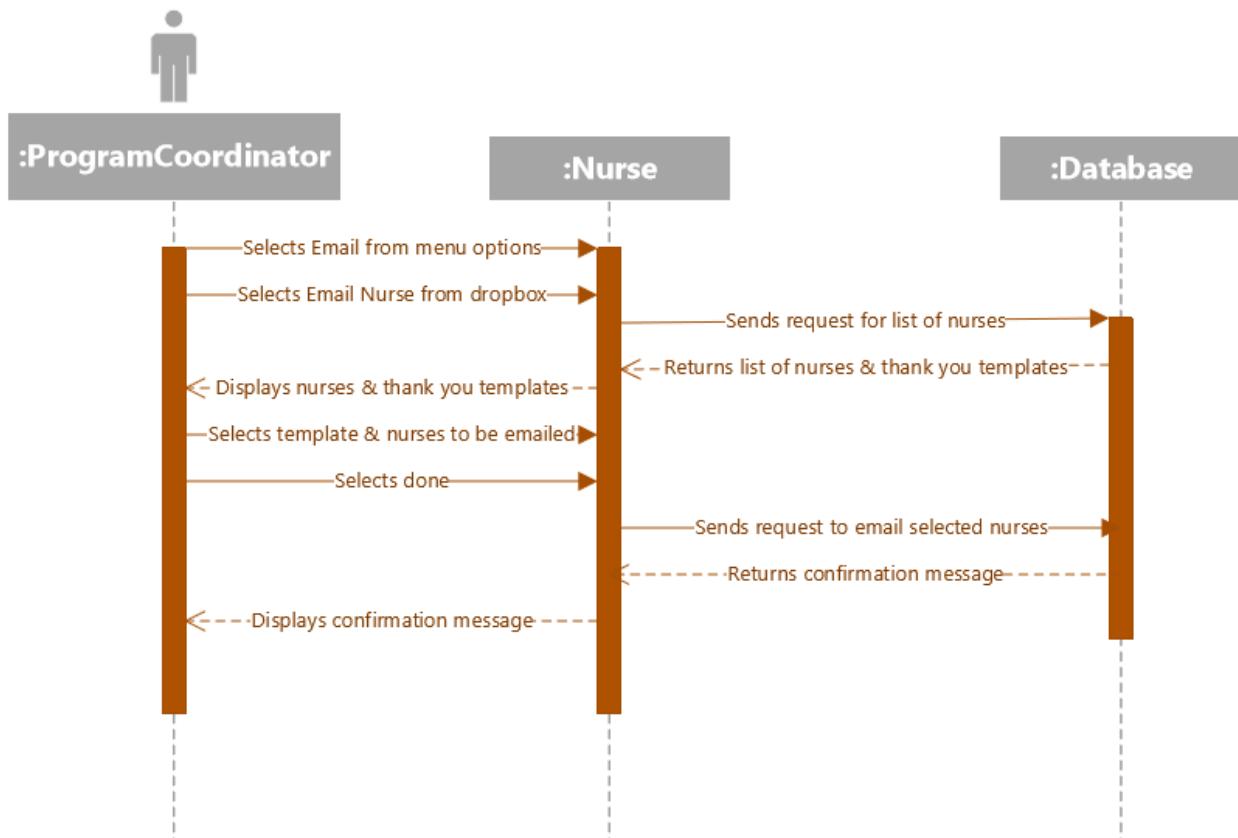
Pre-conditions

The Program Coordinator must have access to the admin portal

The nurse(s) must be in the system

The nurse(s) must have an up-to-date email address in the system

1. The use case starts when the Program Coordinator (PC) logs in to the admin portal.
2. The PC will select Email from the menu options on the homepage.
3. A drop box will appear, and the PC will select Email Nurse.
4. The PC can select a thank-you letter from a list of thank-you letter templates.
5. A list of nurses and their emails will be displayed for selection.
6. The PC will then select which nurse(s) to be emailed.
7. The PC will select Done, to send emails to the specified nurse(s) with personalized greetings by nurse name.



SOS Louisville	Version: 1.0
Use Case Specification: AddNewInterpreter	Date: 10/22/2017
Use Case ID: 26	

Use Case Specification: AddNewInterpreter

AddNewInterpreter

Brief Description

This use case describes how an interpreter's information is added to the system.

Flow of Events

Basic Flow

1. The use case starts when a clinical language interpreter selects Get Involved on SOS Louisville's website.
2. The interpreter will then select the Clinical Language Interpreter Form from a drop-down box.
3. A dynamic webform will be displayed for the interpreter to fill out.
 - a. The interpreter will enter their first name.
 - b. The interpreter will enter their last name.
 - c. The interpreter will enter their city.
 - d. The interpreter will enter their state.
 - e. The interpreter will enter their zip.
 - f. The interpreter will enter their address.
 - g. The interpreter will enter their applicable licenses.
 - h. The interpreter will enter their phone number.
 - i. The interpreter will enter their email address.
 - j. The interpreter will enter their hospital affiliation.
 - k. The interpreter will enter their primary language.
 - l. The interpreter will enter their applicable secondary language(s).
4. The interpreter will hit submit on the webform.
5. The system auto-gathers the information from the webform and uploads it to SOS Louisville's database.
6. The clinical language interpreter's record is added to the system.

Alternative Flows

If the process is canceled

The process is terminated, and the system gathers no information.

If incorrect/invalid information is entered into the system

And is not caught by input validation, the IT coordinator or Executive Director would have to fix the error by manually entering or modifying the appropriate clinical language interpreter's volunteer information.

If the auto-gather function fails

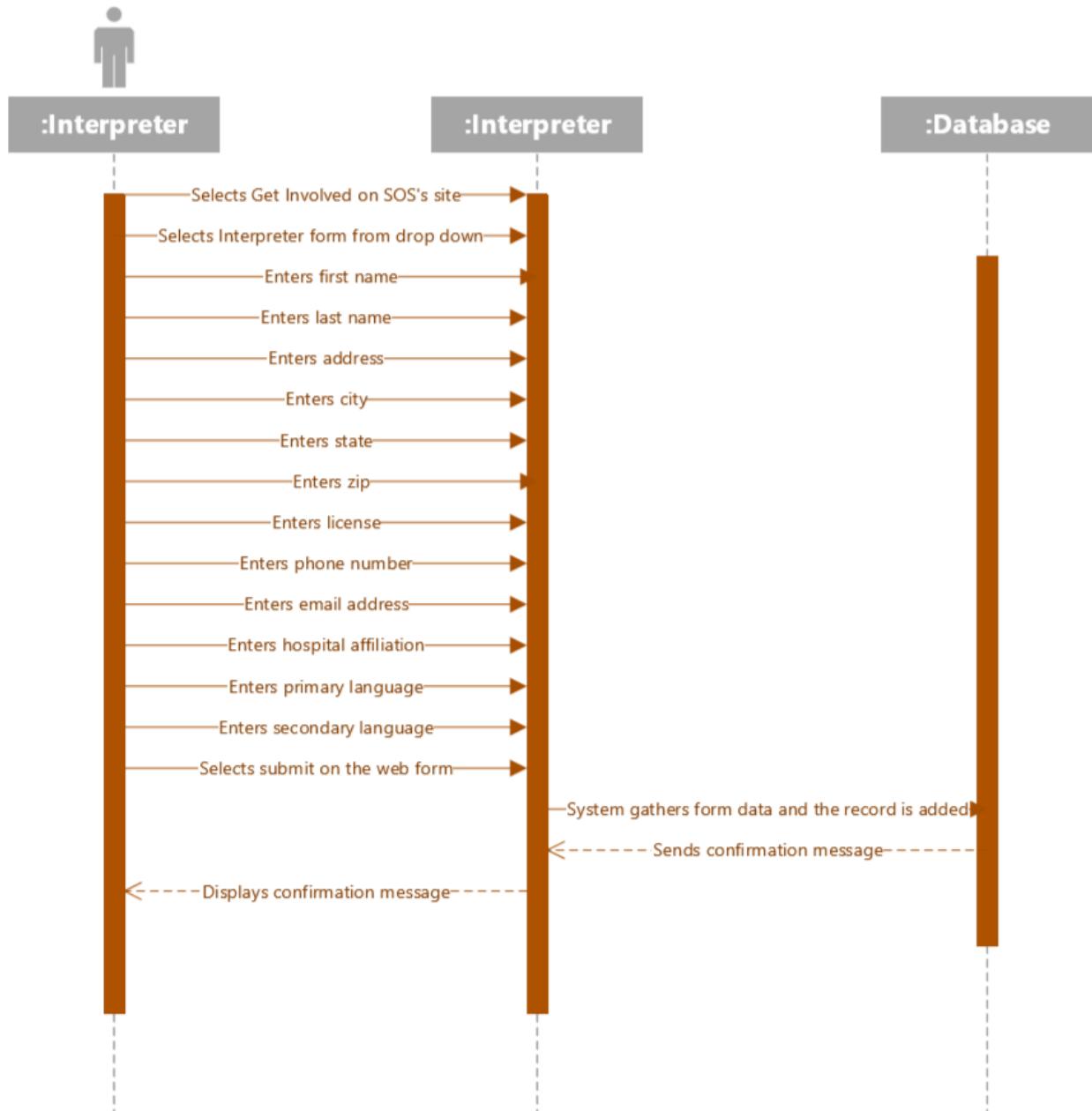
The IT Coordinator or Executive Director would have to manually enter the clinical language interpreter's volunteer information into SOS Louisville's database system.

Pre-conditions

The clinical language interpreter must have the appropriate qualifications

The clinical language interpreter's record must not already exist in the system

1. The use case starts when a clinical language interpreter selects Get Involved on SOS Louisville's website.
2. The interpreter will then select the Clinical Language Interpreter Form from a drop-down box.
3. A dynamic webform will be displayed for the interpreter to fill out.
 - a. The interpreter will enter their first name.
 - b. The interpreter will enter their last name.
 - c. The interpreter will enter their city.
 - d. The interpreter will enter their state.
 - e. The interpreter will enter their zip.
 - f. The interpreter will enter their address.
 - g. The interpreter will enter their applicable licenses.
 - h. The interpreter will enter their phone number.
 - i. The interpreter will enter their email address.
 - j. The interpreter will enter their hospital affiliation.
 - k. The interpreter will enter their primary language.
 - l. The interpreter will enter their applicable secondary language(s).
4. The interpreter will hit submit on the webform.
5. The system auto-gathers the information from the webform and uploads it to SOS Louisville's database.
6. The clinical language interpreter's record is added to the system.



SOS Louisville	Version: 1.0
Use Case Specification: UpdateInterpreterRecord	Date: 10/23/2017
Use Case ID: 27	

Use Case Specification: UpdateInterpreterRecord

UpdateInterpreterRecord

Brief Description

This use case describes how an interpreter's information is updated to the system.

Flow of Events

Basic Flow

1. The use case starts when the IT Coordinator logs into the admin portal.
2. The IT Coordinator selects Maintain Interpreter Records from the menu options on the homepage.
3. The IT Coordinator selects the interpreter to be updated from a list of interpreter records.
4. The IT Coordinator selects Update.
5. The IT Coordinator modifies the required information and submits it.

6. The record is updated within the system.

Alternative Flows

If the process is canceled

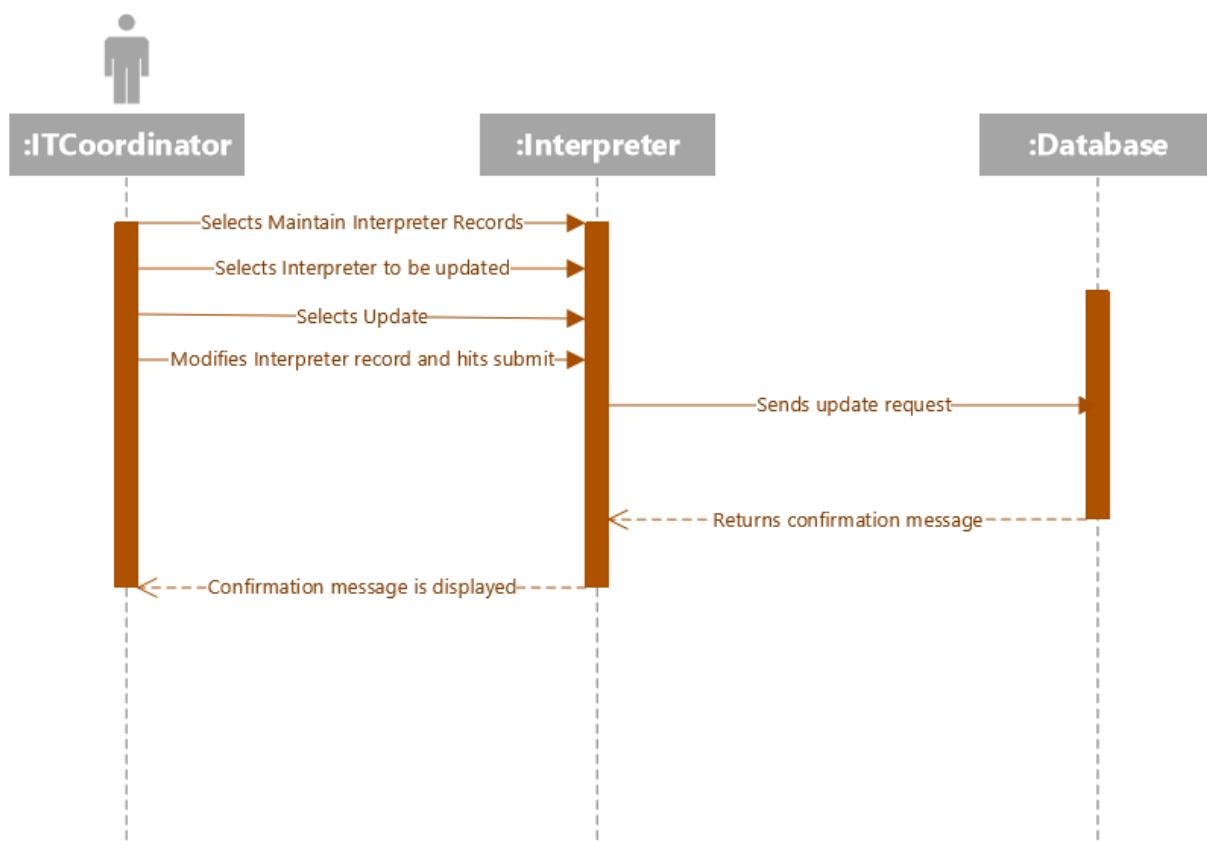
The process is terminated, and the system updates no information.

Pre-conditions

The IT Coordinator must have access to the admin portal

The interpreter record being updated must already exist in the system

1. The use case starts when the IT Coordinator logs into the admin portal.
2. The IT Coordinator selects Maintain Interpreter Records from the menu options on the homepage.
3. The IT Coordinator selects the interpreter to be updated from a list of interpreter records.
4. The IT Coordinator selects Update.
5. The IT Coordinator modifies the required information and submits it.
6. The record is updated within the system.



SOS Louisville	Version: 1.0
Use Case Specification: DeleteInterpreterRecord	Date: 10/22/2017
Use Case ID: 28	

Use Case Specification: DeleteInterpreterRecord

DeleteInterpreterRecord

Brief Description

This use case describes how an interpreter's information is deleted from the system.

Flow of Events

Basic Flow

1. The use case starts when the IT Coordinator logs into the admin portal.
2. The IT Coordinator selects Maintain Interpreter Records from the menu options on the home page.
3. The IT Coordinator selects the interpreter to be removed from a list of interpreters.
4. The IT Coordinator selects Delete.
5. The selected interpreter's record is deleted from the system.

Alternative Flows

If the process is canceled

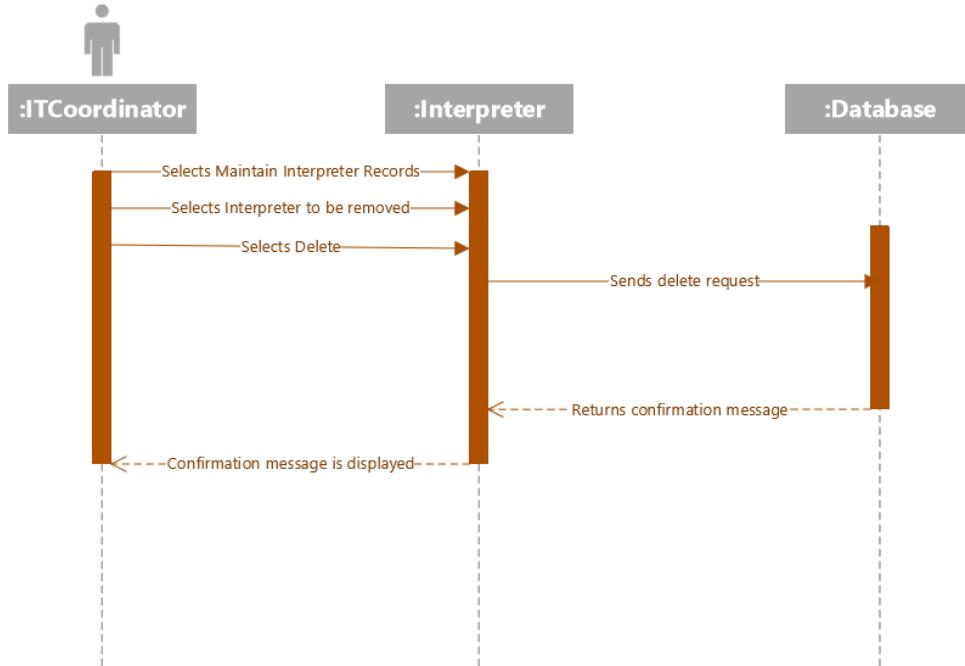
The process is terminated, and the system deletes no information.

Pre-conditions

The IT Coordinator must have access to the admin portal

The interpreter being deleted must already exist in the system

1. The use case starts when the IT Coordinator logs into the admin portal.
2. The IT Coordinator selects Maintain Interpreter Records from the menu options on the home page.
3. The IT Coordinator selects the interpreter to be removed from a list of interpreters.
4. The IT Coordinator selects Delete.
5. The selected interpreter's record is deleted from the system.



SOS Louisville	Version: 1.0
Use Case Specification: AddInterpreterLicenses	Date: 10/22/2017
Use Case ID: 29	

Use Case Specification: AddInterpreterLicenses

AddInterpreterLicenses

Brief Description

This use case describes how an interpreter's license information is added to the system.

Flow of Events

Basic Flow

1. The use case starts when the Program Coordinator (PC) logs into the admin portal.
2. The PC selects Maintain Interpreter Records from the menu options on the home page.
3. The PC selects the interpreter to be updated from a list of interpreter records.
4. The PC selects Add License
5. A form is displayed for the PC to fill out
 - a. The PC will enter the license number.
 - b. The PC will enter the license issue date.
 - c. The PC will enter the license expiration date.
 - d. The PC will enter the licensure state.
 - e. The PC will enter the license type.
6. The PC selects submit on the license form.
7. The system responds by adding the interpreter's associated license record to the system.

Alternative Flows

If the process is canceled

The process is terminated, and the system gathers no information.

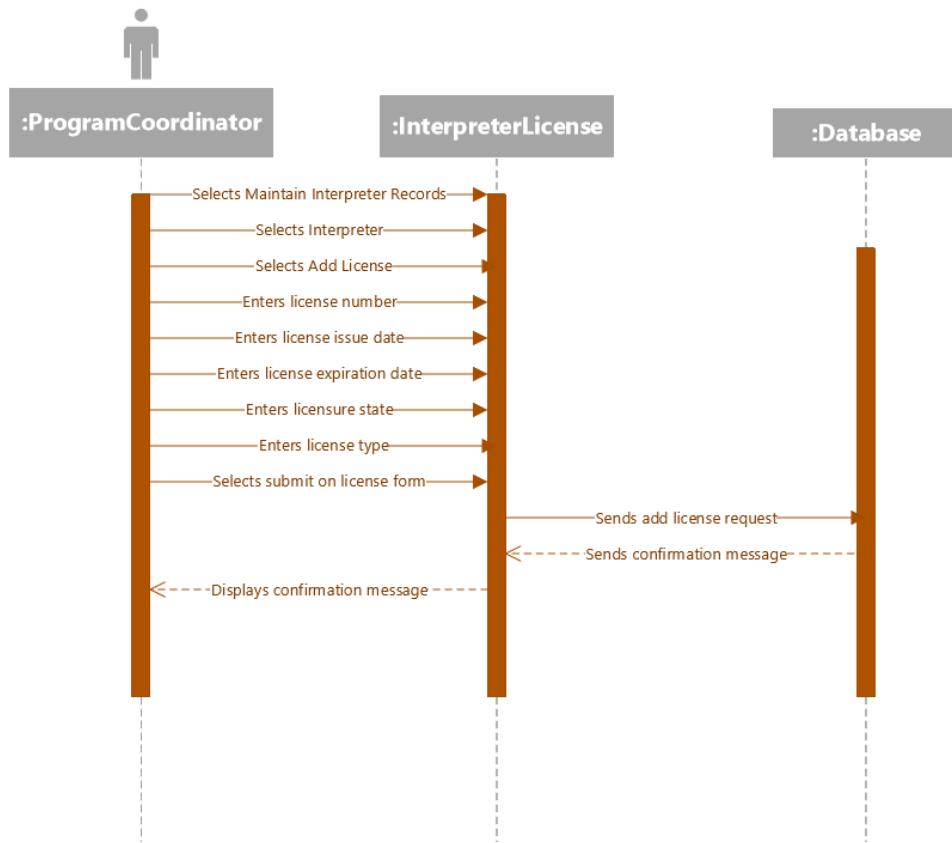
Pre-conditions

The Program Coordinator must have access to the admin portal

The interpreter being updated must already exist in the system

The interpreter must have a valid license and documentation

1. The use case starts when the Program Coordinator (PC) logs into the admin portal.
2. The PC selects Maintain Interpreter Records from the menu options on the home page.
3. The PC selects the interpreter to be updated from a list of interpreter records.
4. The PC selects Add License
5. A form is displayed for the PC to fill out
 - a. The PC will enter the license number.
 - b. The PC will enter the license issue date.
 - c. [The PC will enter the license expiration date.
 - d. The PC will enter the licensure state.
 - e. The PC will enter the license type.
6. The PC selects submit on the license form.
7. The system responds by adding the interpreter's associated license record to the system.



Surgery On Sunday	Version: 1.0
Use Case Specification: Generate Interpreter Report	Date: 10/23/2017
Use Case ID: 30	

Use Case Specification: Generate Interpreter Report

Generate Interpreter Report

Brief Description

The objective of this use case is to show how a program coordinator would go about requesting a report about the available interpreters. This is necessary so that SOS can appropriately assign the correct interpreter based on the patient's surgery and primary language. This way, instead of manually searching through all of the files available on interpreters, they instead can obtain a report showing all available staff or volunteers that are qualified.

Flow of Events

Basic Flow

- A user wants to access interpreter information for a particular surgery's needs
- Program Coordinator logs in with their authorization
- Program Coordinator navigates to reports tab
- Program Coordinator navigates to interpreters within reports
- Program Coordinator saves or prints reports with necessary data
- Program Coordinator logs out of system

Alternative Flows

There are no alternative flows

Special Requirements

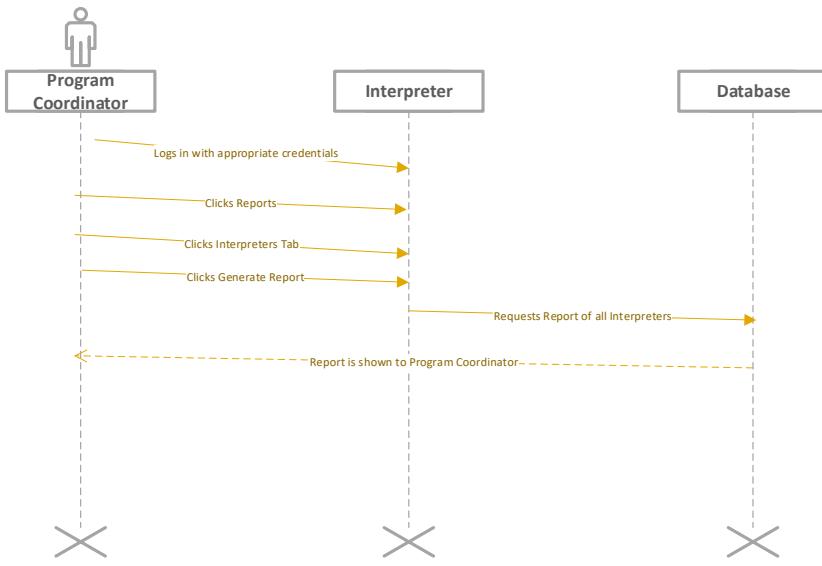
- User must be logged into their account and have the proper authorization to view and access the data.
- Data must already exist in database to be accessed.

Pre-conditions

User must be logged into the system with appropriate credentials.

Post-conditions

Information on interpreters is displayed and may be saved or printed.



Surgery On Sunday	Version: 1.0
Use Case Specification: Send Email To Interpreters	Date: 10/23/2017
Use Case ID: 31	

Use Case Specification: Send Email to Interpreters

Send Email to Interpreters

Brief Description

The objective of this use case is to show how a program coordinator would go about emailing available interpreters. This is necessary so that SOS can appropriately assign the correct interpreter based on the patient's surgery and primary language and make sure they're available and aware of the event. This way, instead of manually searching through all of the files available on interpreters, they instead access all of the email addresses of qualified interpreters and ask them about their availability, or should they have already signed up, remind them of the upcoming event and supply them with further details.

Flow of Events

Basic Flow

- A user wants to access appropriate interpreter email addresses
- Program Coordinator logs in with their authorization
- Program Coordinator navigates email tab
- Program Coordinator navigates to interpreters within email categories
- Program Coordinator finds appropriate interpreters
- Program Coordinator accesses their email addresses
- Program Coordinator inputs them into plug in
- Program Coordinator sends email to all possible/already signed up interpreters
- Program Coordinator logs out of system

Alternative Flows

There are no alternative flows

Special Requirements

- User must be logged into their account and have the proper authorization to view and access the data
- Data must already exist in database to be accessed
- Interpreter must have supplied email address prior to accessing

Pre-conditions

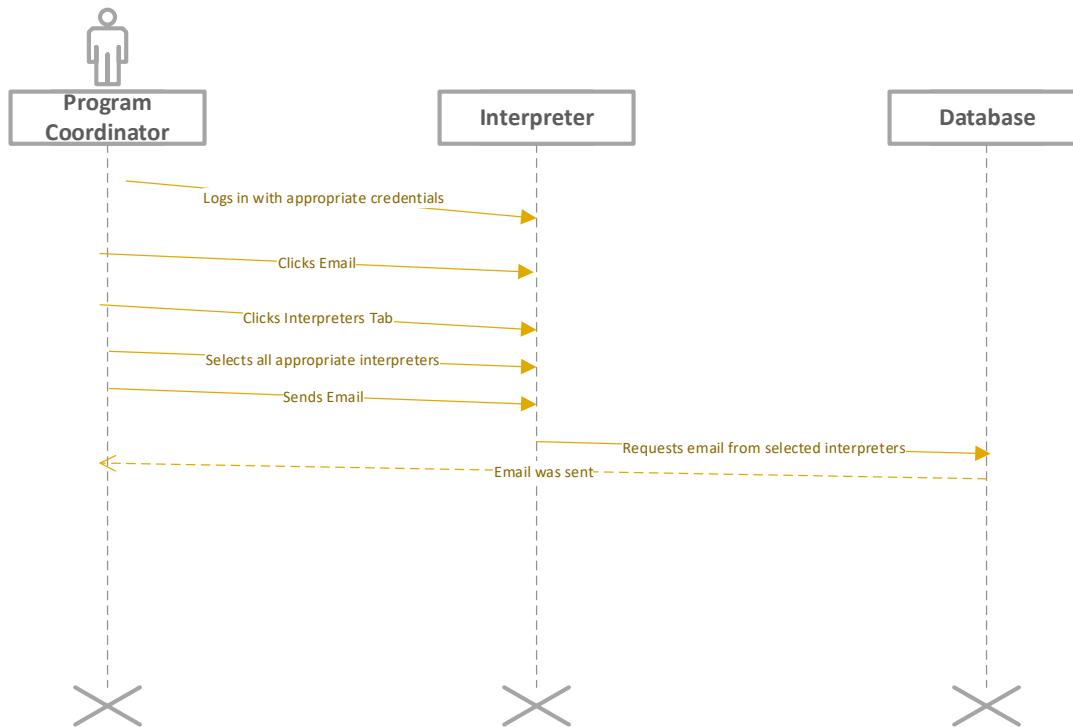
User must be logged into the system with appropriate credentials.

Post-conditions

Email is sent to appropriate email addresses

Extension Points

There are no extension points.



Surgery On Sunday	Version: 1.0
Use Case Specification: Schedule Event With Appropriate Interpreter(s)	Date: 10/23/2017
Use Case ID: 32	

Use Case Specification: Schedule Event With Appropriate Interpreter(s)

Schedule Event With Appropriate Interpreter(s)

Brief Description

The objective of this use case is to show how a program coordinator would go about scheduling an event with the available interpreters. This is necessary so that SOS can appropriately assign the correct interpreter based on the patient's surgery and primary language so that any patient needs will be properly addressed. This way, instead of manually searching through all of the files available on interpreters, they instead access all of the qualified interpreters and can schedule the event with those available and qualified.

Flow of Events

Basic Flow

- A user wants to schedule an event
- Program Coordinator logs in with their authorization
- Program Coordinator navigates to schedule event tab
- Program Coordinator fills out date of event field
- Program Coordinator fills out location of event field
- Program Coordinator fills out patient field
- Program Coordinator fills out selected nurses field
- Program Coordinator fills out selected volunteers field
- Program Coordinator fills out surgeon field
- Program Coordinator fills out selected interpreter field
- Program Coordinator clicks submit
- Program Coordinator logs out

Alternative Flows

There are no alternative flows

Special Requirements

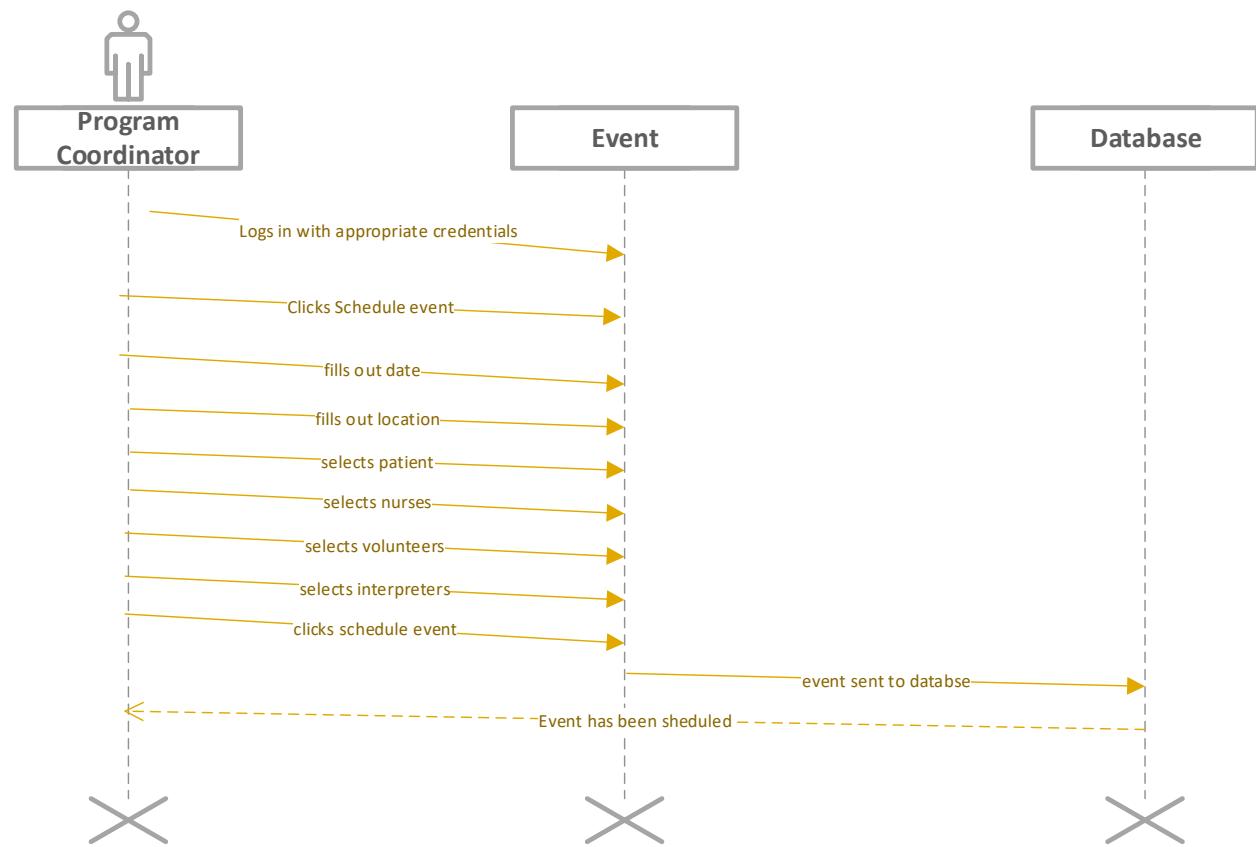
- User must be logged into their account and have the proper authorization to view and access the data
- Data must already exist in database to be accessed
- Interpreter must be qualified in order to select

Pre-conditions

User must be logged into the system with appropriate credentials.

Post-conditions

Event is Scheduled



Surgery On Sunday	Version: 1.0
Use Case Specification: Send Thank You Emails to Appropriate Interpreter(s)	Date: 10/23/2017
Use Case ID: 33	

Use Case Specification: Send Thank You Emails to Appropriate Interpreter(s)

Send Thank You Emails to Appropriate Interpreter(s)

Brief Description

The objective of this use case is to show how a program coordinator would go about sending a thank you email the appropriate interpreters. This is necessary so that SOS can appropriately thank the interpreter(s) that have generously donated their time in order to help SOS and the patient. This way, instead of manually searching through all of the files available on interpreters, they instead access all of interpreters that worked the event and can easily send them emails.

Flow of Events

Basic Flow

- A user wants to show thanks for interpreter volunteer
- Program Coordinator logs in with their authorization
- Program Coordinator navigates to past events tab
- Program Coordinator finds appropriate event
- Program Coordinator navigates to interpreter field
- Program Coordinator selects the appropriate interpreter(s) email address
- Program Coordinator navigates to mail plugin
- Program Coordinator imports these email addresses
- Program Coordinator sends email
- Program Coordinator logs out of plugin
- Program Coordinator logs out of SOS website

Alternative Flows

There are no alternative flows

Special Requirements

- User must be logged into their account and have the proper authorization to view and access the data
- Data must already exist in database to be accessed

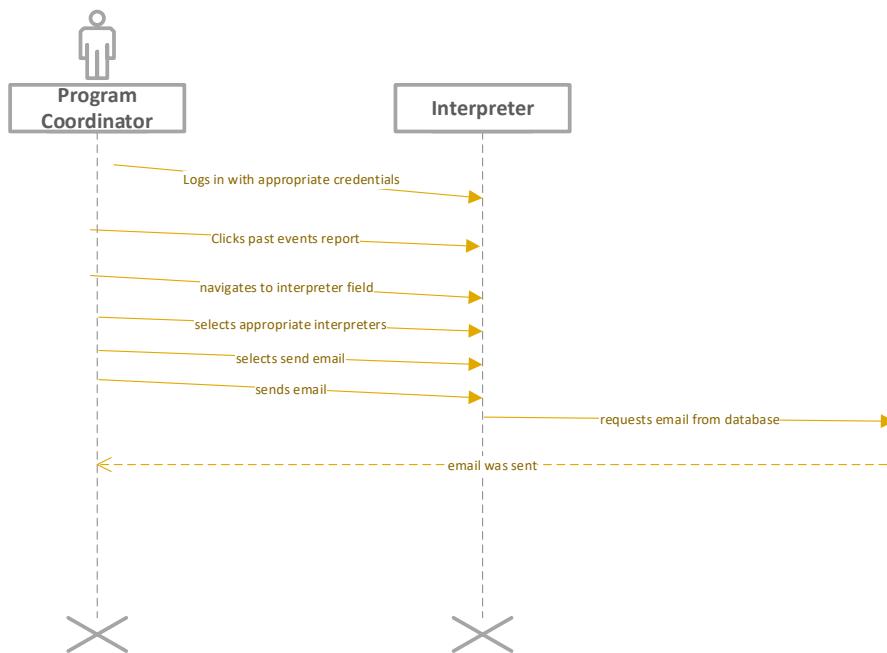
-User must have access to Mail Munch

Pre-conditions

User must be logged into the system with appropriate credentials.

Post-conditions

Email is sent.



Surgery On Sunday	Version: 1.0
Use Case Specification: Add New Non-Clinical Volunteer	Date: 10/23/2017
Use Case ID: 34	

Use Case Specification: Add New Non-Clinical Volunteer

Add New Non-Clinical Volunteer

Brief Description

The objective of this use case is to show how a Volunteer would go about signing up as a non-clinical volunteer, and how the system handles this request. This is necessary so that SOS can appropriately store and receive volunteer information. This way, instead of manually searching through all of the files available on volunteers, the information will be properly stored and will be easier to access.

Flow of Events

Basic Flow

- A user wants to volunteer
- Volunteer navigates to SOS website
- Volunteer navigates to volunteer application
- Volunteer fills out First Name field
- Volunteer fills out Last Name field
- Volunteer fills out City field
- Volunteer fills out State field
- Volunteer fills out Zip field
- Volunteer fills out Address field
- Volunteer fills out Licenses field
- Volunteer fills out Phone number field
- Volunteer fills out Email address field
- Volunteer fills out Hospital affiliation field
- Volunteer fills out Primary language field
- Volunteer fills out Secondary language if exists field
- Volunteer clicks submit
- System stores information in appropriate table
- Information is now accessible by program coordinator
- Volunteer closes tab

Alternative Flows

There are no alternative flows

Special Requirements

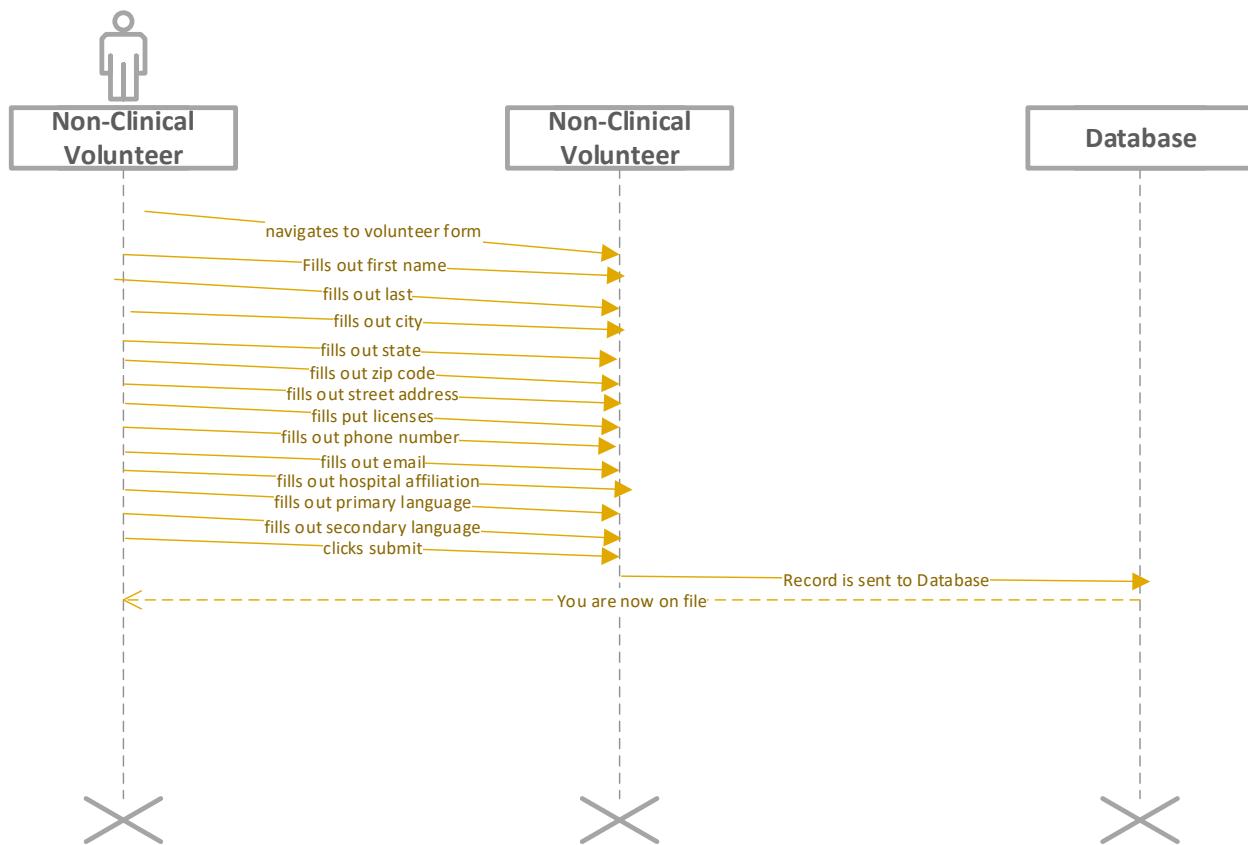
-User must fill out all required elements

Pre-conditions

None

Post-conditions

Information is stored properly.



Surgery On Sunday	Version: 1.0
Use Case Specification: Update Non-Clinical Volunteer Information	Date: 10/23/2017
Use Case ID: 35	

Use Case Specification: Update Non-Clinical Volunteer Information

Update Non-Clinical Volunteer Information

Brief Description

The objective of this use case is to show how an IT coordinator would go about updating a volunteer's information. This is necessary so that SOS can appropriately change the info of a volunteer in case they change phone numbers, move, or many other occurrences that would cause this to be needed. This way, instead of manually searching through all of the files available on volunteers, they instead can access quickly the exact volunteer form, click edit, and change the information directly.

Flow of Events

Basic Flow

- A user wants to update volunteer information
- IT Coordinator logs in with their authorization
- IT Coordinator navigates to volunteers table
- IT Coordinator finds appropriate volunteer
- IT Coordinator clicks edit
- IT Coordinator views First Name field
- IT Coordinator views Last Name field
- IT Coordinator views City field
- IT Coordinator views State field
- IT Coordinator views Zip field
- IT Coordinator views Address field
- IT Coordinator views Licenses field
- IT Coordinator views Phone number field
- IT Coordinator views Email address field
- IT Coordinator views Hospital affiliation field
- IT Coordinator views Primary language field
- IT Coordinator views Secondary language if exists field
- IT Coordinator selects the appropriate field(s) that needs changed
- IT Coordinator types in new attribute
- IT Coordinator clicks save
- IT Coordinator logs out

Alternative Flows

There are no alternative flows

Special Requirements

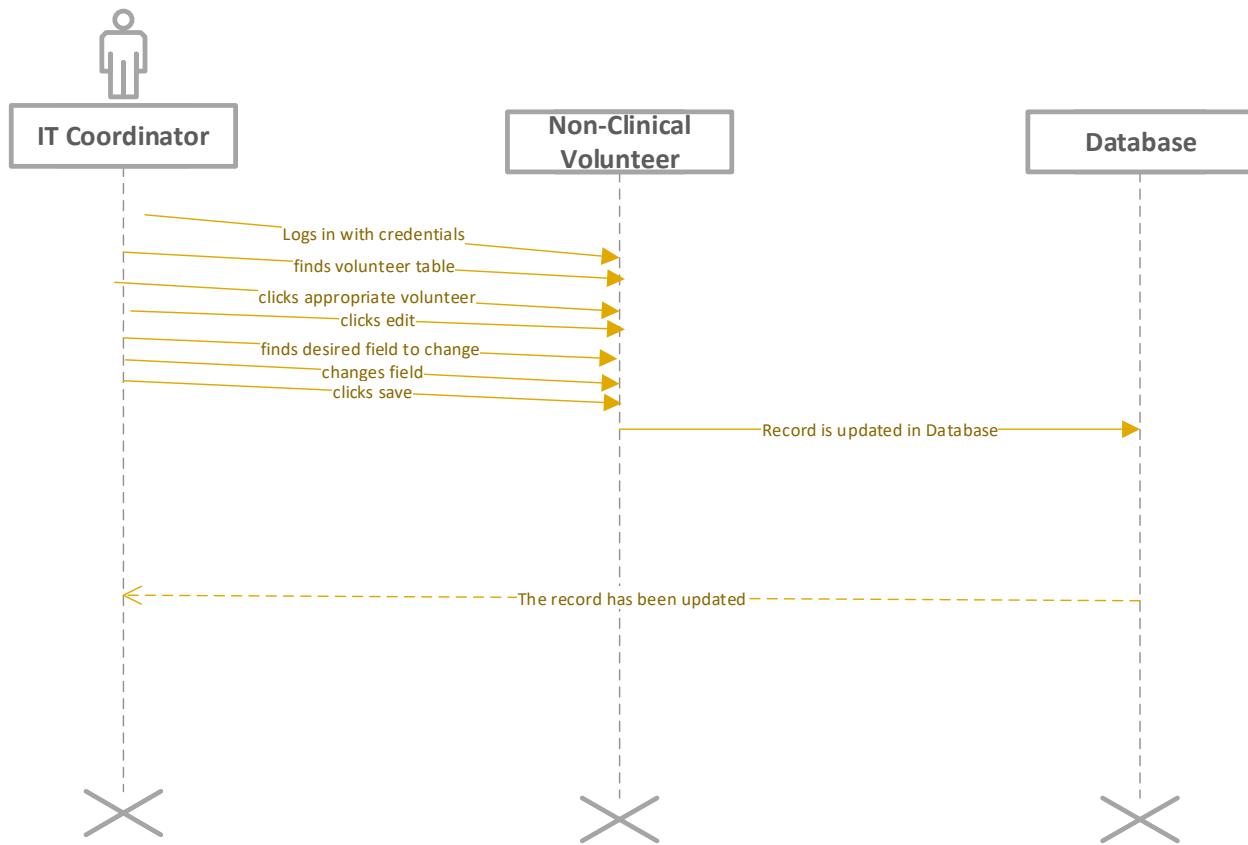
- User must be logged into their account and have the proper authorization to view and access the data
- Data must already exist in database to be accessed

Pre-conditions

User must be logged into the system with appropriate credentials.

Post-conditions

Record is updated.



Surgery On Sunday	Version: 1.0
Use Case Specification: Delete Non-Clinical Volunteer Information	Date: 10/23/2017
Use Case ID: 36	

Use Case Specification: Delete Non-Clinical Volunteer Information

Delete Non-Clinical Volunteer Information

Brief Description

The objective of this use case is to show how an IT coordinator would go about deleting a volunteer's information. This is necessary so that SOS can appropriately remove the info of a volunteer in case they move far away, don't have time to volunteer anymore, or other reasons that would warrant this system capability. This way, instead of manually searching through all of the files available on volunteers, they instead can access quickly the exact volunteer form and delete the record.

Flow of Events

Basic Flow

- A user wants to delete volunteer information
- IT Coordinator logs in with their authorization
- IT Coordinator navigates to volunteers table
- IT Coordinator finds appropriate volunteer record
- IT Coordinator clicks delete
- IT Coordinator clicks save
- IT Coordinator logs out

Alternative Flows

There are no alternative flows

Special Requirements

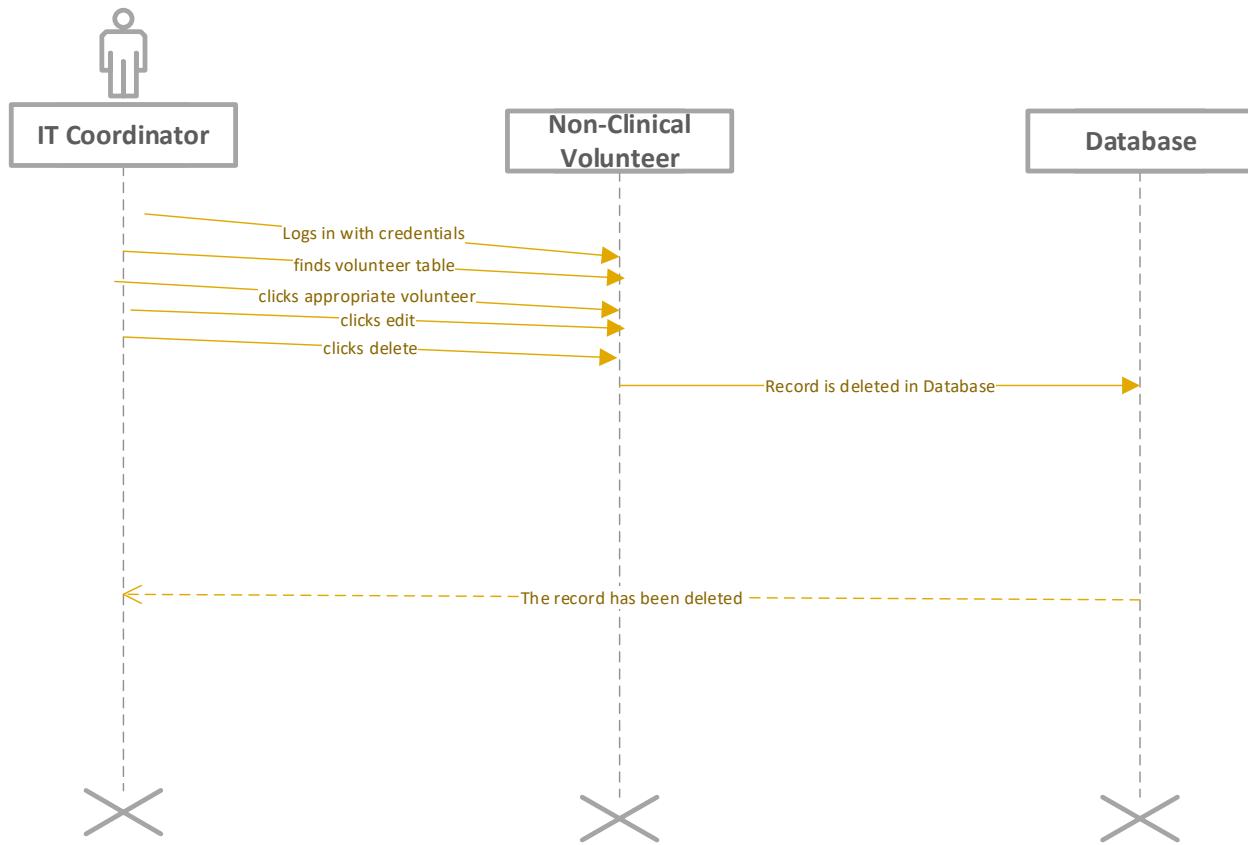
- User must be logged into their account and have the proper authorization to view and access the data
- Data must already exist in database to be accessed

Pre-conditions

User must be logged into the system with appropriate credentials.

Post-conditions

Record is deleted.



Surgery On Sunday	Version: 1.0
Use Case Specification: Generate Non-Clinical Volunteer Report	Date: 10/23/2017
Use Case ID: 37	

Use Case Specification: Generate Non-Clinical Volunteer Report

Generate Non-Clinical Volunteer Report

Brief Description

The objective of this use case is to show how a program coordinator would go about requesting a report about the available non-clinical volunteers. This is necessary so that SOS can appropriately assign the correct volunteers based on the patient's surgery and volunteer availability. This way, instead of manually searching through all of the files available on volunteers, they instead can obtain a report showing all available staff or volunteers that are qualified and available for a certain event.

Flow of Events

Basic Flow

- A user wants to access non-clinical volunteer information for a particular surgery's needs
- Program Coordinator logs in with their authorization
- Program Coordinator navigates to reports tab
- Program Coordinator navigates to non-clinical volunteers within reports
- Program Coordinator saves or prints reports with necessary data
- Program Coordinator logs out of system

Alternative Flows

There are no alternative flows

Special Requirements

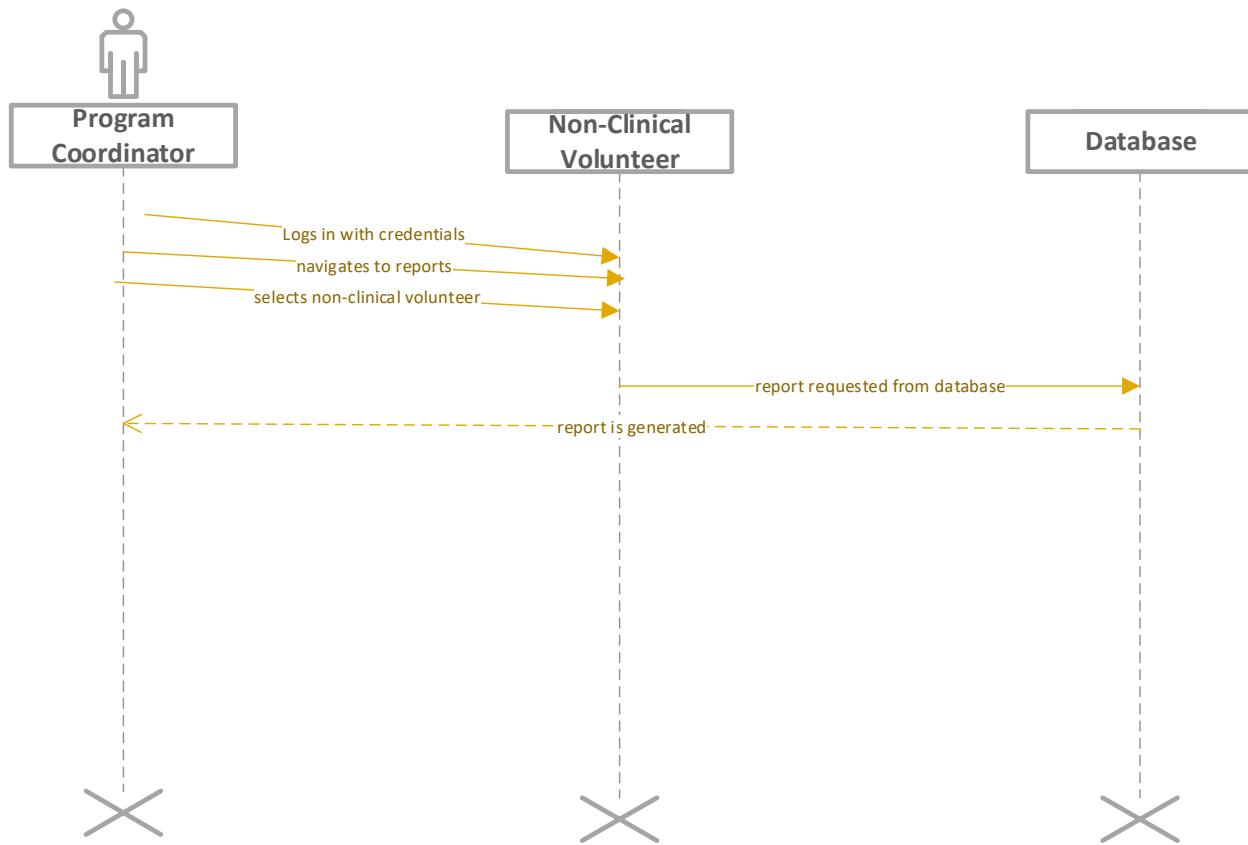
- User must be logged into their account and have the proper authorization to view and access the data.
- Data must already exist in database to be accessed.

Pre-conditions

User must be logged into the system with appropriate credentials.

Post-conditions

Information on volunteers is displayed and may be saved or printed.



Surgery On Sunday	Version: 1.0
Use Case Specification: Send Email to Non-Clinical Volunteers	Date: 10/23/2017
Use Case ID: 38	

Use Case Specification: Send Email to Non-Clinical Volunteers

Send Email to Non-Clinical Volunteers

Brief Description

The objective of this use case is to show how a program coordinator would go about emailing available non-clinical volunteers. This is necessary so that SOS can appropriately assign the correct volunteers based on the patient's surgery and the volunteer's availability. This way, instead of manually searching through all of the files available on volunteers, they instead access all of the email addresses of qualified volunteers and ask them about their availability, or should they have already signed up, remind them of the upcoming event and supply them with further details.

Flow of Events

Basic Flow

- A user wants to access appropriate volunteer email addresses
- Program Coordinator logs in with their authorization
- Program Coordinator navigates to reports tab
- Program Coordinator navigates to non-clinical volunteers within reports
- Program Coordinator finds appropriate volunteers
- Program Coordinator accesses their email addresses
- Program Coordinator imports them to plugin
- Program Coordinator sends email to all possible/already signed up volunteers
- Program Coordinator logs out of system

Alternative Flows

There are no alternative flows

Special Requirements

- User must be logged into their account and have the proper authorization to view and access the data
- Data must already exist in database to be accessed
- Volunteer must have supplied email address prior to accessing

Pre-conditions

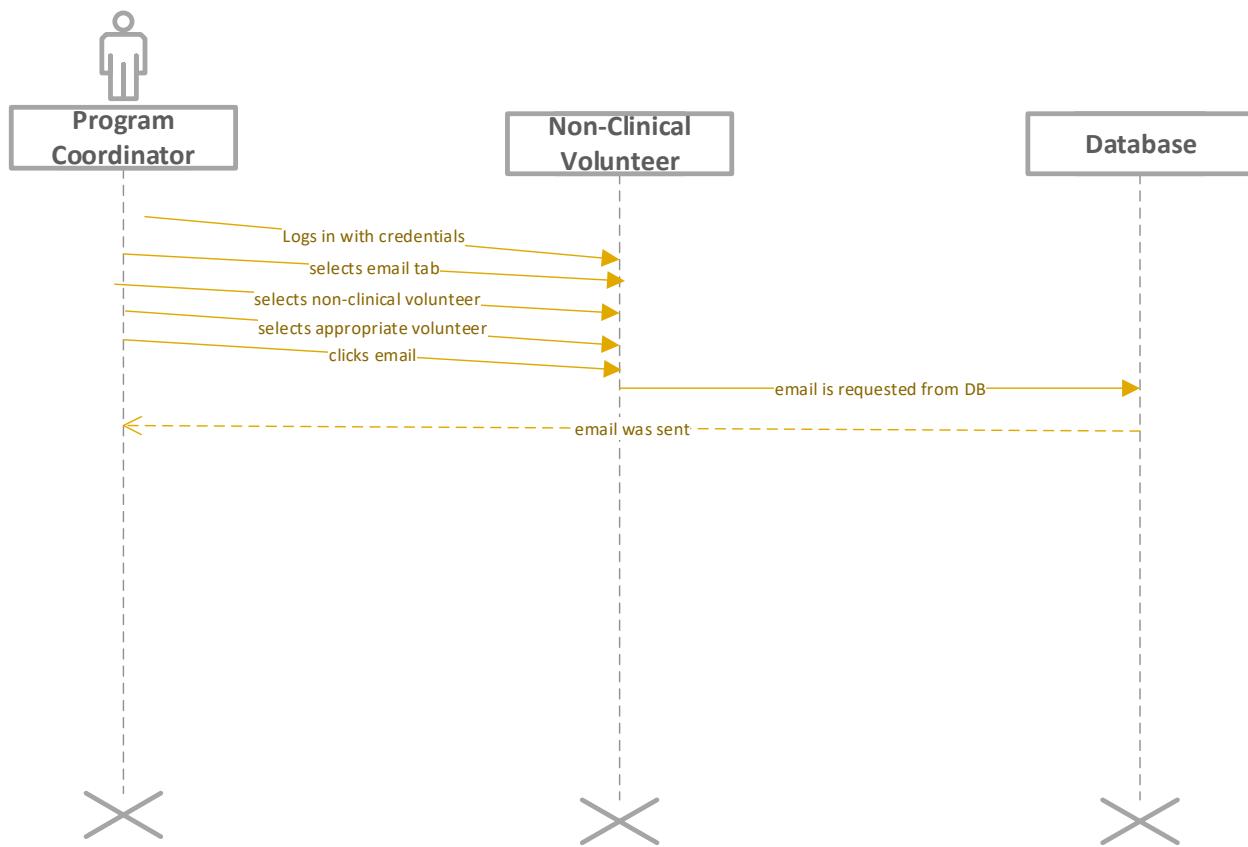
User must be logged into the system with appropriate credentials.

Post-conditions

Email is sent to appropriate email addresses

Extension Points

There are no extension points.



Surgery On Sunday	Version: 1.0
Use Case Specification: Schedule Event With Non-Clinical Volunteers	Date: 10/23/2017
Use Case ID: 39	

Use Case Specification: Schedule Event With Non-Clinical Volunteers

Schedule Event With Non-Clinical Volunteers

Brief Description

The objective of this use case is to show how a program coordinator would go about scheduling an event with the available volunteers. This is necessary so that SOS can appropriately assign the correct non-clinical based on the patient's surgery availability of the volunteer. This way, instead of manually searching through all of the files available on volunteers, they instead access all of the qualified volunteers and can schedule the event with those available and qualified.

Flow of Events

Basic Flow

- A user wants to schedule an event
- Program Coordinator logs in with their authorization
- Program Coordinator navigates to schedule event tab
- Program Coordinator fills out date of event field
- Program Coordinator fills out location of event field
- Program Coordinator fills out patient field
- Program Coordinator fills out selected nurses field
- Program Coordinator fills out surgeon field
- Program Coordinator fills out selected interpreter field
- Program Coordinator navigates to non-clinical volunteer section
- Program Coordinator fills out selected non-clinical volunteers field
- Program Coordinator clicks submit/schedule
- Program Coordinator logs out of system

Alternative Flows

There are no alternative flows

Special Requirements

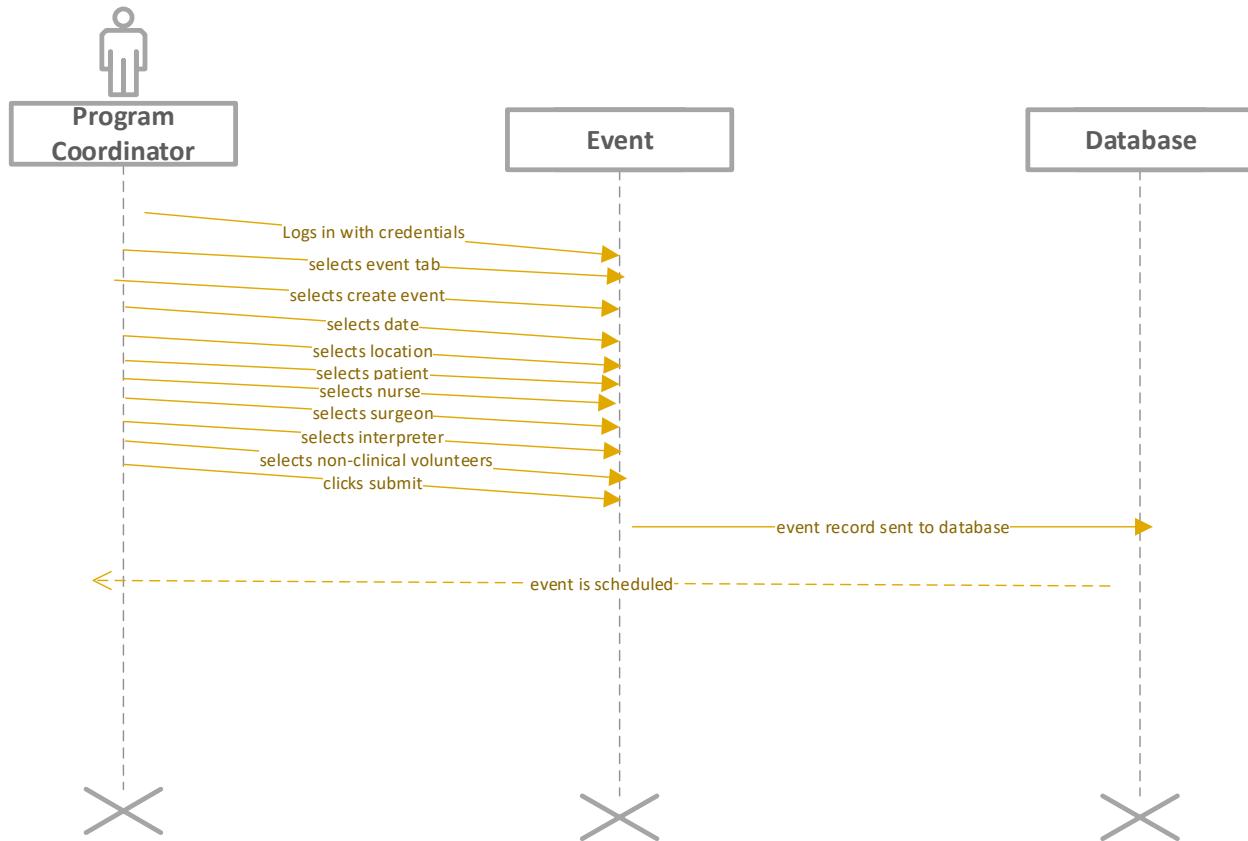
- User must be logged into their account and have the proper authorization to view and access the data
- Data must already exist in database to be accessed

Pre-conditions

User must be logged into the system with appropriate credentials.

Post-conditions

Event is Scheduled



Quality Control	Version: 1.0
Use Case Specification: SendVolunteerThankYouEmails	Date: 10/22/2017
Use Case ID: 40	

Use Case Specification: SendVolunteerThankYouEmails

Use-Case Name

Brief Description

The system sends out thank-you emails to clinical and non-clinical volunteers thanking them for their time and help at surgical events. Sending thank-you emails to volunteers helps build rapport with volunteers. Thank-you emails to volunteers will help in retaining former volunteers because former volunteers may be more likely to volunteer in the future if they feel their volunteer work was appreciated.

Flow of Events

Basic Flow

1. The use case starts when the program coordinator gathers a list of all volunteers from past events.
2. The program coordinator uses an email marketing plugin to send emails. The email marketing plugin already has a list of all volunteers' emails and names. The program coordinator selects the following information from a drop-down menu:
 - 2.1 Volunteer's first name
 - 2.2 Volunteer's last name
 - 2.3 Volunteer's middle name (if applicable) volunteer
 - 2.2 The volunteer ID # will show up on the screen so the program coordinator can ensure the correct volunteer was selected.
3. Once all volunteers are selected, the program coordinator types a thank-you message and clicks "send."
4. Each email will automatically generate each volunteer's name on their respective email and include the generic message.
5. The system displays a message stating emails have been sent and lists the names and email addresses the emails were sent to.
6. A dialog box appears stating emails were sent.
7. Volunteer thank-you emails are sent

Alternative Flows

Cancel

SendVolunteerThankYouEmails: Cancel

The program coordinator cancels the volunteer thank- you email sending process

Special Requirements

The new system will not handle sensitive data about patients

The new system will not violate HIPAA

Pre-conditions

None.

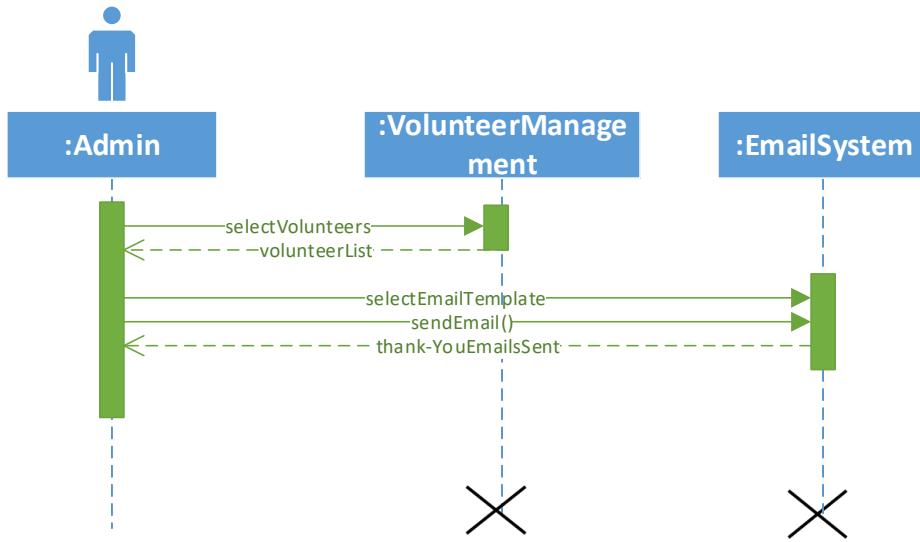
Post-conditions

The system will send Thank You emails to volunteers.

Main Flow:

1. The use case starts when the program coordinator clicks email volunteers.
2. The program coordinator uses an email marketing plugin to send emails. The email marketing plugin already has a list of all volunteers' emails and names. The program coordinator selects the following information from a drop-down menu:
 - 2.1 Volunteer's first name
 - 2.2 Volunteer's last name
 - 2.3 Volunteer's middle name (if applicable) volunteer
 - 2.2 The volunteer ID # will show up on the screen so the program coordinator can ensure the correct volunteer was selected.
3. Once all volunteers are selected, the program coordinator types a thank-you message and clicks "send."
4. Each email will automatically generate each volunteer's name on their respective email and include the generic message.
5. The system displays a message stating emails have been sent and lists the names and email addresses the emails were sent to.
6. A dialog box appears stating emails were sent.

Volunteer thank-you emails are sent



Quality Control	Version: 1.0
Use Case Specification: AddNewEvent	Date: 10/22/2017
Use Case ID: 41	

Use Case Specification: AddNewEvent

Use-Case Name

Brief Description

The program coordinator creates new surgical events in the system. These events assist the program coordinator with keeping track of all events and performing other functions with event information, such as creating reports, scheduling events, and writing grants.

Flow of Events

Basic Flow

1. The use case starts when the program coordinator clicks on ‘Create an Event’ in the admin portal.
2. The program coordinator selects the type of event: It can be a surgical event or a fundraising/other event. The following information is required.
 2. 1 Date and time of the event
 2. 2 Event location, such as city, state, zip
 2. 3 Names of the surgeons and volunteers (clinical and nonclinical) scheduled for the event
 2. 4 For surgical events: type of surgery, the cost of surgery, the total estimated hours
 2. 5 For Fundraising events: the sponsor name, the amount raised, and the budget for the event.
3. After all required information is entered, the program coordinator clicks “Enter”
4. The event is added in the system.

Alternative Flows

Cancel

AddNewEvent: Cancel

Cancel the creation of a new event

Basic Flow

1. The use case starts when the program coordinator clicks on “Cancel” on the Create Event Page in the admin Portal.
2. A dialog box appears asking the user to confirm the cancellation.
3. The user clicks “Yes.”

4. The user cancels event creation.

Post Conditions

A new event has *not* been created

Special Requirements

The new system will not handle sensitive data about patients

The new system will not violate HIPAA

The new system will not interface with the RedCap or Epic Connect patient database

Pre-conditions

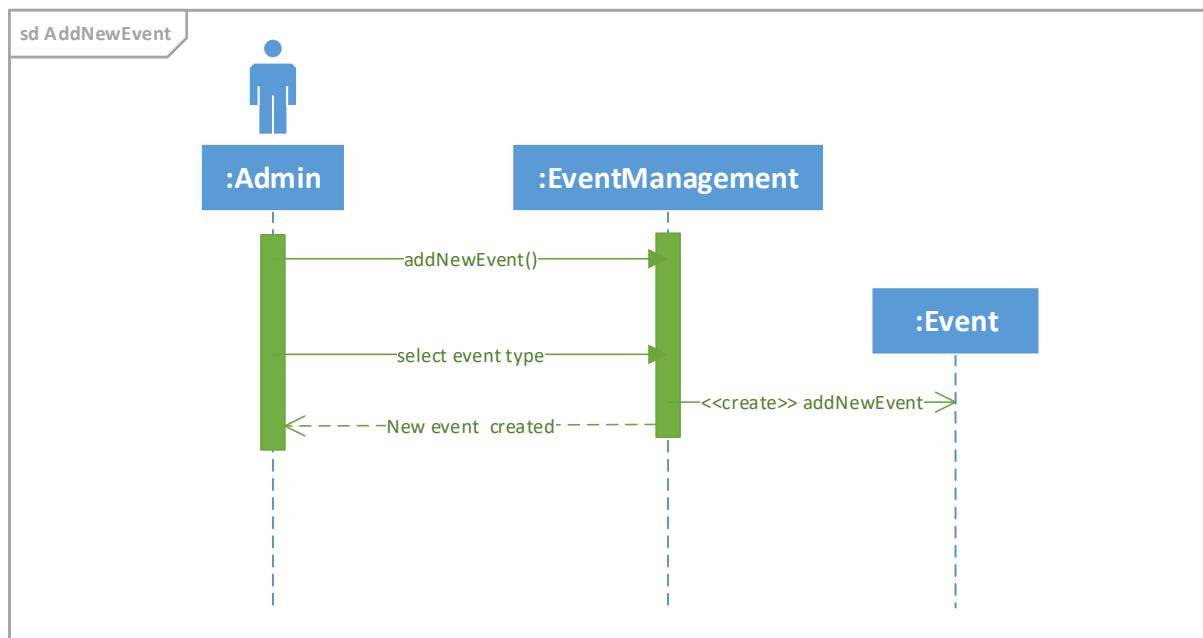
Event information must be gathered, such as the location the event is taking place, the volunteers (clinical and non-clinical) participating, and the total cost of the event.

Post-conditions

A new surgical event or fundraising has been created in the system

Main Flow:

1. The use case starts when the program coordinator clicks on "Create New Event" in the admin portal.
2. The program coordinator enters information about the event. The attributes for the creation of an event are:
 2. 1 Name of the event
 2. 2 Date and time of the surgical event
 2. 3 Surgical event location
 2. 4 Name(s) of the surgeons
 2. 5 Name(s) volunteers (clinical and nonclinical) scheduled for the event
 2. 6 Total cost of the surgical event
 2. 7 Name of Patient
3. After all required information is entered, the program coordinator clicks "Add"
4. The event is added in the system.



Quality Control	Version: 1.0
Use Case Specification: UpdateEventRecord	Date: 10/22/2017
Use Case ID: 42	

Use Case Specification: UpdateEventRecord

Use-Case Name

Brief Description

The program coordinator updates surgical events in the system. Event information may change and the system will need to allow designated users to update and make changes to event information. Updating event records will ensure event information is correct and will help minimize data redundancy.

Flow of Events

Basic Flow

1. The use case starts when the program coordinator clicks on an existing event in the admin portal and clicks 'Update'
2. The system allows the user to update the following information:
 2. 1 Date and/or time
 2. 2 Location information
 2. 3 Volunteer and surgeon information
 2. 4 Event Costs
3. The user clicks update
4. The event is updated in the system

Alternative Flows

Missing Information

UpdateEventRecord: Missing Information.

Enter required missing information

Cancel

UpdateEventRecord: Cancel

Cancel the deletion of an existing event

Special Requirements

The new system will not handle sensitive data about patients

The new system will not violate HIPAA

Pre-conditions

Must select an existing event

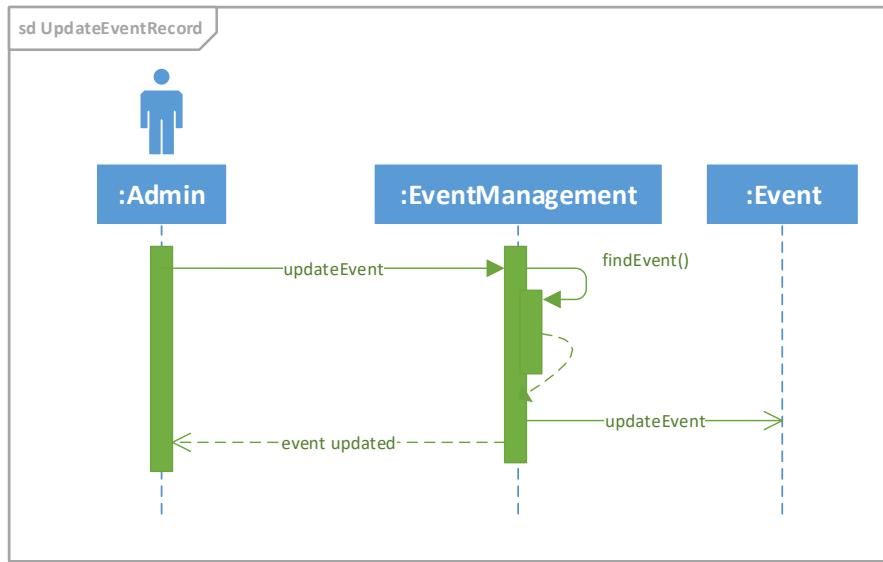
Post-conditions

An event record has been updated in the system

A surgical event is updated in the system

Main Flow:

1. The use case starts when the program coordinator clicks on "Events" in the admin portal and clicks on an existing event
2. The user clicks "Update" to update the event
3. The system allows the user to update the following information: 3.1Date and/or time3.2Location information3.3Volunteer and surgeon information3.4Event Costs
4. The user clicks "Update"
5. The event is updated in the system



Quality Control	Version: 1.0
Use Case Specification: DeleteEventRecord	Date: 10/22/2017
Use Case ID: 43	

Use Case Specification: DeleteEventRecord

Use-Case Name

Brief Description

The program coordinator deletes surgical events in the system. Event information may need to be deleted if an event is canceled or an event was created accidentally. The system needs to allow designated users the ability to delete records to minimize data redundancy. It is important to have data integrity because reports are printed from event information.

Flow of Events

Basic Flow

1. The use case starts when the program coordinator clicks on an existing event in the admin portal.
2. The program coordinator is given an option to update or delete an event
3. The program coordinator selects 'Delete'
4. A dialog box pops up for the user to confirm deletion of the event.
 - 4.1 The dialog box states the action cannot be undone.
5. The user confirms the deletion of the event record by clicking 'yes'
6. The system deletes the event.
7. A dialog box is displayed confirming an event was successfully deleted.
8. An event record is deleted.

Alternative Flows

Cancel

DeleteEventRecord: Cancel

Cancel the deletion of an existing event

Special Requirements

The new system will not handle sensitive data about patients

The new system will not violate HIPAA

Pre-conditions

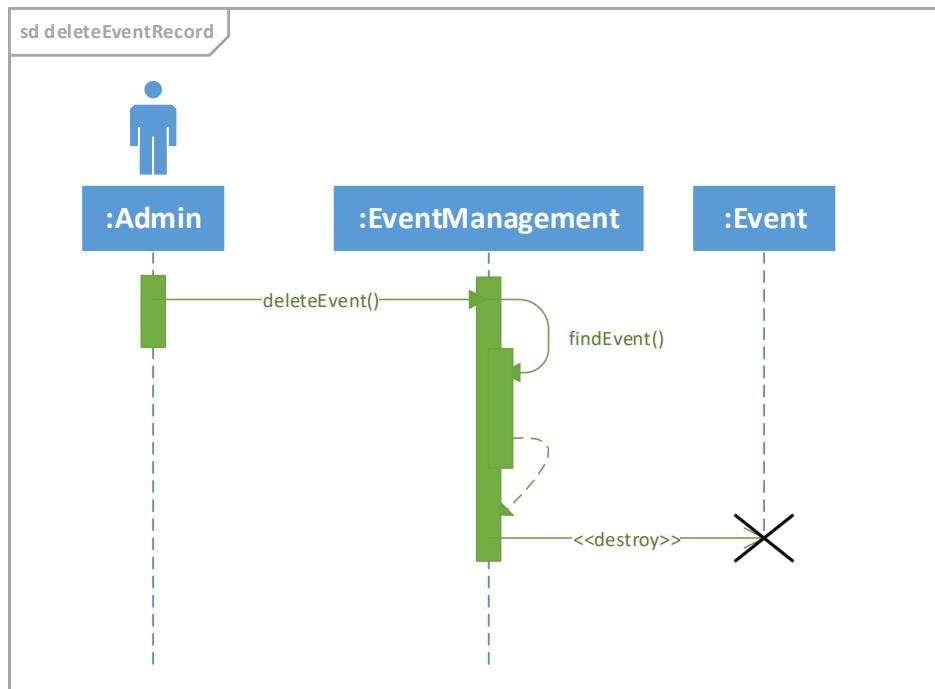
An existing event must be selected. The event must have been entered into the system before it can be selected for deletion.

Post-conditions

An event is deleted from the system

Main Flow:

1. The use case starts when the program coordinator clicks on an existing event in the admin portal.
2. The program coordinator is given an option to update or delete an event
3. The program coordinator selects 'Delete'
4. A dialog box pops up for the user to confirm deletion of the event.
 - 4.1 The dialog box states the action cannot be undone.
5. The user confirms the deletion of the event record by clicking 'yes'
6. The system deletes the event.
7. A dialog box is displayed confirming an event was successfully deleted.
8. An event record is deleted.



Quality Control	Version: 1.0
Use Case Specification: GenerateEventReports	Date: 10/22/2017
Use Case ID: 44	

Use Case Specification: GenerateEventReports

Use-Case Name

Brief Description

The system will generate reports about events for reporting purposes, to assist in writing grants, and to share with management. It is important that the system generate reports so users will not have to manually create time-consuming reports. Reports will assist the program coordinator with writing grants easily and quicker.

Flow of Events

Basic Flow

1. The use case starts when the program coordinator clicks on ‘Generate Reports’ in the admin portal.
2. The program coordinator clicks ‘Events’ from the drop-down menu and selects or types the dates in the date field.
3. There are additional fields the program coordinator can select from when generating a report, such as the following criteria.
 2. 1 Cost parameters
 - 3.2 Type of Surgery performed
4. After all required information is entered, the program coordinator clicks enter and the report is generated and displayed.

Alternative Flows

Cancel

GenerateEventReports: Cancel

Cancel the creation of a new report about events

Special Requirements

The new system will not handle sensitive data about patients

The new system will not violate HIPAA

The new system will not interface with the RedCap or Epic Connect patient database

The new system will allow for disclosure of all volunteers, per the Federal Tort Claims Act

Pre-conditions

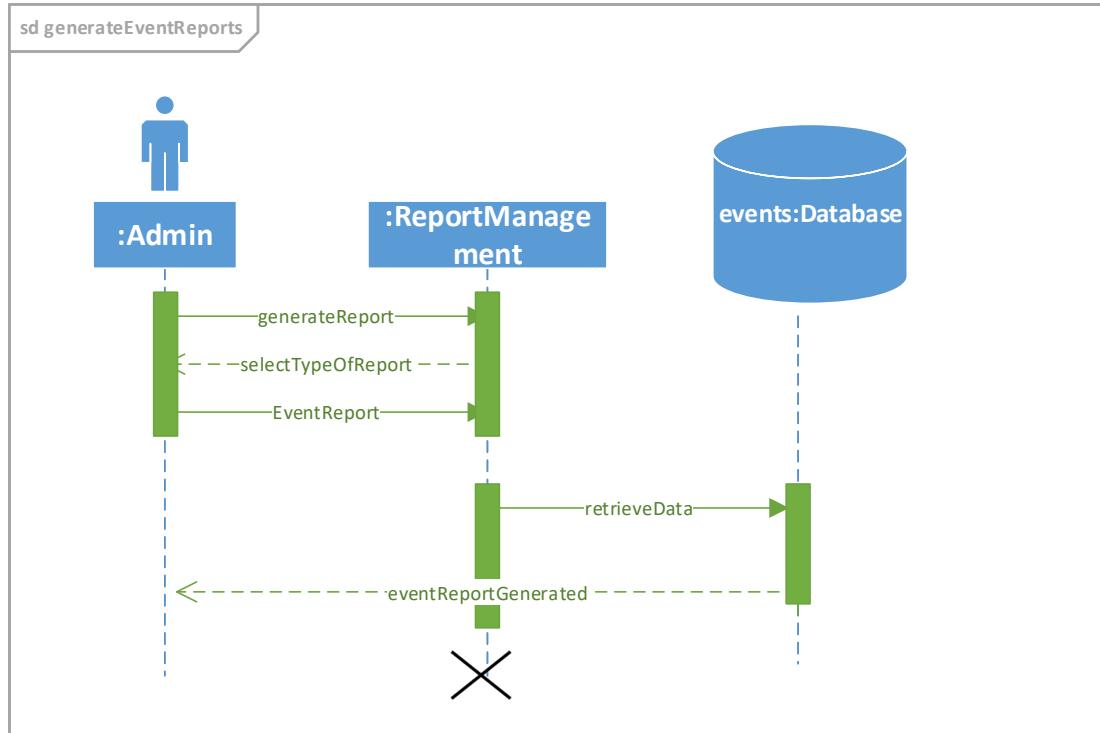
There must be existing event records for the system to generate reports about events Post-conditions

Post-conditions

A new report about events is generated for the user

Main Flow:

1. The use case starts when the program coordinator clicks on 'Generate Reports' in the admin portal.
2. The program coordinator clicks 'Events' from the drop-down menu and selects the type of event: surgical or fundraising event
3. There are additional parameters the program coordinator can select from when generating a report, such as the following:
 - 3.1 Cost parameters for cost of a surgical event
 - 3.2 For surgical events: generate reports by date and time, type of surgery performed. Number of hours for surgery
 - 3.3 For Fundraising events: generate reports by date and time, amount raised, event budget, attendees
4. After entering parameters, the program coordinator clicks enter and the report is generated
The report is displayed on the screen.



Quality Control	Version: 1.0
Use Case Specification: AddAttendance	Date: 10/22/2017
Use Case ID: 45	

Use Case Specification: AddAttendance

Use-Case Name

Brief Description

The program coordinator creates attendance information for volunteers (clinical and nonclinical), surgeons, and other attendees. It is important that the system capture this information for reporting purposes. Keeping track of attendance is also important because other business processes, such as sending thank-you emails, writing grants, and creating reports, depend on this information.

Flow of Events

Basic Flow

1. The use case starts when the program coordinator clicks on ‘Add Attendance’ in the admin portal.
2. The program coordinator selects ‘Sync Attendance data’ so attendance data from I-pad attendance app can be synched into system
 2. 1 Attendance data can be synched via a usb cable connected to a laptop or via Internet.
3. After attendance data is synched, the program coordinator clicks save and attendance data is saved in system

Alternative Flows

None.

Special Requirements

The system will not handle sensitive data about patients

The system will not violate HIPAA

Pre-conditions

I-pad with attendance tracker app must be used during events

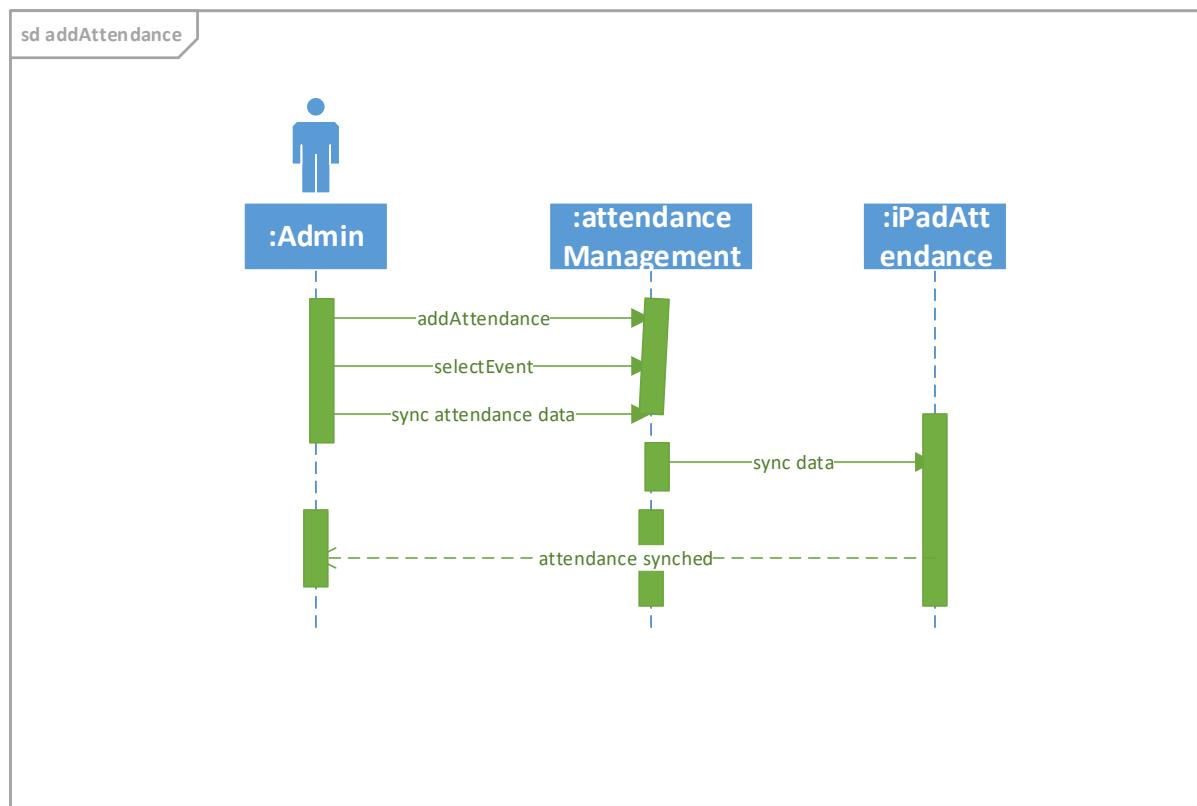
Post-conditions

Attendance data is synced and added in system

Main Flow:

1. The use case starts when the program coordinator clicks on 'Add Attendance' in the admin portal.
2. The program coordinator selects 'Sync Attendance data' so attendance data from I-pad attendance app can be synched into system
3. Attendance data can be synched via a usb cable connected to a laptop or via Internet.

After attendance data is synched, the program coordinator clicks save and attendance data is saved in system



Quality Control	Version: 1.0
Use Case Specification: AddNewPatient	Date: 10/22/2017
Use Case ID: 46	

Use Case Specification: AddNewPatient

Use-Case Name

Brief Description

Patient information is added into the system when a patient's primary care physician submits a patient referral through the online referral system. Adding patient information into the system allows users to track patient non-medical information such as contact information to contact patients for potential surgeries. Users also need the patient name for surgery event reports and other reports.

Flow of Events

Basic Flow

1. The use case starts when a patient's primary care physician submits a patient referral through the online referral system on the soslouisville.org website.
2. The system captures the following patient information:
 - 2.1 First name
 - 2.2 Last name
 - 2.3 Middle name (if applicable)
 - 2.4 Patient's primary Care Physician
 - 2.5 Surgical procedure needed
3. After all required information is entered, the system captures patient information and stores it for event scheduling and planning.

Alternative Flows

MissingInformation

AddNewPatient: MissingInformation

If the physician does not enter all required information, the system informs the user that he or she is missing required information and must add this information in order to submit the patient referral.

Special Requirements

The new system will not handle sensitive data about patients

The new system will not violate HIPAA

Pre-conditions

A patient referral must be filled out by a patient's primary care provider. The patient referral is submitted through the online referral system.

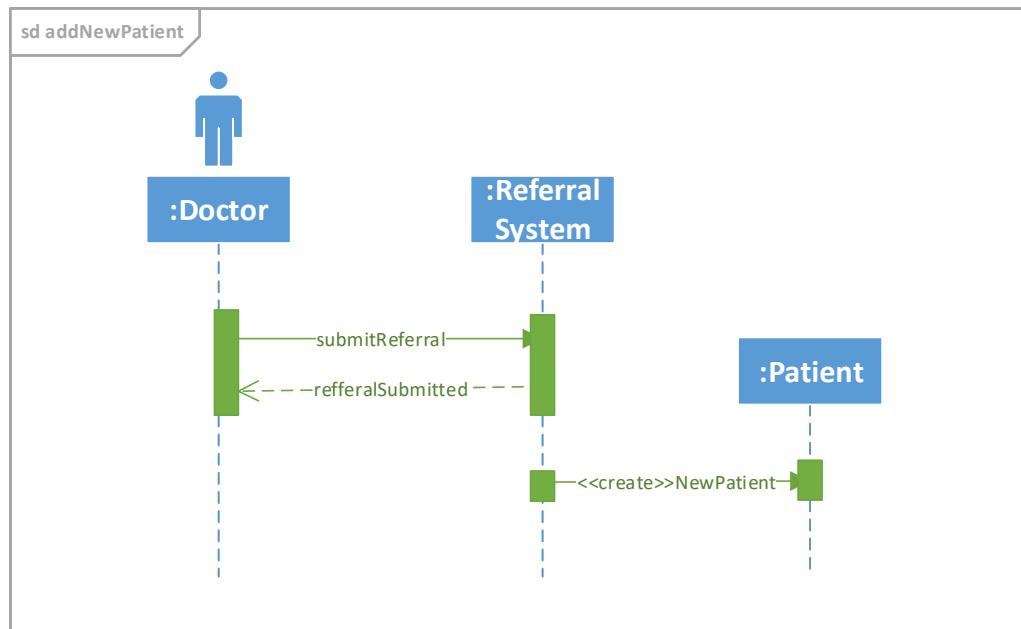
Post-conditions

New patient information is added in the system.

Main Flow:

1. The use case starts when a patient's primary care physician submits a patient referral through the online referral system on the soslouisville.org website.
2. The system captures the following patient information:
 - 2.1 First name
 - 2.2 Last name
 - 2.3 Middle name (if applicable)
 - 2.4 Patient's primary Care Physician
 - 2.5 Surgical procedure needed

After all required information is entered, the system captures patient information and stores it for event scheduling and planning.



Quality Control	Version: 1.0
Use Case Specification: AddNewReferral	Date: 10/22/2017
Use Case ID: 47	

Use Case Specification: AddNewReferral

Use-Case Name

Brief Description

The system creates a new referral record. A referral is submitted by a primary care physician who refers an eligible patient to Surgery on Sunday. It is important to collect this information to keep track of referrals and to collect information about potential surgery patients. Collecting this information will assist with generating reports, scheduling and planning surgeries, and writing grants.

Flow of Events

Basic Flow

4. The use case starts when a patient's primary care physician (PCP) accesses the referral page
5. The primary care physician will key in the code and click "submit"
6. The user will be redirected to the referral submission form
7. The user enters the following information:
 - 4.1 Patient First name
 - 4.2 Patient Last name
 - 4.3 Patient Middle name (if applicable)
 - 4.4 The type of surgery the patient is requesting
 - 4.5 Patient phone number with area code
 - 4.6 Patient email address (if applicable)
 - 4.7 Physician's first name
 - 4.8 Physician's last name
 - 4.9 Physician's middle name (if applicable)
 - 4.10 Physician's phone number with area code
 - 4.11 Physician's email address (if applicable)
8. After the physician has entered the information, the physician will click "submit"
9. The system displays a notification stating the referral was successfully submitted and thanks the physician for their referral
10. The referral is created in the system.

Alternative Flows

InvalidCode

AddNewReferral: InvalidCode

MissingInformation

AddNewReferral: MissingInformation

Cancel

AddNewReferral: Cancel

Special Requirements

The new system will not handle sensitive data about patients

The new system will not violate HIPAA

Pre-conditions

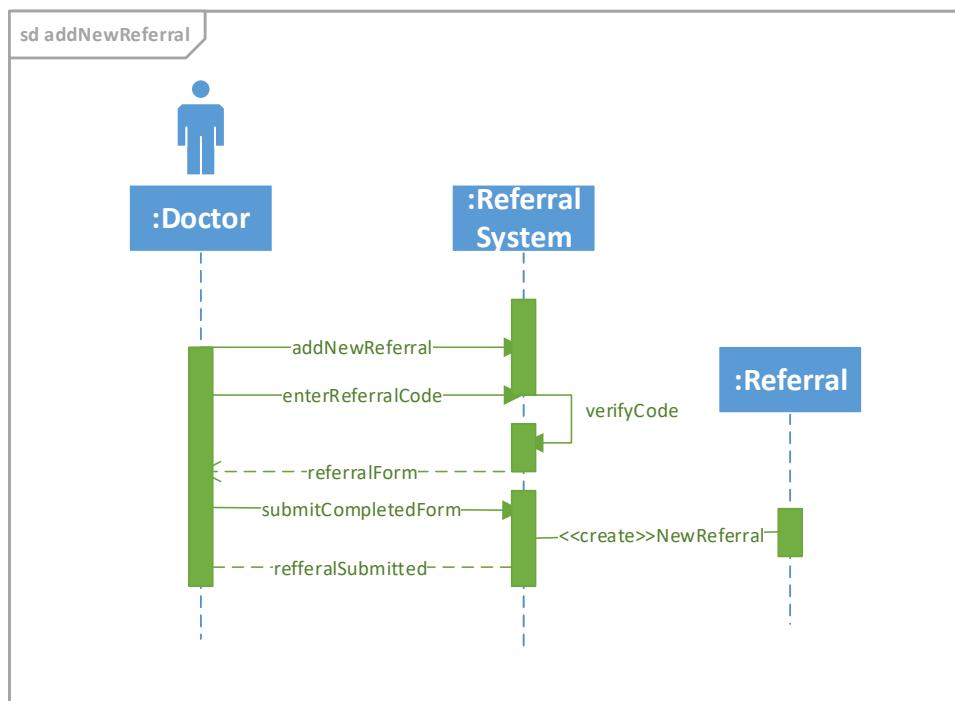
A primary care physician must obtain a code to access the referral form in the online referral system.

Post-conditions

A new referral has been created in the system.

Main Flow:

11. The use case starts when a patient's primary care physician (PCP) accesses the referral page
12. The primary care physician will key in the code and click "submit"
13. The user will be redirected to the referral submission form
14. The user enters the following information:
 - 4.12 Patient First name
 - 4.13 Patient Last name
 - 4.14 Patient Middle name (if applicable)
 - 4.15 The type of surgery the patient is requesting
 - 4.16 Patient phone number with area code
 - 4.17 Patient email address (if applicable)
 - 4.18 Physician's first name
 - 4.19 Physician's last name
 - 4.20 Physician's middle name (if applicable)
 - 4.21 Physician's phone number with area code
 - 4.22 Physician's email address (if applicable)
15. After the physician has entered the information, the physician will click "submit"
16. The system displays a notification stating the referral was successfully submitted and thanks the physician for their referral
17. The referral is created in the system.



Quality Control	Version: 1.0
Use Case Specification: GenerateReferralReports	Date: 10/22/2017
Use Case ID: 48	

Use Case Specification: GenerateReferralReports

Use-Case Name

Brief Description

The system will generate reports about events for reporting purposes, to assist in writing grants, and to share with management. It is important that the system generate reports so users will not have to manually create time-consuming reports. Reports will assist the program coordinator with writing grants easily and quicker.

Flow of Events

Basic Flow

1. The use case starts when the program coordinator clicks on ‘Generate Reports’ in the admin portal.
2. The program coordinator selects ‘Referrals’ from the drop-down menu and
3. There are additional fields the program coordinator can select from when generating a report, such as the following criteria.
 - 3.1 Date(s)
 - 3.2 Type of Surgery performed
4. After all required information is entered, the program coordinator clicks “Enter” and the report is generated
5. The report is displayed on the screen.

Alternative Flows

Cancel

GenerateReferralReports: Cancel

Cancel the creation of a Referrals report

Special Requirements

The new system will not handle sensitive data about patients

The new system will not violate HIPAA

The new system will not interface with the RedCap or Epic Connect patient database

Pre-conditions

There must be existing referral records for the system to generate reports based on these records.

Post-conditions

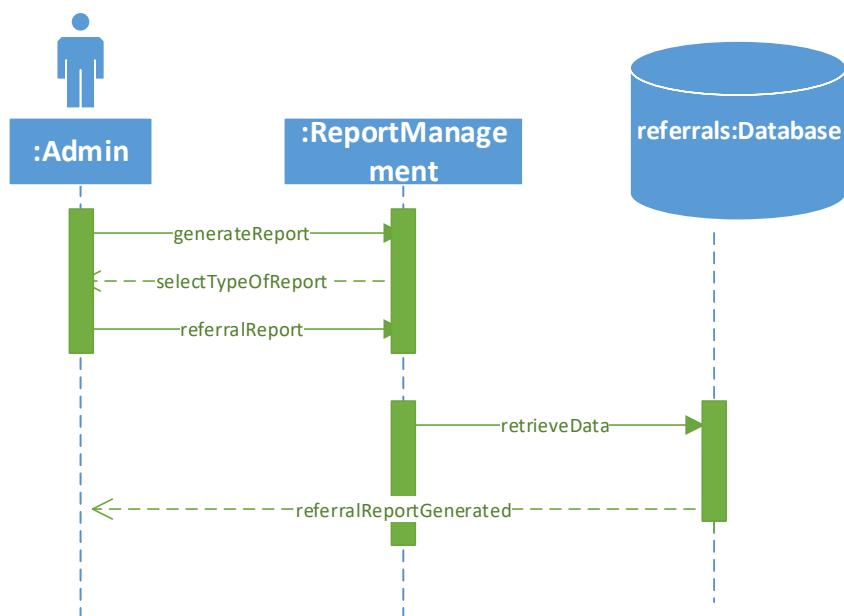
The system will generate a report of referrals

The report will be displayed online which can be exported or saved.

Main Flow:

1. The use case starts when the program coordinator clicks on 'Generate Reports' in the admin portal.
2. The program coordinator selects 'Referrals' from the drop-down menu and
3. There are additional parameters the program coordinator can select from when generating a report, such as the following:
 - 3.1 Date(s)
 - 3.2 Type of Surgery requesting
4. After all required information is entered, the program coordinator clicks "Enter" and the report is generated

The report is displayed on the screen.



Quality Control	Version: 1.0
Use Case Specification: GenerateReferralNotifications	Date: 10/22/2017
Use Case ID: 49	

Use Case Specification: GenerateReferralNotifications

Use-Case Name

Brief Description

The system sends a notification to the program coordinator and/or other users to inform them that a patient referral has been submitted. It is important that notifications are sent so the program coordinator will know a referral was submitted and not find out later on after a period of time has passed. Notifications of referral submissions allows for the program coordinator to plan, review and contact the physician or patient in a timely manner.

Flow of Events

Basic Flow

1. The use case starts when a patient's primary care physician submits a patient referral through the online referral system on the soslouisville.org website.
2. The system will send a notification via text and email to the program coordinator and/or other system users.
3. The email/text notification will state there are new notifications received and pending review.
4. The system will also display a notification when the program coordinator logs into the admin portal. The notification will state there are new notifications received and pending review

Alternative Flows

None

Special Requirements

The new system will not handle sensitive data about patients

The new system will not violate HIPAA

Pre-conditions

A referral is submitted.

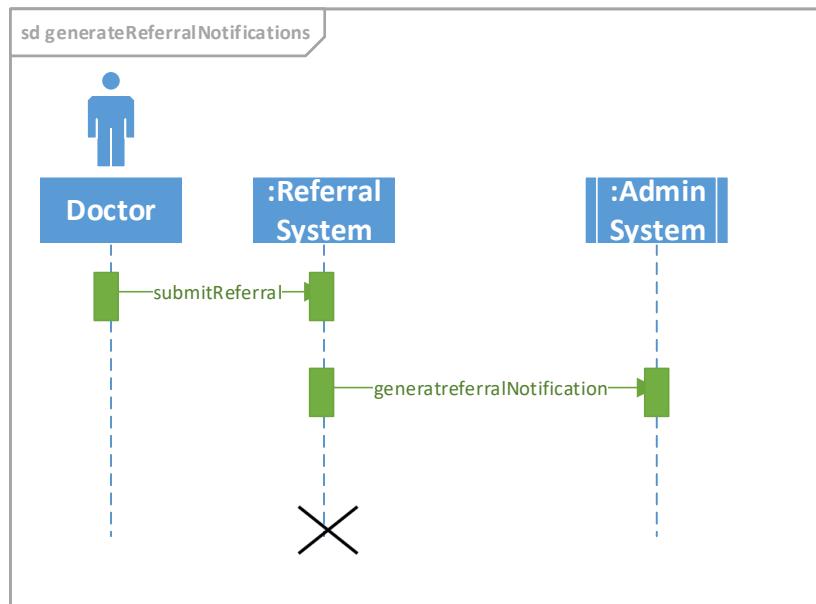
Post-conditions

The system sends referral notifications to the program coordinator via text and email

The system displays notification message on the admin portal after logging in.

Main Flow:

1. The use case starts after a patient's primary care physician submits a patient referral through the online referral system on the soslouisville.org website.
2. The system will send a notification via text and email to the program coordinator and/or other system users.
3. The email/text notification will state there are new notifications received and pending review.
The system will also display a notification when the program coordinator logs into the admin portal. The notification will state there are new notifications received and pending review



SOS Louisville	Version: 1.0
Use Case Specification: LoginAdminPortal	Date: 23/Oct/17
Use Case ID: 50	

Use Case Specification: LoginAdminPortal

1. LoginAdminPortal

1.1 Brief Description

This use case describes how SOS coordinators login to their SOS accounts on the admin portal

2. Flow of Events

2.1 Basic Flow

- 1.** An SOS Admin will enter their Admin Portal username
- 2.** The SOS Admin will enter their Admin Portal password
- 3.** The SOS Admin will click the submit button
- 4.** The SOS Admin is redirected to the Admin Portal

3. Pre-conditions

3.1 The IT Coordinator has authorization to use the Admin Portal

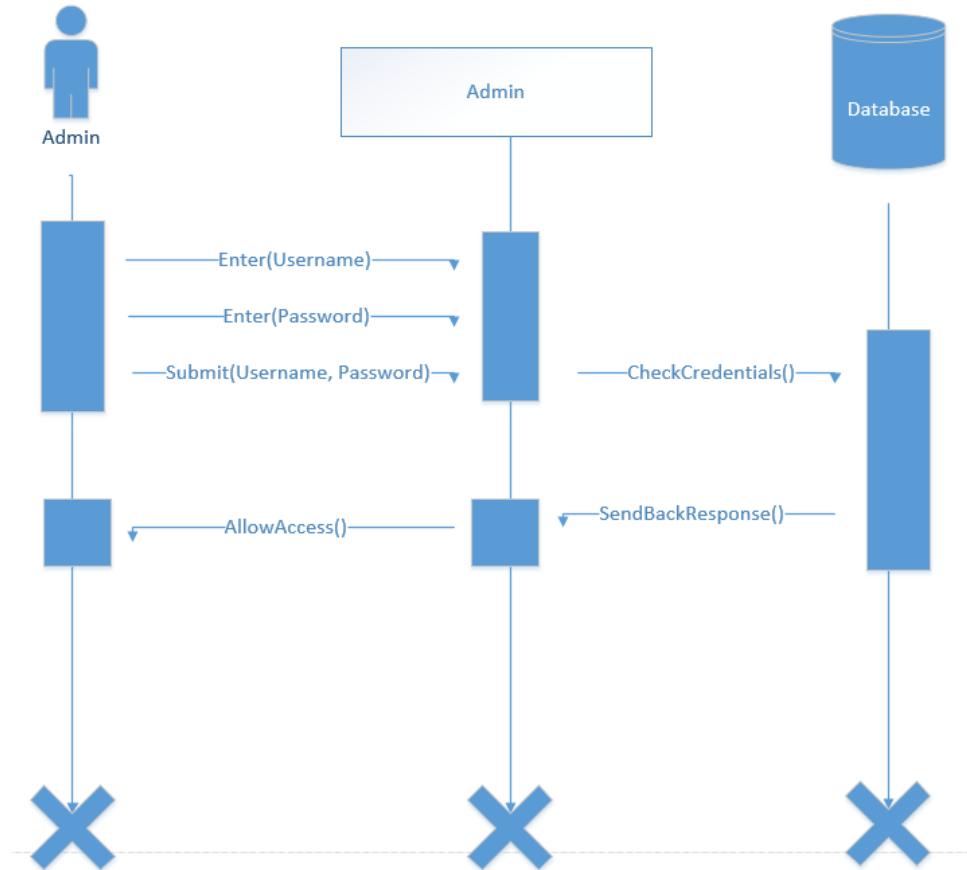
3.2 An Admin Portal exists for SOS Admins

4. Post-conditions

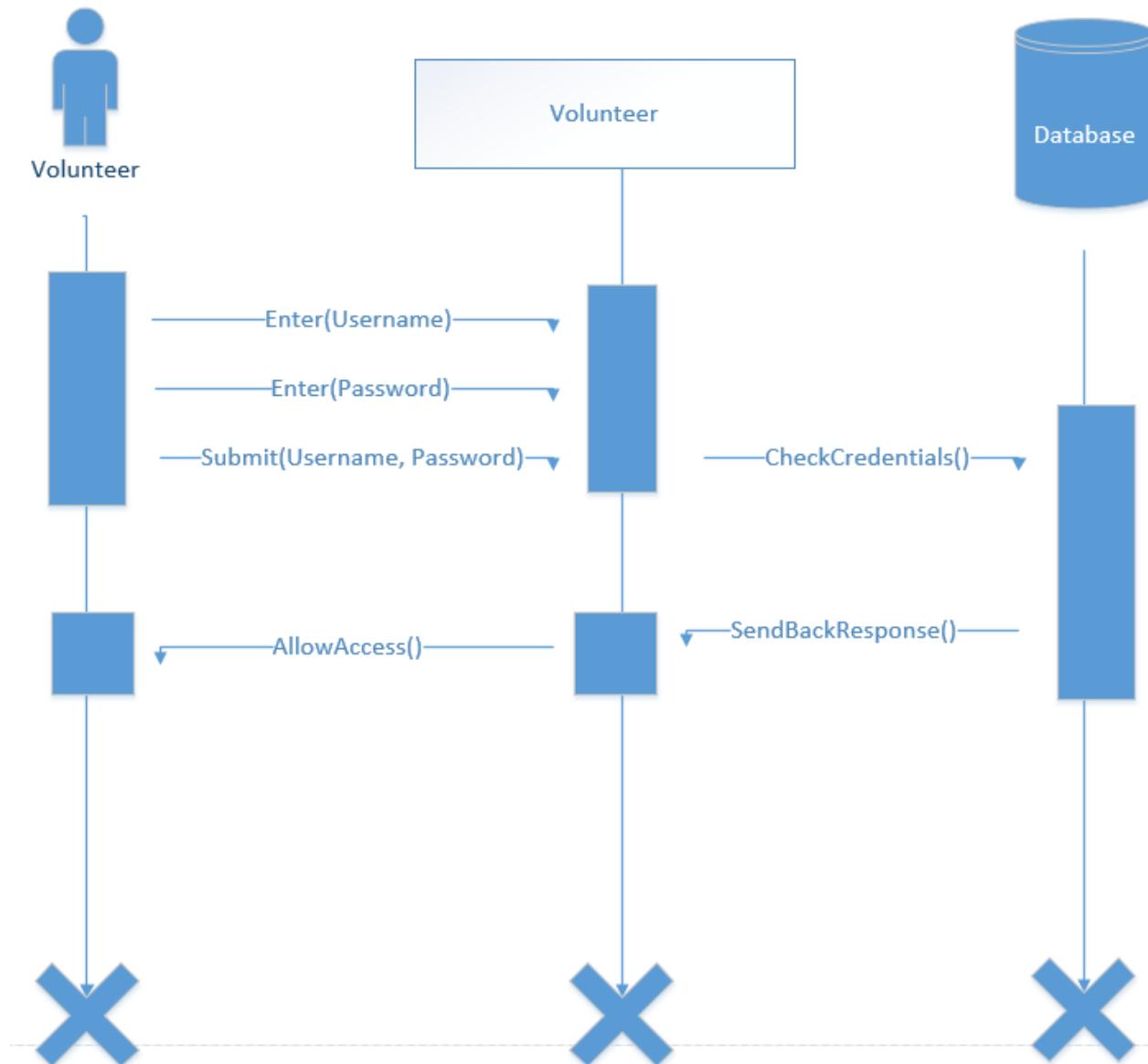
4.1 The SOS staff with Admin privileges will be logged into the Admin portal

The remaining diagrams illustrate the steps taken behind the scenes for the use case to be carried out. The actor fills out and submits fields associated with their class and the object-oriented code communicates with the database or cloud. Then, feedback is provided to the program which gives visual feedback to the user.

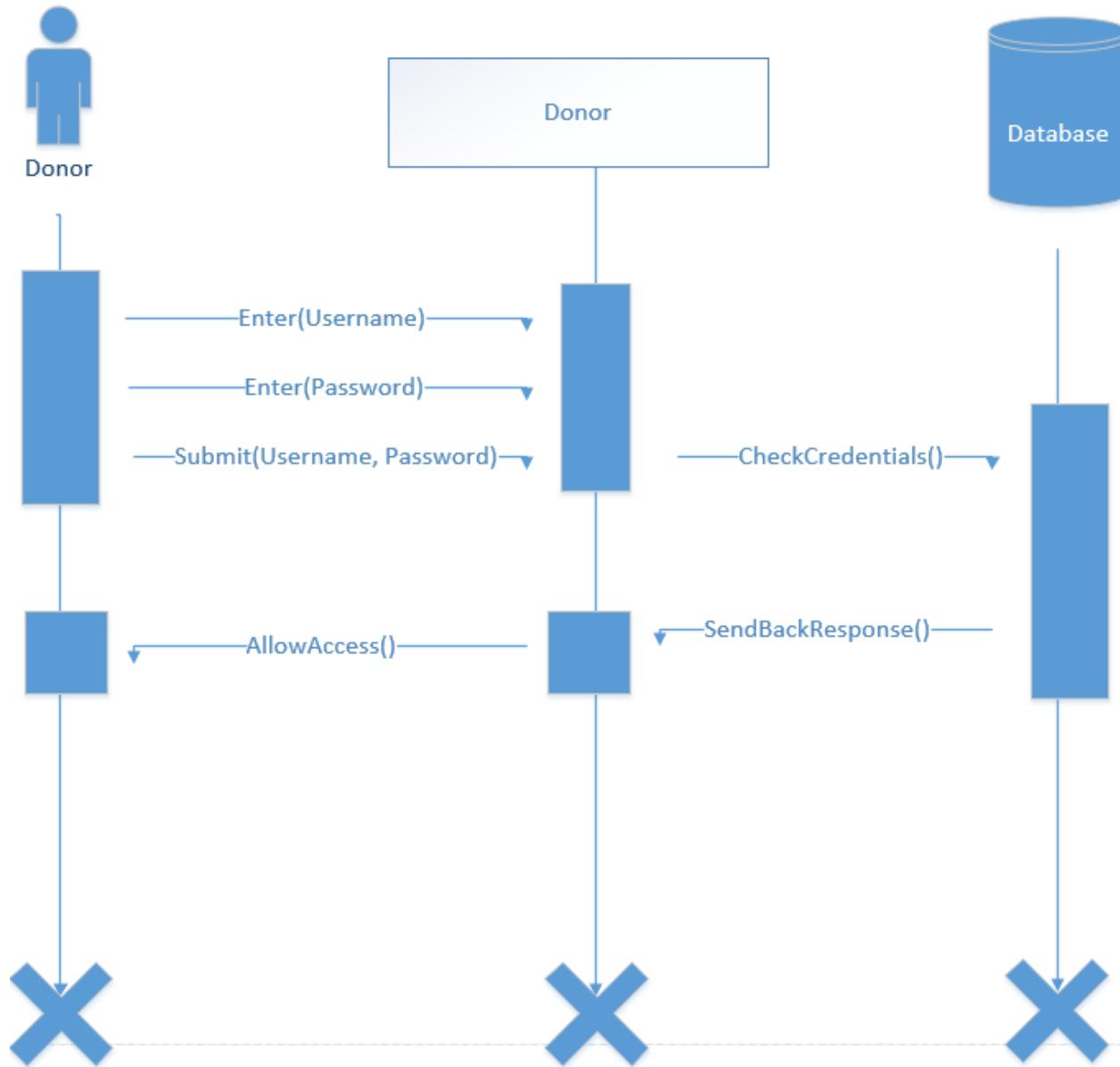
LoginAdminPortal



LoginVolunteers



LoginDonors



Quality Control	Version: 1.0
Use Case Specification:	Date: 10/22/2017
Use Case ID: 53	

Use Case Specification: DisplayReferralInstructions

Use-Case Name

Brief Description

The program coordinator creates attendance information for volunteers (clinical and nonclinical), surgeons, and other attendees. It is important that the system capture this information for reporting purposes. Keeping track of attendance is also important because other business processes, such as sending thank-you emails, writing grants, and creating reports, depend on this information.

Flow of Events

Basic Flow

1. The use case starts when the program coordinator clicks on ‘Add Attendance’ in the admin portal.
2. The program coordinator selects ‘Sync Attendance data’ so attendance data from I-pad attendance app can be synched into system
 2. 1 Attendance data can be synched via a usb cable connected to a laptop or via Internet.
3. After attendance data is synched, the program coordinator clicks save and attendance data is saved in system

Special Requirements

The system will not handle sensitive data about patients

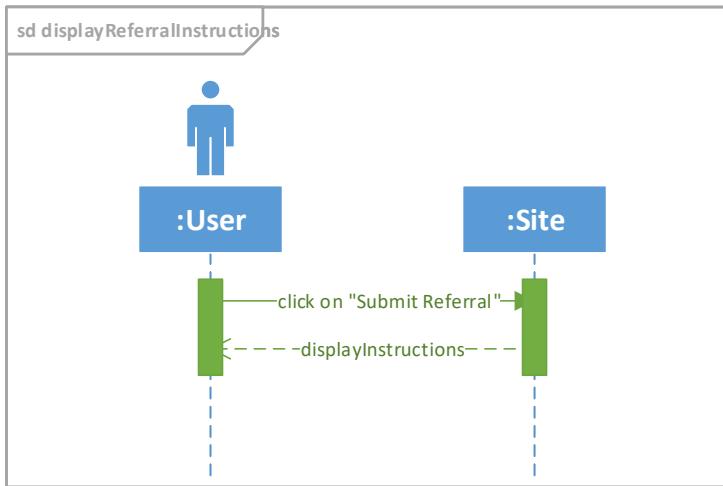
The system will not violate HIPAA

Pre-conditions

I-pad with attendance tracker app must be used during events

Post-conditions

Attendance data is synced and added in system



SOS Louisville	Version: 1.0
Use Case Specification: BackupDatabaseRecords	Date: 23/Oct/17
Use Case ID: 54	

Use Case Specification: BackupDatabaseRecords

1. BackupDatabaseRecords

1.1 Brief Description

This use case describes how SOS data is backed up

2. Flow of Events

2.1 Basic Flow

1. The IT Coordinator opens a batch job or program that automatically backs up data to cloud
2. The IT Coordinator executes batch job or program

2.2 Alternative Flow

1. The IT Coordinator exports non-sensitive data as files
2. The IT Coordinator downloads files as zipped folders in a secure location
3. The IT Coordinator uploads files to secure cloud storage or hard disk location

3. Pre-conditions

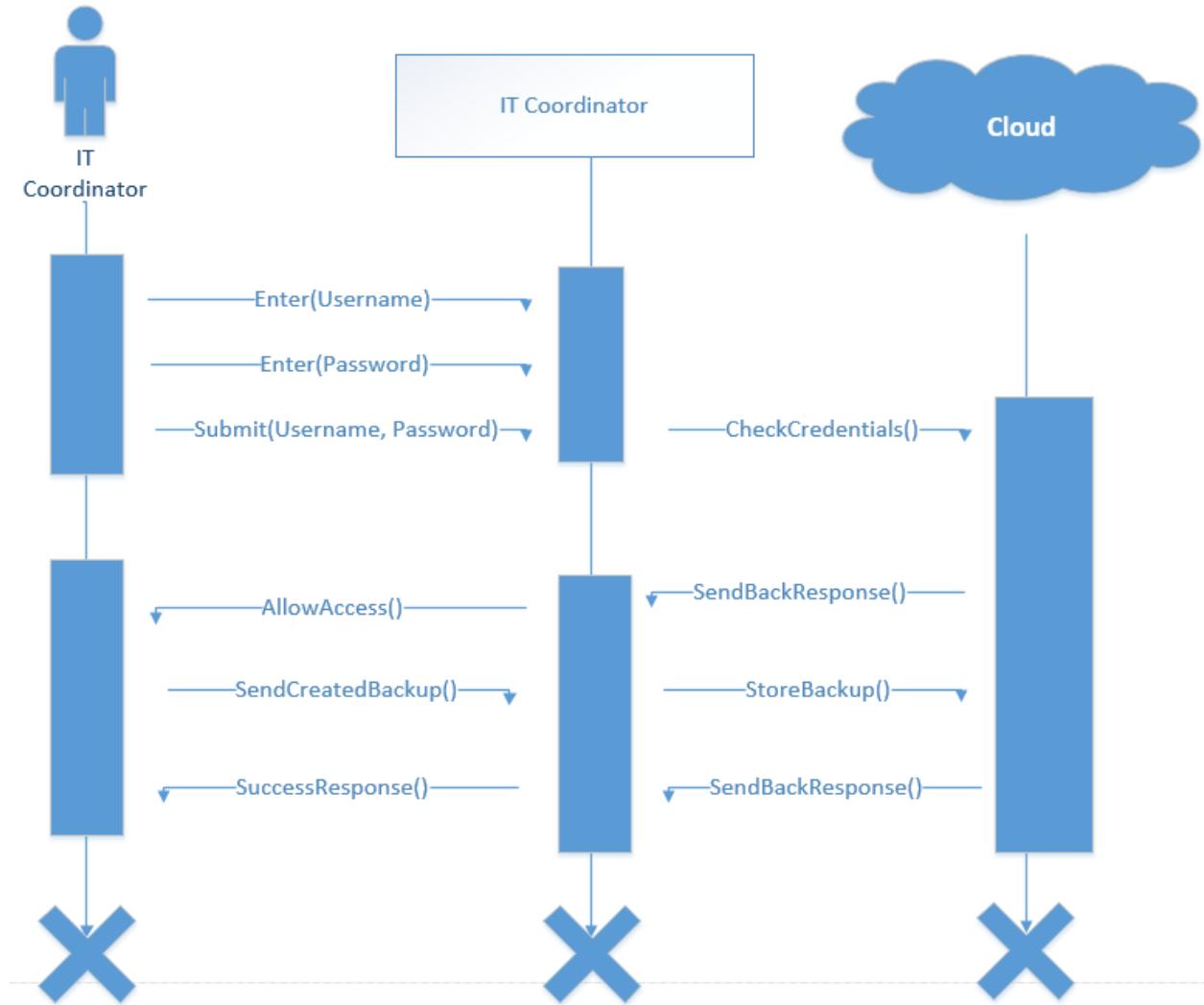
3.1 The IT Coordinator has access to the additional storage

3.2 The IT Coordinator knows how to back up the database to the additional storage

4. Post-conditions

4.1 The database will be backed up in a safe location

The remaining diagrams illustrate the steps taken behind the scenes for the use case to be carried out. The actor fills out and submits fields associated with their class and the object-oriented code communicates with the database or cloud. Then, feedback is provided to the program which gives visual feedback to the user.



SOS Louisville	Version: 1.0
Use Case Specification: RecoverDatabaseRecords	Date: 23/Oct/17
Use Case ID: 55	

Use Case Specification: RecoverDatabaseRecords

1. RecoverDatabaseRecords

1.1 Brief Description

This use case describes how SOS data is recovered in case of data loss

2. Flow of Events

2.1 Basic Flow

1. The IT Coordinator navigates to the batch job or program that automatically recovers database backup from the cloud
2. The IT Coordinator executes the batch job or program
3. The IT Coordinator makes sure data was restored correctly

2.2 Alternative Flow

1. The IT Coordinator navigates to physical backed up files from cloud or hard disk storage
2. The IT Coordinator downloads files from the cloud
3. The IT Coordinator imports the data into the database tables

3. Pre-conditions

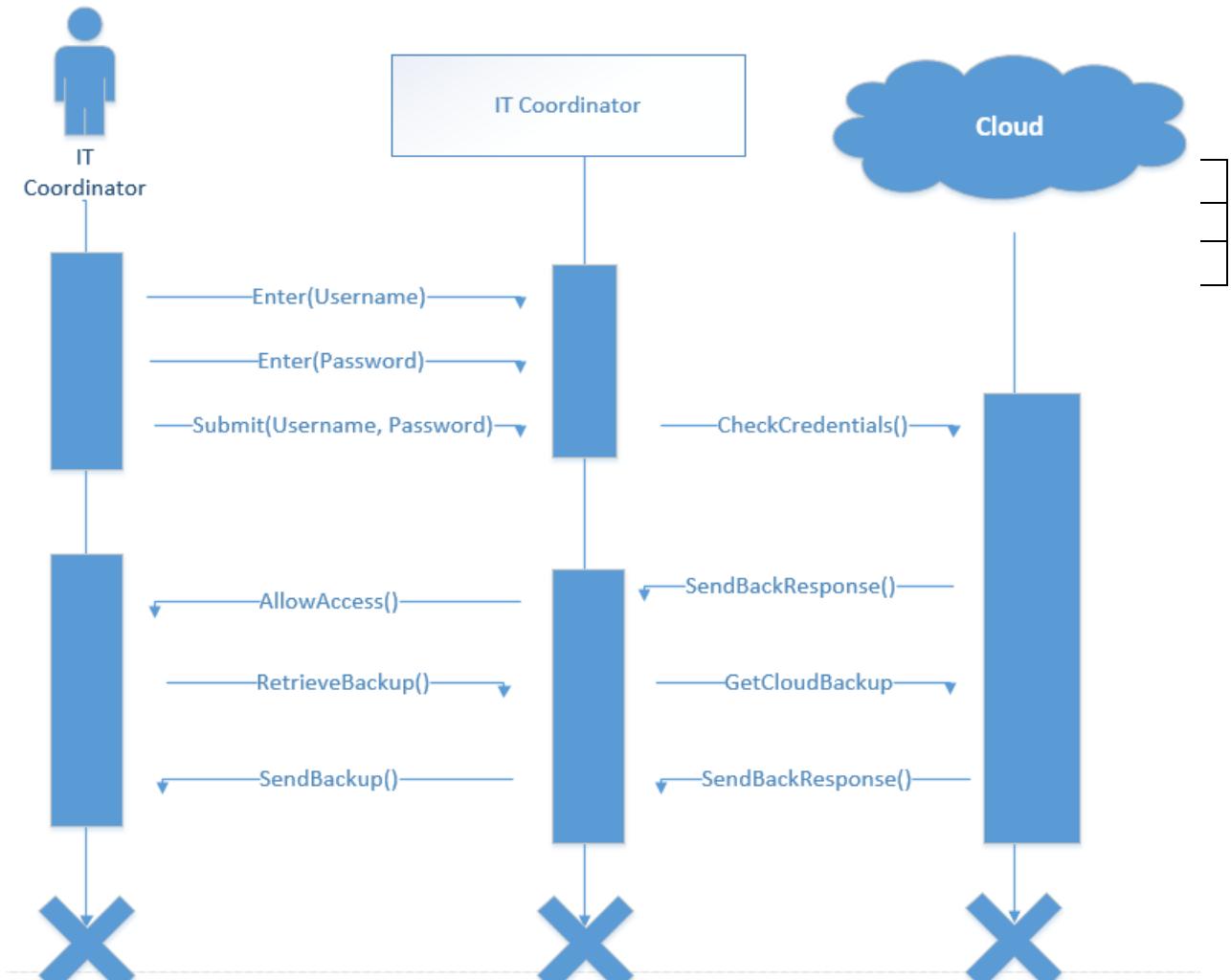
3.1 The database data is damaged and in need of recovery

3.2 The IT Coordinator knows how to recover the database data

4. Post-conditions

4.1 The database will be recovered and restored

The remaining diagrams illustrate the steps taken behind the scenes for the use case to be carried out. The actor fills out and submits fields associated with their class and the object-oriented code communicates with the database or cloud. Then, feedback is provided to the program which gives visual feedback to the user.



SOS Louisville	Version: 1.0
Use Case Specification: AccessDatabaseRecords	Date: 23/Oct/17
Use Case ID: 57	

Use Case Specification: AccessDatabaseRecords

1. AccessDatabaseRecords

1.1 Brief Description

This use case describes how the IT Coordinator will access the database directly if needed

2. Flow of Events

2.1 Basic Flow

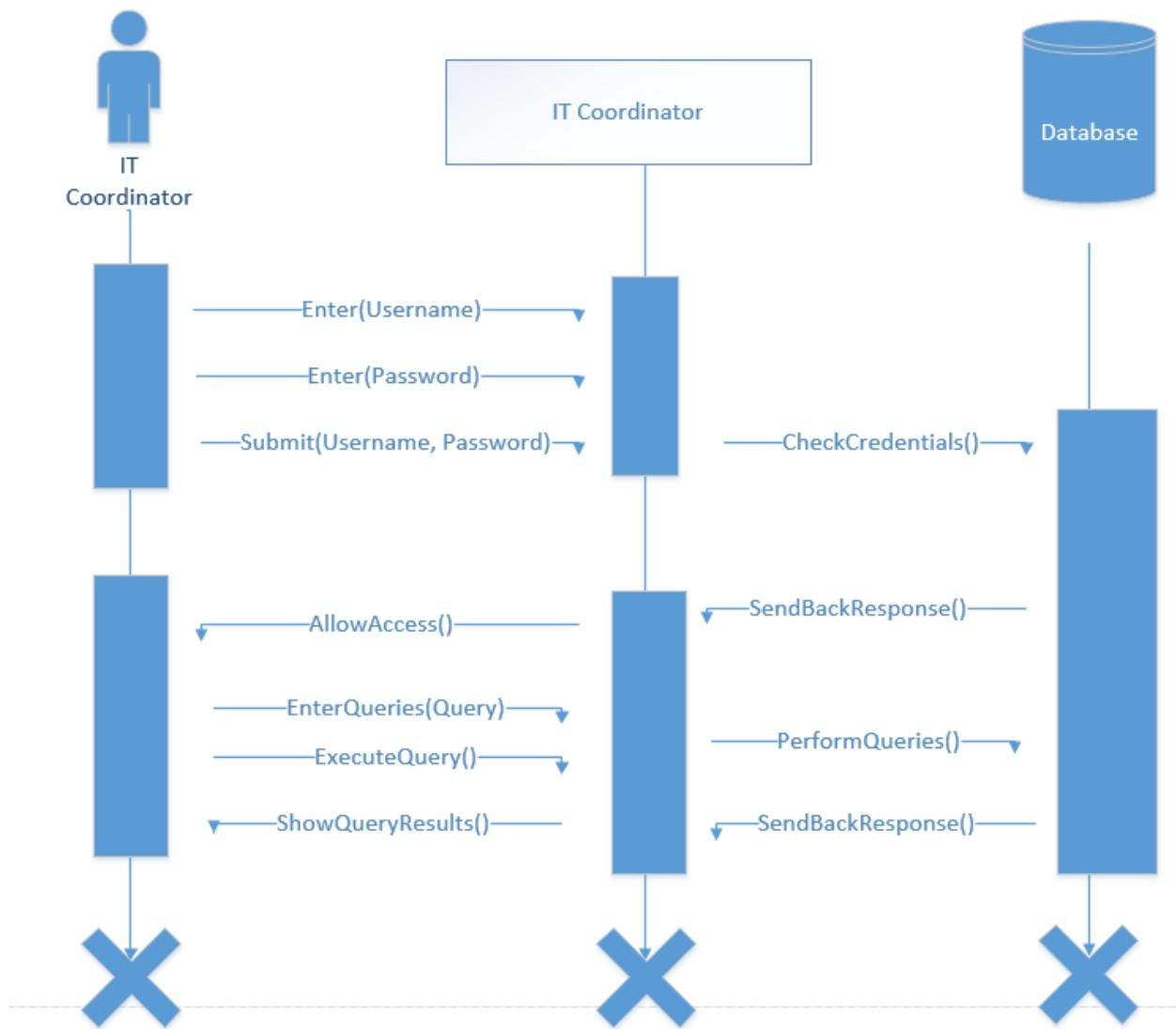
1. The IT Coordinator will enter their MySQL username
2. The IT Coordinator will enter their MySQL password
3. The IT Coordinator clicks the submit button
4. The IT Coordinator enters queries
5. The IT Coordinator executes queries

3. Pre-conditions

3.1 The IT Coordinator must have access privileges with a database login

3.2 The IT Coordinator is familiar with basic queries to access data

The remaining diagrams illustrate the steps taken behind the scenes for the use case to be carried out. The actor fills out and submits fields associated with their class and the object-oriented code communicates with the database or cloud. Then, feedback is provided to the program which gives visual feedback to the user.



HOW TO READ THE CLASS DIAGRAM

The class diagram shows the different classes, the relationships between the classes, and the attributes and operations for each class. The class compartments are shown in Figure1. A *class* represents a kind of person, place or thing about which the system will need to capture and store information. An *attribute* represents properties that describe the state of an object. An *operation* represents the actions or functions that a class can perform. Lastly, the class *relationships* are depicted on the diagram with lines. Lines with arrows depict a generalization relationship with the arrow pointing at the superclass. Lines with a white diamond are used to depict logical a-part-of relationships known as aggregation relationships. An association relationship between classes is depicted by a solid line without arrows or diamonds at the end.

The class diagram was created by performing textual analysis. Our team analyzed use-case descriptions as well as the use-case diagrams to identify the necessary objects, attributes, methods, and relationships for the class diagram. The normal flow of events for use cases were also analyzed and all use-case scenarios were discussed with team members to identify other potential objects, attributes, operations, and relationships.

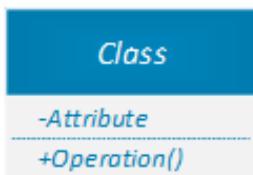
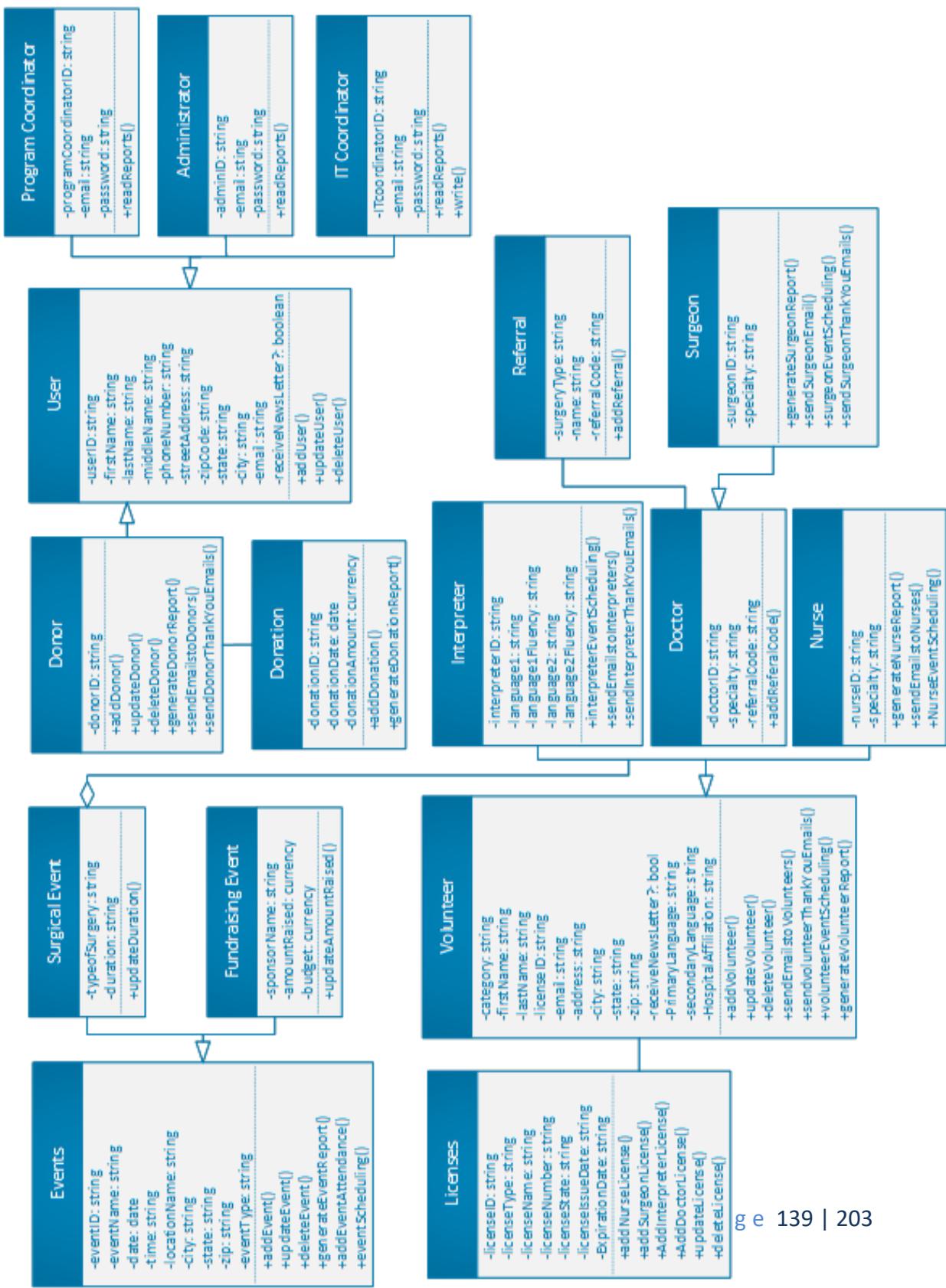


FIGURE 1

CLASS DIAGRAM



SOS DATABASE DESIGN

The following diagram and data types outline the specifications for creating the database that will support the Surgery on Sunday use cases. The database design follows the standard 3rd Normal Form for eliminating redundancy and data inconsistency. This was achieved by removing partial and transitive dependencies (non-key attributes that depend on the primary key or another non-key attribute).

Instead of having separate tables for each kind of volunteer, there is a Category table that will contain fixed volunteer categories such as “surgeon” or “nurse” as the descriptions. The Specialty table referenced by the Volunteer table will contain descriptions that indicate the specialty of the volunteer, such as “anesthesia” or “endoscopic”. This will aid the smart event-creating program in displaying volunteers with the specialties needed for a specific kind of surgery or event. The Events are divided into two types: Surgical and Fundraising. Each has additional attributes to record information specific to the event type. The User table is similar in concept to the Volunteer table, because there are not tables for each type of User, since the same information is stored for all users. Instead, the UserType table will hold descriptions such as “Admin” or “IT Coordinator”. When referenced by the Volunteer table, these user types can be used to give access or authorization to different types of users. The Donor and Donation relation is simplistic and contains a reference to the User table, since Donors may also be users of the SOS website login.

ENTITY RELATIONSHIP DIAGRAM



DATA DEFINITIONS

Patient

Attribute	Type	Size (bytes)
PatientID	Integer	4
FirstName	String (20)	20
LastName	String (30)	30
MiddleName	String (20)	20
Email	String (50)	50
PhoneNumber	String (10)	10
ReceiveNewsletter	Boolean	1

Event

Attribute	Type	Size (bytes)
EventID	Integer	4
EventType	String (20)	20
EventDate	Date	3
StartTime	String (10)	10
Duration	Float	8
Address	String (30)	30
State	String (20)	20
City	String (40)	40
Zip	String (10)	10

SurgicalEvent

Attribute	Type	Size (bytes)
EventID	Integer	4
PatientID	Integer	4
VolunteerTeamID	Integer	4

FundraisingEvent

Attribute	Type	Size (bytes)
EventID	Integer	4
VolunteerTeamID	Integer	4
SponsorName	String (50)	50
AmountRaised	Decimal	9
Budget	Decimal	9

VolunteerTeam

Attribute	Type	Size (bytes)
VolunteerTeamID	Integer	4
VolunteerID	Integer	4

Volunteer

Attribute	Type	Size (bytes)
VolunteerID	Integer	4
CategoryID	Integer	4
SpecialtyID	Integer	4
FirstName	String (20)	20
LastName	String (30)	30
Email	String (50)	50
Address	String (20)	20
City	String (40)	40
Zip	String (10)	10
ReceiveNewsletter	Boolean	1
PrimaryLanguage	String (15)	15
SecondaryLanguage	String (15)	15
HospitalAffiliation	String (50)	40

Category

Attribute	Type	Size (bytes)
CategoryID	Integer	4
CategoryDescr	String (20)	20

Specialty

Attribute	Type	Size (bytes)
SpecialtyID	Integer	4
SpecialtyDescr	String (30)	30

License

Attribute	Type	Size (bytes)
LicenseID	String (30)	30
IssuedDate	Date	3
ExpDate	Date	3
State	String (20)	20

User

Attribute	Type	Size (bytes)
UserID	Integer	4
UserTypeID	Integer	4
FirstName	String (20)	20
LastName	String (30)	30
PhoneNumber	String (10)	10
Email	String (50)	50
ReceiveNewsletter	Boolean	1

UserType

Attribute	Type	Size (bytes)
UserTypeID	Integer	4
UserTypeDescr	String (15)	15

Donor

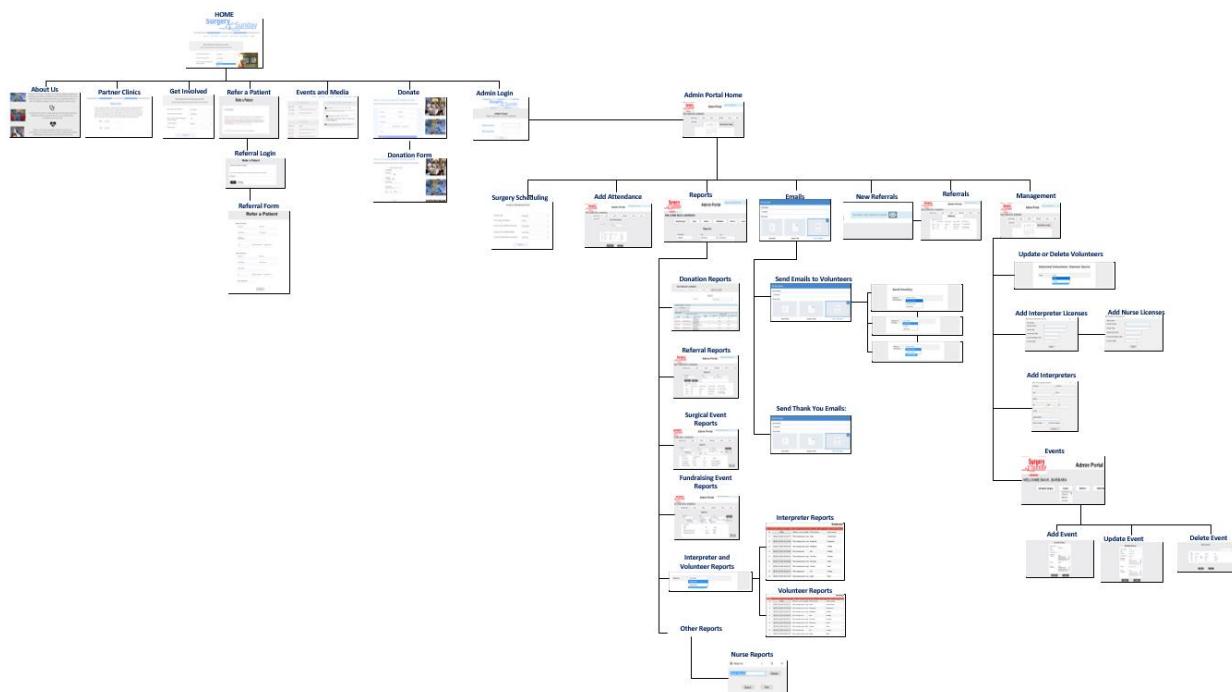
Attribute	Type	Size (bytes)
DonorID	Integer	4
UserID	Integer	4
TotalDonated	Decimal	9

Donation

Attribute	Type	Size (bytes)
DonationID	Integer	4
DonorID	Integer	4
Amount	Decimal	9
DonatedDate	Date	3

USER INTERFACE NAVIGATION DIAGRAM AND SCREEN LAYOUTS

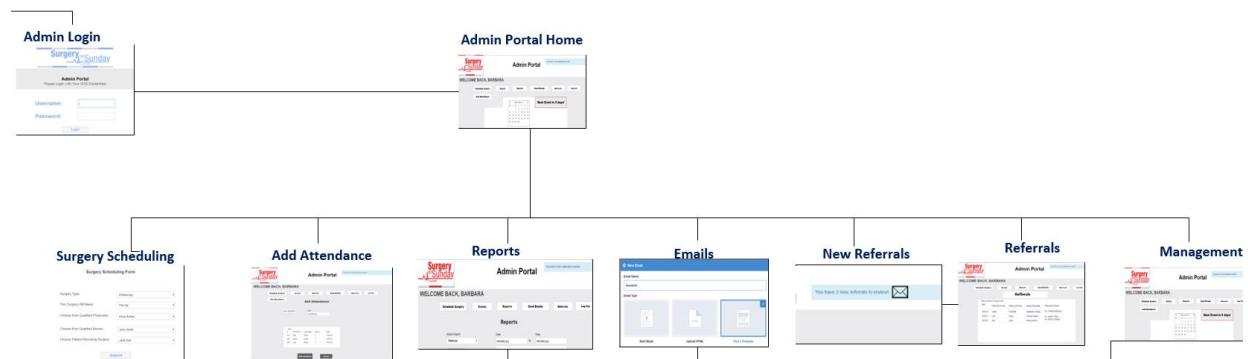
The user interface Navigation diagram models the interactions the user has with the webpage. It shows screen layouts for every link. It displays buttons or links that a user clicks on to navigate the SOS Louisville website. The lines show how screens are connected to each other. The links on the homepage are “About Us,” “Partner Clinics,” “Get Involved,” “Refer a Patient,” “Events and Media,” and “Donate.”



Site Layout - Home

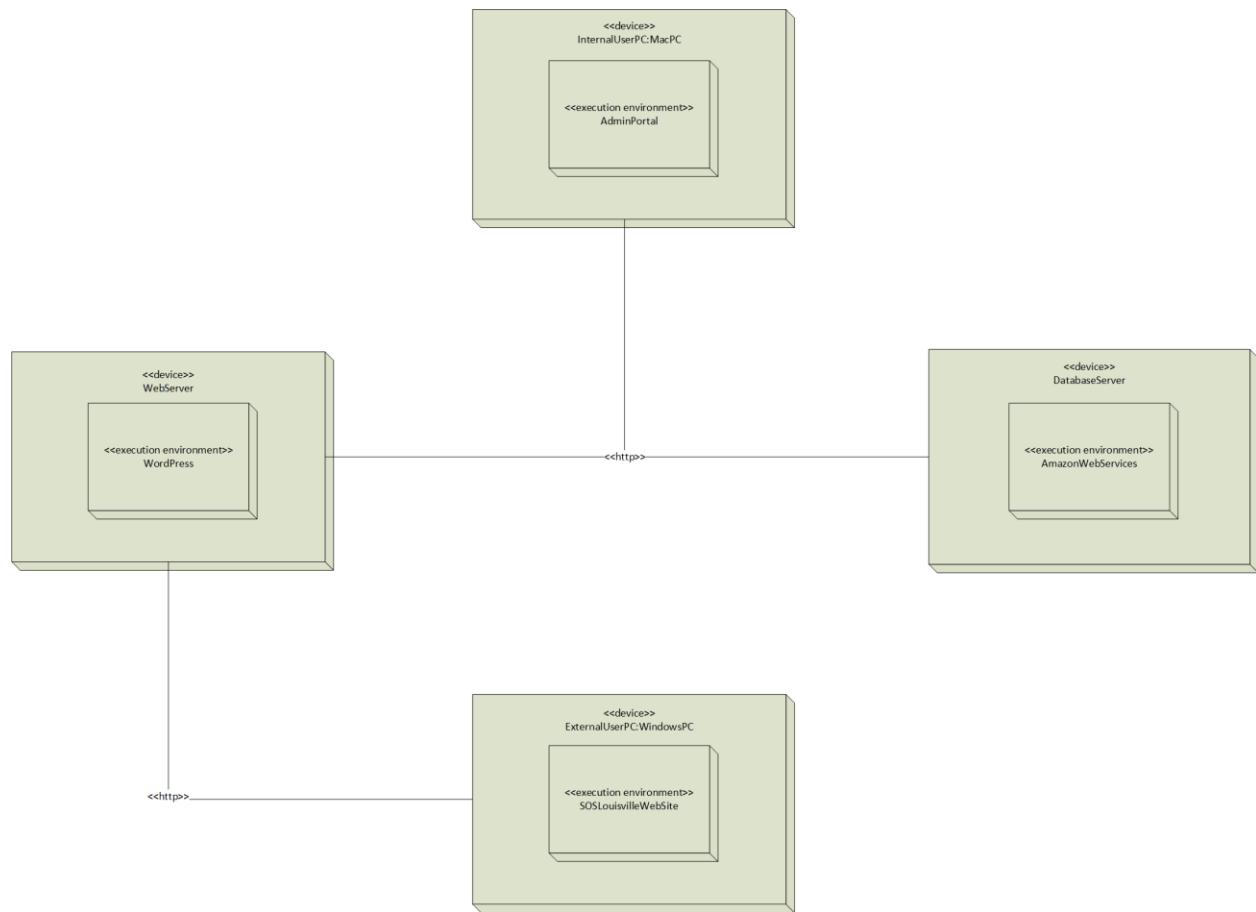


Site Layout – Admin Portal



DEPLOYMENT DIAGRAM

The following diagram illustrates the relationships between the software implemented in the new system. We did not add any hardware because all services are web-based and accessed with SOS's personal computers. The diagram also shows how the different web services communicate to accomplish the goals of your organization.



NON-FUNCTIONAL REQUIREMENTS DESIGN PROCEDURES

Availability

57	The new system will be available every day	High
58	The new system will not crash if website traffic increases	High
59	The new system's website will be available to the public	High
60	The new system will be in place by the end of Spring 2018, unless otherwise stated	Medium

The system will be available every day, will not crash, will be readily available, and will be implemented on time by first using AWS to handle all the server load required to run the system. AWS is very reputable and handles security better than any small server or development team could handle on their end. It also makes sure that these deployable services are always available and ready to use. AWS will ensure that the extra traffic on our system's deployable services will not create a problem by using queuing if the server load is too heavy that way it will instantiate a service in order instead of trying all at once. We will also host these backend services if needed on a software like docker that is not specific to any OS and will not crash based on a user's device operating system.

Storage

61	The new system's data storage will be available to SOS coordinators only	High
62	The new system will provide enough data storage	High
63	The new system will use efficient database design and relationships	Medium
64	The new system's data will be backed up	High

We will create encryption keys in order to access the AWS's S3 storage container. Only people with a certain access key will have writing permissions, while others only have read permissions, while others may not access it at all. AWS provides system storage that is on a basis where you only pay for what you need. The amount of storage you can upgrade to is virtually infinite and will never be an issue for a company/organization such as SOS. The database will

maintain efficiency by eliminating data redundancy and data anomalies by maintaining a strict database design which we will implement, that way data is easily accessible and efficient. The system will also be backed up locally to avoid losing data with a missing internet connection and virtually to avoid loss of the backup incase of device failure.

Maintenance

65	The new system will be easy to maintain	High
66	The new system will be thoroughly tested and not break or have bugs	High
67	The new system maintenance will be available to the IT coordinator	Medium

The system will be updatable using versioning of the system that we can test and manipulate without effecting the current in-use system. Then when we have written enough tests to ensure the update is in fact creating a better system, we can deploy that version to the in-use system. Bugs will be eliminated by testing the system manually and also automatically through the testing framework's available like Nunit. This will create a system that is easily maintenance and will have a limited amount of bugs.

Constraints

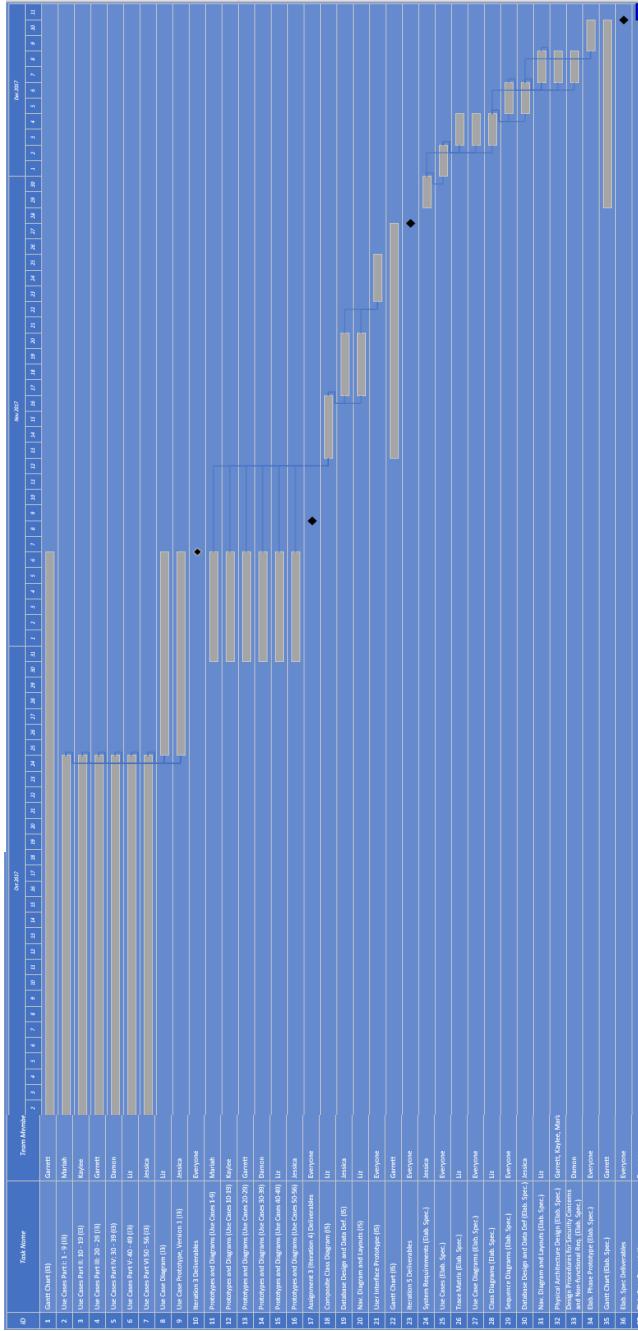
68	The new system will not handle sensitive data about patients	High
69	The new system will not violate HIPAA	High
70	The new system will not interface with the RedCap or Epic Connect patient database	High
71	The new system will allow for disclosure of all volunteers, per the Federal Tort Claims Act	High

The system will only handle data that is specifically needed to create events and pull in volunteers. This way the system doesn't have sensitive patient data that could be accessed by people that shouldn't access it. It will not violate HIPAA do to this acceptance of only must-have data. The system will accurately and efficiently store data to allow for full disclosure of volunteer information in case the need arises.

GANTT CHART

The Gantt chart determines the tasks of a project and how long each task will take as well as task dependencies. The Gantt chart includes the task name, team member responsible, the start and end dates, and the duration in days for each task. Each activity is represented by a bar and the position and length of the bar reflect the start, duration, and end of an activity. Below is the Gantt chart for Iteration 3. The duration of Iteration 3 was 35 days, started on October 2nd, 2017 and ended November 4th. Key dates are marked with a diamond-shaped symbol as shown below. The Gantt chart for Iterations 3 - 5, as well as the elaboration phase are shown on the next page.

Iterations 3 – 5 and Elaboration Phase



PROTOTYPES

Interactive prototypes demo -

<https://trial.dc.irise.com:443/iRise/ds/play/5f84e2f15d9c427b964c8159c8747704>

Dynamic Volunteer Form (Insert Volunteer Records) - Instead of using a dropdown that points to different volunteer forms, we envision one dynamic form. This will simultaneously declutter the website, provide a sleek user experience, and capture only the information that is necessary for each type of volunteer.

Donor Form (Insert Donor Records) – Donor information will be captured at the time of first donation, although the financial information will not be stored. The first time someone makes a donation, their donor profile will be created in the SOS database. Subsequent donation records will be tied to the same donor. This organization helps improves reports for writing grants or news publications.

Braintree Donations (Submit Donation Payment) - Our team recommends that the client switch from PayPal to Braintree. We will introduce the donor to a warm webpage that invites and inspires them to become a part of something exciting. Happy photos and possibly quotes from real SOS patients will assure donors that they are making a difference.

User Portals (Logins) - We envision a centralized portal for SOS staff and coordinators to schedule surgeries, events, run reports, and manage SOS data without needing to frequently access the MySQL database directly. Additionally, SOS volunteers and donors will have the capability to login to the SOS website to view information that's relevant to them. The login form will look just like the Admin login.

Surgery Scheduling (Schedule Surgeries and Create Surgical Events) – Within the Admin portal, admins have the ability to quickly schedule surgeries and events using dropdowns populated by the SOS database. The smart scheduling functionality will only suggest volunteers qualified for the specific jobs needed.

MySQL Data Forms (Update and Delete Records) - Although the IT Coordinator will have direct access to the MySQL database, we will provide forms that allow for user-friendly updates or deletions to all types of SOS records (volunteers, donors, events, etc).

Update Delete Database Record

The screenshot shows a Windows application window titled "Person Details". The window contains a form with the following fields:

Field	Value
ID	986230914
First Name	Peter
Middle Name	J
Last Name	Thompson
Email	pt11@gmail.com
Phone	7439874366

Below the form is a row of buttons: New..., Update, Delete, First, Previous, Next, Last.

Surgery Scheduling Form

Surgery Type:

Endoscopy

This Surgery Will Need:

Pre-Op

Choose from Qualified Physicians

Erica Sutton

Choose from Qualified Nurses

John Smith

Choose Patient Receiving Surgery:

Jane Doe

Submit

Donation Submission

Thank you so much for helping the Louisville community!

Your generosity allows Surgery on Sunday Louisville to continue helping those in need of surgeries

I would like this donation to recur

Your Donation

Amount (\$)

Card Type

Please select... 

Name on Card

Card Number

MM YY Code



[Donate](#)

About Us



Surgery On Sunday - Louisville, Inc. provides outpatient surgical and endoscopic procedures to patients who are within 250% of the current federal poverty line, have medical need, are symptomatic and/or in a high risk healthcare category, and are uninsured or under-insured.



Our licensed physicians, medical and non-clinical support staff, and healthcare partners provide all services on a volunteer basis.

Our patients never see a bill.



We are a non-profit organization based in Louisville, KY. Our program was founded in 2013 by a group of passionate medical professionals and members of the Louisville community.

Send Emails

New Email

Email Name
Newsletter

Email Type

 Start Blank

 Upload HTML

 Pick a Template

Events

About Us Partner Clinics Get Involved Refer a Patient Events and Media **Donate**

Upcoming Events

Date and Time	Location
Oct. 15, 2017 6 a.m. - 2 p.m.	Metro Specialty Health Jeffersonville Indiana
Nov. 12, 2017 6 a.m. - 2 p.m.	Norton Downtown, Louisville KY

What People are Saying about SOS:

Chanelle Helm reviewed Surgery on Sunday Louisville — 5*
September 25 at 1:44pm ·

I'm telling you! I had surgery on 9.24.17 at Premier Surgery. Everyone was so warm and welcoming. It was amazing. Thank you so much.

Beunca Graffree > Surgery on Sunday Louisville
September 21 at 5:32pm ·

Hello! When is the next event? I'd love to get involved again 😊

Dave Bell So proud of the work you do and the Colon Cancer Prevention Project is honored to partner with you in preventing colon cancer.
Like · Reply · 2 · March 1 at 10:18pm

Get Involved/Volunteer Form

Surgery on Sunday

Louisville

About Us Partner Clinics Get Involved Refer a Patient Events and Media Donate

We would love to have you join us!

The form below will walk you through becoming an SOS volunteer

My volunteer work would be...

I would like to help out with...

Are you fluent in other languages?
(besides English)
 Yes



Donation

Thank you so much for helping the Louisville community!

Your generosity allows Surgery on Sunday Louisville to continue helping those in need of surgeries

First Name *	Last Name *	
<input type="text"/>	<input type="text"/>	
Email Address *	Phone Number	
<input type="text"/>	<input type="text"/>	
Street Address		
<input type="text"/>		
City	State/Province/Region	Zip/Postal Code
<input type="text"/>	<input type="text"/>	<input type="text"/>
Country		
<input type="text"/>		

Donation Details



Admin Portal

WELCOME BACK, BARBARA

Schedule Surgery Create Event Reports Log Out

Surgery Scheduling Form

Surgery Type:

This Surgery Will Need:

Choose from Qualified Physicians

Choose from Qualified Nurses

Choose Patient Receiving Surgery:

Submit

Surgery on Sunday
Louisville

WELCOME BACK, BARBARA

[Schedule Surgery](#) [Create Event](#) [Reports](#) [Log Out](#)

Reports

Show Me... [Donors Report](#)

Donation Reports > View Report

- View Report -

Donations By Donor Apr 21, 2006 11:58:56 AM
Page 1 of 1 [First](#) | [Previous](#) | [Next](#) | [Last](#)

Report Results						
Donor Contact Information				Donation Totals		
Name	Email	Donor Address	Phone Number	Total Transactions (#)	Total Transactions (\$)	Average Transaction (\$)
Benson, Sally	sasbenson@convio.com	43 Maple Cv Johnson, AK 98765 United States		1	60.00	60.00
Brown, Jane	jjbrown@convio.com	123 Oak Tree Cv Austin, VA 23456 United States		1	35.00	35.00
Johnson, Ben	bajohnson@convio.com	2349 Elmwood Ave Johnson City, FL 34567 United States		1	60.00	60.00
Milson, Jack	jjmilson@convio.com	21 Pear Ct Evans, AR 98765 United States		2	95.00	47.50



Admin Portal
Please Login with Your SOS Credentials

Username:

Password:

Login

Dynamic Volunteer Form

We would love to have you join us!

The form below will walk you through becoming an SOS volunteer

My volunteer work would be...

Non-Clinical ▾

I would like to help out with...

Fundraising ▾

Are you fluent in other languages?
(besides English)

Yes ▾

I am also fluent in...

Spanish ▾

Please specify:

Submit

Donation Form

First Name *	Last Name *	
<input type="text"/>	<input type="text"/>	
Email Address *	Phone Number	
<input type="text"/>	<input type="text"/>	
Street Address		
<input type="text"/>		
City	State/Province/Region	Zip/Postal Code
<input type="text"/>	<input type="text"/>	<input type="text"/>
Country		
<input type="text"/>		
Donation Details		
Donation Amount *		
<input type="text"/>		
Donation Comments *		
<input type="text"/>		

[Continue >>](#)

Maintain Nurse Records

Maintain Nurse Records

Person Details	
ID	146589034
First Name	Judah
Middle Name	H.
Last Name	Johnson
Email	jhj01@gmail.com
Phone	5025554789

New... Update Delete First Previous Next Last

Maintain Nurse Records

Person Details	
ID	
First Name	
Middle Name	
Last Name	
Email	
Phone	

New... Update Delete First Previous Next Last

Maintain Nurse Records

Add License

License Number	<input type="text" value="123564897"/>
License Type	<input type="text" value="RN License"/>
License Issue Date	<input type="text" value="02/15/2008"/>
Licensure Expiration Date	<input type="text" value="02/15/2018"/>
Licensure State	<input type="text" value="KY"/>

Maintain Nurse Records

Add License

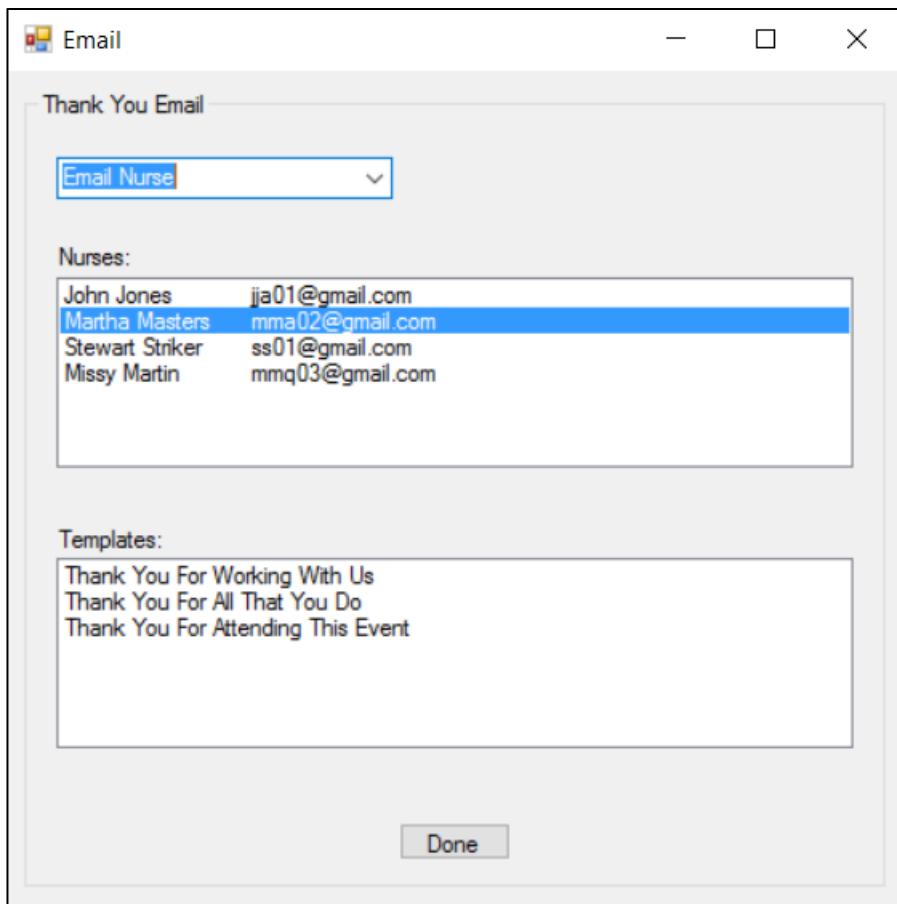
License Number	<input type="text" value=" "/>
License Type	<input type="text"/>
License Issue Date	<input type="text"/>
Licensure Expiration Date	<input type="text"/>
Licensure State	<input type="text"/>

Schedule Surgery

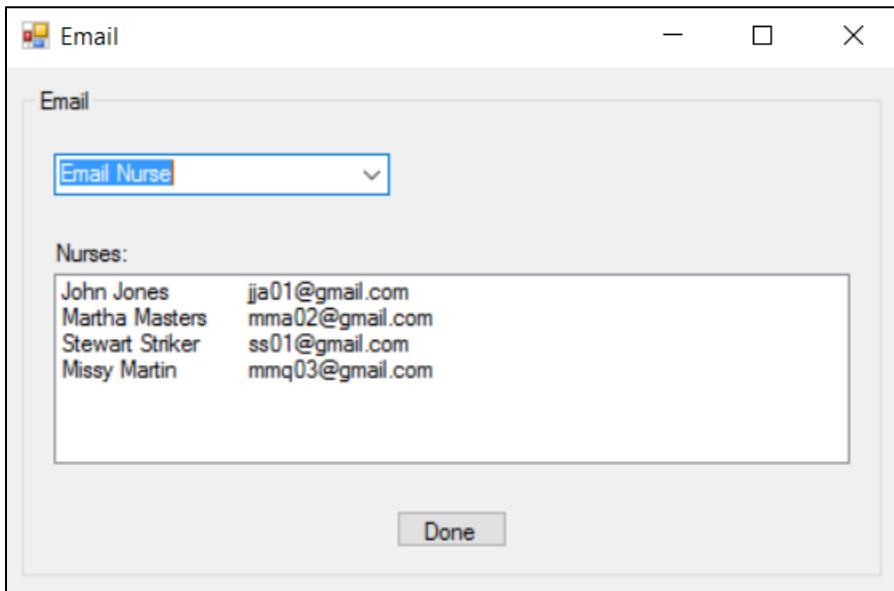
Surgery Scheduling Form

Surgery Type:	Colonoscopy
This Surgery Will Need:	Nurse Anesthetis
Choose from Qualified Nurses:	James B. Johnson
Choose Patient Receiving Surgery:	John H. Jones

Thank You Email



Email Nurses



Add Interpreter

Clinical Language Interpreter

First Name Last Name

Email Phone

Address

City State Zip

License

Hospital Affiliation

Primary Language Secondary Language

[Continue >>](#)

Clinical Language Interpreter

First Name	Last Name	
John	Jones	
Email	Phone	
bj01@gmail.com	5025554714	
Address		
1234 West Ave.		
City	State	Zip
Louisville	KY	40207
License		
123459123		
Hospital Affiliation		
Baptist Hospital		
Primary Language	Secondary Language	
English	Spanish	
Continue >>		

Reports

Reports

Nurse Reports	Disolvav
Export	Print

Reports

Nurse Reports	Disolvav
Export	Print

Maintain Interpreter

Maintain Interpreter Records

Person Details

ID	564892145
First Name	Trudy
Middle Name	B.
Last Name	Jones
Email	tbj02@gmail.com
Phone	5025556987

New... Update Delete First Previous Next Last

Maintain Interpreter Records

Person Details

ID	<input type="text"/>
First Name	<input type="text"/>
Middle Name	<input type="text"/>
Last Name	<input type="text"/>
Email	<input type="text"/>
Phone	<input type="text"/>

New... Update Delete First Previous Next Last

Maintain Nurse Records

Maintain Nurse Records

Person Details	
ID	146589034
First Name	Judah
Middle Name	H.
Last Name	Johnson
Email	jhj01@gmail.com
Phone	5025554789

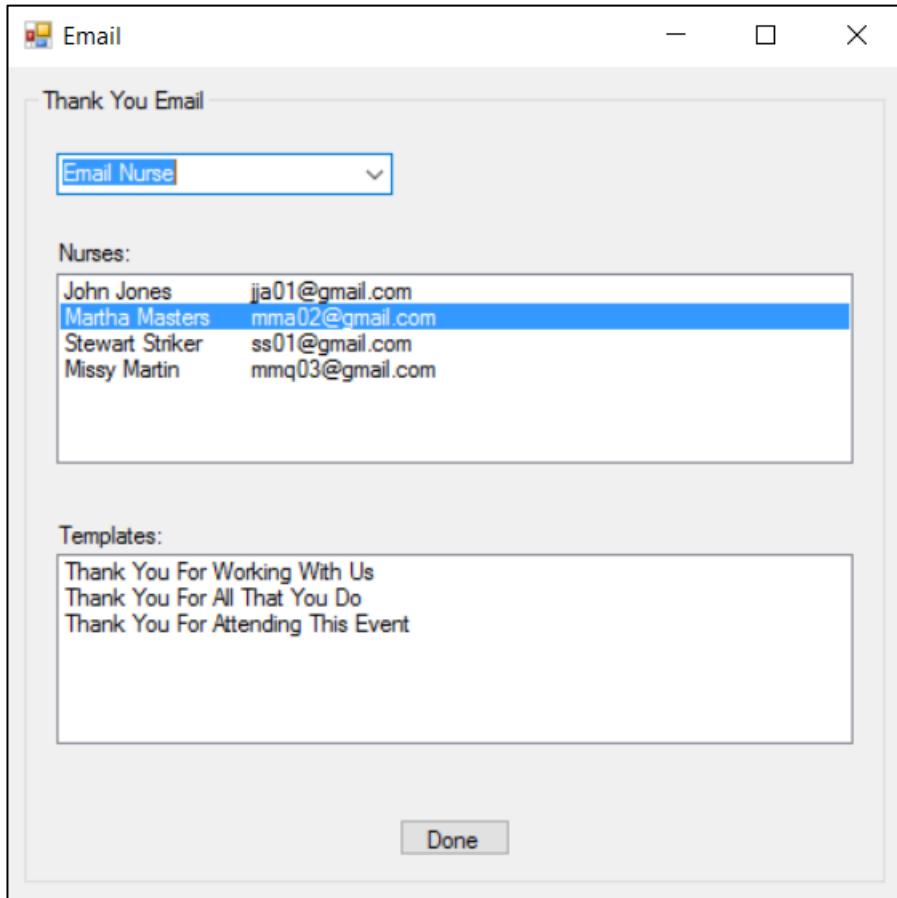
New... Update Delete First Previous Next Last

Maintain Nurse Records

Person Details	
ID	
First Name	
Middle Name	
Last Name	
Email	
Phone	

New... Update Delete First Previous Next Last

Thank You Emails - Nurses



Maintain Interpreter Records

Maintain Interpreter Records

Add License

License Number	<input type="text" value="235456985"/>
License Type	<input type="text" value="Clinical Language Interpreter"/>
License Issue Date	<input type="text" value="02/15/2008"/>
Licensure Expiration Date	<input type="text" value="02/15/2018"/>
Licensure State	<input type="text" value="KY"/>

Maintain Interpreter Records

Add License

License Number	<input type="text" value=" "/>
License Type	<input type="text"/>
License Issue Date	<input type="text"/>
Licensure Expiration Date	<input type="text"/>
Licensure State	<input type="text"/>

REFERRALS REPORTS

The screenshot shows the Admin Portal interface for Surgery on Sunday Louisville. At the top right, a message says "You have 2 new referrals to review!". Below the header, it says "WELCOME BACK, BARBARA". A navigation bar includes links for Schedule Surgery, Events, Reports, Send Emails, Referrals, and Log Out. The main section is titled "Reports" and contains a form to "Select Report". The dropdown menu shows "Referrals" selected. There are date pickers for "Date" (from mm/dd/yyyy to mm/dd/yyyy) and a dropdown for "Type of Surgery" with "ALL" selected. Below the form is a table titled "Referrals Report (Sample data)". The table has columns for Date, Patient First Name, Patient Last Name, Surgery Requested, and Physician Name. It lists three rows of sample data.

Date	Patient First Name	Patient Last Name	Surgery Requested	Physician Name
10/20/15	James	Locksmith	Gallbladder Surgery	Dr. Cinthia Roberts
10/20/15	Lana	Zuess	Cataract Surgery	Dr. James Zhao
10/25/15	Alex	Collin	Hernia removal	Dr. Jenny Ortega

New Referrals

The screenshot shows the Admin Portal interface for Surgery on Sunday Louisville. At the top left is the logo "Surgery on Sunday Louisville". The top right features the text "Admin Portal" and a blue box stating "You have 2 new referrals to review!". Below the header is a welcome message "WELCOME BACK, BARBARA". A navigation bar includes links for "Schedule Surgery", "Events", "Reports", "Send Emails", "Referrals", and "Log Out". A central box is titled "New Referrals" and contains a table titled "Referrals Report (Sample data)". The table has columns for Date, Patient First Name, Patient Last Name, Surgery Requested, and Physician Name. It lists three entries:

Date	Patient First Name	Patient Last Name	Surgery Requested	Physician Name
10/20/15	James	Locksmith	Gallbladder Surgery	Dr. Cinthia Roberts
10/20/15	Lana	Zuess	Cataract Surgery	Dr. James Zhao
10/25/15	Alex	Collin	Hernia removal	Dr. Jenny Ortega

Referral Form

Refer a Patient

Physician Information:

First Name * Last Name *
Email Address * Phone Number *
Hospital Street Address *
City * State/Province/Region * Zip/Postal Code *

Patient Information:

First Name * Last Name *
Email Address * Phone Number *
Street Address *
City * State/Province/Region * Zip/Postal Code *
Surgery Requesting: *

Referral Login

The screenshot shows a website's 'Refer a Patient' section. At the top, there's a horizontal navigation bar with links: 'About Us', 'Partner Clinics', 'Get Involved', 'Refer a Patient', 'Events & Media', and a red 'DONATE' button. Below the navigation is a large, bold title 'Refer a Patient'. Underneath the title, a gray header bar contains the text 'PROTECTED: REFER A PATIENT'. The main content area is a white box containing the message 'This content is password protected. To view it please enter your password below:'. Below this message is a password input field with the placeholder 'Password:' and a black 'Enter' button next to it. To the right of the input field is a link labeled 'Instructions' with a small arrow icon pointing to it.

This content is password protected. To view it please enter your password below:

Password:

Enter [Instructions](#)

Event Report

Surgery on Sunday
Louisville

Admin Portal

You have 2 new referrals to review!

COME BACK, BARBARA

Schedule Surgery Events Reports Send Emails Referrals Log Out

Reports

Select Report: Events Date: mm/dd/yyyy to mm/dd/yyyy Generate

Event Type: Surgical Event Cost: Greater than: Less than: Equal to: Type of Surgery: ALL Clear

Events Report (Sample data)				
Type of surgery	Duration	Date	Location	...
Arthroscopy	45 min	12/03/17	Norton Healthcare	
Breast Biopsy	20 min	11/12/17	Norton Healthcare	
Cataract Surgery	60 min	10/22/17	Baptist Health	

F **+**

Add Attendance

The screenshot shows the "Admin Portal" interface for "Surgery on Sunday Louisville". At the top right, a message says "You have 2 new referrals to review!". Below the header, it says "WELCOME BACK, BARBARA". A navigation bar includes "Schedule Surgery", "Events", "Reports", "Send Emails", "Referrals", and "Log Out", with "Add Attendance" being the active button. The main content area is titled "Add Attendance" and contains a "Sync Attendance" button and a date input field set to "mm/dd/yyyy". A data grid labeled "Data" lists three entries:

#	First Name	Last Name	Hours	Date
102	Mark	Smith	5	11/05/16
065	Jacob	Jones	3	11/05/16
008	Larry	Hardy	6	11/05/16

At the bottom are "ADD Attendance" and "Cancel" buttons.

Thank You Email

The screenshot shows a software application window titled "WELCOME BACK, BARBARA". At the top, there are several menu options: "Schedule Surgery", "Events", "Reports", "Send Emails", "Referrals", and "Log Out". Below the menu, the title "Delete Event" is centered. To the right of the title is a search bar with the placeholder "search" and a magnifying glass icon. A table below the search bar displays three rows of event data:

Event Name	First Name	Last Name	Date
JL042	James	Locksmith	12/05/17
LZ055	Lana	Zuess	02/05/2018
AC05	Alex	Collin	05/05/16

At the bottom of the dialog box are two buttons: "DELETE" and "Cancel". A cursor arrow is positioned over the "DELETE" button.

Refer a Patient

The screenshot shows the 'Refer a Patient' page of the Surgery on Sunday Louisville website. At the top, there is a banner featuring several smiling healthcare professionals. Below the banner, a navigation bar includes links for 'About Us', 'Partner Clinics', 'Get Involved', 'Refer a Patient' (which is the active page), 'Events & Media', and 'DONATE'. The main title 'Refer a Patient' is prominently displayed in a large, bold font. A sub-section titled 'PATIENT REFERRAL' contains text explaining the referral process and providing an email address for inquiries. A note at the bottom of this section indicates that existing logon information can be used to access the referral portal.

PATIENT REFERRAL

Patient referrals to Surgery on Sunday Louisville may be made by any licensed healthcare provider through that provider's office, or by a registered health entity (such as a physician's office or a community clinic). If you are in need of access to our patient referral system, please send an email to info@soslouisville.org on your office's letterhead and someone from our organization will contact you within five business days to provide to you the logon credentials.

If you already have the logon information for our referral portal, [click here](#).

Fundraising

Surgery
on Sunday
Louisville

Welcome Back, BARBARA

You have 2 new referrals to review!

Admin Portal

Schedule Surgery Events Reports Send Emails Referrals Log Out

Reports

Select Report: Events Date: mm/dd/yyyy to mm/dd/yyyy Generate

Event Type: Fundraising Amount Raised: Greater than: _____, Less than: _____, Equal to: _____ Budget: Greater than: _____, Less than: _____, Equal to: _____ Sponsor: ALL Clear

Events Report

Event Name	Date	Budget
Family Community Clinic	12/05/17	\$1000
Family Community Clinic	02/05/2018	\$1000

Referral Notification

The screenshot shows the 'Admin Portal' interface for 'Surgery on Sunday Louisville'. At the top left is the logo 'Surgery on Sunday Louisville' with a red heart rate line graphic. To the right, the title 'Admin Portal' is displayed in large black font. A blue header bar at the top right contains the message 'You have 2 new referrals to review!' next to an envelope icon. Below the header, a welcome message 'WELCOME BACK, BARBARA' is shown in bold capital letters. A horizontal navigation bar at the bottom includes five buttons: 'Schedule Surgery', 'Events', 'Reports', 'Send Emails', and 'Log Out'. The main content area below the navigation bar is currently empty.

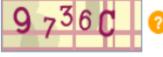
Volunteer Is Updated or Deleted:

The screenshot shows a software interface titled "Selected Volunteer: Damon Quire". Below the title, there is a "Task:" label followed by a dropdown menu. The dropdown menu has three options: "Edit" (which is highlighted with a blue background), "Delete", and another option that is partially visible. The background of the interface is light gray.

Volunteer is Added:

We thank you for your interest in volunteering and supporting our organization's cause.

Personal Information

Volunteer Category	Surgeon
Volunteer's Name *	<input type="text"/>
Email *	<input type="text"/>
Phone *	<input type="text"/>
Address	<input type="text"/> Street Address <input type="text"/> Street Address Line 2 <input type="text"/> City • <input type="text"/> Region • <input type="text"/> Postal / Zip Code • <input type="text"/> United States
Would you like to receive our newsletter?	<input type="radio"/> Yes <input type="radio"/> No
Secondary Language	<input type="text"/>
Verification *	  <input type="text"/>

Submit your Volunteer Application Form

Send Thank You Emails:

Past Events

Report Filters (click to expand)

Past Events

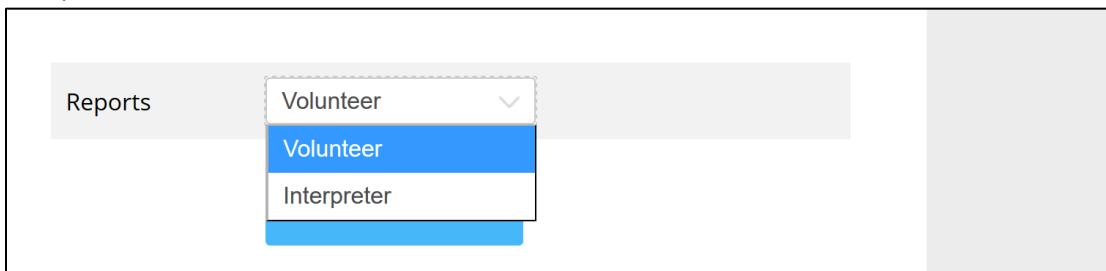
Finished

Desired Interpreter:

Schedule Event with Interpreter and Volunteers Included:

Event Date	MM/DD/YYYY
Event Date	HH MM AM <input type="button" value="▼"/>
Address	Street Address
	Street Address Line 2
City	Region
Postal / Zip Code	United States <input type="button" value="▼"/>
Patient	Choice1 <input type="button" value="▼"/>
Volunteer Team	Choice1 <input type="button" value="▼"/>
SUBMIT EVENT	

Interpreter or Volunteer



Interpretation

This is a demo image. Your submissions table will appear once you finish.

#	Date	What is your position?	First name:	Last name:
1	2015-12-09 13:12:17	The employee's supervisor	Alan	Ackerman
2	2015-12-09 13:11:26	The employee's manager	Marjorie	Kingston
3	2015-12-09 10:54:19	The employee's peer	Matthew	White
4	2015-12-09 10:53:56	The employee	Ian	Kellog
5	2015-12-09 10:53:31	The employee's peer	Tommy	Green
6	2015-12-09 10:53:08	The employee's supervisor	George	Kohl
7	2015-12-09 10:52:37	The employee's peer	Jesse	Alto
8	2015-12-09 10:52:11	The employee	Uri	Galup
9	2015-12-09 10:51:34	The employee's manager	Kelly	Pea

Volunteer

This is a demo image. Your submissions table will appear once you finish.

#	Date	What is your position?	First name:	Last name:	Department:
1	2015-12-09 13:12:17	The employee's supervisor	Alan	Ackerman	HR
2	2015-12-09 13:11:26	The employee's manager	Marjorie	Kingston	SA
3	2015-12-09 10:54:19	The employee's peer	Matthew	White	CO
4	2015-12-09 10:53:56	The employee	Ian	Kellog	CO
5	2015-12-09 10:53:31	The employee's peer	Tommy	Green	IT
6	2015-12-09 10:53:08	The employee's supervisor	George	Kohl	SA
7	2015-12-09 10:52:37	The employee's peer	Jesse	Alto	MM
8	2015-12-09 10:52:11	The employee	Uri	Galup	SA
9	2015-12-09 10:51:34	The employee's manager	Kelly	Pea	LR

Send Emails to Interpreters and Volunteers:

