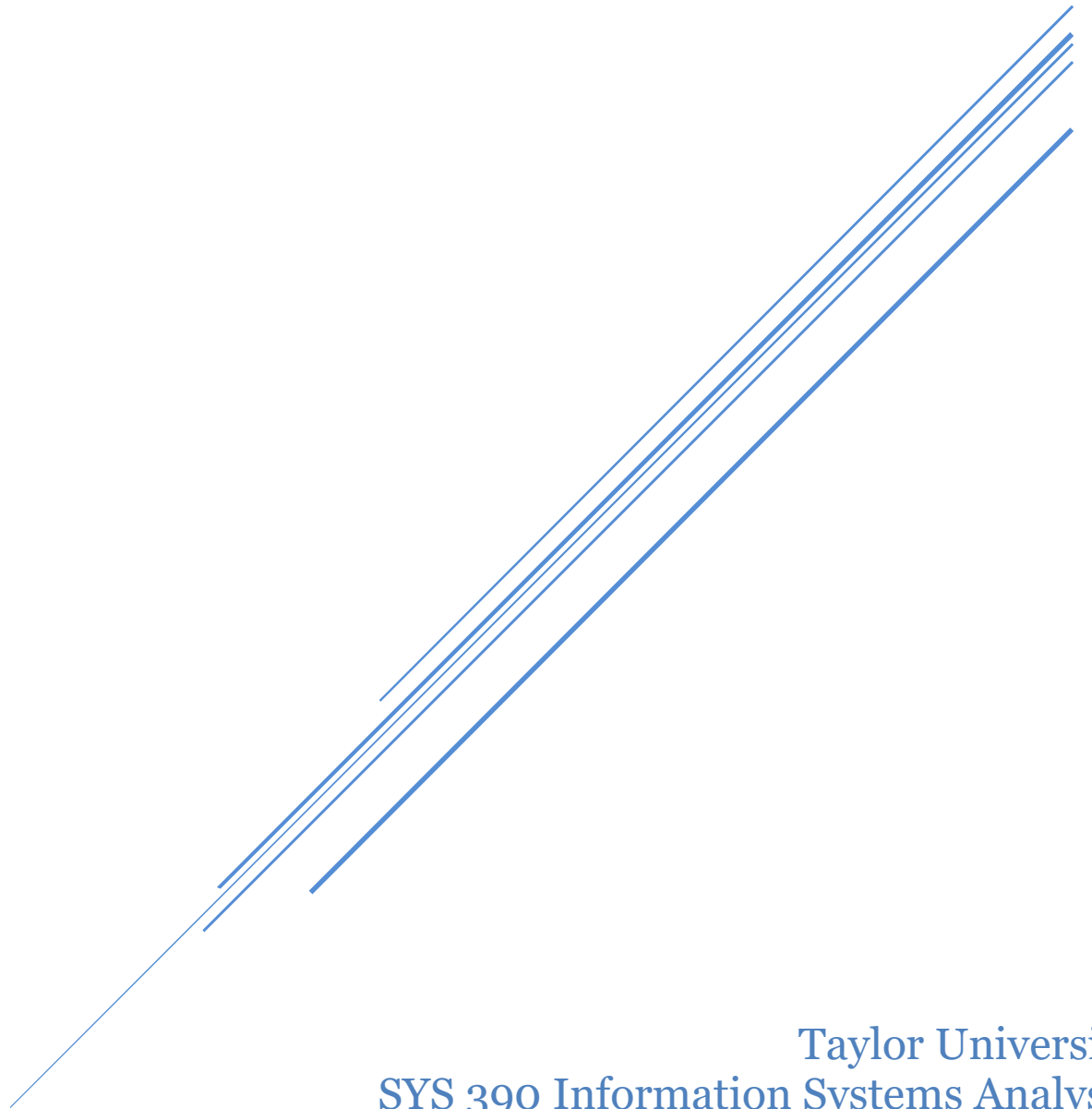


SYSTEM REQUIREMENTS SPECIFICATION

Verbo Church Cuenca Attendance Tracker



Taylor University
SYS 390 Information Systems Analysis
Dr. Nurkkala & Dr. Stanley
Team ISA V

Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu

TABLE OF CONTENTS

1	GENERAL.....	2
1.1	USER ROLES	2
1.2	COLLABORATING SYSTEMS & OTHER SOFTWARE.....	3
1.3	SCHEDULE.....	3
2	FUNCTIONAL REQUIREMENTS.....	5
2.1	SYSTEM RESPONSE TABLE (SRT)	5
2.2	LOGICAL DATA MODEL.....	9
2.3	DATA DICTIONARY.....	10
2.4	LOGICAL PROCESS MODEL.....	12
2.4.1	Data Descriptions	12
2.4.2	Data Flow Diagrams and Process Specifications	15
2.5	USER EXPERIENCE (UX) DIAGRAMS	79
2.6	WIREFRAMES	81
3	NON-FUNCTIONAL REQUIREMENTS.....	86
4	GLOSSARY	89

1 GENERAL

This System Requirements Specification (SRS) outlines the functional and non-functional requirements of the system to be built for Verbo Church Cuenca. This system will be built to accomplish the following goals:

- To improve the completeness and consistency of Verbo's attendance-taking system.
- To provide church members with a map of Verbo's home churches, complete with categories (18 and under, women only, etc.)
- To store the information of all members of Verbo (contact information, date of birth, marital status, etc.)
- To answer questions such as "Is there growth in the home churches?" and "Do home church members stick around?"
- To provide a user-friendly interface for anybody using the system.

This system will provide the completeness and consistency Verbo needs and that Microsoft Excel was not designed to provide.

1.1 User Roles

The following defines the different users who will interact with the system.

User Title	Role
Visitor	Any web user without a record in the database
Member	An attendee of Verbo Church, but not necessarily a home church
	Has record in the database
	Does not have user account
Home Church Leader	A leader of at least one home church
	Has user account
	Has limited permissions
Administrator	Only church leaders and information technology (IT) support
	Has unlimited permissions
User	Encompassing term for both Home Church Leaders and Administrators

1.2 Collaborating Systems & Other Software

Heroku:

Heroku is a platform as a service provider. This system would be used to store data assuming Verbo Church Cuenca does not have their own servers.

Google Maps and Calendar APIs:

Google Maps and Calendar APIs would be used to support the application's home church maps and home church calendar functionalities. The Google APIs would receive the addresses of home churches to display them on the home church maps and event dates and information to be displayed on the home church calendars.

1.3 Schedule

Milestone	Estimated Completion Date
Initial Planning	February 3, 2017
Member and User Functionality	February 17, 2017
Home church integration with members	March 10, 2017
Home church categories	March 17, 2017
Home church events	April 7, 2017
Leadership classes	April 21, 2017
Miscellaneous requirements	April 28, 2017
Testing, bug fixes, and deployment	May 12, 2017
Estimated Completion Date	May 12
Absolute Deadline	May 18

The schedule for building and implementing our solution is detailed in the table above, to be completed by May 18, 2017. By February 3, all initial planning will be completed, which includes establishing new teams, dividing up work between team members, planning team meetings, and miscellaneous initial planning. By February 17, all member and user functionality will be implemented, including logging in and out, user roles, user attributes (name, contact information, etc.), and user location. By March 10, home church functionality will be completed, along with the integration with home church members. This involves home church attributes (name, meeting time, location), individual home church meetings, members and roles within the home church, and attendance. By March 17, home church categories will be implemented. By April 7, everything related to home churches, including events, will be fully functional. By April 21, leadership classes and their integration with members will be operational, including tracking which members have completed which classes. By May 12, all testing and bug fixes will be completed, along with the deployment of the

Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu

Team: ISA V

SYS 390 Information Systems Analysis

Dr. Nurkkala and Dr. Stanley

website. We have set an absolute deadline of May 18 in case the Taylor students implementing this solution fall behind schedule, for whatever reason.

2 FUNCTIONAL REQUIREMENTS

2.1 System Response Table (SRT)

Administrator

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
AD1	Administrator	Give administrator role to member	Administrator role given to member	Updated member information to have administrator role	Database
				New administrator role notification	New administrator
AD2	Administrator	Revoke administrator role from member	Administrator role revoked from member	Updated member information to no longer have administrator role	Database

Attendance

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
A1	User	View home church statistics	Home church statistics viewed	Home church statistics and graphs	User
A2	User	View member attendance history	Member attendance history viewed	Member attendance history	User
A3	Home church leader	Record home church attendance	Home church attendance recorded	New home church attendance information	Database
A4	Home church leader	Update home church attendance record	Home church attendance record updated	Updated home church attendance information	Database

Event

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
E1	Visitor	View home church events	Home church events viewed	Home church events	Visitor
E2	Visitor	View home church event details	Home church event details viewed	Home church event details	Visitor
E3	Home church leader	Add event to home church	Event added to home church	New home church event	Database
				New home church event notification	Home church members
E4	User	Update home church event details	Home church event details updated	Updated home church event details	Database
E5	User	Archive home church event	Home church event archived	Updated home church event information to be archived	Database
E6	User	Restore home church event	Home church event restored	Updated home church event information to no longer be archived	Database

Home Church

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
C1	Visitor	View home church map	Home church map viewed	Home church map	Visitor
C2	Visitor	View home church details	Home church details viewed	Home church details	Visitor
C3	Administrator	Create home church	Home church created	New home church	Database
				New home church notification	New home church leader
				Updated member information to be leader of new home church	Database
C4	User	Update home church information	Home church information updated	Updated home church information	Database
C5	Administrator	Archive home church	Home church archived	Updated home church information to be archived	Database
C6	Administrator	Restore home church	Home church restored	Updated home church information to no longer be archived	Database

Home Church Leader

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
CL1	Administrator	Add home church leader to home church	Home church leader added to home church	Updated member information to be new home church leader of home church	Database
				New home church leader notification	New home church leader
CL2	Administrator	Remove home church leader from home church	Home church leader removed from home church	Updated member information to no longer be home church leader of home church	Database

Home Church Member

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
CM1	User	View home church members	Home church members viewed	Home church members	User
CM2	User	Add member to home church	Member added to home church	Member updated	Database
				Added to home church notification	Member
CM3	User	Remove member from home church	Member removed from home church	Updated member information to be removed from the home church	Database

Leadership Class

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
LC1	Visitor	View list of leadership classes	List of leadership classes viewed	List of leadership classes	Visitor
LC2	Visitor	View leadership class details	Leadership class details viewed	Leadership class details	Visitor
LC3	Administrator	Add new leadership class	New leadership class added	New leadership class	Database
LC4	Administrator	Update leadership class details	Leadership class details updated	Updated leadership class details	Database
LC5	Administrator	Archive leadership class	Leadership class archived	Updated leadership class information to be archived	Database
LC6	Administrator	Restore leadership class	Leadership class restored	Updated leadership class information to no longer be archived	Database

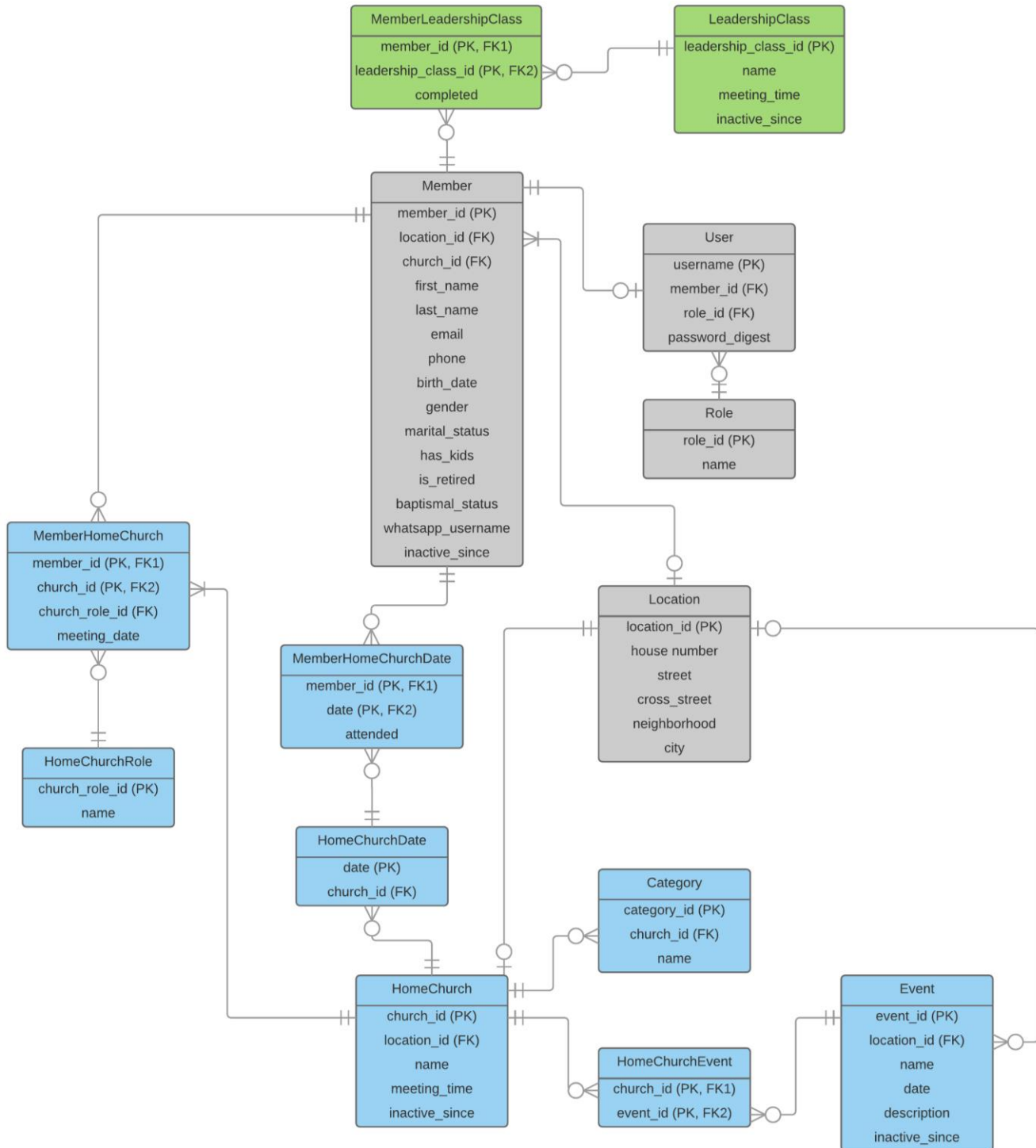
Logging In and Out

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
L1	User	Log in	Logged in	User home page	User
				Request for new session	Server
L2	User	Log out	Logged out	Log-in page	User
				Request to delete user session	Server

Member

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
M1	Administrator	View all members	All members viewed	All members	Administrator
M2	User	Filter members	Members filtered	List of filtered members	User
M3	User	View member profile page	Member profile page viewed	Member profile page	User
M4	User	Create member	Member created	New member	Database
				Welcome message	Member
M5	User	Update member profile	Member profile updated	Updated member information	Database
M6	Administrator	Archive member	Member archived	Updated member information to be archived	Database
M7	Administrator	Restore member	Member restored	Updated member information to no longer be archived	Database

2.2 Logical Data Model



2.3 Data Dictionary

Entity	Attributes	Relationships
Category	Category ID	Has zero to many Home Church Categories.
	Church ID	
	Name	
Event	Event ID (PK)	Has zero to many Home Church Events.
	Location ID (FK)	Has zero or one Locations.
	Name	
	Date	
	Description	
	Inactive Since	
Home Church	Church ID (PK)	Has zero to many Home Church Attachments.
	Location ID (FK)	Has zero to many Home Church Dates.
	Name	Has zero to many Home Church Categories.
	Meeting Time	Has zero to many Home Church Events.
	Inactive Since	Has exactly one Location.
		Has one to many Member Home Churches.
Home Church Date	Date (PK)	Has zero to many Member Home Church Dates.
	Church ID (FK)	Has exactly one Home Church.
Home Church Event	Church ID (PK, FK1)	Has exactly one Home Church.
	Event ID (PK, FK2)	Has exactly one Event.
Home Church Role	Church Role ID (PK)	Has zero to many Member Home Churches.
	Name	
Leadership Class	Leadership Class ID (PK)	Has zero to many Member Leadership Classes.
	Name	
	Meeting Time	
	Inactive Since	
Location	Location ID (PK)	Has zero to many Events.
	Cross Street	Has zero or one Home Churches.
		Has one to many Members.

Entity	Attributes	Relationships
Member	Member ID (PK)	Has zero or one Locations.
	Location ID (FK)	Has zero to many Member Home Churches.
	Church ID (FK)	Has zero to many Member Home Church Dates.
	First Name	Has zero to many Member Leadership Classes.
	Last Name	
	Email	
	Phone	
	Birth Date	
	Gender	
	Marital Status	
	Has Kids	
	Is Retired	
	Baptismal Status	
	WhatsApp Username	
	Inactive Since	
Member Home Church	Member ID (PK, FK1)	Has exactly one Member.
	Church ID (PK, FK2)	Has exactly one Home Church.
	Church Role ID (FK)	Has exactly one Home Church Role.
	Meeting Date	
Member Home Church Date	Member ID (PK, FK1)	Has exactly one Member.
	Date (PK, FK2)	Has exactly one Home Church Date.
	Attended	
Member Leadership Class	Member ID (PK, FK1)	Has exactly one Member.
	Leadership Class ID (PK, FK2)	Has exactly one Leadership Class.
	Completed	
Role	Role ID (PK)	Has zero to many Users.
	Name	
User	Username	Has exactly one Member.
	Member ID (FK)	Has exactly one Role.
	Role ID (FK)	
	Password Digest	

2.4 Logical Process Model

Logical process models include data flow diagrams, data descriptions, and process specifications. Each data flow diagram corresponds to an entry in the [system response table \(SRT\)](#) and provides a visual of the data flow for each event process. Data can be in the form of a trigger, response, input, output, retrieval, or storing. Other components of the data flow diagrams are the external agents (who or what is interacting with the system), and the data store, which defines what the data is stored under.

The [data description](#) defines broader groups of data that will be passed through each process. The data group names are what will be referred to in the data flow diagrams as well as in the process specification. The data within each group will refer to attributes within the [entity relationship diagram \(ERD\)](#).

The process specification defines the steps of each event process. The steps are written in pseudo-code to accompany each data flow diagram, describing what data (described in the [data description](#)) is used in each event process.

2.4.1 Data Descriptions

Italicized data refers to another data description entry. Content in braces {} corresponds with the attribute immediately before the braces. Content in parentheses () describe how the attribute immediately before it should be specified or chosen. Content in quotation marks (") specifies the value of the specified attribute.

<u>Home Church Map</u>	= Church ID (Inactive Since = null) {Name, Location ID {House Number, Street, Cross Street, Neighbourhood, City}}
<u>Home Church Details</u>	= Church ID {Name, Location ID {House Number, Street, Cross Street, Neighbourhood, City}, Meeting Time, Category, Inactive Since, Member ID (Church Role Name = "Home Church Leader" and Church ID = "X") {First Name, Last Name, Email, Phone, WhatsApp Username}}
<u>New Home Church Notification</u>	= <i>Home Church Details</i>
<u>Updated Church Role Information</u>	= Member ID {Church ID, Church Role ID, Username{Role ID {Name}}}
<u>Updated Church Role Notification</u>	= Member ID {First Name, Last Name, Email, Church Role ID, Church Role Name, Church ID {Name}, Username{Role ID {Name}}}

<u>Home Church Active Status</u>	= Church ID {Inactive Since}
<u>Member Details</u>	= Member ID {First Name, Last Name, Location ID {House Number, Street, Cross Street, Neighbourhood, City}, Church ID, Email, Phone, Birth Date, Gender, Marital Status, Has Kids, Is Retired, Baptism Status, Inactive Since, WhatsApp Username, Church ID, Church Role ID {Name}, Username {Role ID {Name}}, Leadership Class ID {Name, Completed}, Inactive Since}
<u>Home Church Members List</u>	= Member ID (Church ID = "X", Inactive Since = null) {First Name, Last Name, Email, Phone}
<u>Updated Member Notification</u>	= Member ID {First Name, Last Name, Location ID {House Number, Street, Cross Street, Neighbourhood, City}, Church ID, Email, Phone, Birth Date, Gender, Marital Status, Has Kids, Is Retired, Baptism Status, Inactive Since, WhatsApp Username, Church ID, Church Role ID {Name} Username{Role ID {Name}}}
<u>Member List</u>	= Member ID (may be filtered on any Member attributes, Inactive Since = null) {First Name, Last Name, Email, Phone}
<u>Home Church Stats</u>	= <i>Home Church Attendance for Date</i> (Date = "ALL")
<u>Member Home Church Attendance History</u>	= Member ID {First Name, Last Name, Church ID {Date, Attended}}
<u>Home Church Attendance for Date</u>	= Church ID (Date = "X") {Date ("X") {Member ID, Attended}}
<u>Home Church Events List</u>	= Church ID (Inactive Since = null){Event ID {Location ID {House Number, Street, Cross Street, Neighborhood, City}, Name, Date}}
<u>Home Church Event Details</u>	= Event ID {Church ID, Location ID {House Number, Street, Cross Street, Neighborhood, City}, Name, Date, Description, Inactive Since}
<u>Home Church Event Active Status</u>	= Church ID {Event ID {Church ID, Inactive Since}}

Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu

Team: ISA V

SYS 390 Information Systems Analysis

Dr. Nurkkala and Dr. Stanley

<u>User Credentials</u>	= Username {Password Digest}
<u>Leadership Classes List</u>	= Leadership Class ID (Inactive Since = null) {Name, Meeting Time}
<u>Leadership Class Details</u>	= Leadership Class ID {Name, Meeting Time, (would include additional Leadership Class attributes as available), Inactive Since}
<u>Leadership Class Active Status</u>	= Leadership Class ID {Inactive Since}

2.4.2 Data Flow Diagrams and Process Specifications

Data as described in [Data Descriptions](#) is shown in red.

Data entities from the database are shown in green.

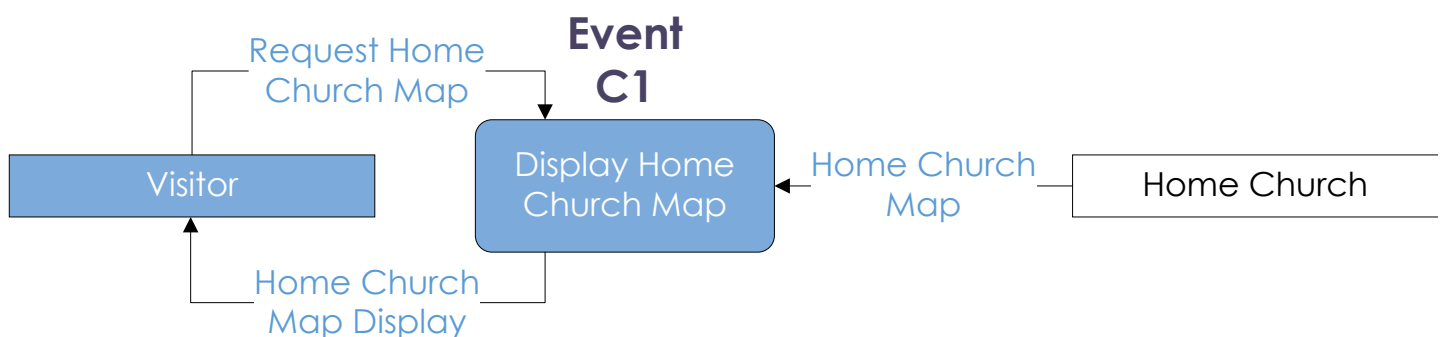
Data Stores include Member, User, Home Church, and Leadership Class entity groups from the ERD.

Member includes entities Member, User, Role, and MemberHomeChurch.

2.4.2.1 View Home Church Map

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
C1	Visitor	View home church map	Home church map viewed	Home church map	Visitor

ID	C1			
Name	Viewing home church map			
Primary Actor	Visitor			
Description	A visitor views the map of home churches.			
	Actor Action		System Response	
Typical Event Flow	1. Visitor navigates to home church map page			
			2. System displays home church map	
	Business Focus		System Focus	
Before Event	Trigger	View home church map	Preconditions	None
After Event	Conclusion	Home church map viewed	Postconditions	None



#Display **Home Church Map**

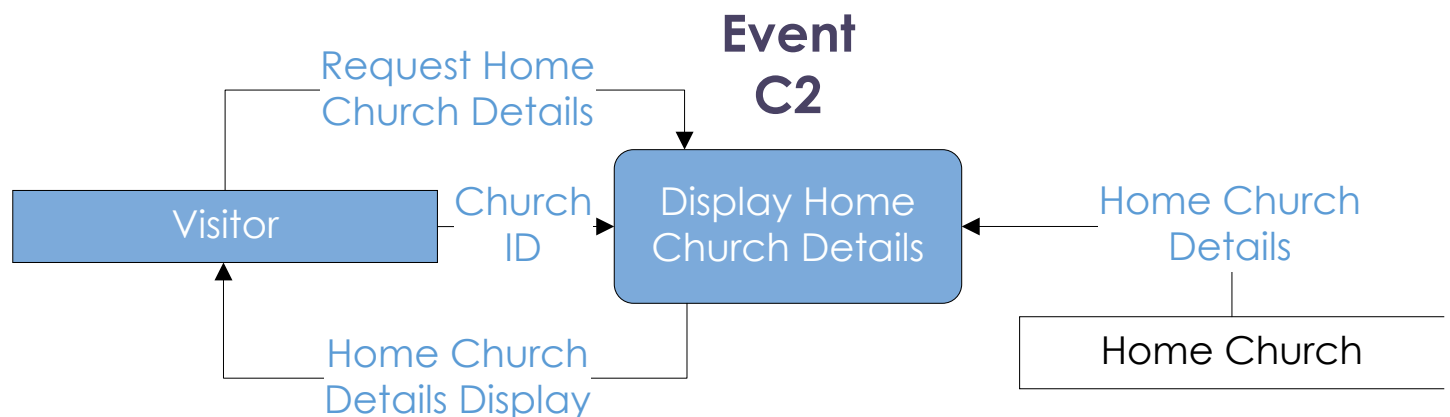
Get **Home Church Map** from **Home Church**

Display **Home Church Map**

2.4.2.2 View Home Church Details

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
C2	Visitor	View home church details	Home church details viewed	Home church details	Visitor

ID	C2				
Name	Looking up home church information				
Primary Actor	Visitor				
Description	A visitor views details about a home church of interest.				
Typical Event Flow	Actor Action			System Response	
	1. Visitor navigates to home church map page				
	2. Visitor clicks on home church of interest				
				3. System displays home church details	
Before Event	Business Focus			System Focus	
	Trigger	View home church details		Preconditions	Home church previously added to system
After Event	Conclusion	Home church details viewed		Postconditions	None



#Get **Church ID**

Get **Church ID** from Visitor

#Display **Home Church Details**

Using **Church ID**, Get corresponding **Home Church Details** from **Home Church**

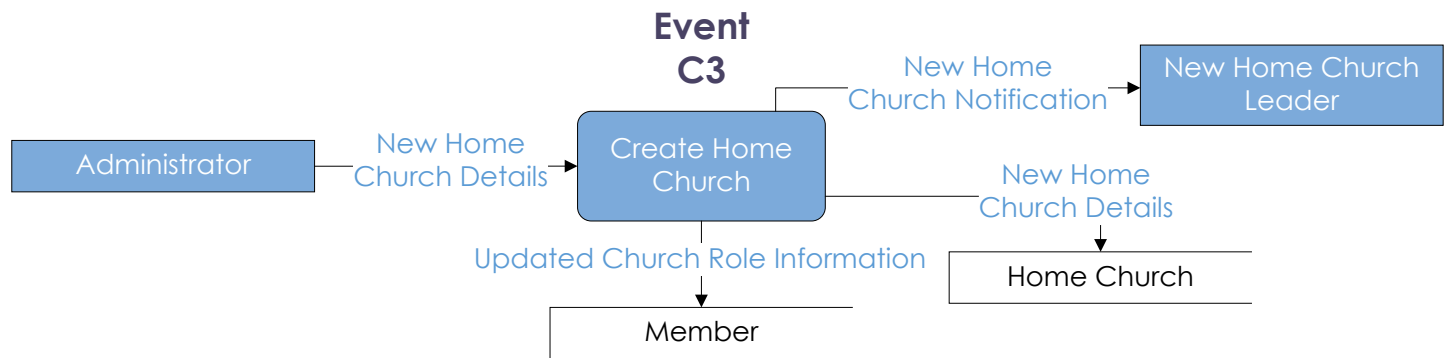
Display **Home Church Details**

2.4.2.3 Create Home Church

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
C3	Administrator	Create home church	Home church created	New home church	Database
				New home church notification	New home church leader
				Updated member information to be leader of new home church	Database

ID	C3				
Name	Adding a new home church				
Primary Actor	Administrator				
Other Actors	Leader of new home church				
Description	An administrator adds a new home church to the system.				
	Actor Action			System Response	
Typical Event Flow	1. Administrator enters new church details				
				2. System verifies all necessary fields have been entered correctly	
	3. Administrator searches and selects home church leader to be added				
				4. System creates user account for new home church leader	
				5. System sends confirmation email to new home church leader	
Alternate Event Flows	1. Administrator enters new church information				
				2. System verifies all necessary fields have been entered correctly	
				3. System displays error message	
	4. Administrator updates new church information to fix error message				
	Business Focus			System Focus	
Before Event	Trigger	Create home church		Preconditions	Administrator logged in to system
					Home church leader previously added to system as member
After Event	Conclusion	Home church created		Postconditions	Database updated to reflect new home church and updated member information

			Notification sent to new home church leader
--	--	--	---



#Create new Home Church Details

Get new Home Church Details from Administrator

Using new Church ID and Member ID, Store new Home Church Details in Home Church and Update entry in Member

#Notify new Home Church Leader

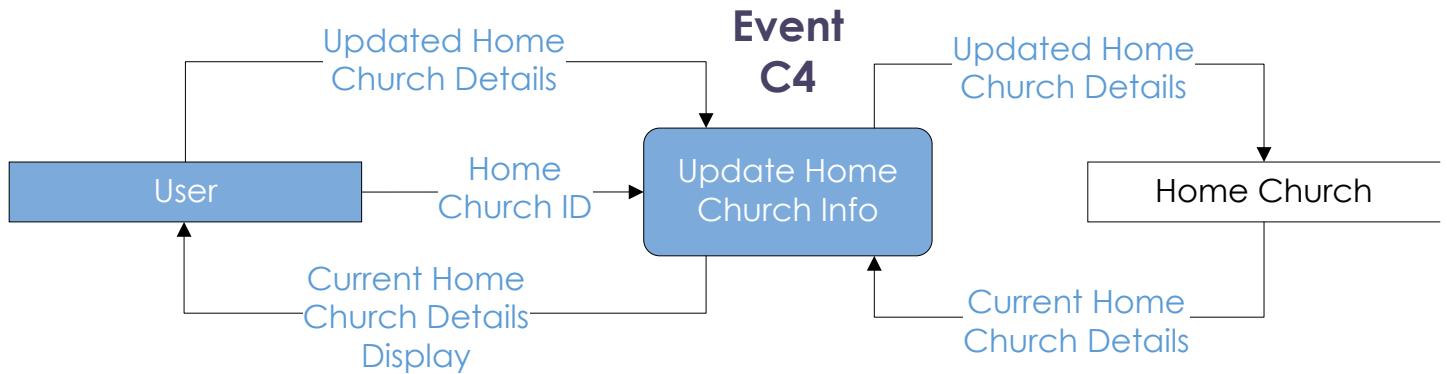
Using Church ID, Get corresponding Member ID Email of new Home Church Leader from Home Church

Using Email, send New Home Church Notification to new Home Church Leader Email

2.4.2.4 Update Home Church Information

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
C4	User	Update home church information	Home church information updated	Updated home church information	Database

ID	C4				
Name	Updating home church information				
Primary Actor	User				
Description	An administrator or home church leader updates a home church's information.				
	Actor Action			System Response	
Typical Event Flow	1. User edits a home church's information				
				2. System validates all necessary fields are still properly filled in	
				3. System updates home church information in database	
Alternate Event Flows	1. User edits a home church's information				
				2. System validates all necessary fields are still properly filled in	
				3. System displays error message	
	4. User updates home church information to fix error message				
	Business Focus			System Focus	
Before Event	Trigger	Update home church information	Preconditions	User logged in to system	
				Home church previously added to system	
After Event	Conclusion	Home church information updated	Postconditions	Database updated to reflect modified home church information	



#Get **Church ID**

Get **Church ID** from User

#Display current **Home Church Details**

Using **Church ID**, Get corresponding **Home Church Details** from **Home Church**

Display **Home Church Details**

#Update **Home Church Details**

Get updated **Home Church Details** from User

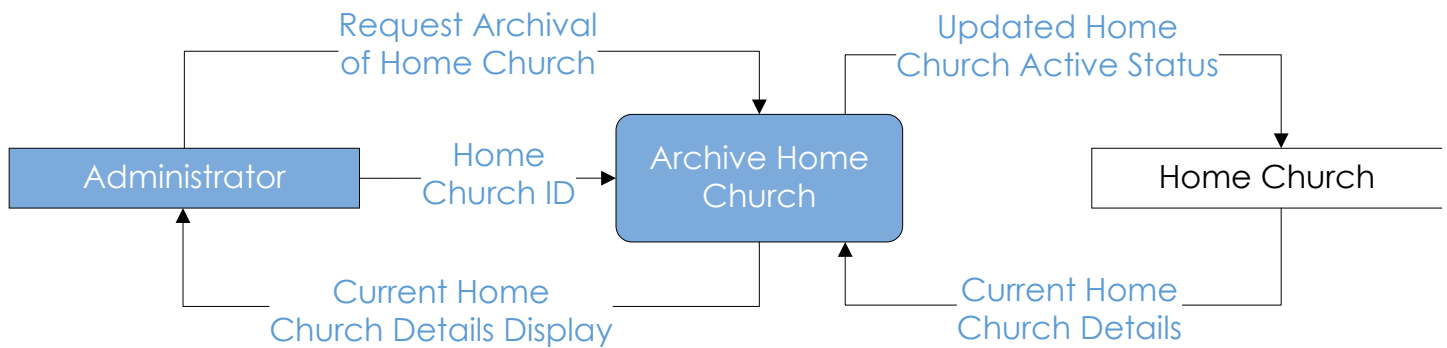
Using **Church ID**, Store updated **Home Church Details** in **Home Church**

2.4.2.5 Archive Home Church

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
C5	Administrator	Archive home church	Home church archived	Updated home church information to be archived	Database

ID	C5			
Name	Archiving a home church			
Primary Actor	Administrator			
Other Actors	Home church leader			
Description	Administrator navigates to a list of all home churches and archives a church.			
	Actor Action		System Response	
Typical Event Flow	1. Administrator navigates to list of home churches page			
	2. Administrator selects home church to be archived			
			3. System asks for confirmation	
	4. Administrator confirms			
			5. System updates home church to be archived	
			6. System deactivates leader account if no longer leader of at least one home church	
Alternate Event Flows	1. Administrator navigates to list of home churches			
	2. Administrator archives a home church			
			3. System asks for confirmation	
	4. Administrator declines			
	Business Focus		System Focus	
Before Event	Trigger	Archive home church	Preconditions	Administrator logged in to system
				Home church previously added to system
After Event	Conclusion	Home church archived	Postconditions	Database updated to reflect home church being archived
				Database updated to reflect home church leader no longer being leader of home church

Event C5



#Get Church ID

Get Church ID from User

#Display current Home Church Details

Using Church ID, Get corresponding Home Church Details from Home Church

Display Home Church Details

#Update Home Church Active Status

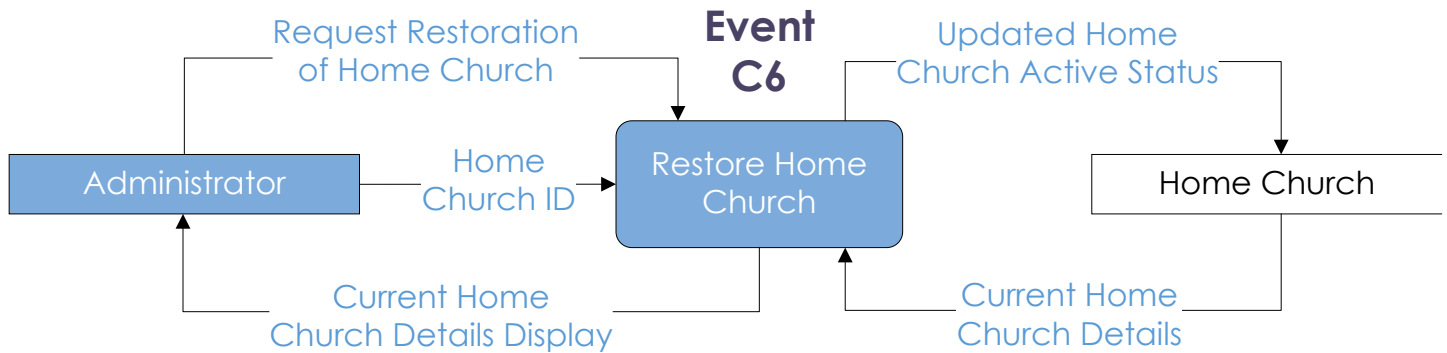
Get updated Home Church Active Status from User

Using Church ID, Store updated Home Church Active Status in Home Church

2.4.2.6 Restore Home Church

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
C6	Administrator	Restore home church	Home church restored	Updated home church information to no longer be archived	Database

ID	C6				
Name	Restoring home church				
Primary Actor	Administrator				
Other Actors	Home church leader				
Description	An administrator restores a home church that was previously archived.				
	Actor Action			System Response	
Typical Event Flow	1. Administrator navigates to list of archived home churches				
				2. System displays list of archived home churches	
	3. Administrator selects desired home church to be restored				
				4. System updates home church to no longer be archived	
				5. System reassigns former home church leader to restored home church	
				6. System reactivates former home church leader's account if previously deactivated	
	Business Focus			System Focus	
Before Event	Trigger	Restore home church	Preconditions	Administrator logged in to system	
				Home church previously archived	
After Event	Conclusion	Home church restored	Postconditions	Database updated to reflect home church no longer being archived	



#Get **Church ID**

Get **Church ID** from User

#Display current **Home Church Details**

Using **Church ID**, Get corresponding **Home Church Details** from **Home Church**

Display **Home Church Details**

#Update **Home Church Active Status**

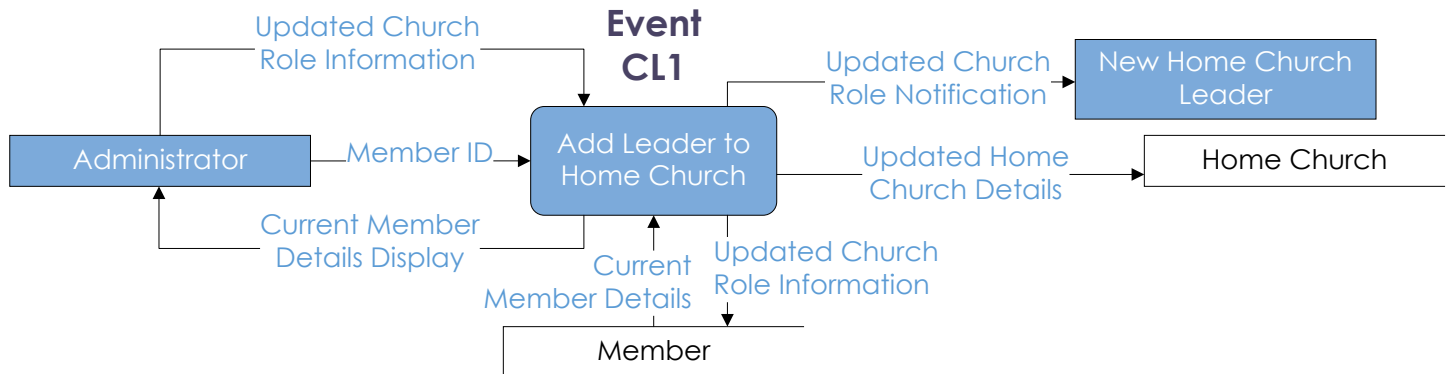
Get updated **Home Church Active Status** from User

Using **Church ID**, Store updated **Home Church Active Status** in **Home Church**

2.4.2.7 Add Leader to Home Church

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
CL1	Administrator	Add home church leader to home church	Home church leader added to home church	Updated member information to be new home church leader of home church	Database
				New home church leader notification	New home church leader

ID	CL1				
Name	Assigning a home church leader to a home church				
Primary Actor	Administrator				
Other Actors	New home church leader				
Description	An administrator updates an existing member to be the home church leader of an existing home church.				
	Actor Action			System Response	
Typical Event Flow	1. Administrator navigates to home church members page				
				2. System displays list of members for home church	
	3. Administrator selects desired member to be assigned as home church leader				
				4. System creates user account for new home church leader	
	Business Focus			System Focus	
Before Event	Trigger	Add leader to home church	Preconditions	Administrator logged in	
				Home church leader previously added to system as member	
After Event	Conclusion	Leader added to home church	Postconditions	Database updated to reflect new user account	
				Notification sent to new home church leader	



#Get Church ID

Get Church ID from Administrator

#Get Member ID

Get Member ID from Administrator

#Display current Member Details

Using Member ID, Get corresponding current Member Details from Member

Display current Member Details

#Update Church Role Information

Get updated Church Role Information from Administrator

Using Member ID, Store updated Church Role Information in Member

#Update Home Church Details

Get updated Home Church Details from Administrator

Using Church ID, Store updated Home Church Details in Home Church

#Notify new Home Church Leader

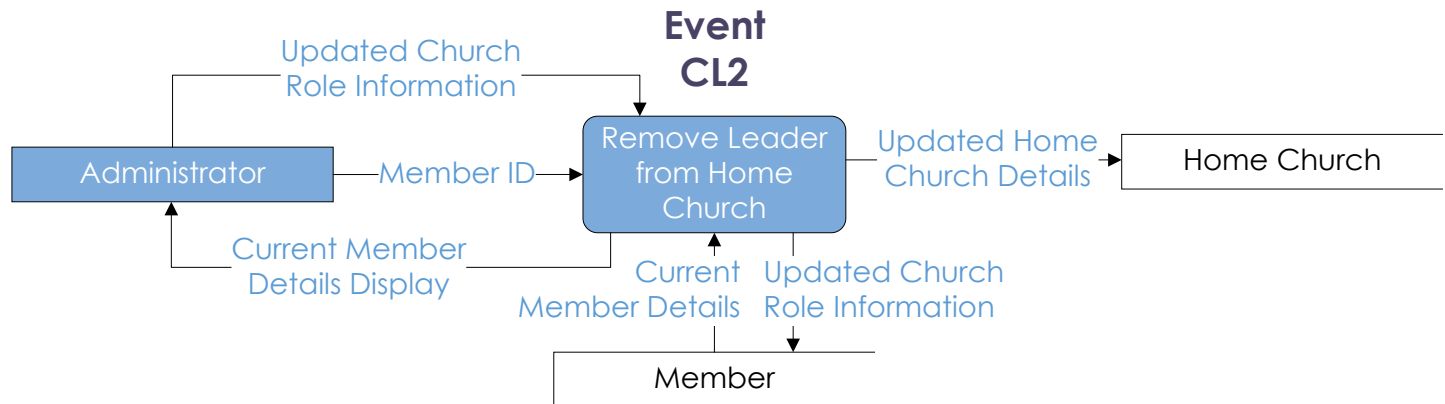
Using Member ID, Get corresponding Email of new Home Church Leader from Member

Using Email, send updated Church Role Notification to new Home Church Leader Email

2.4.2.8 Remove Leader from Home Church

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
CL2	Administrator	Remove home church leader from home church	Home church leader removed from home church	Updated member information to no longer be home church leader of home church	Database

ID	CL2				
Name	Removing home church leader from home church				
Primary Actor	Administrator				
Other Actors	Home church leader				
Description	An administrator views the list of home church members, and removes a home church leader from the home church.				
	Actor Action			System Response	
Typical Event Flow	1. Administrator navigates to home church members page				
				2. System displays list of members for home church	
	3. Administrator selects desired leader to be removed from home church				
				4. System removes leader from home church	
				5. System deactivates leader account if no longer leader of at least one home church	
	Business Focus			System Focus	
Before Event	Trigger	Remove home church leader from home church	Preconditions	Administrator logged in to system	
				Member previously added as home church leader	
After Event	Conclusion	Home church leader removed from home church	Postconditions	Database updated to reflect leader being removed from home church	



#Get **Church ID**

Get **Church ID** from Administrator

#Get **Member ID**

Get **Member ID** from Administrator

#Display current **Member Details**

Using **Member ID**, Get corresponding current **Member Details** from **Member**

Display current **Member Details**

#Update **Church Role Information**

Get updated **Church Role Information** from Administrator

Using **Member ID**, Store updated **Church Role Information** in **Member**

#Update **Home Church Details**

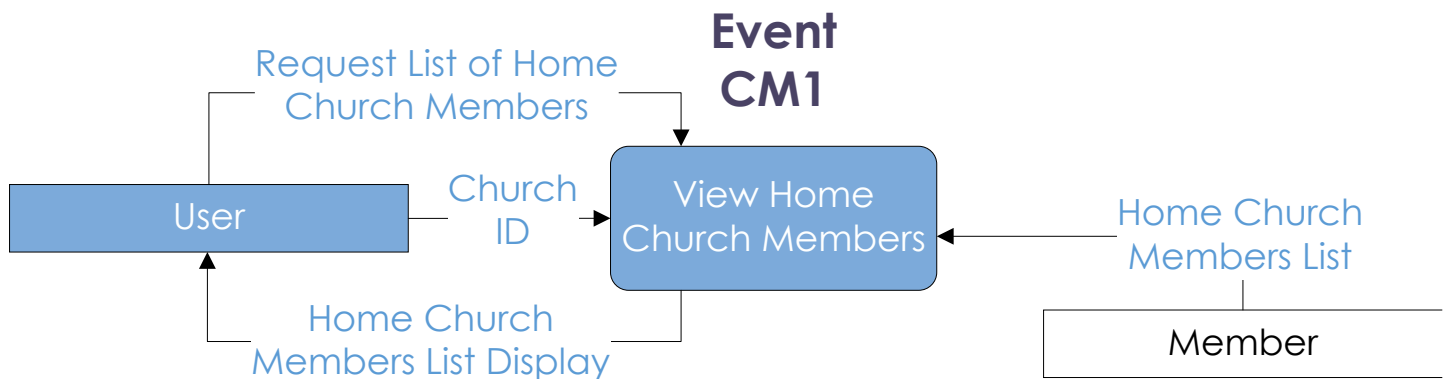
Get updated **Home Church Details** from Administrator

Using **Church ID**, Store updated **Home Church Details** in **Home Church**

2.4.2.9 View Home Church Members

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
CM1	User	View home church members	Home church members viewed	Home church members	User

ID	CM1				
Name	Viewing list of home church members				
Primary Actor	User				
Description	The home church leader or an administrator views the list of members for a home church.				
	Actor Action			System Response	
Typical Event Flow	1. User navigates to list of home church members page				
				2. System displays list of home church members	
Alternate Event Flows	1. Home church leader navigates to a different home church's list of members page				
				2. System gives error message	
	Business Focus			System Focus	
Before Event	Trigger	View home church members		Preconditions	User logged in to system
After Event	Conclusion	Home church members viewed		Postconditions	None



#Get **Church ID**

Get **Church ID** from User

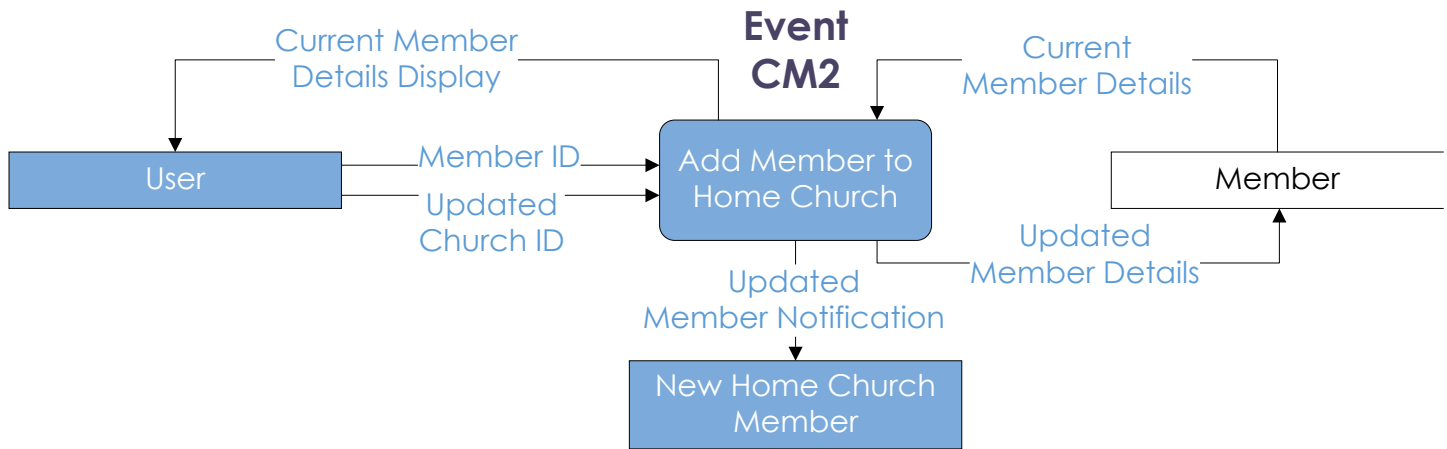
#Display **Home Church Members**

Using **Church ID**, Display corresponding **Home Church Members** from **Member**

2.4.2.10 Add Member to Home Church

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
CM2	User	Add member to home church	Member added to home church	Member updated	Database
				Added to home church notification	Member

ID	CM2				
Name	Adding a member to a home church				
Primary Actor	User				
Other Actors	Administrator				
Description	When a member starts going to a new home church, the home church leader or an administrator can add the member to the home church.				
	Actor Action			System Response	
Typical Event Flow	1. User searches for member				
				2. System displays list of members	
	3. User selects member's name to be added				
				4. System adds member to home church	
				5. System sends confirmation email to new home church member	
Alternate Event Flows	1. User searches for member				
				2. System displays list of members	
	3. User opts to add a new member				
	4. User provides member information				
				5. System sends new member added notification email to administrator	
	Business Focus			System Focus	
Before Event	Trigger	Add member to home church	Preconditions	User logged in to system	
				Member previously added to system by administrator	
After Event	Conclusion	Member added to home church	Postconditions	Database updated to reflect member being added to home church	
				Added to home church notification sent to member	



#Get **Member ID**

Get **Member ID** from User

#Display current **Member Details**

Using **Member ID**, Get corresponding current **Member Details** from **Member**

Display current **Member Details**

#Update **Member Details**

Get updated **Church ID** from User

Using **Member ID**, Store updated **Member Details** in **Member**

#Notify new Home Church Member

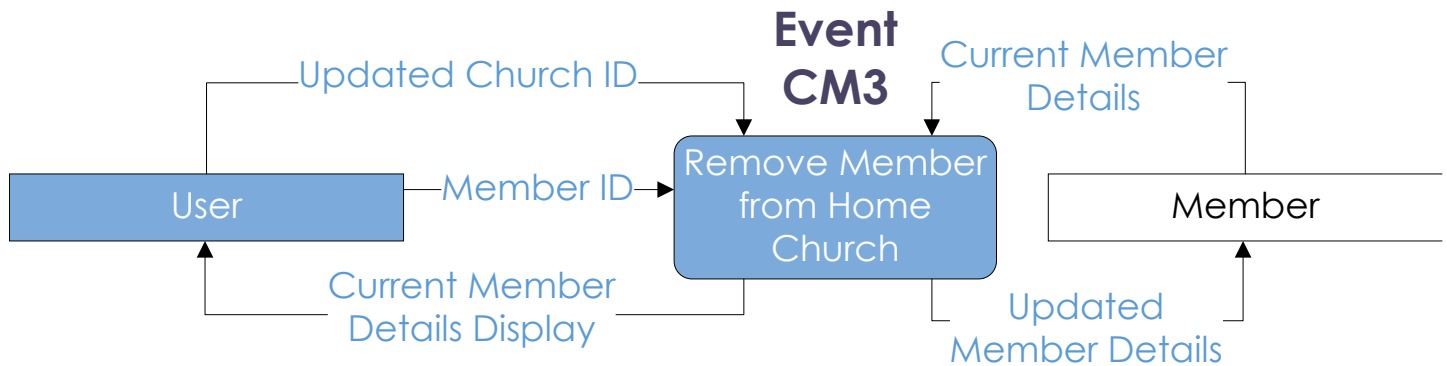
Using **Member ID**, Get corresponding **Email** of new Home Church Member from **Member**

Using **Email**, send **Updated Member Notification** to new Home Church Member **Email**

2.4.2.11 Remove Member from Home Church

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
CM3	User	Remove member from home church	Member removed from home church	Updated member information to be removed from the home church	Database

ID	CM3				
Name	Removing member from home church				
Primary Actor	User				
Other Actors	Member				
Description	The home church leader or an administrator removes a member from the home church.				
	Actor Action			System Response	
Typical Event Flow	1. User navigates to list of home church members page				
				2. System displays list of home church members	
	3. User selects desired member to be removed from home church				
				4. System removes member from home church	
Alternate Event Flows	1. Home church leader navigates to a different home church's list of members page				
				2. System gives error message	
	Business Focus			System Focus	
Before Event	Trigger	Remove member from home church	Preconditions	User logged in to system	
				Member previously added to home church	
After Event	Conclusion	Member removed from home church	Postconditions	Database updated to reflect member removed from home church	



#Get **Member ID**

Get **Member ID** from User

#Display current **Member Details**

Using **Member ID**, Get corresponding current **Member Details** from **Member**

Display current **Member Details**

#Update **Member Details**

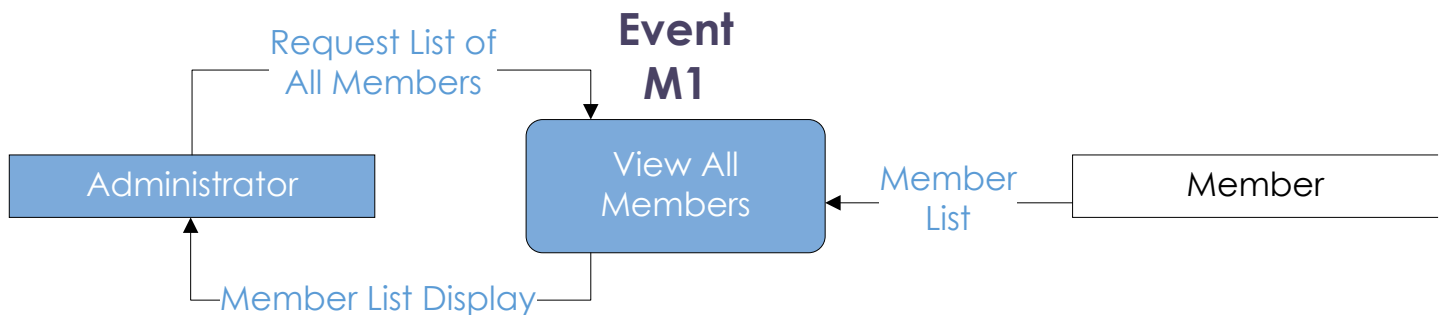
Get updated **Member Details** from User

Using **Member ID**, Store updated **Member Details** in **Member**

2.4.2.12 View All Members

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
M1	Administrator	View all members	All members viewed	All members	Administrator

ID	M1				
Name	Viewing all members from every home church				
Primary Actor	Administrator				
Description	An administrator views a list of all of members from every home church.				
	Actor Action			System Response	
Typical Event Flow	1. Administrator navigates to list of all members page				
				2. System displays list of all members	
	Business Focus			System Focus	
Before Event	Trigger	View all members	Preconditions	Administrator logged in to system	
After Event	Conclusion	All members viewed	Postconditions	None	



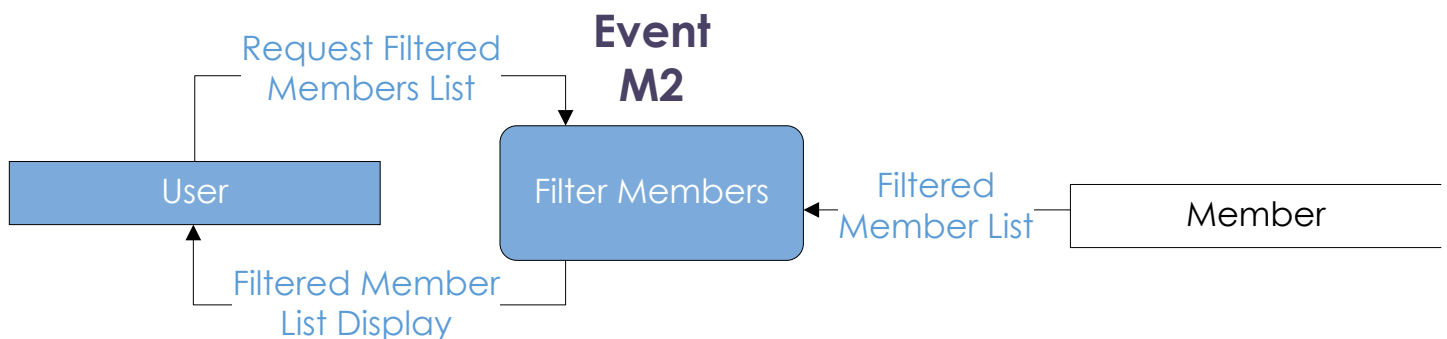
#Display **Member List**

Display **Member List** from **Member**

2.4.2.13 Filter Members

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
M2	User	Filter members	Members filtered	List of filtered members	User

ID	M2				
Name	Filtering members				
Primary Actor	User				
Description	A user filters members based on certain demographics, qualifications, or criteria.				
	Actor Action			System Response	
Typical Event Flow	1. User selects criteria in which to filter members				
				2. System displays list of members	
Alternate Event Flows	1. User selects criteria in which to filter members				
				2. System displays message that no members meet the criteria	
	Business Focus			System Focus	
Before Event	Trigger	Filter members	Preconditions	User logged in to system	
After Event	Conclusion	Members filtered	Postconditions	None	



#Display filtered **Member List**

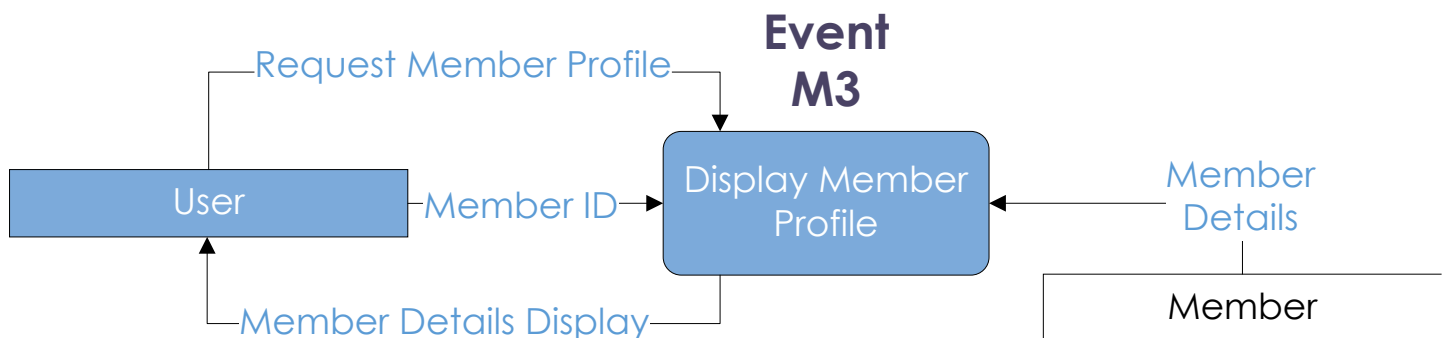
Using requested filter, Get filtered **Member List** from **Member**

Display filtered **Member List**

2.4.2.14 View Member Profile

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
M3	User	View member profile page	Member profile page viewed	Member profile page	User

ID	M3				
Name	Viewing member information				
Primary Actor	User				
Description	An administrator or home church leader views the information of a member.				
Typical Event Flow	Actor Action			System Response	
	1. User navigates to member profile page			2. System displays member information	
Before Event	Business Focus			System Focus	
	Trigger	View member information	Preconditions	User logged in to system Member previously added to system	
After Event	Conclusion	Member information viewed	Postconditions	None	



#Get **Member ID**

Get **Member ID** from User

#Display **Member Details**

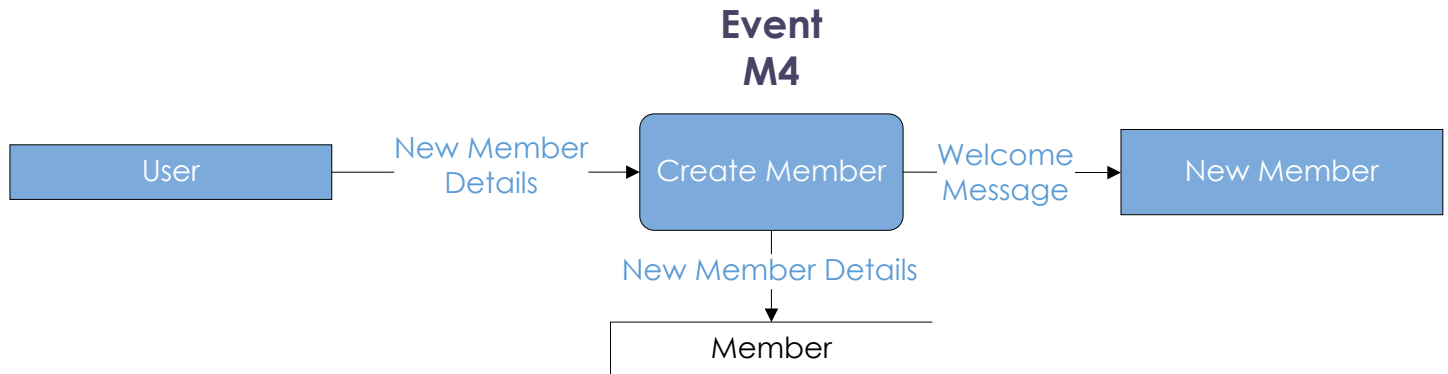
Using **Member ID**, Get corresponding **Member Details** from **Member**

Display **Member Details**

2.4.2.15 Create Member

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
M4	User	Create member	Member created	New member	Database
				Welcome message	Member

ID	M4			
Name	Adding new member			
Primary Actor	User			
Other Actors	New member			
Description	Using a new member's completed contact card, a user adds the member's name, phone number, and email to the system. The system then sends a welcome email to the member.			
	Actor Action		System Response	
Typical Event Flow	1. User inputs new member information			
			2. System verifies all necessary fields have been entered correctly	
			3. System adds new member to database	
			4. System sends welcome email to new member	
Alternate Event Flows	1. User inputs new member information			
			2. System verifies all necessary fields have been entered correctly	
			3. System displays error message	
	4. User updates new member information to fix error message			
	Business Focus		System Focus	
Before Event	Trigger	Create member	Preconditions	User logged in to system
After Event	Conclusion	Member created	Postconditions	Database updated to reflect member being added



#Create new **Member Details**

Get new **Member Details** from User

Using new **Member ID**, Store new **Member Details** in **Member**

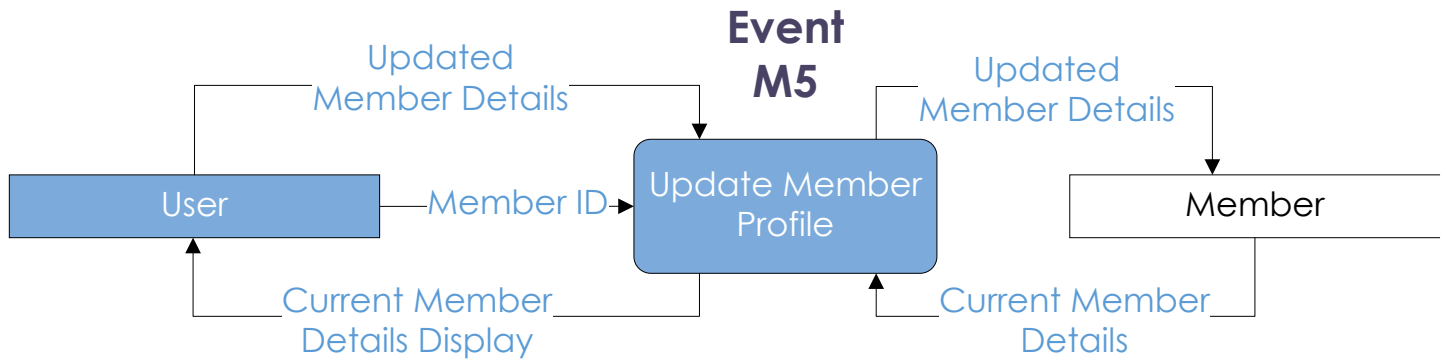
#Notify new Member

Using new Member **Email**, send Welcome Message to new Member **Email**

2.4.2.16 Update Member Profile

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
M5	User	Update member profile	Member profile updated	Updated member information	Database

ID	M5			
Name	Updating member information			
Primary Actor	User			
Other Actors	Member			
Description	A user updates the information of a member.			
	Actor Action		System Response	
Typical Event Flow	1. User enters updated member information			
			2. System verifies all necessary fields have been entered correctly	
			3. System updates member information	
Alternate Event Flows	1. User enters updated member information			
			2. System verifies all necessary fields have been entered correctly	
			3. System displays error message	
	4. User updates member information to fix error message			
	Business Focus		System Focus	
Before Event	Trigger	Update member information	Preconditions	User logged in to system
				Member previously added to system
After Event	Conclusion	Member information updated	Postconditions	Database updated to reflect modified member information



#Get **Member ID**

Get **Member ID** from User

#Display current **Member Details**

Using **Member ID**, Get corresponding **Member Details** from **Member**

Display **Member Details**

#Update **Member Details**

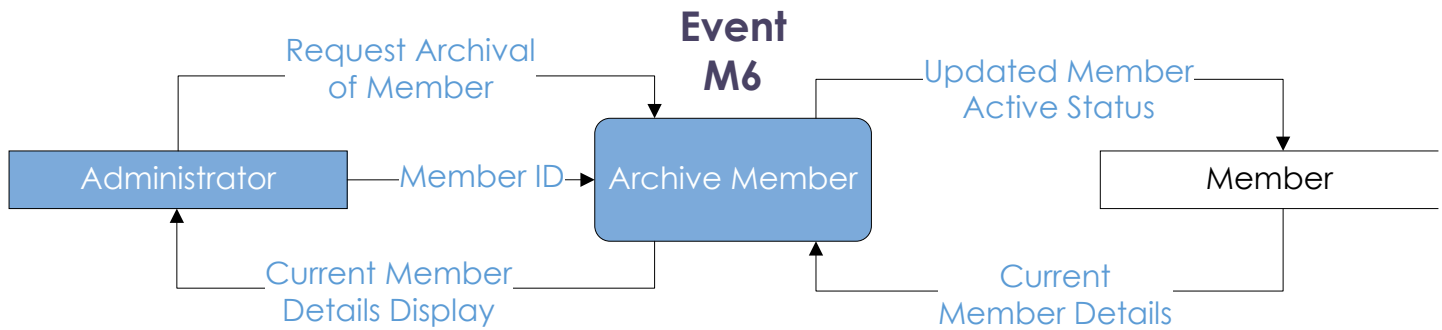
Get updated **Member Details** from User

Using **Member ID**, Store updated **Member Details** in **Member**

2.4.2.17 Archive Member

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
M6	Administrator	Archive member	Member archived	Updated member information to be archived	Database

ID	M6			
Name	Archiving a member			
Primary Actor	Administrator			
Other Actors	Member			
Description	Administrator navigates to a list of all members and archives a member.			
	Actor Action		System Response	
Typical Event Flow	1. Administrator navigates to list of all members page			
			2. System displays list of all members	
	3. Administrator selects desired member to be archived			
			4. System asks for confirmation	
	5. Administrator confirms			
			6. System updates member to be archived	
Alternate Event Flows	1. Administrator navigates to list of all members page			
			2. System displays list of all members	
	3. Administrator selects desired member to be archived			
			4. System asks for confirmation	
	4. Administrator declines			
	Business Focus		System Focus	
Before Event	Trigger	Archive member	Preconditions	Administrator logged in to system
				Member previously added to system
After Event	Conclusion	Member archived	Postconditions	Database updated to reflect member being archived



#Get **Member ID**

Get **Member ID** from Administrator

#Display current **Member Details**

Using **Member ID**, Get corresponding **Member Details** from **Member**

Display current **Member Details**

#Update **Member Active Status**

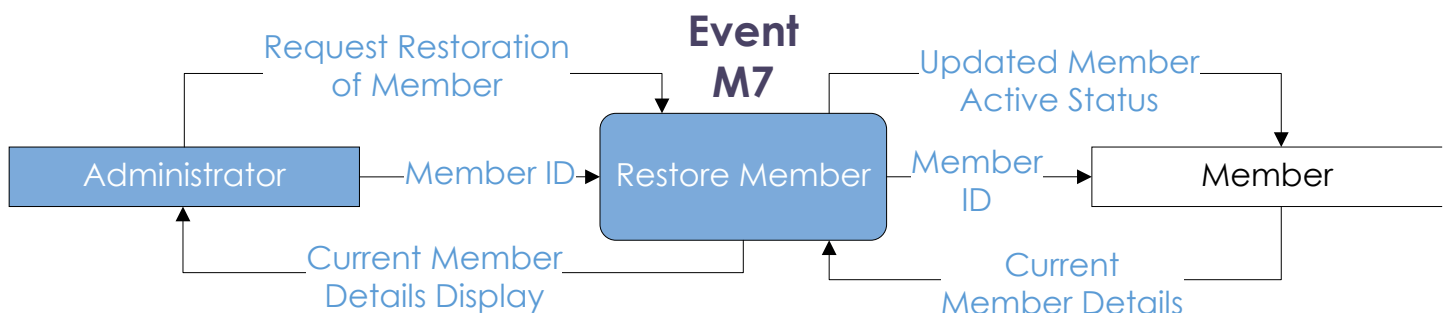
Get updated **Member Active Status** from Administrator

Using **Member ID**, Store updated **Member Active Status** in **Member**

2.4.2.18 Restore Member

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
M7	Administrator	Restore member	Member restored	Updated member information to no longer be archived	Database

ID	M7				
Name	Restoring member				
Primary Actor	Administrator				
Other Actors	Member				
Description	An administrator restores a member who was previously archived.				
	Actor Action			System Response	
Typical Event Flow	1. Administrator navigates to list of archived members				
				2. System displays list of archived members	
	3. Administrator selects desired member to be restored				
				4. System updates member to no longer be archived	
	Business Focus			System Focus	
Before Event	Trigger	Restore member	Preconditions	Administrator logged in to system	
				Member previously archived	
After Event	Conclusion	Member restored	Postconditions	Database updated to reflect member no longer being archived	



#Get **Member ID**

Get **Member ID** from Administrator

Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu

Team: ISA V

SYS 390 Information Systems Analysis

Dr. Nurkkala and Dr. Stanley

#Display current **Member Details**

Using **Member ID**, Get corresponding **Member Details** from **Member**

Display **Member Details**

#Update **Member Active Status**

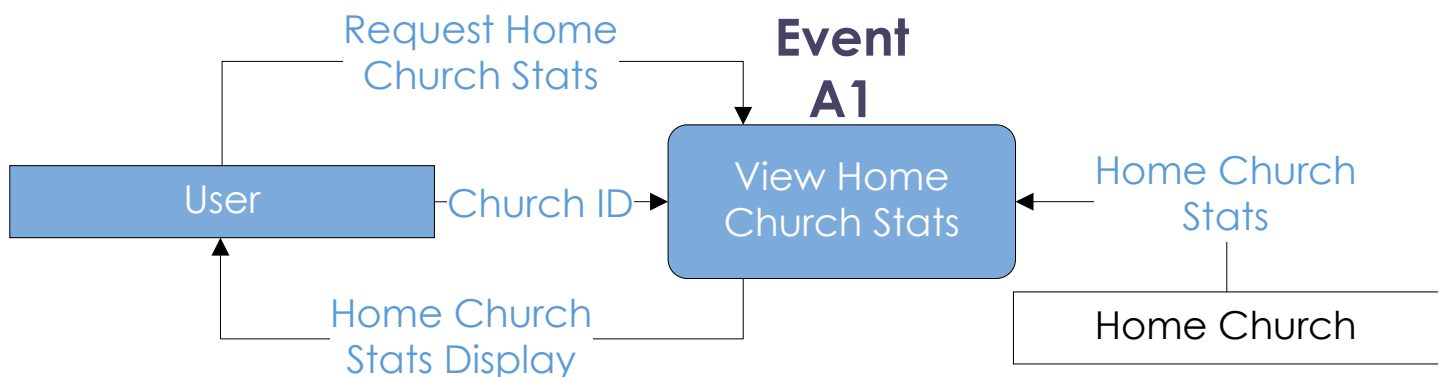
Get updated **Member Active Status** from Administrator

Using **Member ID**, Store updated **Member Active Status** in **Member**

2.4.2.19 View Home Church Statistics

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
A1	User	View home church statistics	Home church statistics viewed	Home church statistics and graphs	User

ID	A1			
Name	Viewing home church statistics			
Primary Actor	User			
Description	When logged in, a user views the statistics and graphs of a home church.			
	Actor Action		System Response	
Typical Event Flow	1. User navigates to home church statistics page			
			2. System displays home church statistics and graphs	
Alternate Event Flows	1. Home church leader navigates to a different home church's attendance page			
			2. System gives an error message	
	Business Focus		System Focus	
Before Event	Trigger	View home church statistics	Preconditions	User logged in to system
				Home church has recorded attendance at least once
After Event	Conclusion	Home church statistics viewed	Postconditions	None



#Get **Church ID**

Get **Church ID** from User

#Display **Home Church Stats**

Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu

Team: ISA V

SYS 390 Information Systems Analysis

Dr. Nurkkala and Dr. Stanley

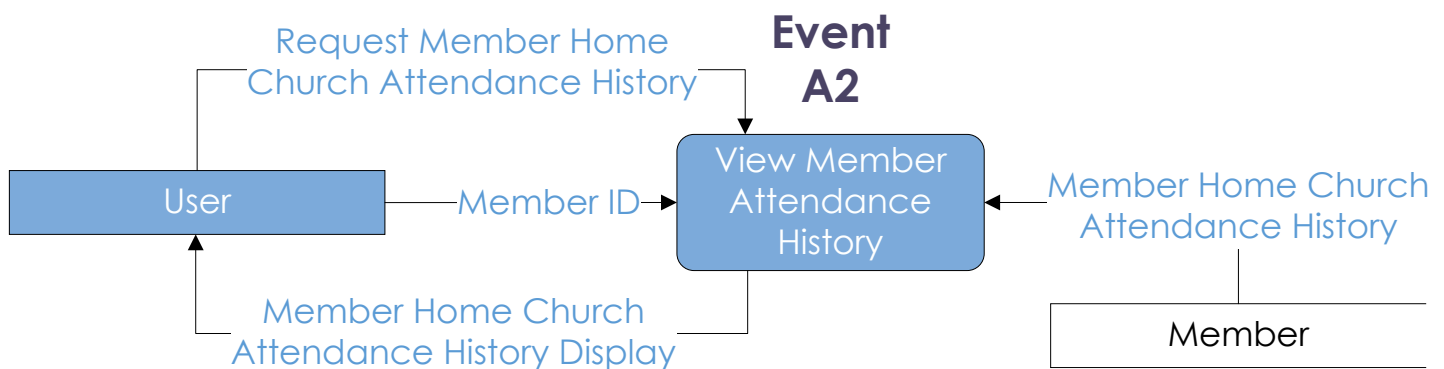
Using **Church ID**, Get corresponding **Home Church Stats** from **Home Church**

Display **Home Church Stats**

2.4.2.20 View Member Attendance History

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
A2	User	View member attendance history	Member attendance history viewed	Member attendance history	User

ID	A2				
Name	Viewing attendance history of member				
Primary Actor	User				
Description	The home church leader or an administrator views the attendance history of a home church member.				
	Actor Action		System Response		
Typical Event Flow	1. User navigates to member attendance history page				
			2. System displays attendance history of member		
Alternate Event Flows	1. Home church leader navigates to attendance history of different home church member				
			2. System gives error message		
	Business Focus		System Focus		
Before Event	Trigger	View member attendance history	Preconditions	User logged in to system	
				Home church has recorded attendance at least once	
After Event	Conclusion	Member attendance history viewed	Postconditions	None	



#Get **Member ID**

Get **Member ID** from User

Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu
Team: ISA V
SYS 390 Information Systems Analysis
Dr. Nurkkala and Dr. Stanley

#Display **Member Home Church Attendance History**

Using **Member ID**, Get corresponding **Member Home Church Attendance History** from
Member

Display **Member Home Church Attendance History**

2.4.2.21 Record Home Church Attendance

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
A3	Home church leader	Record home church attendance	Home church attendance recorded	New home church attendance information	Database

ID	A3			
Name	Taking attendance			
Primary Actor	Home church leader			
Other Actors	Member			
Description	Home church leader marks whether or not home church member attends home church on a particular day. System updates attendance statistics accordingly.			
	Actor Action		System Response	
Typical Event Flow	1. Home church leader navigates to home church attendance page			
			2. System displays list of home church members	
	3. Home church leader marks present members as present			
			4. System saves home church attendance	
	Business Focus		System Focus	
Before Event	Trigger	Home church leader	Preconditions	Home church leader logged in to system
After Event	Conclusion	Home church attendance recorded	Postconditions	Database updated to reflect modified home church statistics



#Get new Home Church Attendance for Date

Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu
Team: ISA V
SYS 390 Information Systems Analysis
Dr. Nurkkala and Dr. Stanley

Get new **Home Church Attendance for Date** from Home Church Leader

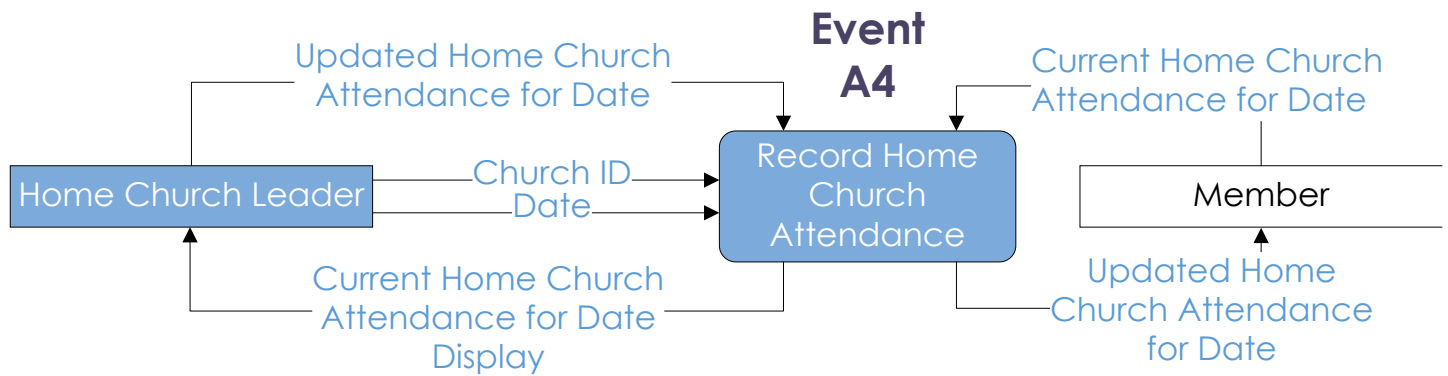
#Create new **Home Church Attendance for Date**

Using **Church ID** and **Member ID**, Store new **Home Church Attendance for Date** in **Member**

2.4.2.22 Update Home Church Attendance

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
A4	Home church leader	Update home church attendance record	Home church attendance record updated	Updated home church attendance information	Database

ID	A4				
Name	Updating attendance record of home church				
Primary Actor	Home church leader				
Other Actors	Member				
Description	A home church leader updates an attendance record from its previous value.				
	Actor Action		System Response		
Typical Event Flow	1. Home church leader navigates to home church attendance page				
			2. System displays list of attendance records		
	3. Home church leader edits an attendance record				
			4. System updates attendance record		
Alternate Event Flows	1. Home church leader navigates to attendance page of different home church				
			2. System gives error message		
	Business Focus		System Focus		
Before Event	Trigger	Update home church attendance record	Preconditions	Home church leader logged in to system	
				Home church previously added to system	
After Event	Conclusion	Home church attendance record updated	Postconditions	Database updated to reflect modified attendance record information	



#Get **Church ID**

Get **Church ID** from Home Church Leader

#Get **Date**

Get **Date** from Home Church Leader

#Display current **Home Church Attendance for Date**

Using **Church ID** and **Date**, Display current **Home Church Attendance for Date** from **Member**

#Update **Home Church Attendance for Date**

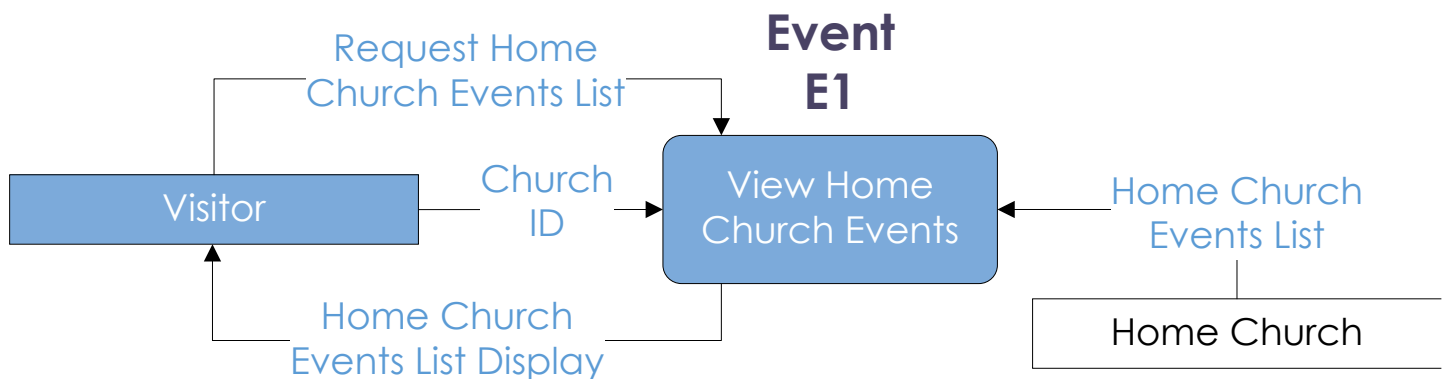
Get updated **Home Church Attendance for Date** from Home Church Leader

Using **Church ID** and **Date**, Store updated **Home Church Attendance for Date** in **Member**

2.4.2.23 View Home Church Events

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
E1	Visitor	View home church events	Home church events viewed	Home church events	Visitor

ID	E1			
Name	Viewing home church events			
Primary Actor	Visitor			
Description	A visitor views the events of a home church.			
	Actor Action		System Response	
Typical Event Flow	1. Visitor navigates to list of home church events page for the home church			
			2. System displays list of home church events	
Alternate Event Flows	1. Visitor navigates to list of home church events page for the home church			
			2. System displays message that there are no events for home church	
	Business Focus		System Focus	
Before Event	Trigger	View home church events	Preconditions	None
After Event	Conclusion	Home church events viewed	Postconditions	None



#Get **Church ID**

Get **Church ID** from Visitor

#Display **Home Church Events List**

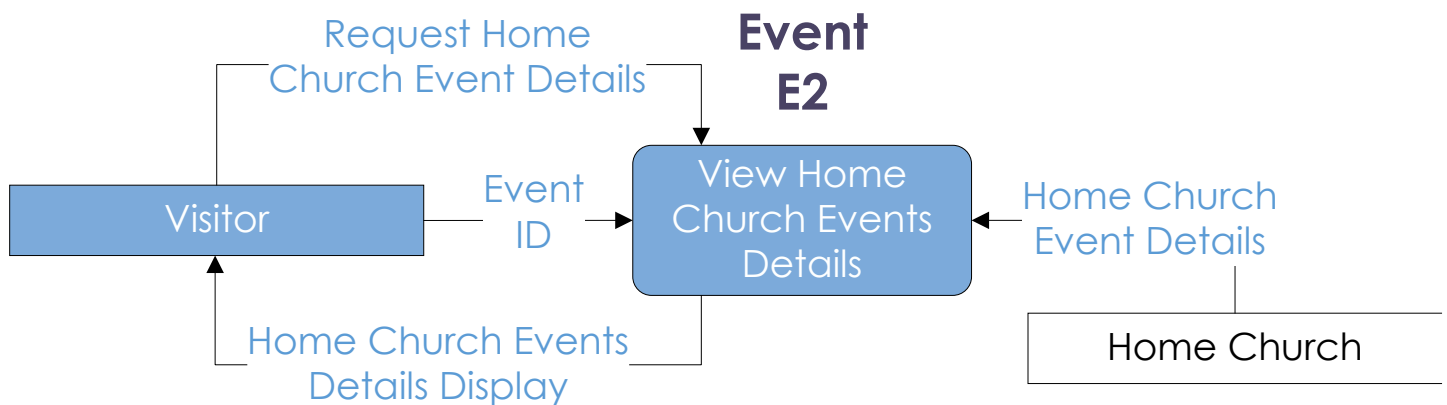
Using **Church ID**, Get corresponding **Home Church Events List** from **Home Church**

Display **Home Church Events List**

2.4.2.24 View Home Church Event Details

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
E2	Visitor	View home church event details	Home church event details viewed	Home church event details	Visitor

ID	E2				
Name	Viewing details of home church event				
Primary Actor	Visitor				
Description	A visitor views the details of a home church event.				
	Actor Action			System Response	
Typical Event Flow	1. Visitor navigates to home church event page				
				2. System displays home church event details	
	Business Focus			System Focus	
Before Event	Trigger	View home church event details	Preconditions	Home church event previously added to system	
After Event	Conclusion	Home church event details viewed	Postconditions	None	



#Get **Event ID**

Get **Event ID** from Visitor

#Display **Home Church Event Details**

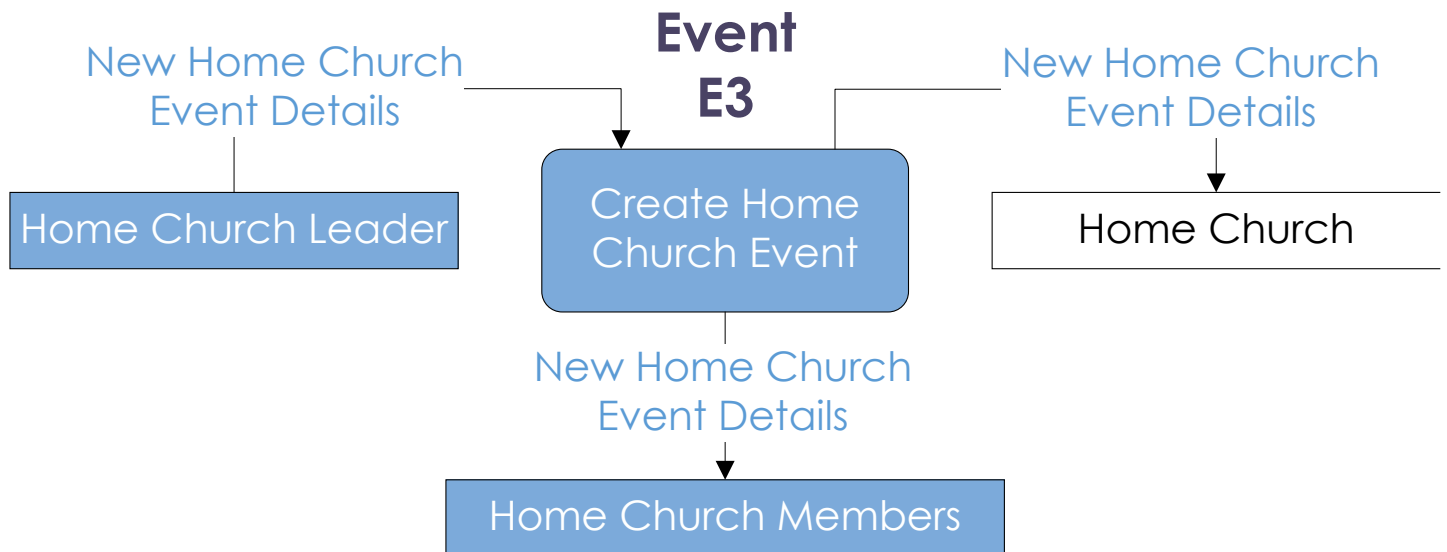
Using **Event ID**, Get corresponding **Home Church Event Details** from **Home Church**

Display **Home Church Event Details**

2.4.2.25 Add Event to Home Church

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
E3	Home church leader	Add event to home church	Event added to home church	New home church event	Database
				New home church event notification	Home church members

ID	E3				
Name	Adding an event				
Primary Actor	Home church leader				
Description	A home church leader adds an event to the home church calendar.				
	Actor Action			System Response	
Typical Event Flow	1. Home church leader enters event information				
				2. System verifies all necessary fields have been entered correctly	
				3. System adds new home church event to home church	
				4. System sends new home church event notification to home church members	
Alternate Event Flows	1. Home church leader enters event information				
				2. System verifies all necessary fields have been entered correctly	
				3. System displays error message	
	4. Home church leader updates event information to fix error message				
	Business Focus			System Focus	
Before Event	Trigger	Add event to home church calendar		Preconditions	Home church leader logged in to system
After Event	Conclusion	Event added to home church calendar		Postconditions	Database updated to reflect home church event being added
					New home church event notification sent to home church members



#Create new **Home Church Event Details**

Get new **Home Church Event Details** from Home Church Leader

Using **Church ID** and new **Event ID**, Store new **Home Church Event Details** in **Home Church**

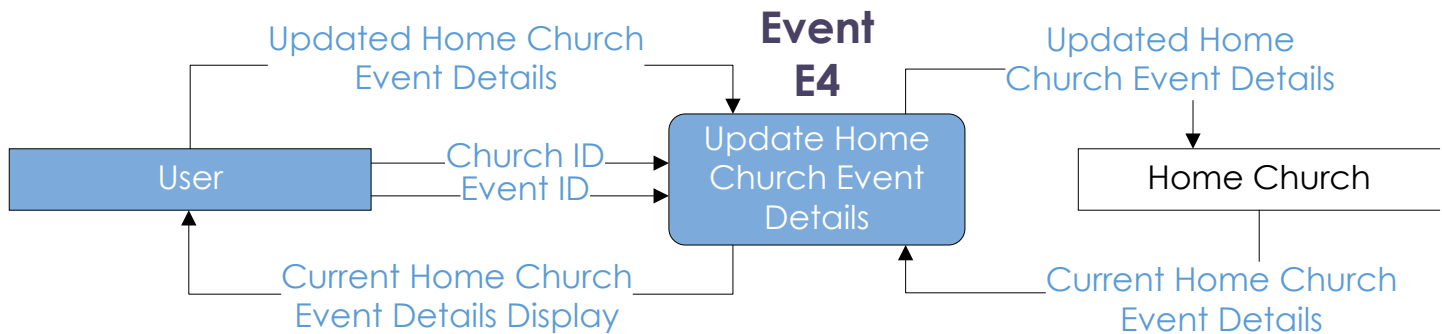
#Notify Home Church Members

Using new **Church ID**, send new **Home Church Event Details** notification to Home Church Member **Emails**

2.4.2.26 Update Home Church Event Details

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
E4	User	Update home church event details	Home church event details updated	Updated home church event details	Database

ID	E4				
Name	Updating home church event details				
Primary Actor	Home church leader				
Description	A home church leader updates the details of a home church event.				
	Actor Action			System Response	
Typical Event Flow	1. Home church leader edits a home church event's details				
				2. System validates all necessary fields are still properly filled in	
				3. System updates home church event information in database	
Alternate Event Flows	1. Home church leader edits a home church event's details				
				2. System validates all necessary fields are still properly filled in	
				3. System displays error message	
	4. Home church leader updates home church event information to fix error message				
	Business Focus			System Focus	
Before Event	Trigger	Update home church events details	Preconditions	Home church leader logged in to system	
				Home church event previously added to system	
After Event	Conclusion	Home church event details updated	Postconditions	Database updated to reflect modified home church event details	



#Get **Church ID**

Get **Church ID** from Home Church Leader

#Get **Event ID**

Get **Event ID** from Home Church Leader

#Display **Home Church Event Details**

Using **Church ID** and **Event ID**, Get corresponding **Home Church Event Details** from **Home Church**

Display current **Home Church Event Details**

#Update **Home Church Event Details**

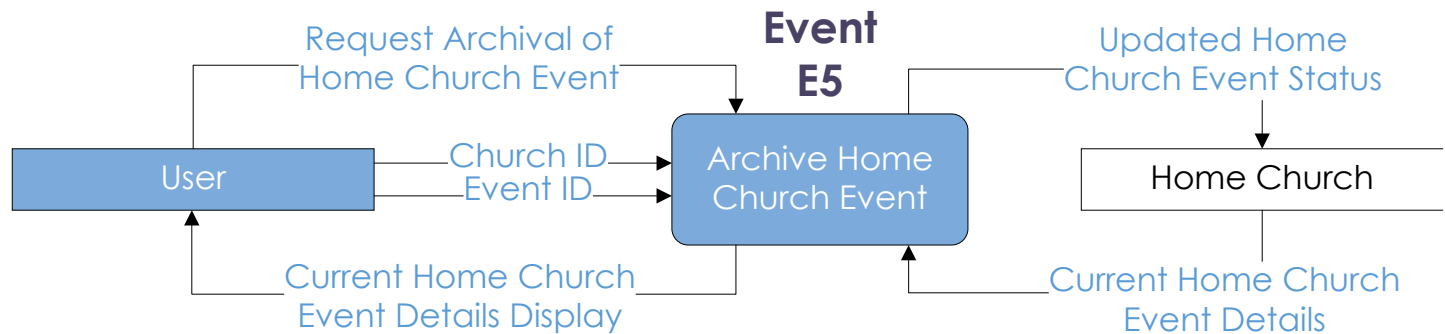
Get updated **Home Church Event Details** from User

Using **Event ID**, Store updated **Home Church Event Details** in **Home Church**

2.4.2.27 Archive Home Church Event

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
E5	User	Archive home church event	Home church event archived	Updated home church event information to be archived	Database

ID	E5				
Name	Archiving home church event				
Primary Actor	Home church leader				
Description	A home church leader archives a home church event.				
	Actor Action			System Response	
Typical Event Flow	1. Home church leader navigates to list of home church events page				
				2. System displays list of home church events	
	3. Home church leader selects desired home church event to archive it				
				4. System asks for confirmation	
	5. Home church leader confirms				
				6. System updates home church event to be archived	
Alternate Event Flows	1. Home church leader navigates to list of home church events page				
				2. System displays list of home church events	
	3. Home church leader selects desired home church event to archive it				
				4. System asks for confirmation	
	5. Home church leader denies				
	Business Focus			System Focus	
Before Event	Trigger	Archive home church event	Preconditions	Home church leader logged in to system	
				Home church event previously added to system	
After Event	Conclusion	Home church event archived	Postconditions	Database updated to reflect home church event being archived	



#Get **Church ID**

Get **Church ID** from Home Church Leader

#Get **Event ID**

Get **Event ID** from Home Church Leader

#Display **Home Church Event Details**

Using **Church ID** and **Event ID**, Get corresponding **Home Church Event Details** from **Home Church**

Display current **Home Church Event Details**

#Update **Home Church Event Active Status**

Get updated **Home Church Event Active Status** from User

Using **Event ID**, Store updated **Home Church Event Active Status** in **Home Church**

2.4.2.28 Restore Home Church Event

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
E6	User	Restore home church event	Home church event restored	Updated home church event information to no longer be archived	Database

ID	E6				
Name	Restoring home church event from archives				
Primary Actor	Home church leader				
Description	A home church leader restores a home church event that was previously archived.				
	Actor Action		System Response		
Typical Event Flow	1. Home church leader navigates to list of archived home church events				
			2. System displays list of archived home church events		
	3. Home church leader selects desired home church event				
			4. System updates home church event to no longer be archived		
	Business Focus		System Focus		
Before Event	Trigger	Restore home church event	Preconditions	Home church leader logged in to system	
				Home church event previously archived	
After Event	Conclusion	Home church event restored	Postconditions	Database updated to reflect home church event no longer being archived	



Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu
Team: ISA V
SYS 390 Information Systems Analysis
Dr. Nurkkala and Dr. Stanley

#Get **Church ID**

Get **Church ID** from Home Church Leader

#Get **Event ID**

Get **Event ID** from Home Church Leader

#Display **Home Church Event Details**

Using **Church ID** and **Event ID**, Get corresponding **Home Church Event Details** from **Home Church**

Display current **Home Church Event Details**

#Update **Home Church Event Active Status**

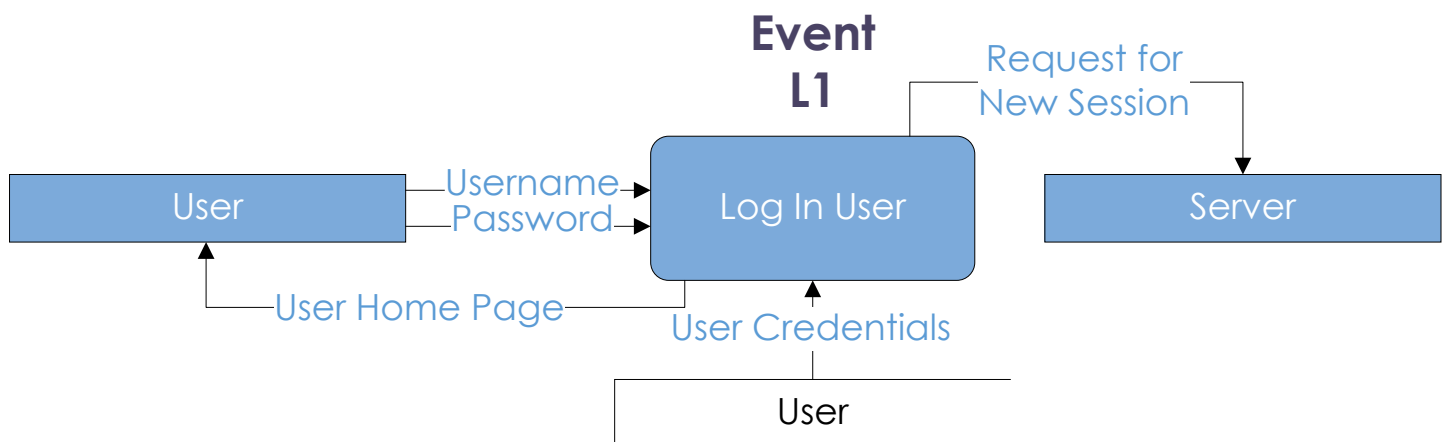
Get updated **Home Church Event Active Status** from User

Using **Event ID**, Store updated **Home Church Event Active Status** in **Home Church**

2.4.2.29 Log In

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
L1	User	Log in	Logged in	User home page	User
				Request for new session	Server

ID	L1			
Name	User log in			
Primary Actor	User			
Description	Once a user has been added in the database, the user logs in to the system.			
	Actor Action		System Response	
Typical Event Flow	1. User navigates to login page			
	2. User enters username and password			
			3. System validates username and password	
			4. System authenticates home church leader	
Alternate Event Flows	1. User navigates to login page			
	2. User enters username and password			
			3. System validates username and password	
			4. System displays error message	
	5. User reenters username and password			
	Business Focus		System Focus	
Before Event	Trigger	Log in	Preconditions	User has account
After Event	Conclusion	Logged in	Postconditions	User session is created



#Identify User

Get **Username** and **Password** from User

Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu
Team: ISA V
SYS 390 Information Systems Analysis
Dr. Nurkkala and Dr. Stanley

Using **Username**, Get **User Credentials** from **User**

Authenticate **Username** and **Password** with **User Credentials**

#Log In User

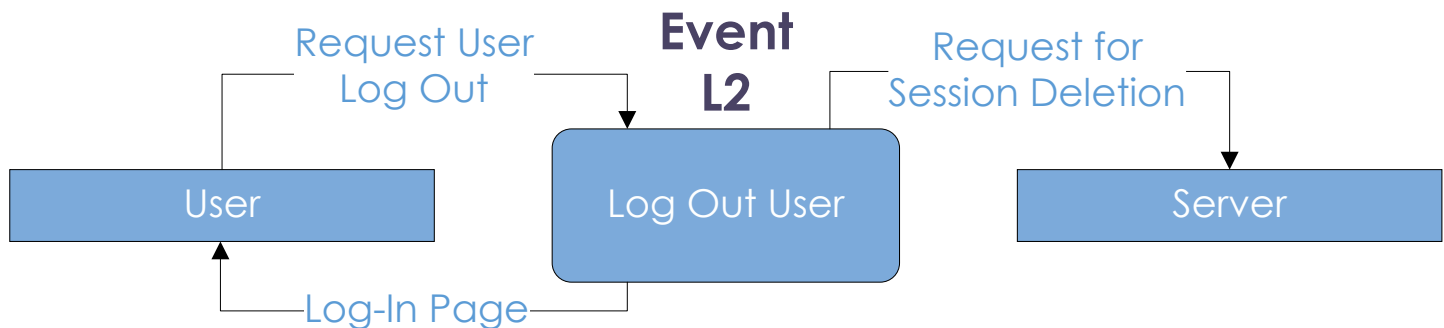
Display User Home Page

Send Request for New Session to Server

2.4.2.30 Log Out

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
L2	User	Log out	Logged out	Log-in page	User
				Request to delete user session	Server

ID	L2			
Name	User log out			
Primary Actor	User			
Description	A user logs out of the system.			
	Actor Action		System Response	
Typical Event Flow	1. User clicks button to log out			
			2. System logs the user out	
	Business Focus		System Focus	
Before Event	Trigger	Log out	Preconditions	User logged in to system
After Event	Conclusion	Logged out	Postconditions	User session is deleted



#Log Out User

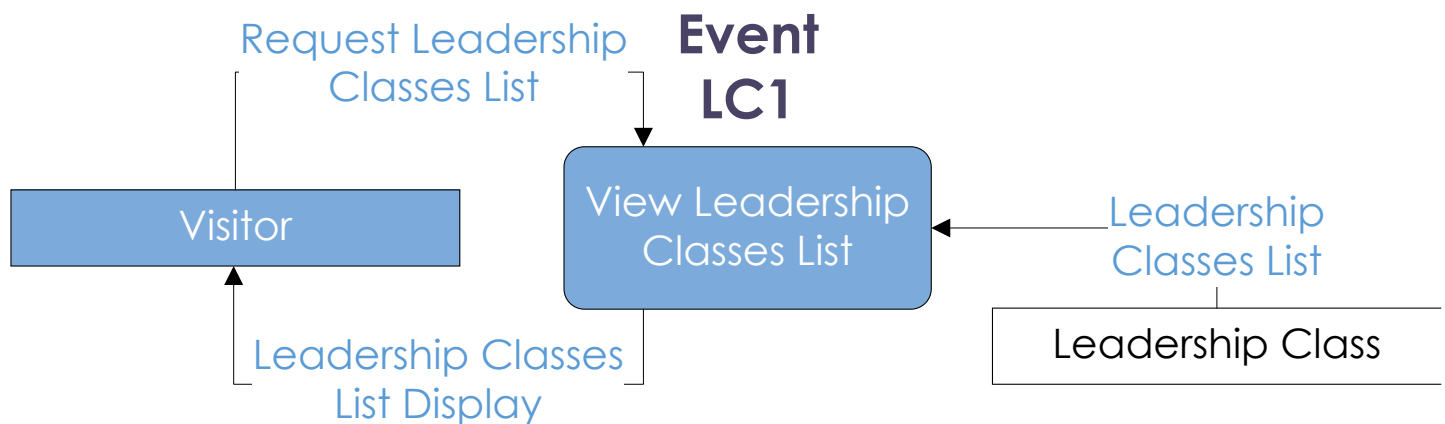
Using **Username** from User, Send Request for Session Deletion to Server

Display User Home Page

2.4.2.31 View List of Leadership Classes

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
LC1	Visitor	View list of leadership classes	List of leadership classes viewed	List of leadership classes	Visitor

ID	LC1				
Name	Viewing leadership classes				
Primary Actor	Visitor				
Description	A visitor views a list of available leadership classes.				
	Actor Action		System Response		
Typical Event Flow	1. Visitor navigates to list of leadership classes page				
			2. System displays list of leadership classes		
Alternate Event Flows	1. Visitor navigates to list of leadership classes page				
			2. System displays no leadership classes available message		
	Business Focus		System Focus		
Before Event	Trigger	View list of leadership classes	Preconditions	None	
After Event	Conclusion	List of leadership classes viewed	Postconditions	None	



#Display Leadership Classes List

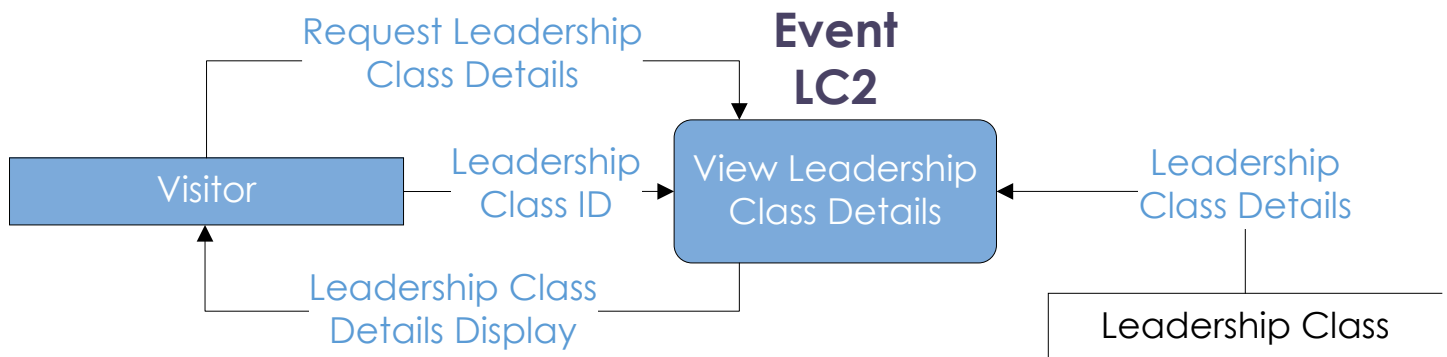
Get Leadership Classes List from Leadership Class

Display Leadership Classes List

2.4.2.32 View Leadership Class Details

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
LC2	Visitor	View leadership class details	Leadership class details viewed	Leadership class details	Visitor

ID	LC2			
Name	Viewing details of leadership class			
Primary Actor	Visitor			
Description	A visitor views the details of a leadership class.			
	Actor Action		System Response	
Typical Event Flow	1. Visitor navigates to leadership class page			
			2. System displays leadership class details	
	Business Focus		System Focus	
Before Event	Trigger	View leadership class details	Preconditions	Leadership class previously added to system
After Event	Conclusion	Leadership class details viewed	Postconditions	None



#Get **Leadership Class ID**

Get **Leadership Class ID** from Visitor

#Display **Leadership Class Details**

Using **Leadership Class ID**, Get corresponding **Leadership Class Details** from **Leadership Class**

Display **Leadership Class Details**

2.4.2.33 Add New Leadership Class

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
LC3	Administrator	Add new leadership class	New leadership class added	New leadership class	Database

ID	LC3				
Name	Adding a leadership class				
Primary Actor	Administrator				
Description	An administrator adds a leadership class.				
	Actor Action		System Response		
Typical Event Flow	1. Administrator enters leadership class information				
			2. System verifies all necessary fields have been entered correctly		
			3. System adds new leadership class to database		
Alternate Event Flows	1. Administrator enters leadership class information				
			2. System verifies all necessary fields have been entered correctly		
			3. System displays error message		
	4. Administrator updates leadership class information to fix error message				
	Business Focus		System Focus		
Before Event	Trigger	Add new leadership class	Preconditions	Administrator logged in to system	
After Event	Conclusion	New leadership class added	Postconditions	Database updated to reflect leadership class being added	

Event LC3



#Create new **Leadership Class Details**

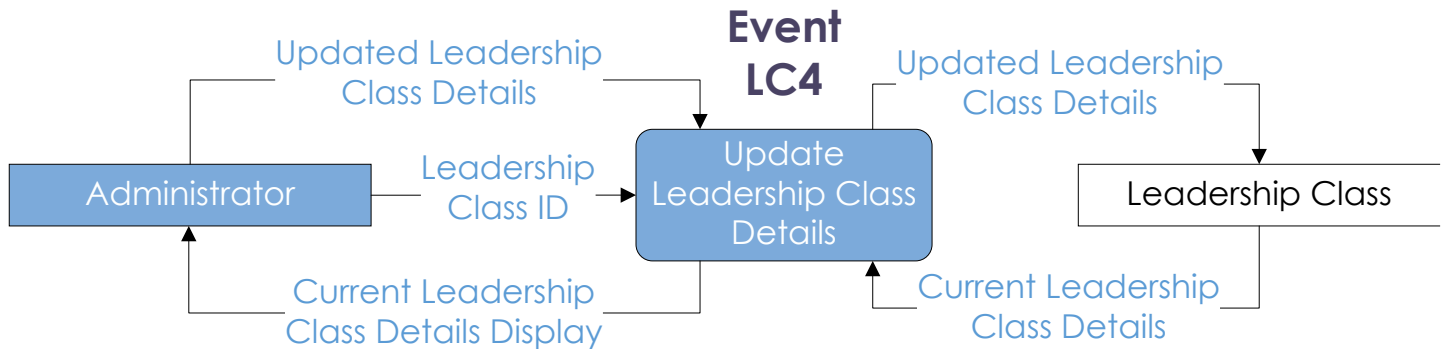
Get new **Leadership Class Details** from Administrator

Using new **Leadership Class ID**, Store new **Leadership Class Details** in **Leadership Class**

2.4.2.34 Update Leadership Class Details

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
LC4	Administrator	Update leadership class details	Leadership class details updated	Updated leadership class details	Database

ID	LC4				
Name	Updating leadership class details				
Primary Actor	Administrator				
Description	An administrator updates the details of a leadership class.				
	Actor Action		System Response		
Typical Event Flow	1. Administrator edits details of a leadership class				
			2. System validates all necessary fields are still properly filled in		
			3. System updates leadership class information in database		
Alternate Event Flows	1. Administrator edits details of a leadership class				
			2. System validates all necessary fields are still properly filled in		
			3. System displays error message		
	4. Administrator updates leadership class information to fix error message				
	Business Focus		System Focus		
Before Event	Trigger	Update leadership class details	Preconditions	Administrator logged in to system	
				Leadership class previously added to system	
After Event	Conclusion	Leadership class details updated	Postconditions	Database updated to reflect modified leadership class details	



#Get **Leadership Class ID**

Get **Leadership Class ID** from Administrator

#Display **Leadership Class Details**

Using **Leadership Class ID**, Get corresponding **Leadership Class Details** from **Leadership Class**

Display current **Leadership Class Details**

#Update **Leadership Class Details**

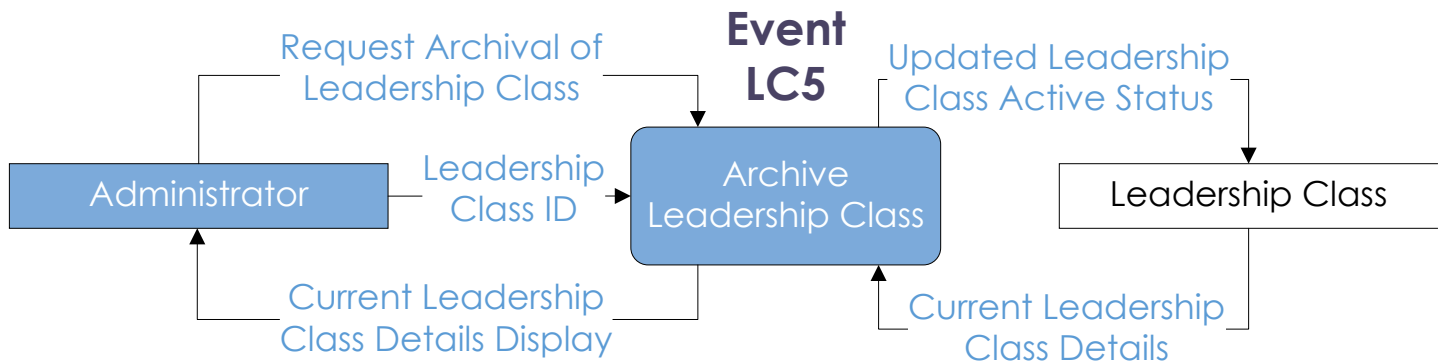
Get updated **Leadership Class Details** from User

Using **Leadership Class ID**, Store updated **Leadership Class Details** in **Leadership Class**

2.4.2.35 Archive Leadership Class

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
LC5	Administrator	Archive leadership class	Leadership class archived	Updated leadership class information to be archived	Database

ID	LC5				
Name	Archiving leadership class				
Primary Actor	Administrator				
Description	An administrator archives a leadership class.				
	Actor Action		System Response		
Typical Event Flow	1. Administrator navigates to list of leadership classes page				
			2. System displays list of leadership classes		
	3. User selects desired leadership class				
			4. System asks for confirmation		
	5. Administrator confirms				
			6. System updates leadership class to be archived		
Alternate Event Flows	1. Administrator navigates to list of leadership classes page				
			2. System displays list of leadership classes		
	3. User selects desired leadership class				
			4. System asks for confirmation		
	5. Administrator denies				
	Business Focus		System Focus		
Before Event	Trigger	Archive leadership class	Preconditions	Administrator logged in to system	
				Leadership class previously added to system	
After Event	Conclusion	Leadership class archived	Postconditions	Database updated to reflect leadership class being archived	



#Get **Leadership Class ID**

Get **Leadership Class ID** from Administrator

#Display **Leadership Class Details**

Using **Leadership Class ID**, Get corresponding **Leadership Class Details** from **Leadership Class**

Display current **Leadership Class Details**

#Update **Leadership Class Active Status**

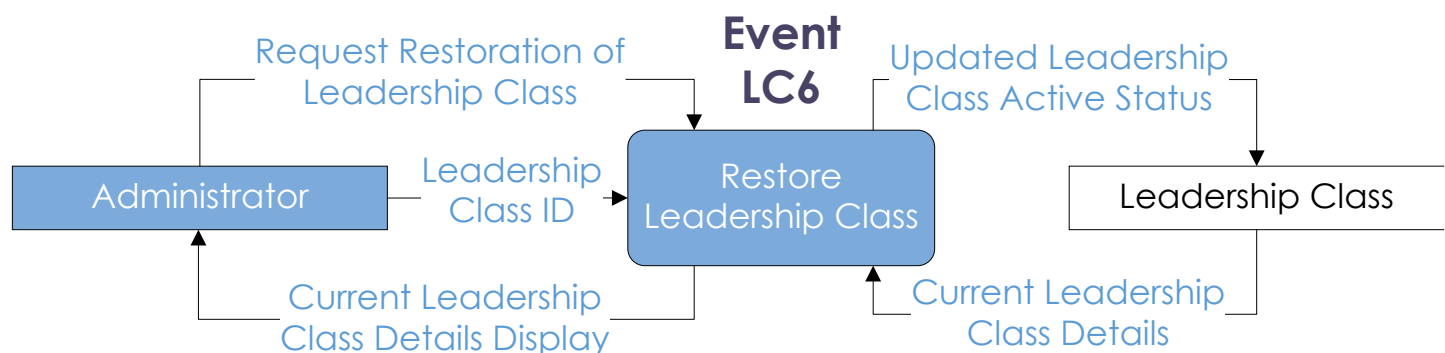
Get updated **Leadership Class Active Status** from User

Using **Leadership Class ID**, Store updated **Leadership Class Active Status** in **Leadership Class**

2.4.2.36 Restore Leadership Class

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
LC6	Administrator	Restore leadership class	Leadership class restored	Updated leadership class information to no longer be archived	Database

ID	LC6				
Name	Restoring leadership class from being archived				
Primary Actor	Administrator				
Description	An administrator restores a leadership class that was previously archived.				
	Actor Action		System Response		
Typical Event Flow	1. Administrator navigates to list of archived leadership classes				
			2. System displays list of archived leadership classes		
	3. User selects desired leadership class				
			4. System updates leadership class to no longer be archived		
	Business Focus		System Focus		
Before Event	Trigger	Restore leadership class	Preconditions	Administrator logged in to system	
				Leadership class previously archived	
After Event	Conclusion	Leadership class restored	Postconditions	Database updated to reflect leadership class no longer being archived	



#Get **Leadership Class ID**

Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu
Team: ISA V
SYS 390 Information Systems Analysis
Dr. Nurkkala and Dr. Stanley

Get **Leadership Class ID** from Administrator

#Display **Leadership Class Details**

Using **Leadership Class ID**, Get corresponding **Leadership Class Details** from **Leadership Class**

Display current **Leadership Class Details**

#Update **Leadership Class Active Status**

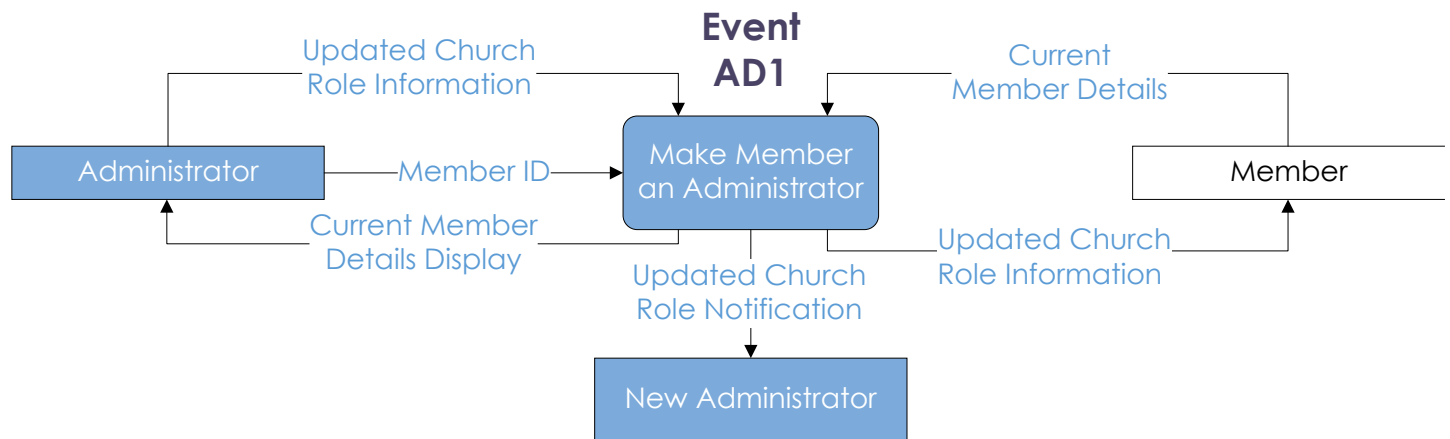
Get updated **Leadership Class Active Status** from User

Using **Leadership Class ID**, Store updated **Leadership Class Active Status** in **Leadership Class**

2.4.2.37 Give Administrator Role to Member

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
AD1	Administrator	Give administrator role to member	Administrator role given to member	Updated member information to have administrator role	Database
				New administrator role notification	New administrator

ID	AD1				
Name	Giving administrator role to member				
Primary Actor	Administrator				
Other Actors	New administrator				
Description	An administrator updates an existing member to have an administrator role.				
Typical Event Flow	Actor Action			System Response	
	1. Administrator navigates to list of all members				
				2. System displays list of all members	
	3. Administrator selects desired member				
				4. System updates member to have administrator role	
Before Event	Business Focus			System Focus	
	Trigger	Give administrator role to member	Preconditions	Administrator logged in to system	
				New administrator already in system as member	
	Conclusion	Administrator role given to member	Postconditions	Database updated to reflect member being given administrator role	
After Event				New administrator role notification sent to new administrator	



#Get **Member ID**

Get **Member ID** from Administrator

#Display current **Member Details**

Using **Member ID**, Get corresponding **Current Member Details** from **Member**

Display **Current Member Details**

#Update **Church Role Information**

Get updated **Church Role Information** from Administrator

Using **Member ID**, Store updated **Church Role Information** in **Member**

#Notify new Administrator

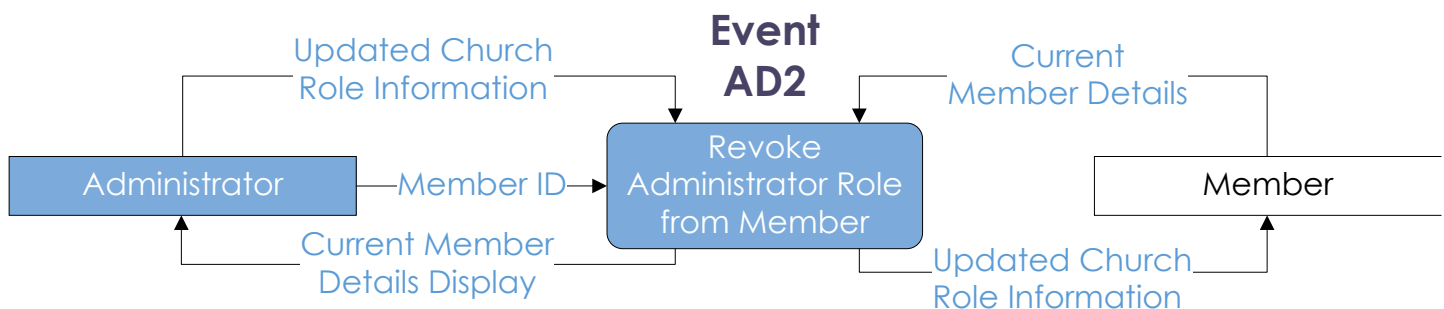
Using **Member ID**, Get corresponding **Email** of new Administrator from **Member**

Using **Email**, send **Updated Church Role Notification** to new Administrator **Email**

2.4.2.38 Revoke Administrator Role

ID	Source	Trigger	Response	Major Outputs	Ext Destinations
AD2	Administrator	Revoke administrator role from member	Administrator role revoked from member	Updated member information to no longer have administrator role	Database

ID	AD2			
Name	Revoking administrator role			
Primary Actor	Administrator			
Other Actors	Administrator whose role is getting revoked			
Description	An administrator revokes the administrator role from another administrator.			
Typical Event Flow	Actor Action		System Response	
	1. Administrator navigates to list of all members		2. System displays list of all members	
	3. Administrator selects desired administrator			
			4. System updates former administrator to no longer have administrator role	
Before Event	Business Focus		System Focus	
	Trigger	Revoke administrator role from member	Preconditions	Administrator logged in to system Other administrator previously had administrator role
After Event	Conclusion	Administrator role revoked from member	Postconditions	Database updated to reflect administrator getting role revoked



#Get **Member ID**

Get **Member ID** from Administrator

Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu

Team: ISA V

SYS 390 Information Systems Analysis

Dr. Nurkkala and Dr. Stanley

#Display current **Member Details**

Using **Member ID**, Get corresponding **Current Member Details** from **Member**

Display **Current Member Details**

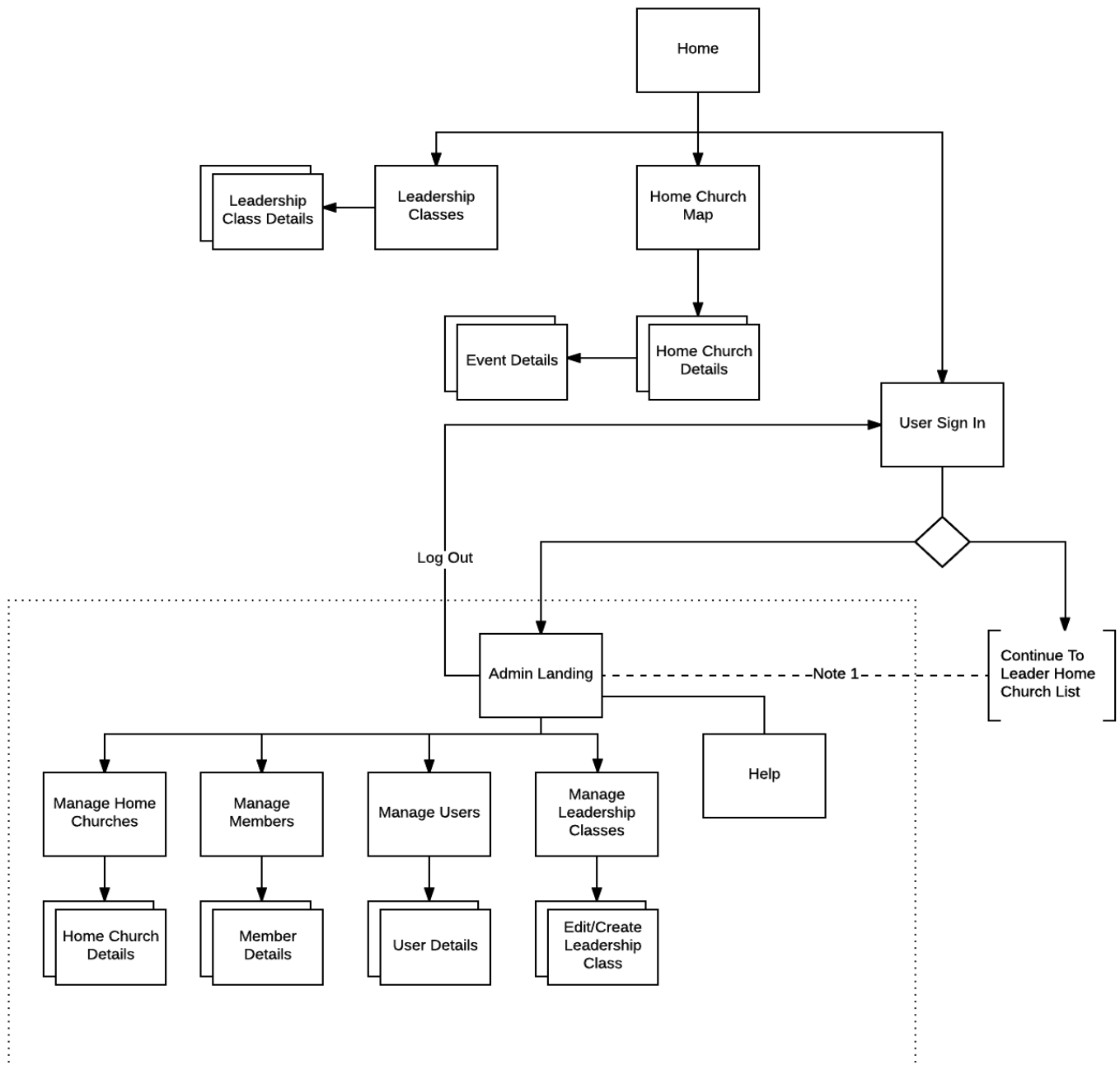
#Update **Church Role Information**

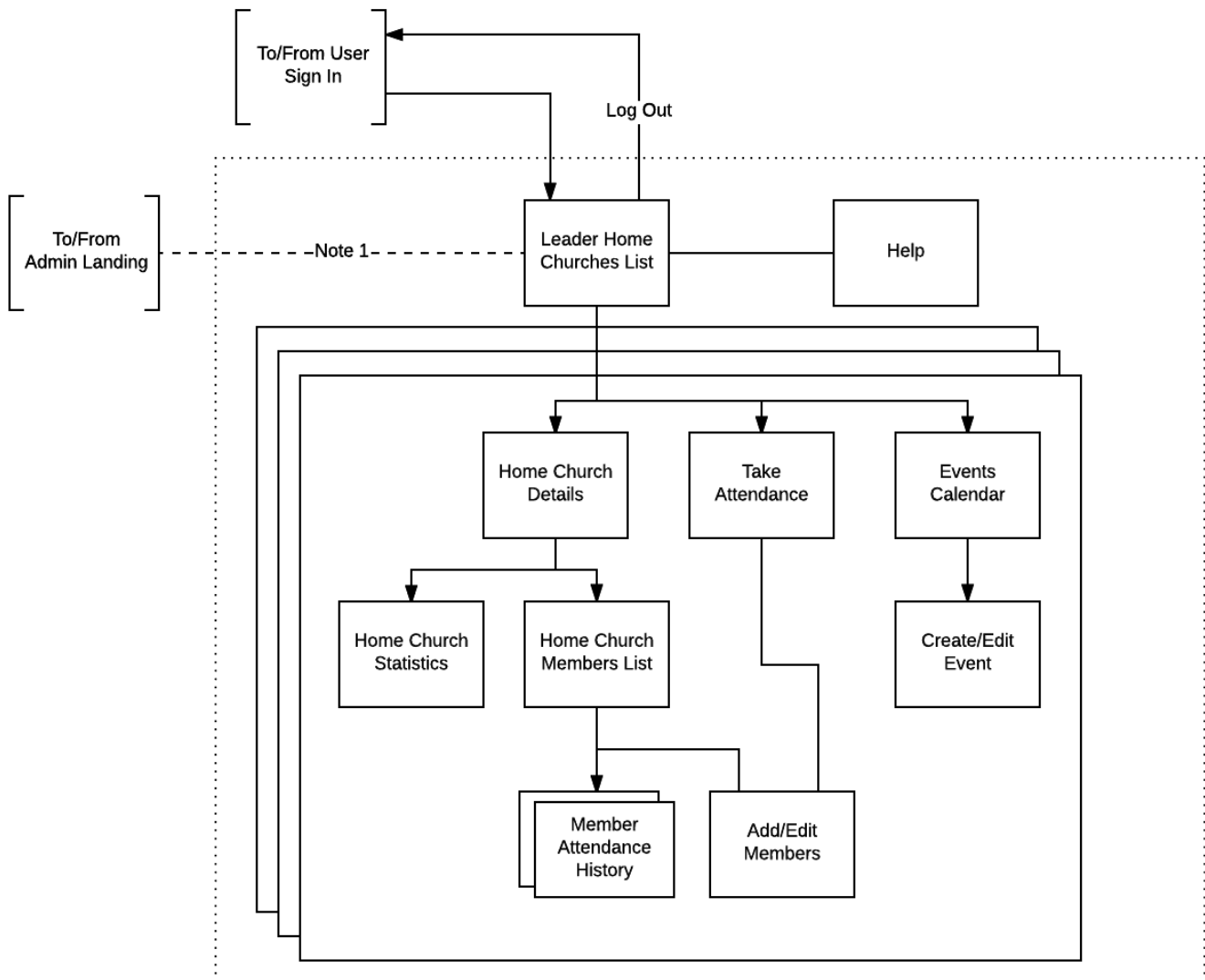
Get updated **Church Role Information** from Administrator

Using **Member ID**, Store updated **Church Role Information** in **Member**

2.5 User Experience (UX) Diagrams

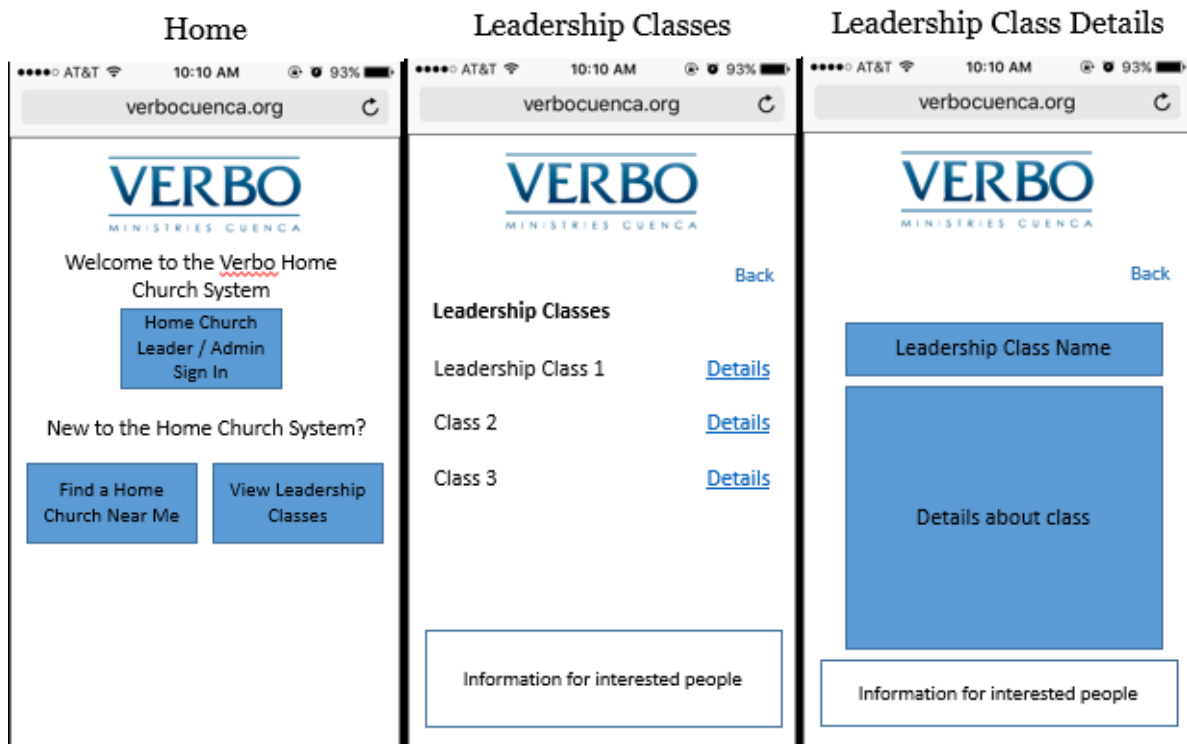
The User Experience Diagrams show the flows of a user interacting with the system. The page names will have corresponding wireframes.





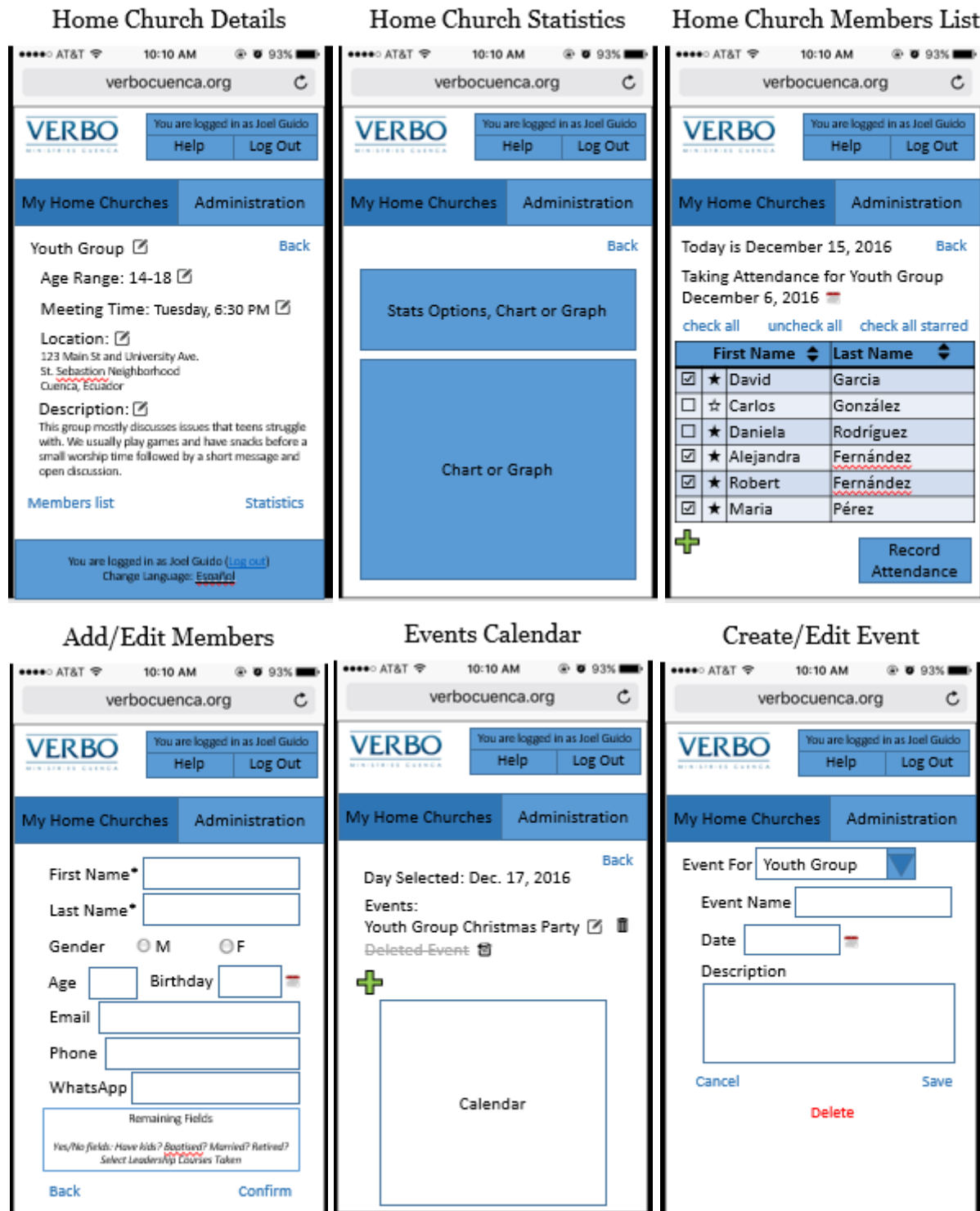
Note 1: If an admin is not assigned as the leader to a Home Church, he may not reach the Leader Home Churches List. If a home church leader is not an admin, he cannot reach the Admin Landing.

2.6 Wireframes



Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu
Team: ISA V
SYS 390 Information Systems Analysis
Dr. Nurkkala and Dr. Stanley





Home Church Members List

verbocuenca.org

VERBO MINISTRIES CUENCA

You are logged in as Joel Guido
[Help](#) [Log Out](#)

My Home Churches Administration

Youth Group Members [Back](#)

First Name	Last Name	Records
David	Garcia	<input type="checkbox"/>
Carlos	González	<input type="checkbox"/>
Daniela	Rodríguez	<input type="checkbox"/>
Alejandra	Fernández	<input type="checkbox"/>
Robert	Fernández	<input type="checkbox"/>
Maria	Pérez	<input type="checkbox"/>

+

Member Attendance History

verbocuenca.org

VERBO MINISTRIES CUENCA

You are logged in as Joel Guido
[Help](#) [Log Out](#)

My Home Churches Administration

David Garcia [Back](#)

Attendance Details

Admin Landing

verbocuenca.org

VERBO MINISTRIES CUENCA

You are logged in as Joel Guido
[Help](#) [Log Out](#)

My Home Churches Administration

Manage Home Churches

Manage Users

Manage Members

Manage Leadership Classes

You are logged in as Joel Guido ([Log out](#))
 Change Language: [Español](#)

Manage Home Churches

verbocuenca.org

VERBO MINISTRIES CUENCA

You are logged in as Joel Guido
[Help](#) [Log Out](#)

My Home Churches Administration

Manage Home Churches [Back](#)

[Add New](#) [Include Archived](#)

Search

ID	Leader	Group Name
1	Joel Guido	Youth Group
2	Leader 2	Group 1
3	Leader 2	Group 2
4	Leader 3	Group 3
5	Leader 3	Group 4
6	Leader 4	Group 5
7	Leader 5	Group 6

Home Church Details

verbocuenca.org

VERBO MINISTRIES CUENCA

You are logged in as Joel Guido
[Help](#) [Log Out](#)

My Home Churches Administration

Create New Home Church [Back](#)

Leader*

Location*

Description

Remaining Optional Fields

[Cancel](#) [Confirm](#)

Manage Users

verbocuenca.org

VERBO MINISTRIES CUENCA

You are logged in as Joel Guido
[Help](#) [Log Out](#)

My Home Churches Administration

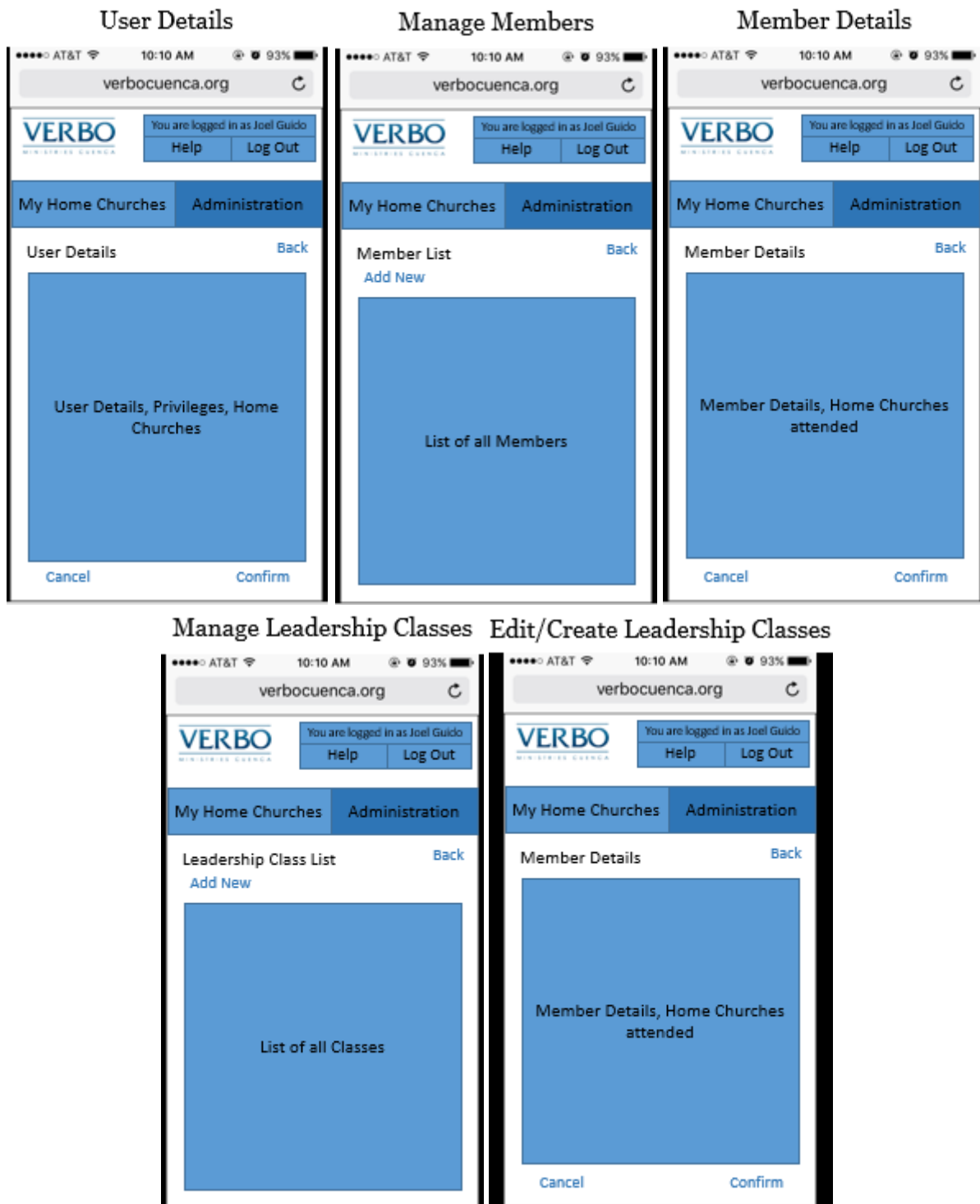
Manage Users [Back](#)

[Add New](#) [Include Archived](#)

Search

ID	Name	L	A
1	Joel Guido	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	User 2	<input type="checkbox"/>	<input checked="" type="checkbox"/>
3	Leader 2	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	Leader 3	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	Leader 3	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	Leader 4	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	Leader 5	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu
Team: ISA V
SYS 390 Information Systems Analysis
Dr. Nurkkala and Dr. Stanley



3 NON-FUNCTIONAL REQUIREMENTS

3.1 User Requirements

1. Usability
 - 1.1. The system shall be user-friendly to accommodate users with little technological experience.
 - 1.1.1. The system shall have a simple and intuitive user interface.
 - 1.2. The system shall be easy to use such that visitors can perform all operations without assistance on their first attempt.
2. Accessibility
 - 2.1. The system shall be culturally accessible for the people of Verbo Church Cuenca.
 - 2.1.1. The system shall have the capability of displaying all text in Spanish.
 - 2.1.2. The system shall store and show street addresses by cross streets as is standard in Ecuador.
 - 2.1.3. The system shall store and show dates in DD/MM/YYYY format as is standard in Ecuador.
 - 2.1.4. The system shall not use any exclusively American terminology, idioms, conventions, etc.
 - 2.2. The system shall be accessible on mobile devices and computers.
 - 2.3. The system shall accommodate users with visual impairments.
 - 2.3.1. The system shall have adjustable text sizes to assist readability.
 - 2.3.2. The system shall differentiate objects on the user interface by means other than (or in addition to) color to accommodate colorblindness.
3. Availability
 - 3.1. The system shall be available for use 24/7 such that users can access and modify data at any given time.
 - 3.2. The system shall have minimal downtime for maintenance.
 - 3.2.1. The system shall only be maintained once every two weeks.
 - 3.2.2. The system shall not be maintained during peak periods of usage – Sundays and Tuesday nights.
 - 3.3. The system shall minimize impact on users during upgrade periods.
 - 3.3.1. The system shall only be upgraded overnight so as to limit usage interruptions.
 - 3.3.2. The system shall not be upgraded more than once every other month.
4. Documentation and Training
 - 4.1. The design team shall provide project documentation at the request of the customer.
 - 4.2. The customer shall be provided documentation (updated user and training manuals when applicable) with every upgrade during the lifetime of the system.
 - 4.3. The design team shall provide the customer with a user manual.
 - 4.4. The design team shall provide the customer with materials for training users on the new system.
 - 4.4.1. The design team shall provide the customer with a training manual.
 - 4.4.2. The design team shall provide the customer with training videos upon request.
 - 4.4.3. The visitor role shall not require any training. All functionality for visitors must be self-explanatory.
 - 4.4.4. The home church leader role shall require one hour-long training session.

4.4.5. The administrator role shall require one half-day training session upon becoming an administrator and sporadic brief training sessions as upgrades are released.

3.2 System Requirements

1. Performance
 - 1.1. The system shall be interactive and responsive.
 - 1.1.1. All pages shall load in less than one second per page.
 - 1.1.2. The system shall give feedback when a user performs an invalid action.
 - 1.1.3. The system shall give feedback when a page is loading.
 - 1.2. The system shall be timely upon performing operations.
 - 1.2.1. The system shall process an entry in less than one second.
 - 1.2.2. The system shall update data in less than one second.
 - 1.2.3. The system shall load attendance statistics in less than one second.
 - 1.3. The system shall be able to accommodate 1,000 simultaneous users without noticeable degradation of service or performance.
2. Capacity
 - 2.1. The system shall be capable of handling five hundred users and expand as needed.
 - 2.2. The system shall be capable of handling two hundred home churches and expand as needed.
 - 2.3. The system shall store both current and historical data.
3. Security
 - 3.1. The system shall restrict access depending on the user role.
 - 3.1.1. A visitor shall only be able to access the home church map (and related information), home church calendar (and related information), and the list of leadership classes (and related information). A visitor shall not be allowed to log in to the system.
 - 3.1.2. A home church leader shall be able to access all information a visitor can access. In addition, a home church leader shall be able to create, edit, and archive attendance records; create, edit, and archive members; create, edit, and archive home church events in the calendar; update home church information. A home church leader shall only be allowed to perform these actions on a church he or she is the leader of.
 - 3.1.3. An administrator shall be able to access all functionality of the system.
 - 3.2. The system shall manage access through the database.
 - 3.3. The system shall track access for auditing purposes.
 - 3.4. The system shall allow the administrator to bring the application offline to prevent unauthorized and malicious access.
 - 3.5. The system shall not provide a user account for visitors and members, so that privacy and confidentiality of data is maintained.
 - 3.5.1. An administrator shall grant access to confidential data by creating a user account for only qualified members.
 - 3.5.2. The system shall provide an administrator with an audit trail in order to document unauthorized access.
4. Longevity

Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu

Team: ISA V

SYS 390 Information Systems Analysis

Dr. Nurkkala and Dr. Stanley

- 4.1. The system shall be able to run in its current state for at least ten years.
- 4.2. The system shall run continuous backups to insure that data is preserved after the system goes out of service.

4 GLOSSARY

Home Church	A group of members, along with at least one home church leader, who meet at a determined location on a Tuesday.
Visitor	Any web user without a record in the database.
Member	An attendee of Verbo Church, but not necessarily a home church.
Home Church Leader	A leader of at least one home church, who has a user account, but with limited permissions.
Administrator	A user with unlimited permissions.
User	Encompassing term that refers to both home church leaders and administrators.
Collaborating System	Any external system that is used in order to provide full functionality.
Other Software	Any external software that is used in order to provide full functionality.
System Response Table (SRT)	Table that captures the result of event-driven modeling.
Logical Data Model	Model that expresses entities and their attributes, and relationships between entities in the enterprise.
Entity Relationship Diagram (ERD)	The most widely used logical data model.
Data Dictionary	Documentation that expresses what entities are present in the system, as well as their attributes and relationships with other entities.
Logical Process Model	Model that organizes and documents structure, flow of data, logic, policy, and procedures to implement.
Data Description	Detailed content of data flow between external agents and data stores.
Data Flow Diagram	A simple restatement of a system response table entry that captures the essence of an event.
Process Specification	The middle ground between describing the parts of the system in plain words and writing code.

Cassie Long, John Luscombe, Evan Miyakawa, Edric Yu, Joyce Yu

Team: ISA V

SYS 390 Information Systems Analysis

Dr. Nurkkala and Dr. Stanley

User Experience (UX) Diagram	Diagram that shows how each user interacts with a system to accomplish specific tasks that contribute business value.
Wireframe	Bare-bones representation of the user experience and interaction design that includes only essential user interface elements, and that uses placeholders for everything else.