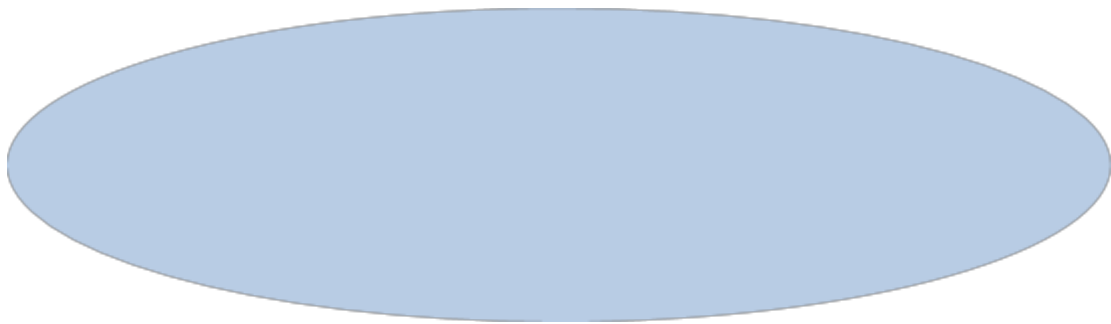


# Acceptance Test

Requested by: Ms. Mary Partridge-Brown  
Ms. Roberta Sandler  
Co-Directors  
Grassroot Givers' Community Store



# SMARK Solutions

Be intuitive. Be efficient. Be SMARK.

Prepared by: Salvatore Baisley - Database Manager  
Mary Ritchie - Webmaster  
Anna Grant - Programmer  
Ryan Martin – Team Lead

April 20, 2015

S.W.I.F.T. (Simple Web Inventory for Tracking)  
Acceptance Test

# Contents

1. Product Overview and Summary	5
2. Use Case Narratives	6
2.1 Volunteer	6
2.2 Director	7
3. UML Diagrams	8
3.1 Use Case Diagram	8
3.1.1 Use Case Legend	8
3.1.2 Use Case Diagram	9
3.2 Deployment Diagram	10
3.2.1 Deployment Legend	10
3.2.2 Deployment Diagram	11
3.3 Activity Diagrams	12
3.3.1 Activity Diagram Legend	12
3.3.2 Activity Diagram: Login	13
3.3.3 Activity Diagram: Add Customer	14
3.3.4 Activity Diagram: Search Customer	15
3.3.5 Activity Diagram: Check Out	16
3.3.6 Activity Diagram: Record Donation	17
3.3.7 Activity Diagram: Edit Customer/Donor	18
3.3.8 Activity Diagram: Add Volunteer	19

3.4 Website Maps	20
3.4.1 Website Map Legend	20
3.4.2 Website Map: Landing Page	21
3.4.3 Website Map: Volunteer	22
3.4.4 Website Map: Director	23
4. Data Flow Diagrams	24
4.1 Data Flow Legend	24
4.2 Context Diagram	25
4.3 Level 0 Diagram	26
4.4 Level 1 Diagrams	27
4.4.1 Log In	27
4.4.2 Add Customer	28
4.4.3 Search Customer	29
4.4.4 Checkout	30
4.4.5 Record Donation	31
4.4.6 Edit Customer/Donor	32
4.4.7 Add Volunteer	33
5. Functional Requirements Inventory	34
5.1 Volunteer	34
5.2 Director	34
6. External Design Specifications	35
6.1 User Displays & Report Format	35
6.1.1 Login Screen	35
6.1.2 Home Screen	36
6.1.3 Search Customer Screen	37
6.1.4 Add Customer Screen	38
6.1.5 Record Donation Screen	39
6.1.6 Search Donor Screen	40

6.1.7 Manage Volunteers Screen	41
6.1.8 Edit Inventory Screen	42
6.2 Logical Data Dictionary	43
6.3 ER Diagram	47
6.3.1 User Displays & Report Format	47
6.3.2 User Displays & Report Format	48
6.4 Relational Schema	49
6.5 Database Tables	50
7. Testing Plan	53
7.1 Overview and Strategy	53
7.2 Acceptance Test	53
7.3 Unit Tests	53
7.4 Test Cases	54
7.5 System Test	54
7.6 Integration Test	54
7.7 Regression Test	54
8. Development and Production Environments	
55	
8.1 Development Environment	55
8.2 Operating Environment	55
8.3 Maintenance	
55	
9. Deliverables	
56	
9.1 Source Code	56
10. Future Enhancements	57
11. Appendices	58
Appendix A: Test Results	58
Appendix B: Sources of Information	
59	

Appendix C: Glossary of Terms	60
Appendix D: Timeline	61

## 1. Product Overview and Summary

Grassroot Givers is a non-profit organization devoted to bridging the gap between those in need and those seeking to donate. One of the ways in which this is achieved is through the Community Store within their facilities at the GWU Center in Albany, NY. The mission of this store is to create a boutique-like atmosphere so that customers can “shop with dignity”. Co-directors of Grassroot Givers, Mary Partridge-Brown and Roberta Sandler would like to develop an easy to use, web-based application to supplement their everyday functions of the store. S.W.I.F.T. (Simple Web Inventory For Tracking) is a web-based application that will allow Grassroot Givers to track incoming items through the creation of donor and customer profiles, database searching, and receipts.

## 2. Use Case Narratives

### 2.1 Volunteer

The volunteer will login on to S.W.I.F.T. using a unique username and password. The volunteer will have access to a page where the volunteer will choose to enter a new customer, look up the history of a specific customer, check out a specific customer, or take in a donation. If the volunteer wishes to enter a new customer into the database, there will be a form for creating a customer profile. The form will require the volunteer to enter the name of the customer, address of the customer, number of family members in the customer's household including each member's age, and other agencies which the customer is affiliated with, additionally the date that the profile was created will be stored in the customer's profile. The volunteer will have the ability to search the customer records by name and address to view the customer's profile. The volunteer will have the ability to search the inventory to see the quantity of the items that are in high demand. The volunteer will be able to check out customers, during which the customer's history within the past three months will be reviewed. If the customer is eligible to take the items the customer has selected, the volunteer will record and store the items in the customer's profile along with the date of the transaction, and the name of the volunteer doing the checkout. If the volunteer is taking a donation, the volunteer will record the number of bags and boxes being donated and the contents of the packages. The customer will tell the volunteer whether or not the customer desires a receipt and if so, a form will be filled by the volunteer indicating the items donated and their value. Regardless of whether the customer desires a receipt, the number of boxes and bags will still be recorded. Multiple volunteers will be able to be logged in at once.

## 2.2 Director

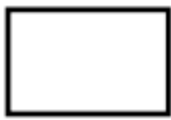
The director will login on to S.W.I.F.T. using a username and password specific to being a director. The director will have access to a page where the director will choose to either enter a new customer, look up the history of a specific customer, check out a specific customer, editing and deleting customer information, adding a new volunteer, or take in a donation. If the director wishes to enter a new customer into the database, there will be a form for creating a customer profile. The form will require the director to enter the name of the customer, address of the customer, number of family members in the customer's household including each member's age, and other agencies which the customer is affiliated with, additionally the date that the profile was created will be stored in the customer's profile. The director will have the ability to search the customer records by name and address to view, edit, or delete the customer's profile. The director will have the ability to search the inventory to see the quantity of the items that are in high demand. The director will be able to check out customers, during which the customer's history within the past three months will be reviewed. If the customer is eligible to take the items the customer has selected, the director will record and store the items in the customer's profile along with the date of the transaction, and the name of the director doing the checkout. If the director is creating a new volunteer account, the director will submit a form with the volunteer's information to S.W.I.F.T.. If the director is taking a donation, the director will record the number of bags and boxes being donated and the contents of the packages. The customer will tell the director whether or not the customer desires a receipt and if so, a form will be filled by the director indicating the items donated and their value. Regardless of whether the customer desires a receipt, the number of boxes and bags will still be recorded. Multiple directors will be able to be logged in at once.



## 3. UML Diagrams

### 3.1 Use Case Diagram

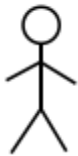
#### 3.1.1 Use Case Legend



System Boundary: where interactions between both use cases inside the system and actors outside the system are shown



Use Case: the activities that actors interact with inside the system



Actor: human or non human users that interact with the system.



Participation line: lines that connect use cases and actors to show what actors participate in



Extends: used to represent items

that may be included in a

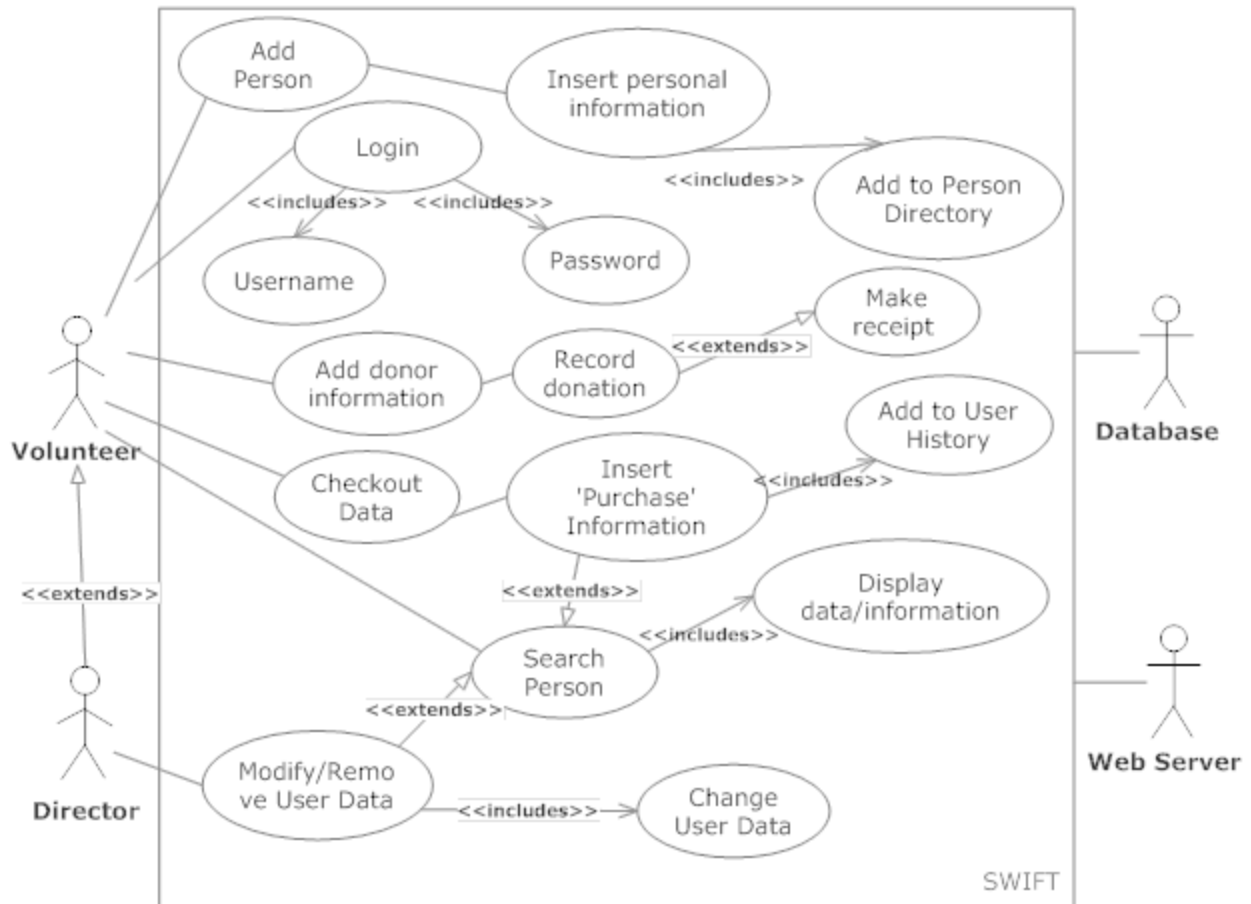
use case



Includes: used to represent items that must be included in a use case



### 3.1.2 UML Use Case Diagram



## 3.2 Deployment Diagram

### 3.2.1 Deployment Legend

<HTTP>

Hypertext Transfer Protocol - application protocol used in distributing, collaborating, and hypermedia information systems.

<SCP>

Secure Copy - A way to securely transfer computer files between a local host and a remote host.

<ODBC>

Open Database Connectivity - Standard programming language API for DBMS.

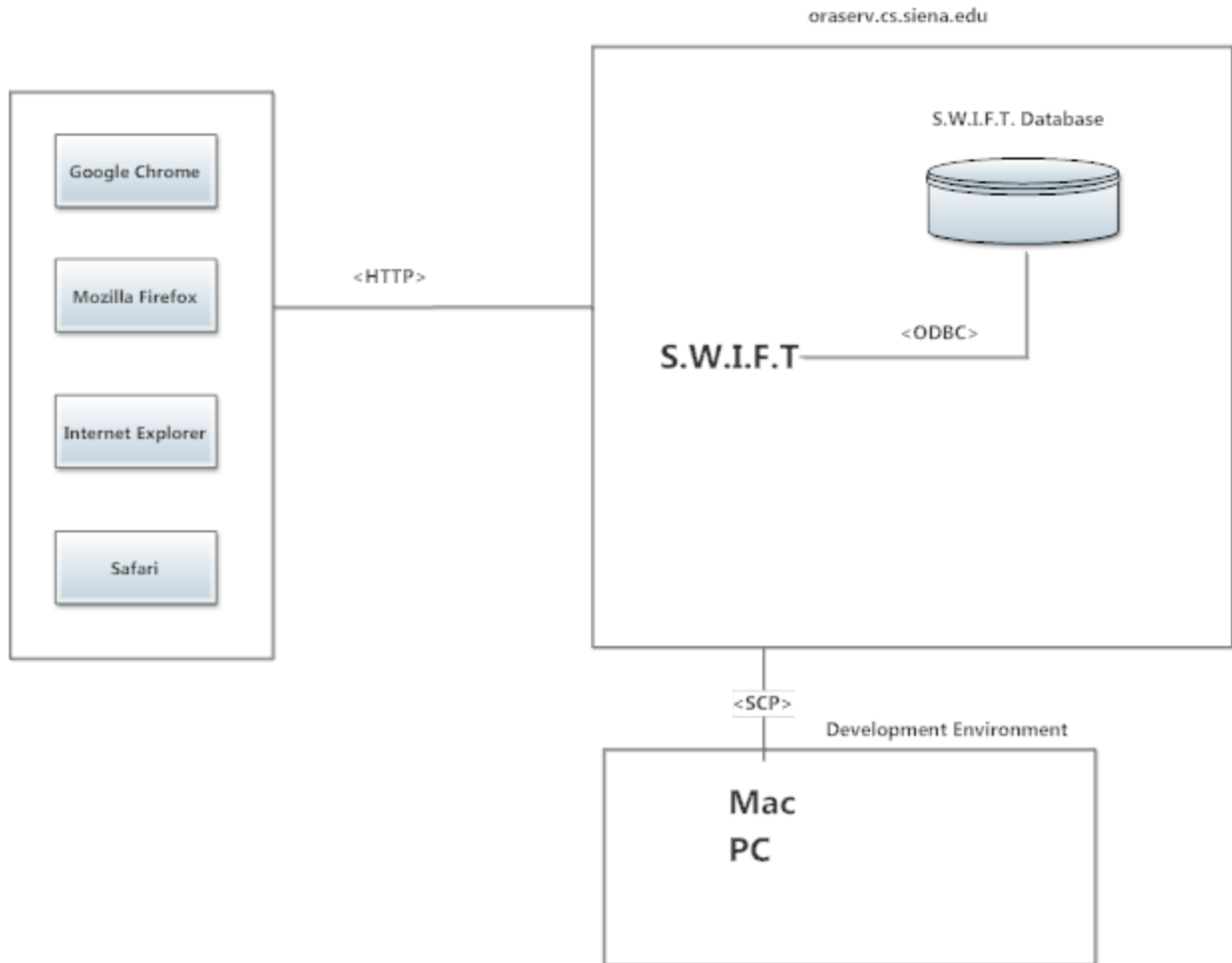


System Boundary - Represents the divide between the inside of the system where the interactions occur and the outside.




Connection - Represents a relation between system boundaries.


### 3.2.2 Deployment Diagram

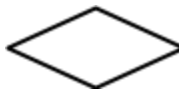



### 3.3 Activity Diagrams

#### 3.3.1 Activity Diagram Legend

 Initial Node - The starting point for the movement through the activity.

 Final Node - The ending point, where the process is over.

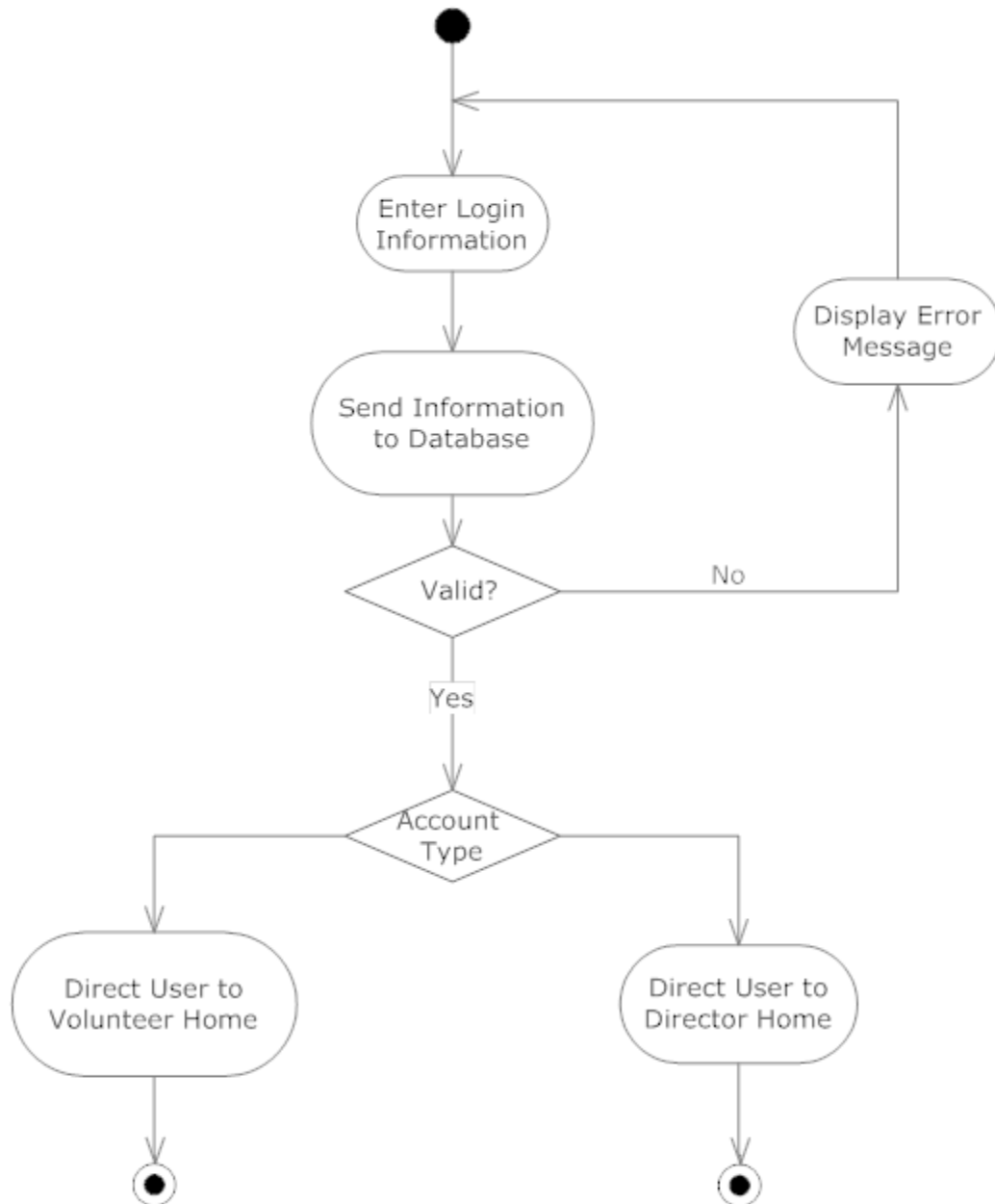
 Decision Node - Represents a branching in the flow of the activity. Each branch represents a unique answer.

 Activity Node - Describes the next step completed or that needs to be completed by the system to continue.

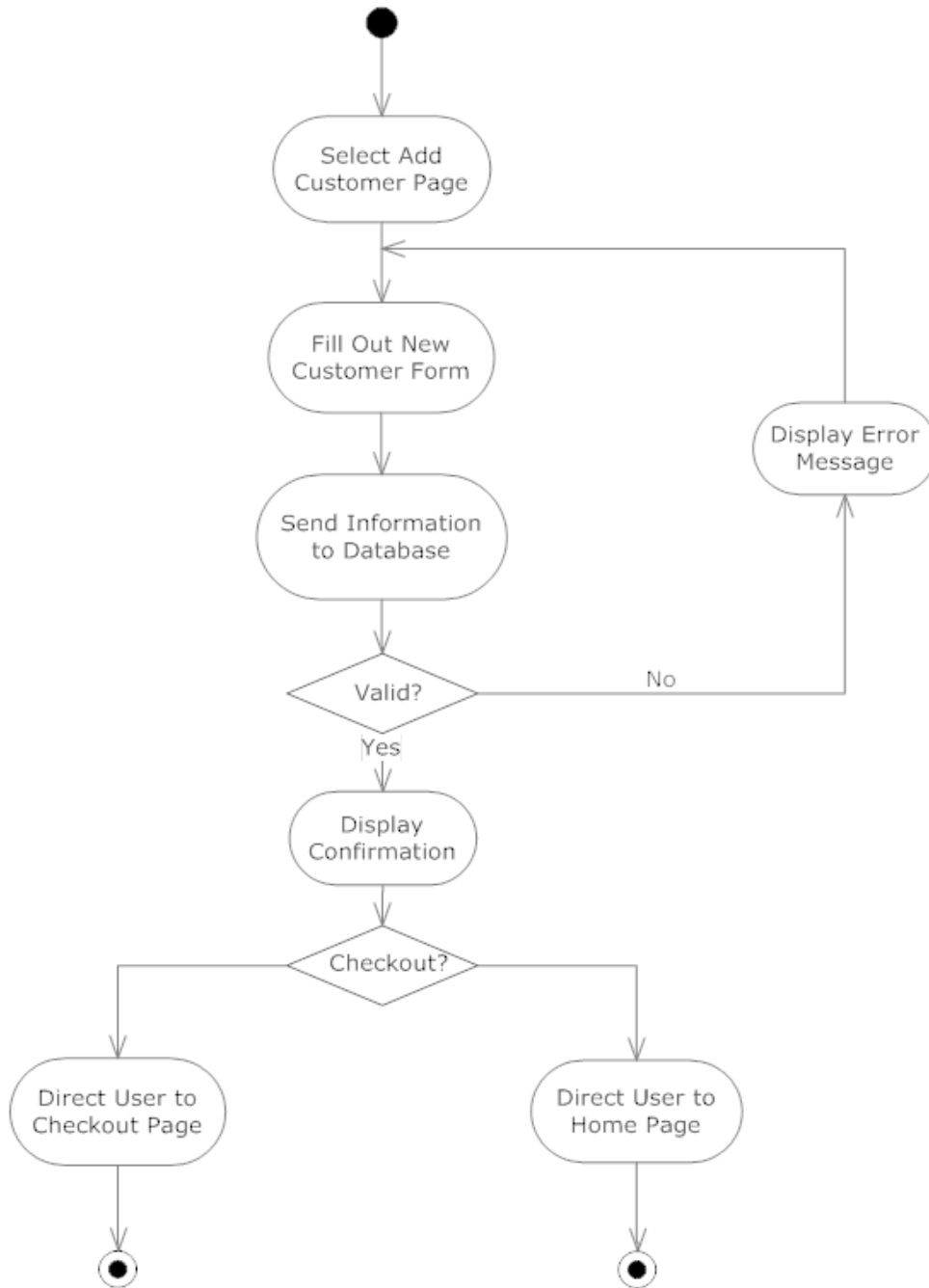
 Split/Join - Separates or joins several different flows.

 Flow - Demonstrates the movement of the activity.

## 3.3.2 Activity Diagram: Login

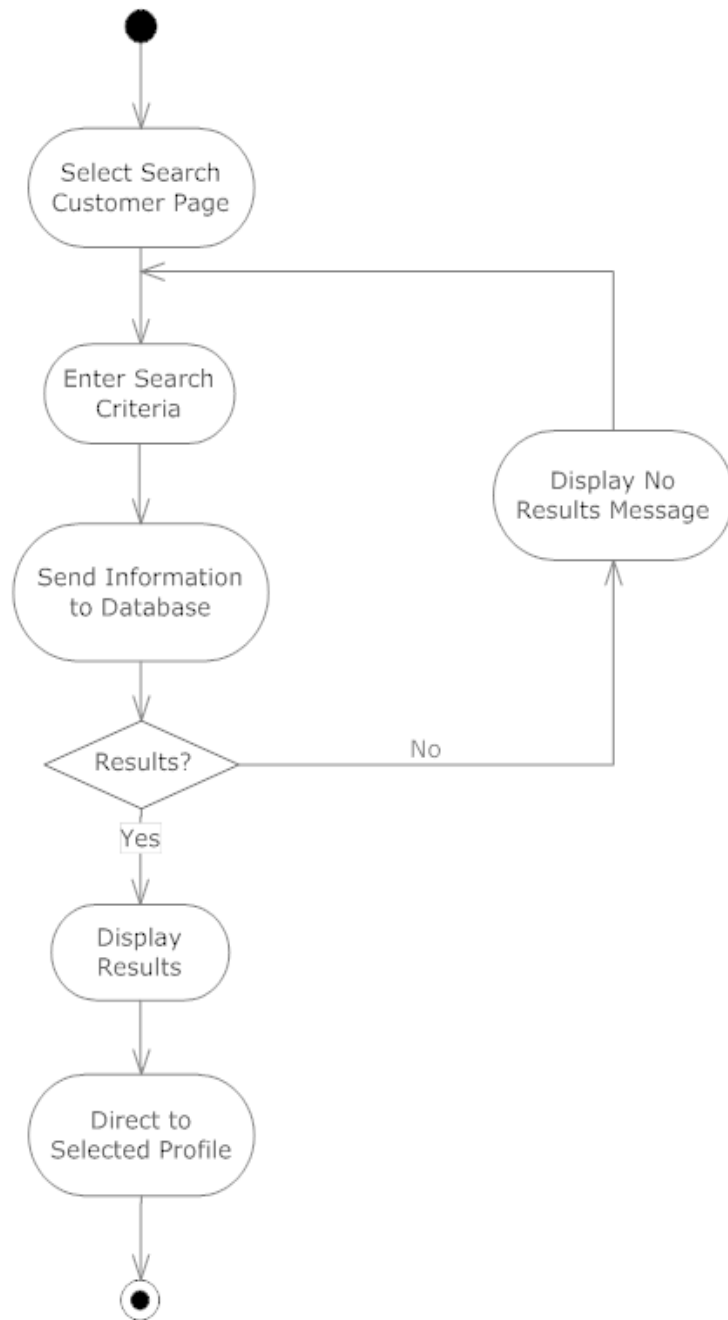


## 3.3.3 Activity Diagram: Add Customer

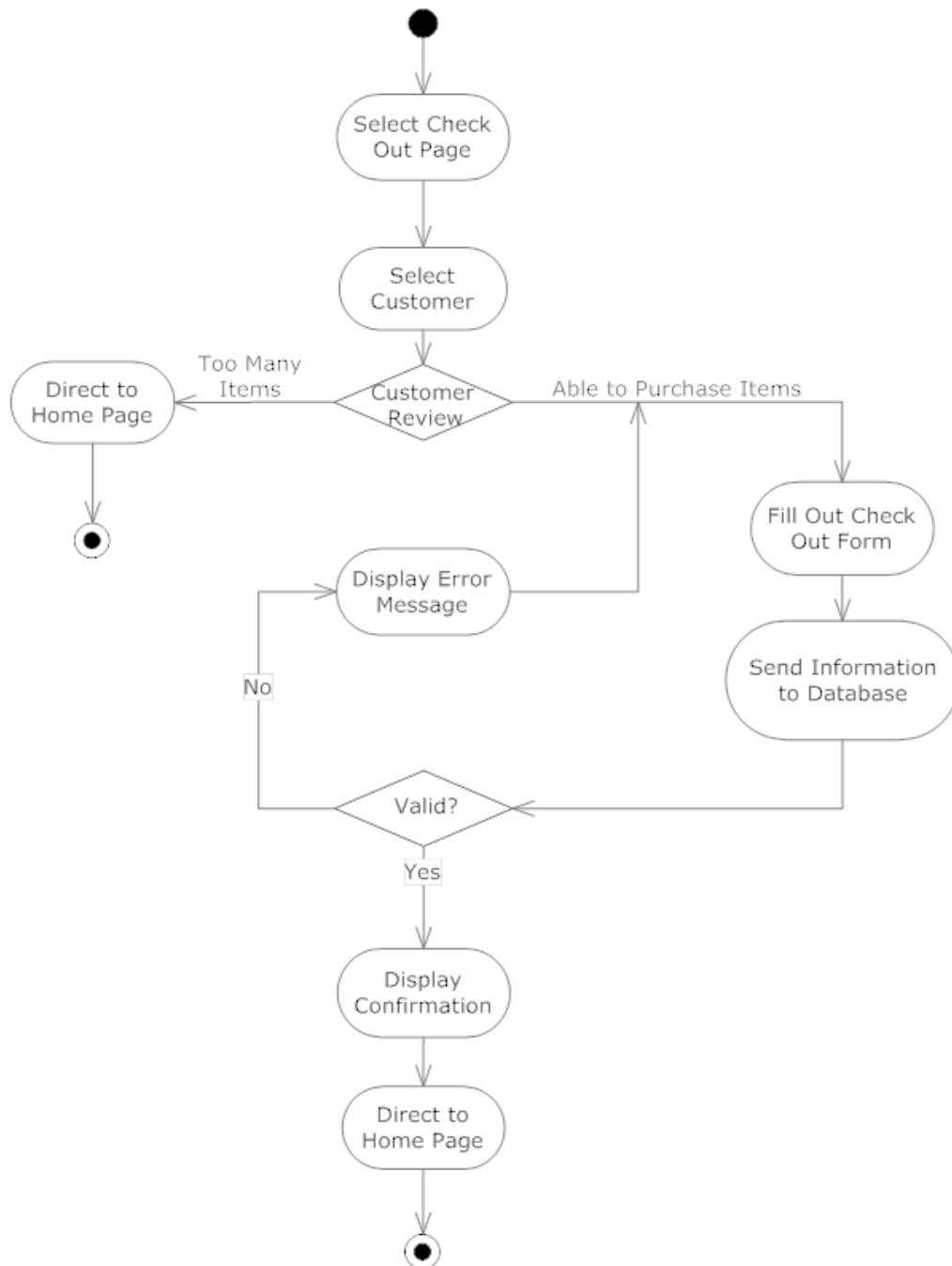


3.3.4 Activity Diagram: Search Customer

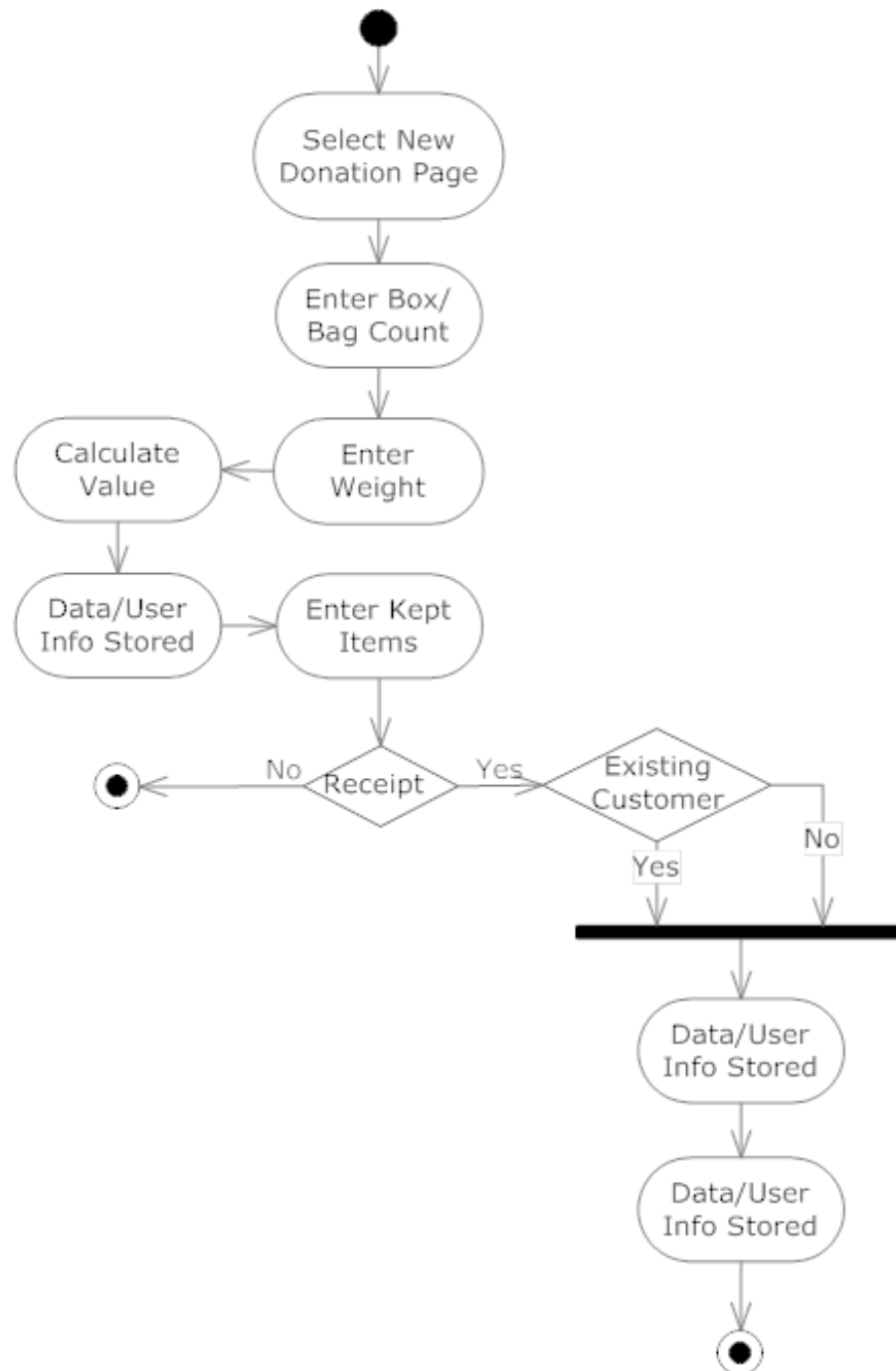




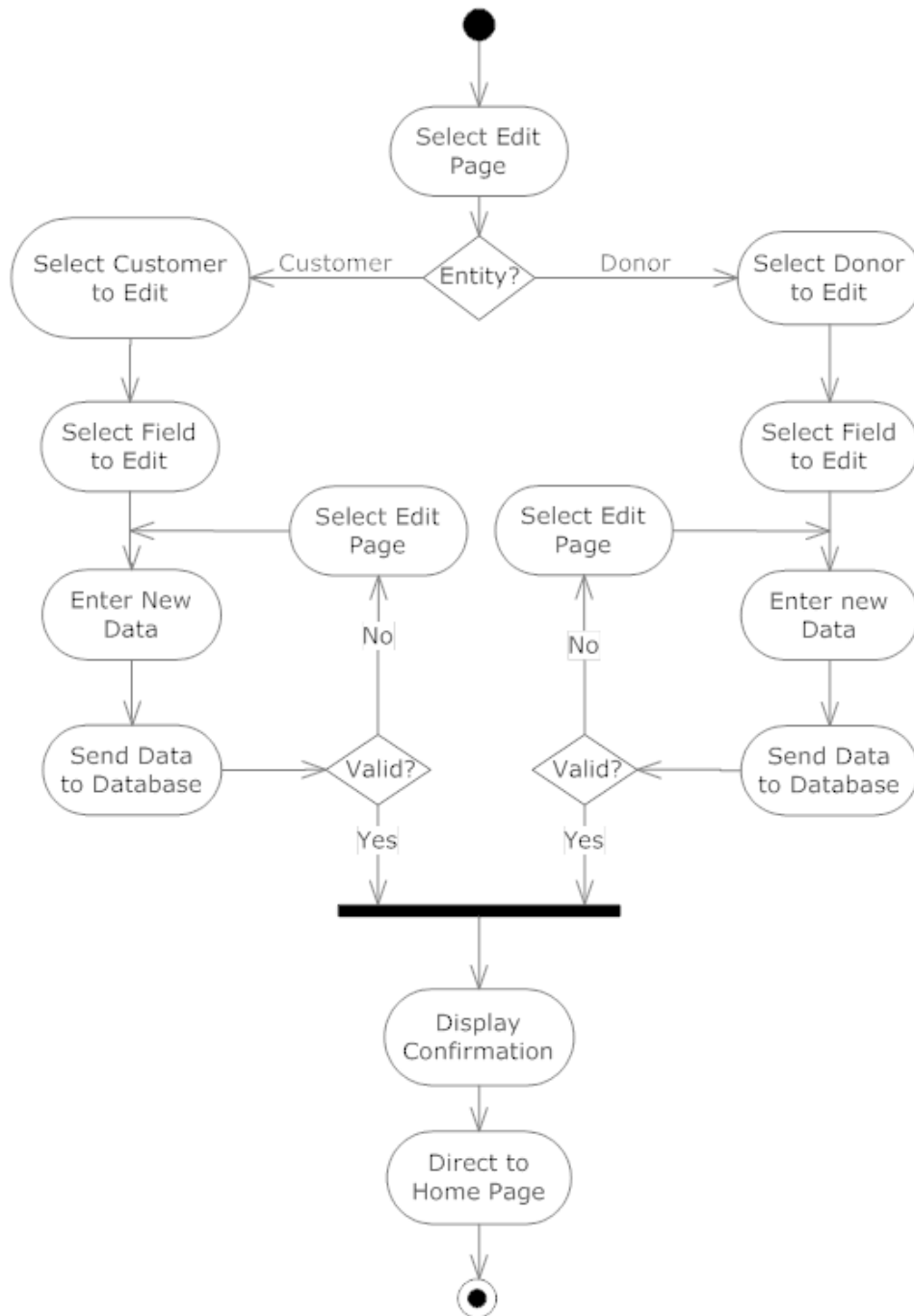
## 3.3.5 Activity Diagram: Check Out



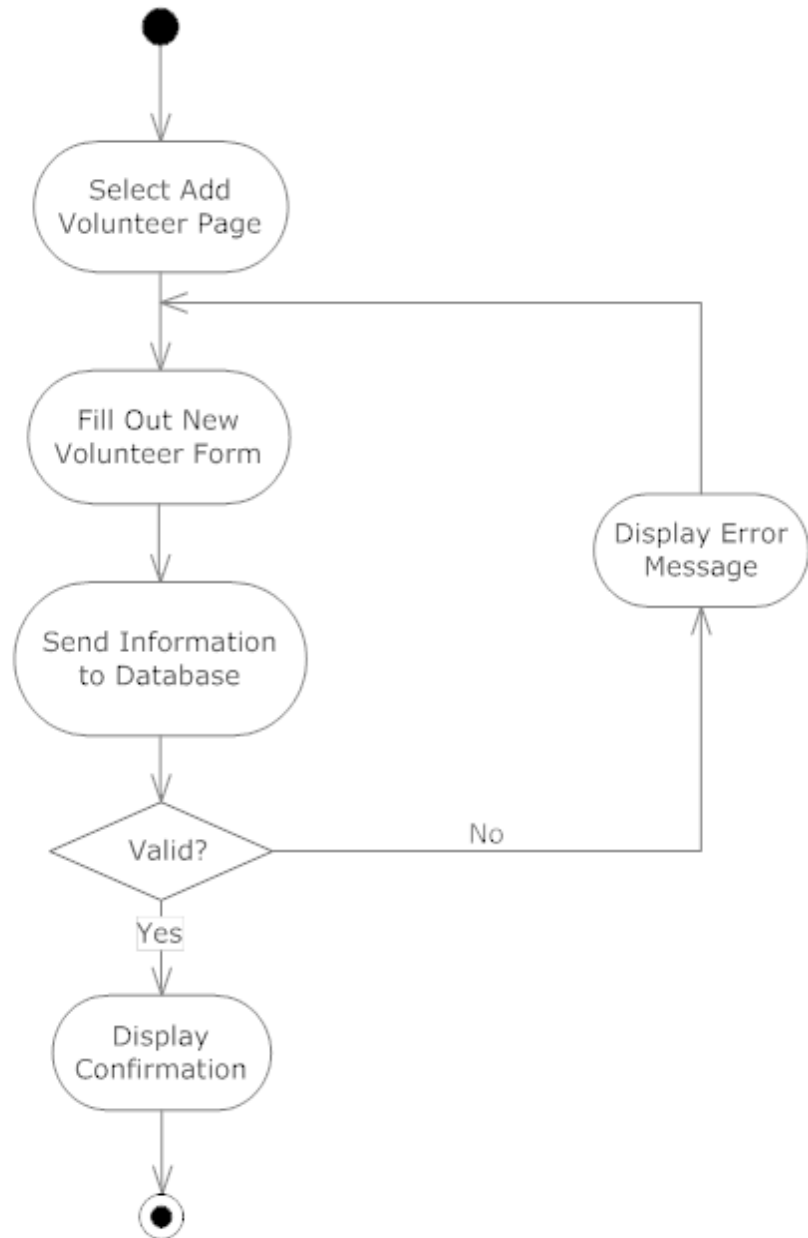
## 3.3.6 Activity Diagram: Record Donation



3.3.7 Activity Diagram: Edit Customer/Donor



3.3.8 Activity Diagram: Add Volunteer



### 3.4 Website Maps

### 3.4.1 Website Map Legend



Web Page - Represents a web page in the system.



Page Redirect - Forced direct to another page.

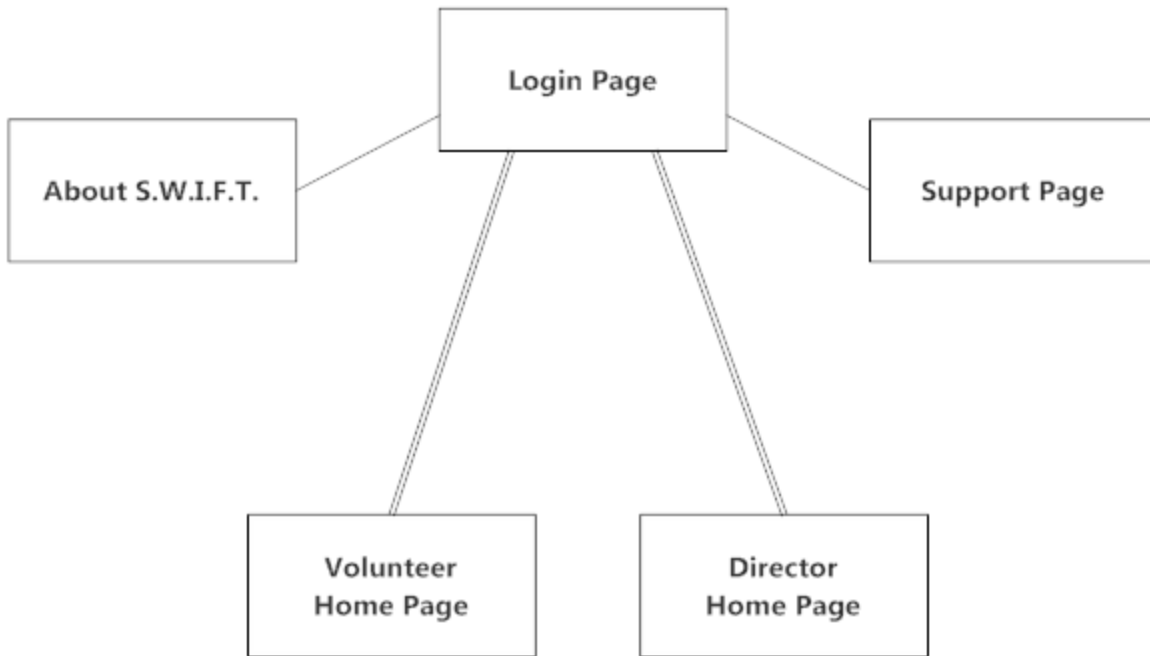


Link - Represents the ability to access one page from the other.

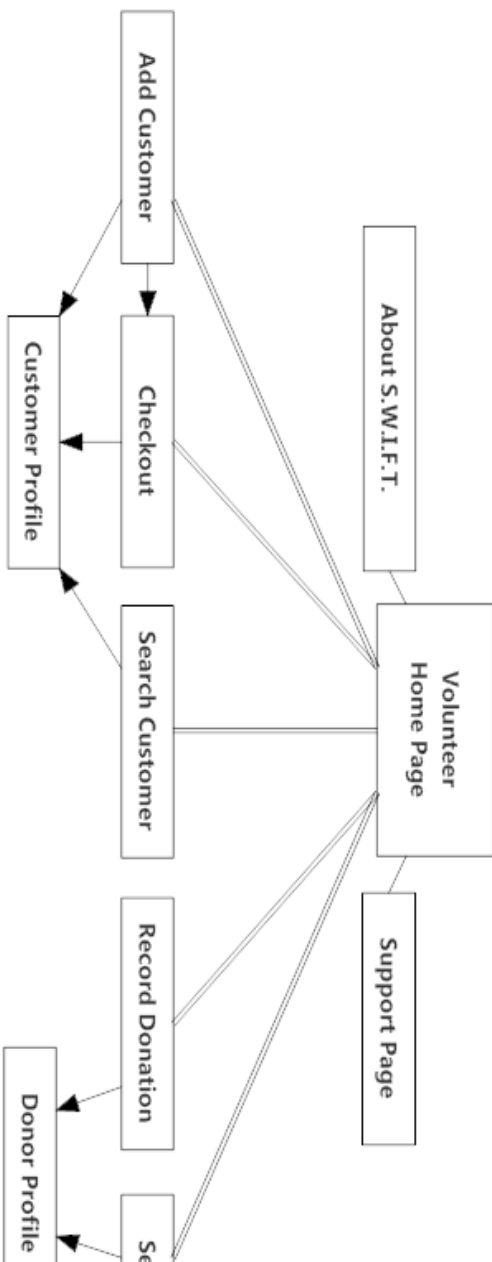


Double Link - Ability to go back and forth between the two connected pages.

### 3.4.2 Website Map: Landing Page

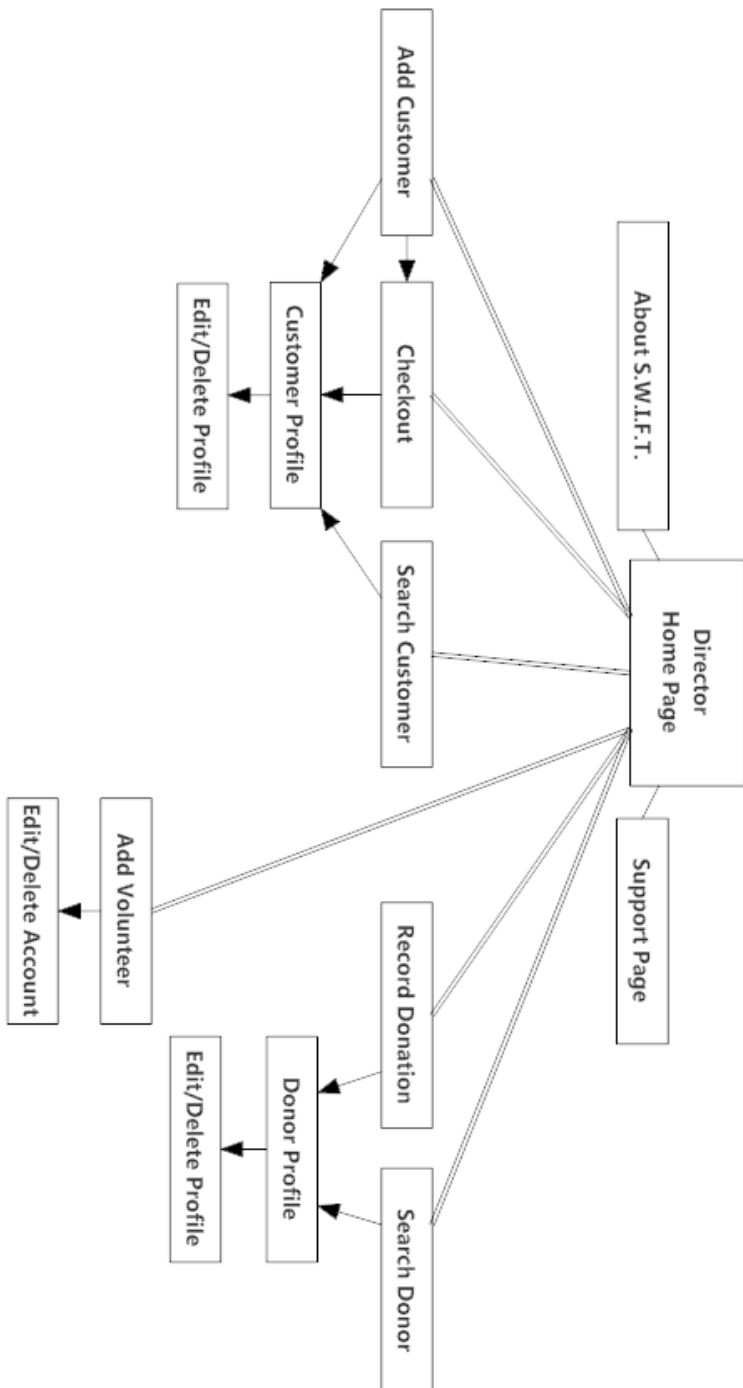


### 3.4.3 Website Map: Volunteer



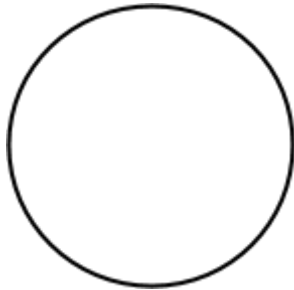


### 3.4.4 Website Map: Director



## 4. Data Flow Diagrams

### 4.1 Data Flow Legend



Process - Functions which are able to receive, modify, and output data.



Entity - Contributes data to and receives data from the system.



Data Store - A place where data is stored permanently or temporarily.

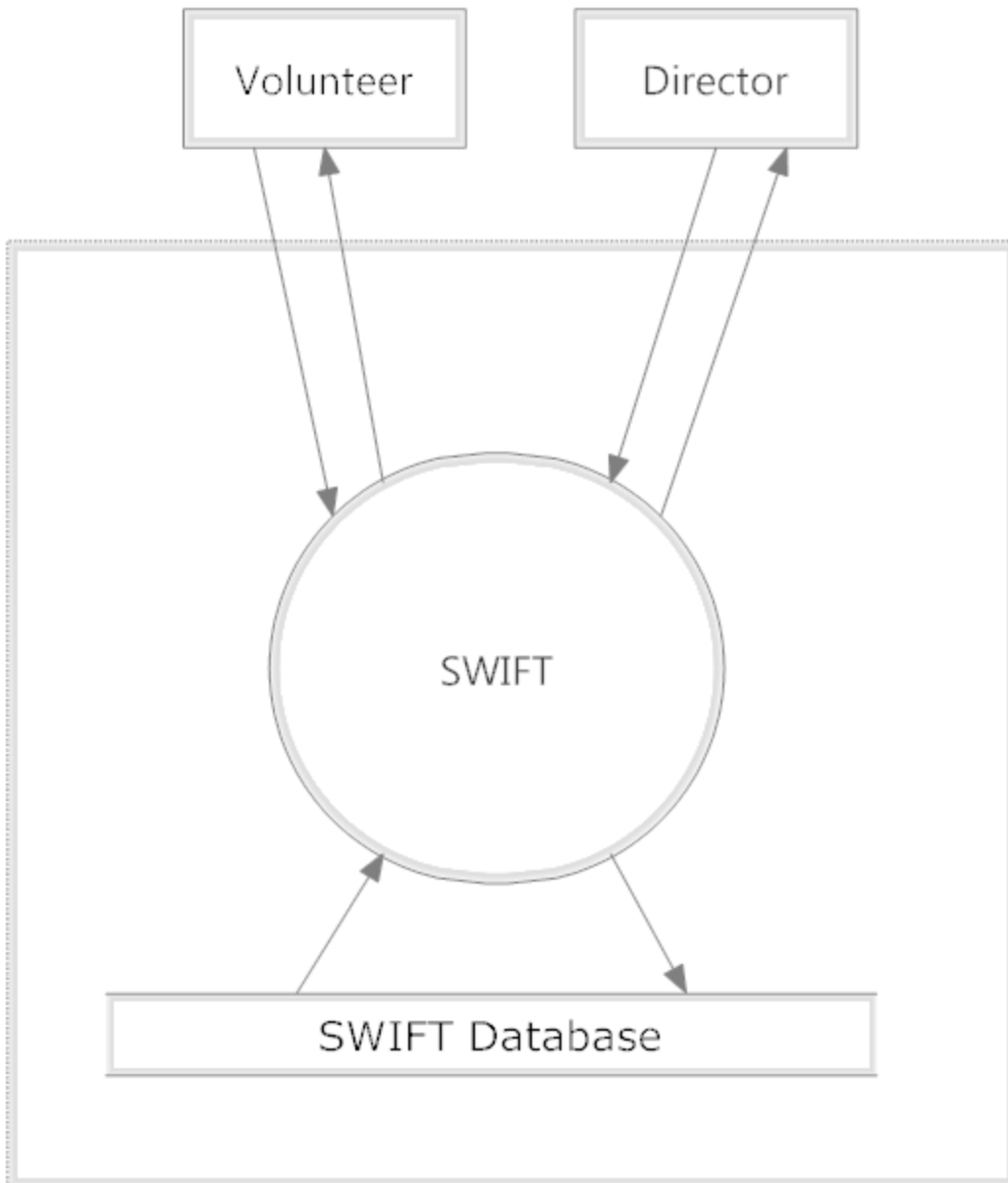


Data Flow - Represents the movement and direction of data.

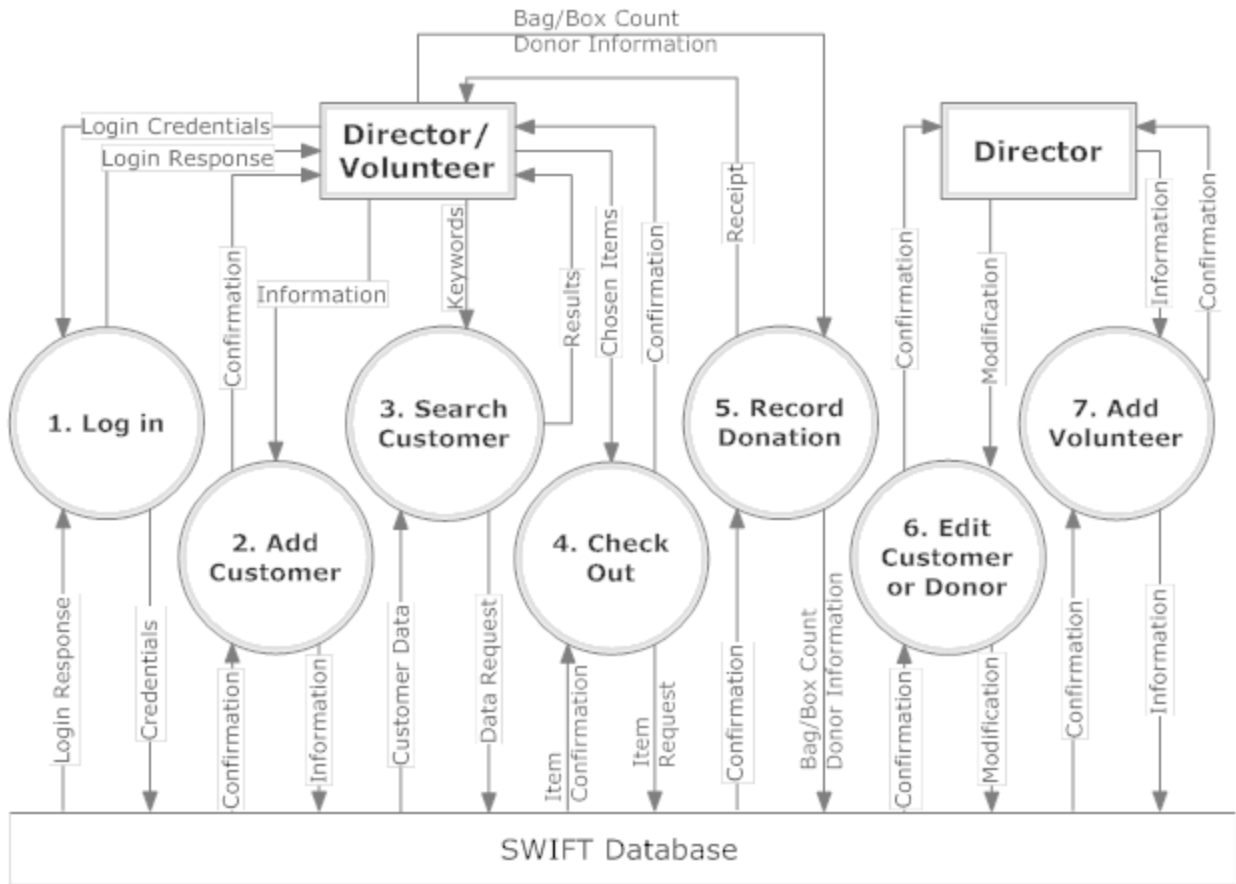


System Boundary - represents the boundary showing the data stores and the system processes inside the system.

## 4.2 Context Diagram

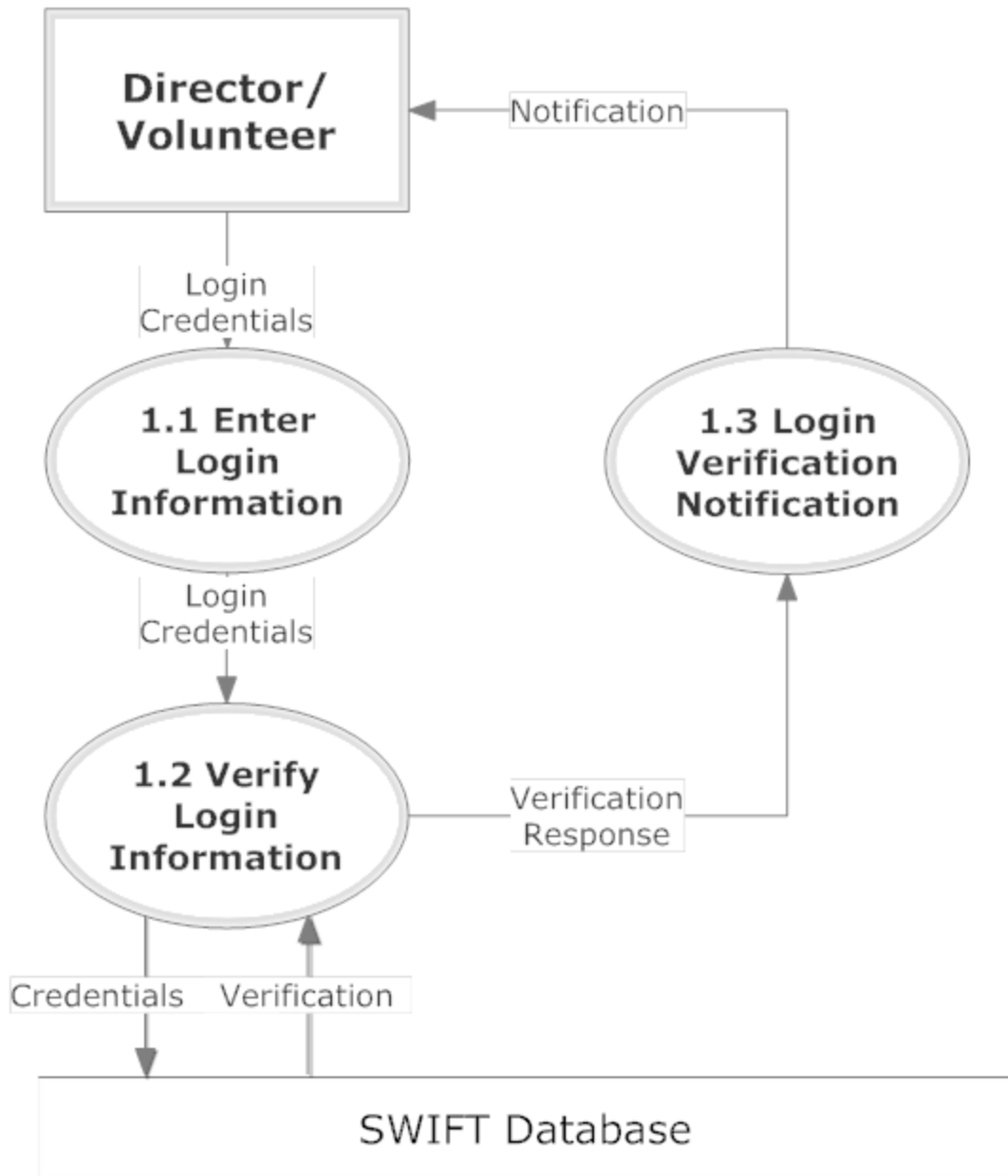


### 4.3 Level 0 Diagram

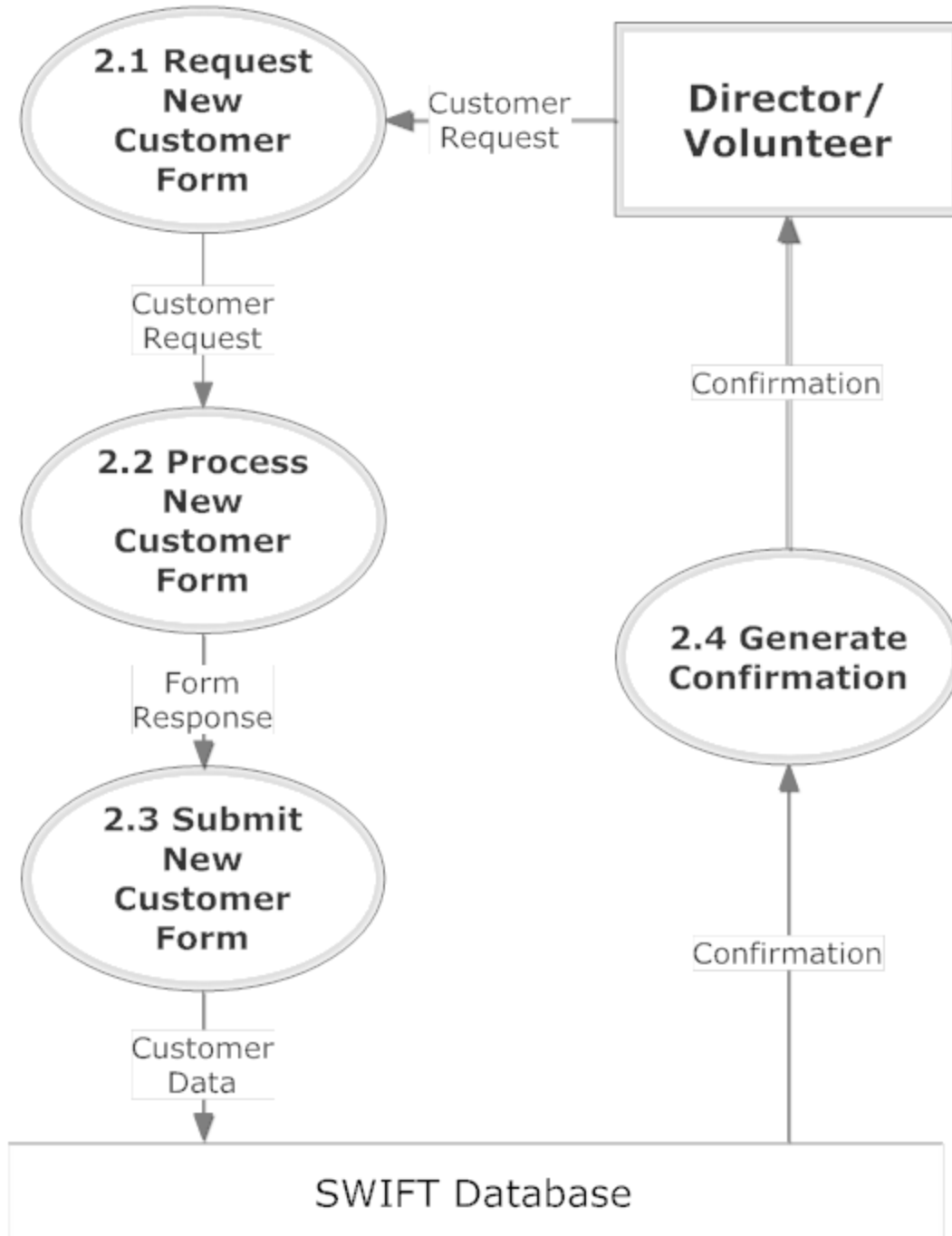


#### 4.4 Level 1 Diagrams

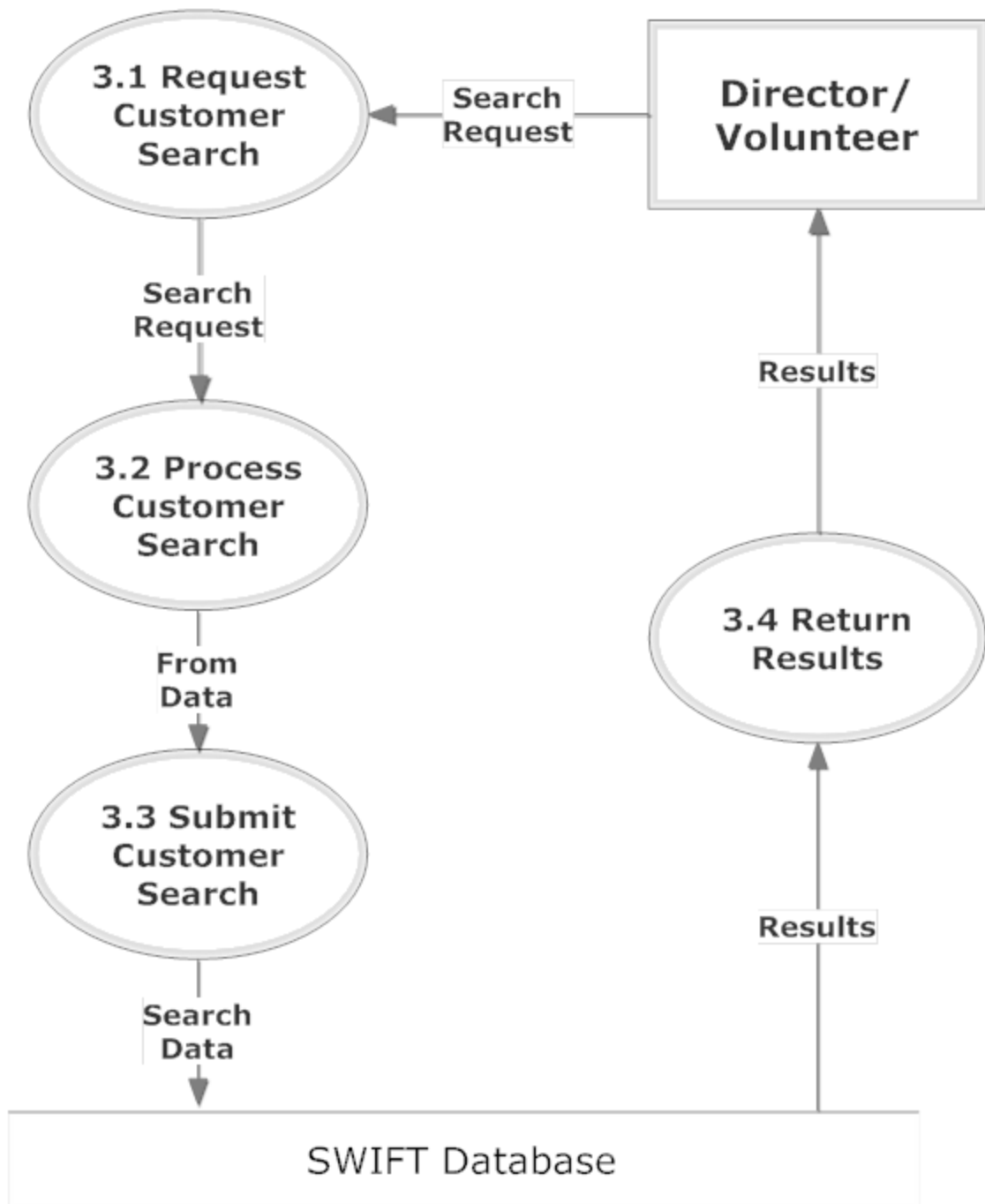
##### 4.4.1 Level 1: Log In



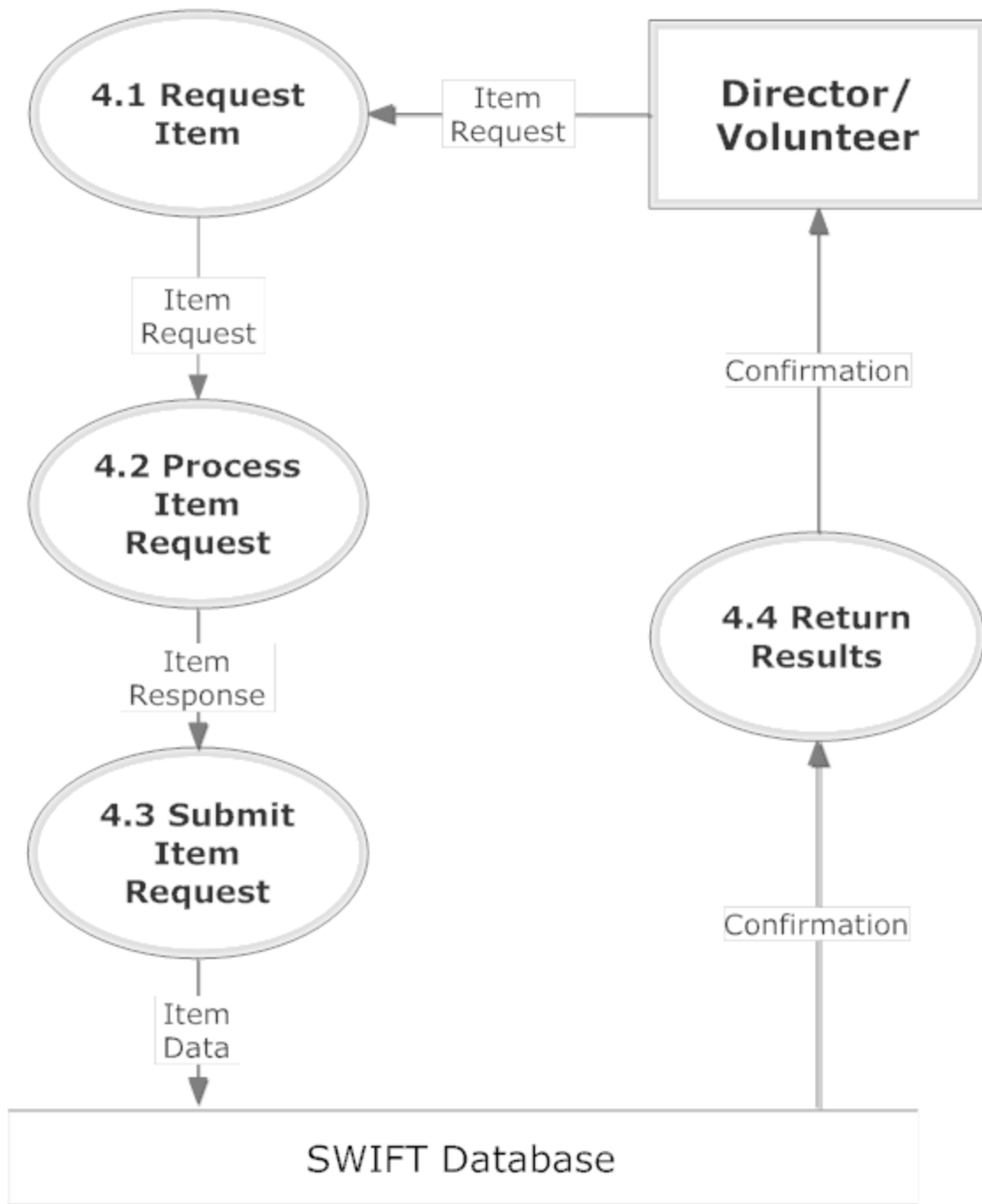
#### 4.4.2 Level 1: Add Customer



#### 4.4.3 Level 1: Search Customer

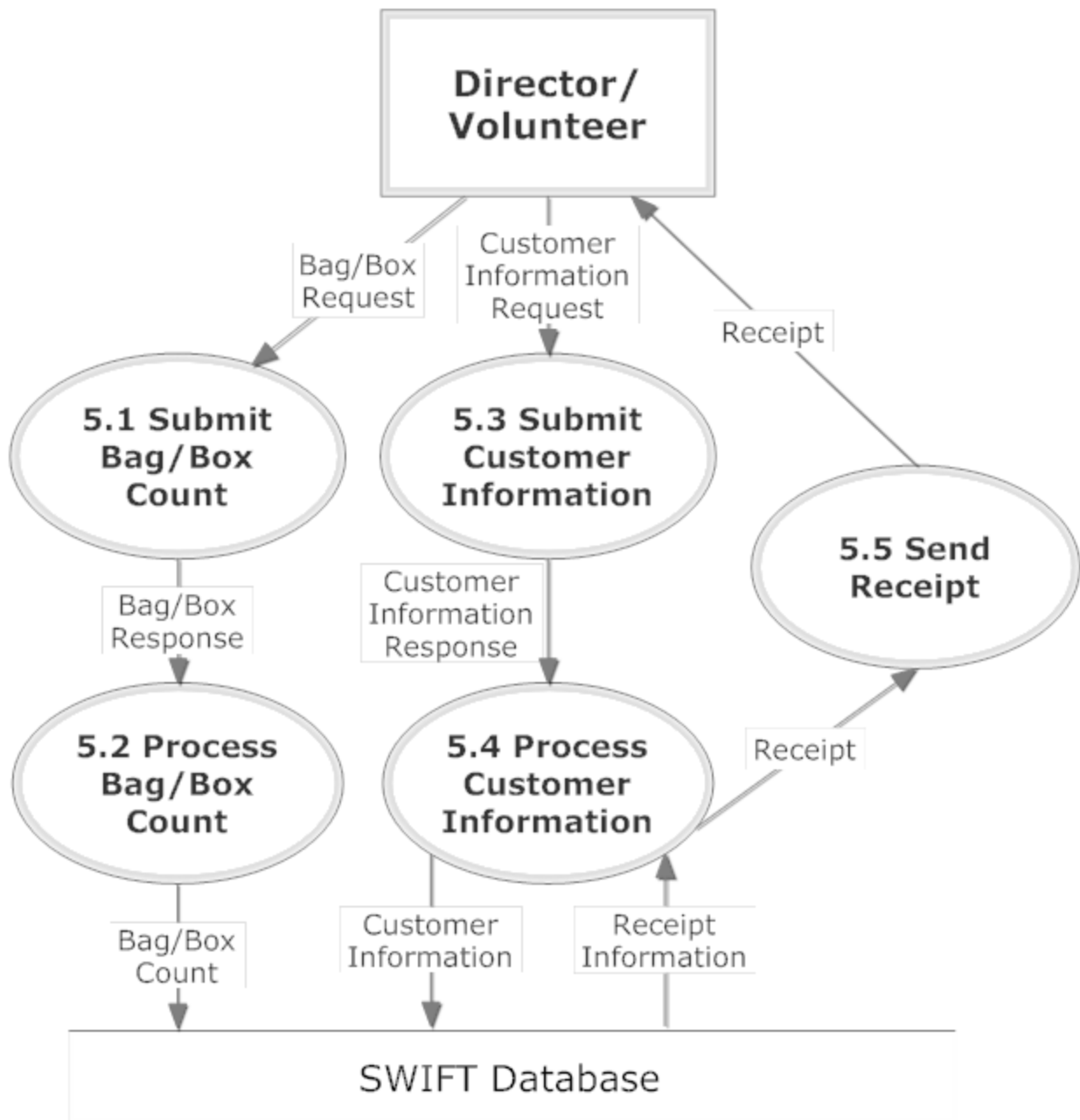


4.4.4 Level 1: Checkout

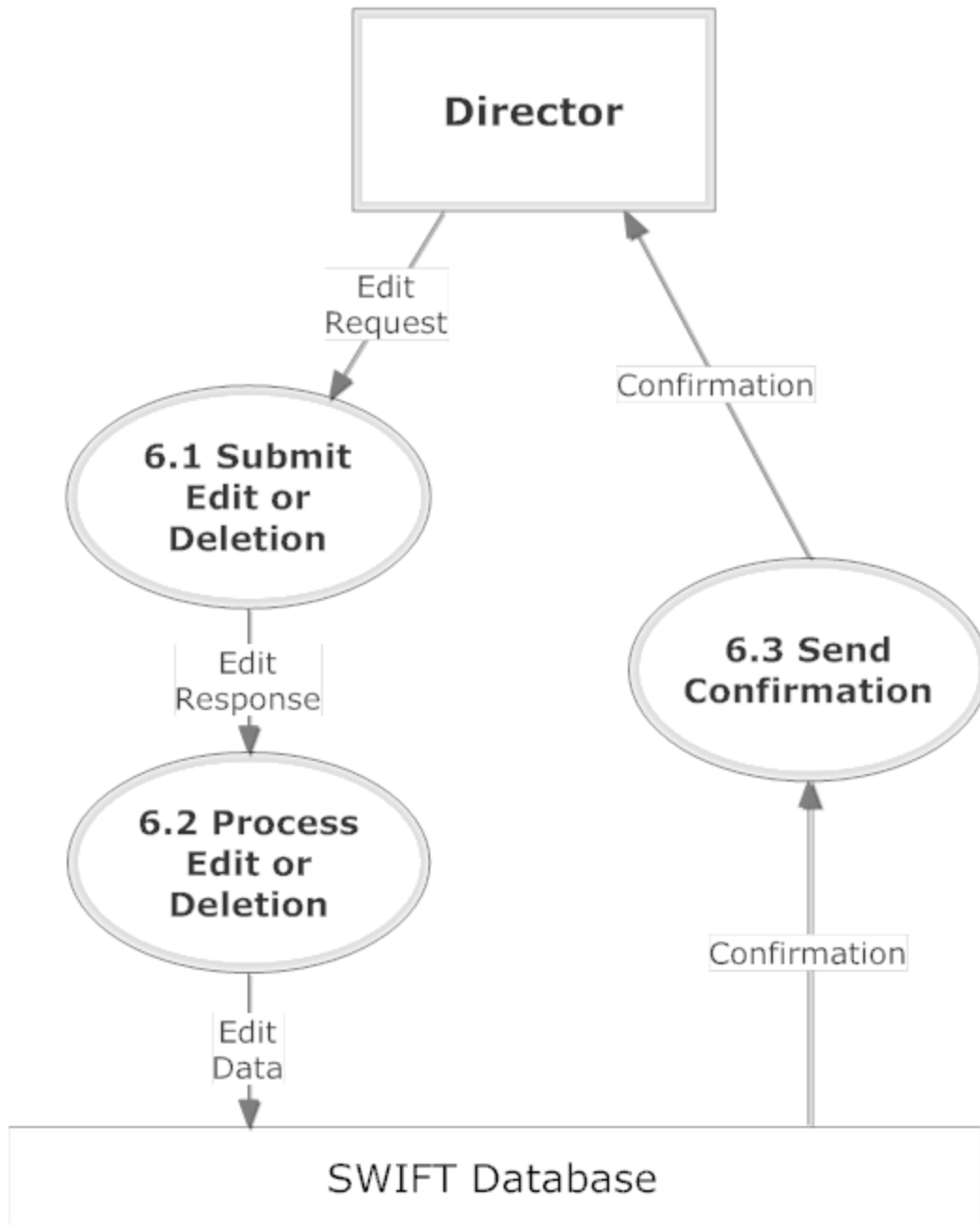


4.4.5 Level 1: Record Donation

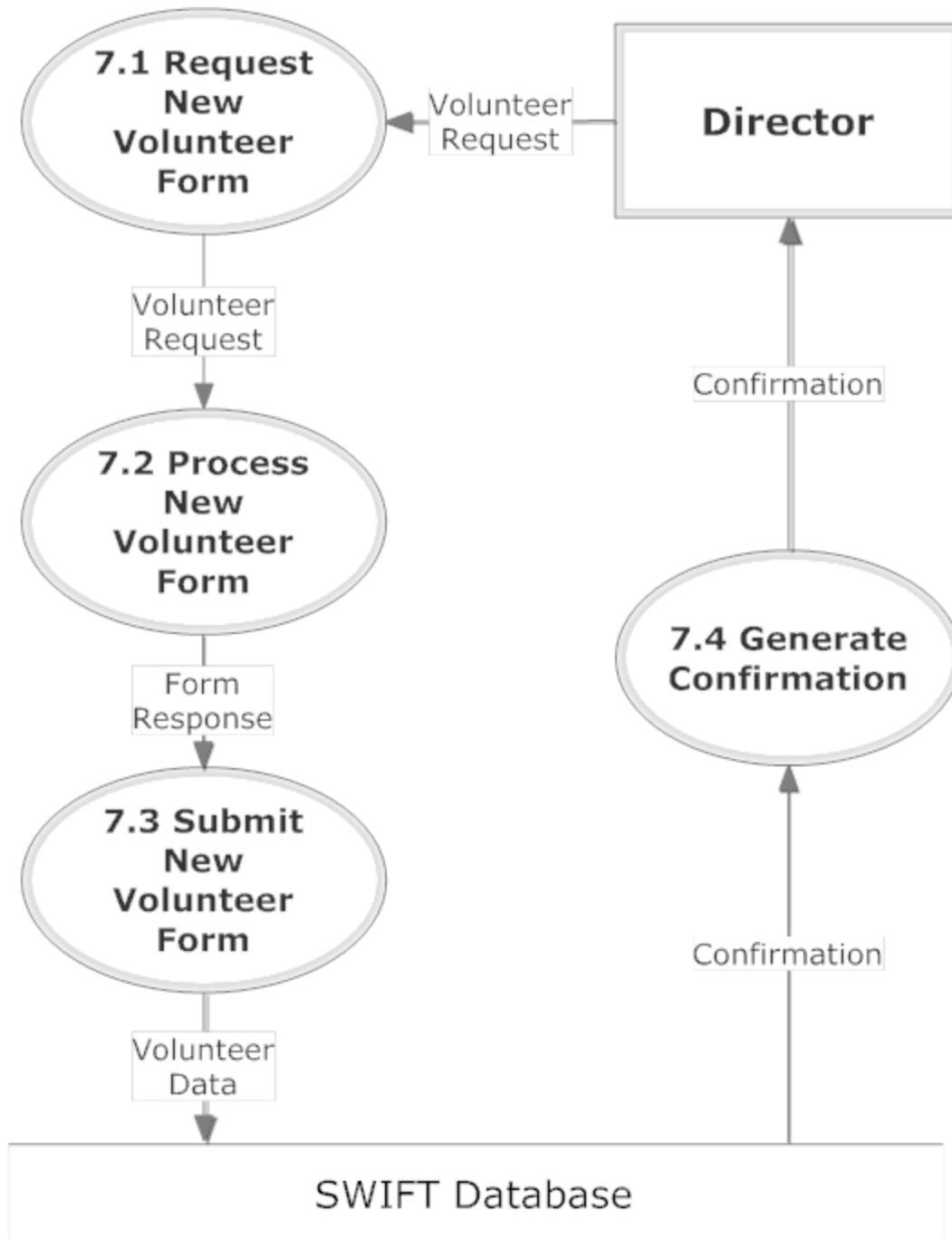




4.4.6 Level 1: Edit Customer/Donor



4.4.7 Level 1: Add Volunteer



## 5. Functional Requirements Inventory

The list below will provide a general outline for the users involved in the system and what they will have access to do. Since the software will be a user friendly web application, it will be able to be used on all major web browsers. The browsers that the software will be compatible with include Google Chrome, Safari, Mozilla Firefox, and Internet Explorer.

## 5.1 Volunteer

- Will be able to login
  - Logins will be individualized names and passwords
- Will be able to log out
- Will be able to search for a person
- Will be able to checkout items
- Will be able to create donation receipts
- Will be able to insert
  - customer information
  - purchase information
  - donor information

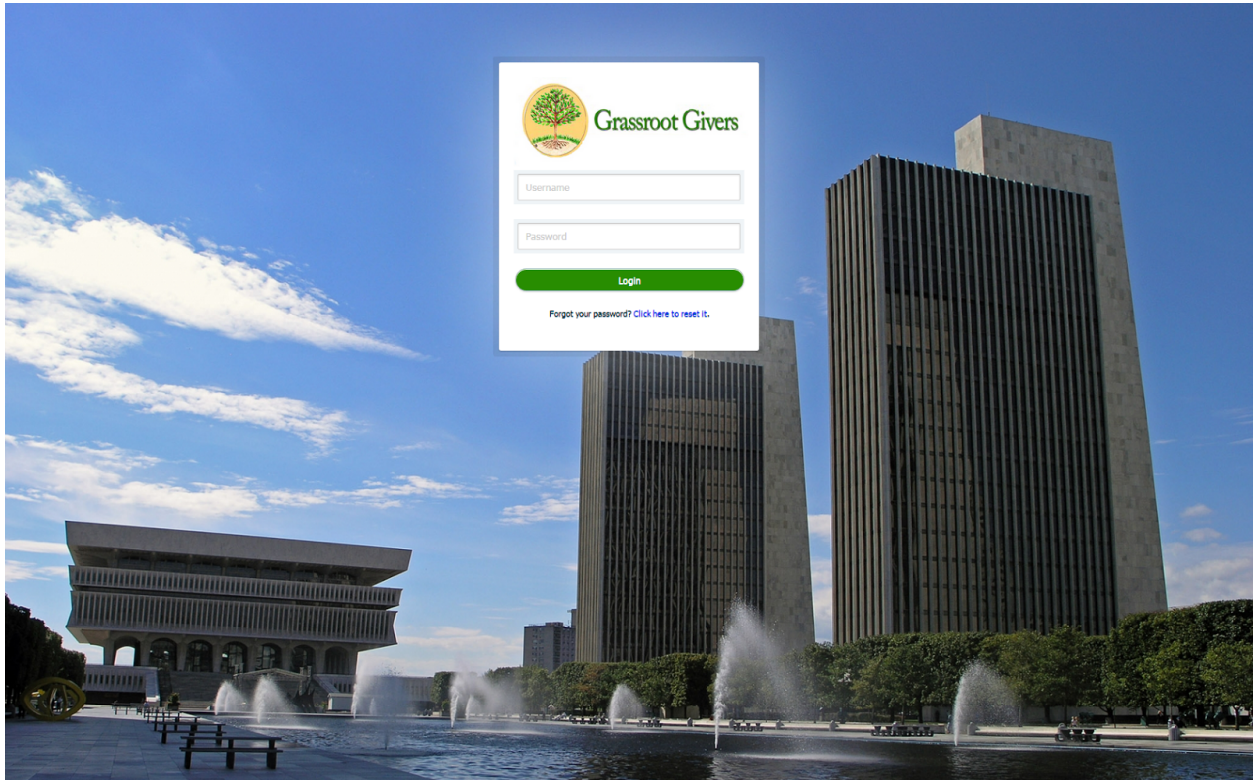
## 5.2 Director

- Will inherit all functional requirements for the volunteer
- Will be able to edit existing data on the system
- Will be able to delete data no longer wanted on the system
- Will be able to add volunteer accounts
- Will be able to delete volunteer accounts

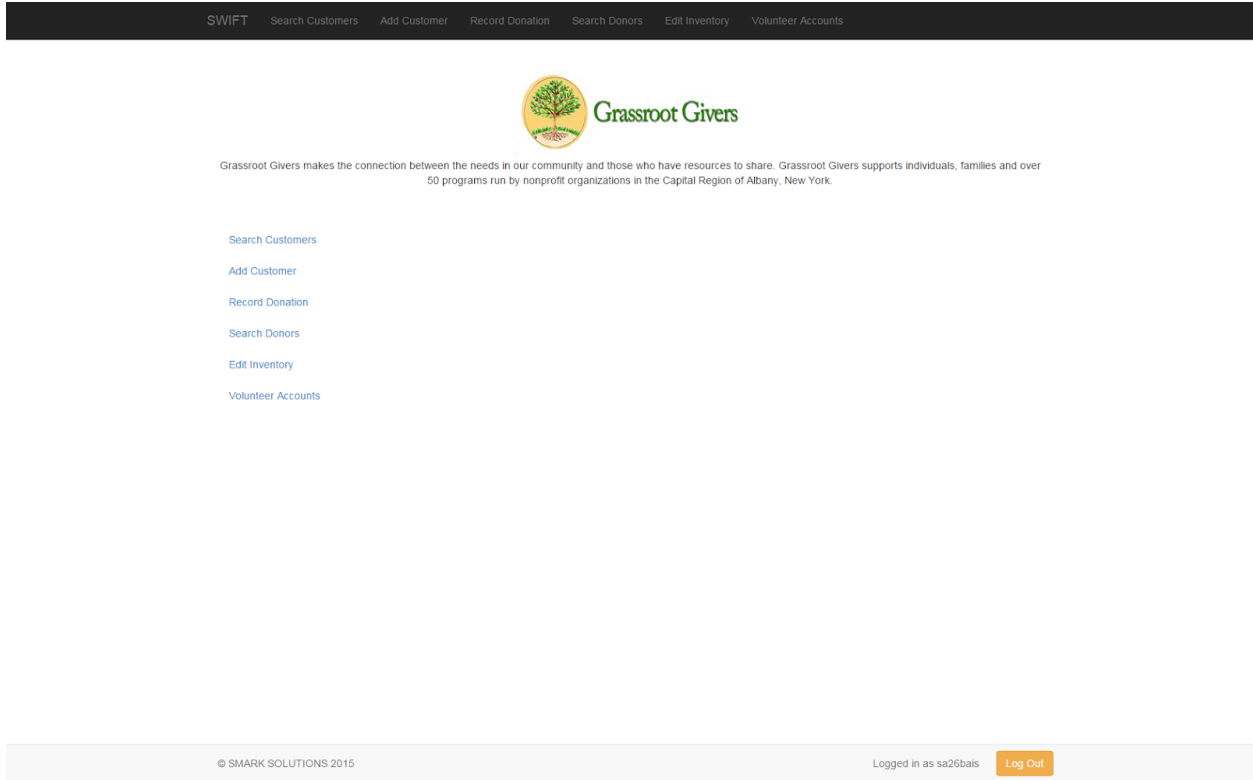
## 6. External Design Specifications

### 6.1 User Displays & Report Format

#### 6.1.1 Login Screen



## 6.1.2 Home Screen



The screenshot shows the home screen of the SWIFT application. At the top, a dark navigation bar contains the following menu items: SWIFT, Search Customers, Add Customer, Record Donation, Search Donors, Edit Inventory, and Volunteer Accounts. Below the navigation bar is the Grassroot Givers logo, which features a tree icon and the text "Grassroot Givers". Underneath the logo is a paragraph of text: "Grassroot Givers makes the connection between the needs in our community and those who have resources to share. Grassroot Givers supports individuals, families and over 50 programs run by nonprofit organizations in the Capital Region of Albany, New York." Below this text is a vertical list of links: Search Customers, Add Customer, Record Donation, Search Donors, Edit Inventory, and Volunteer Accounts. At the bottom of the page, a light gray footer bar contains the copyright notice "© SMARK SOLUTIONS 2015" on the left, the text "Logged in as sa26bais" in the center, and a "Log Out" button on the right.

## 6.1.3 Search Customer Screen

SWIFT Search Customers Add Customer Record Donation Search Donors Edit Inventory Volunteer Accounts

### Search Customers

**By ID:**

**By First Name:**

**By Last Name:**

**By Address:**

© SMARK SOLUTIONS 2015 Logged in as sa26bais

## 6.1.4 Add Customer Screen

SWIFT Search Customers Add Customer Record Donation Search Donors Edit Inventory Volunteer Accounts

### Add New Customer

**Name:**

**Address:**

**Affiliations:**  or:

© SMARK SOLUTIONS 2015 Logged in as sa26bais



## 6.1.5 Record Donation Screen

SWIFT Search Customers Add Customer Record Donation Search Donors Edit Inventory Volunteer Accounts

### Record Donation

New Donor Information (optional)

**Name:** first name last name

**Contact:** email address phone number

Create Donor

Already a donor? select

### Donation

About box count box value

Clothing choose gender

**Tops:** choose item amount size color

Add Item(s)

**Bottoms:** choose item amount size color

Add Item(s)

**Footwear:** choose item amount size color

Add Item(s)

**Accessories:** choose item amount

© SIMARK SOLUTIONS 2015 Logged in as sa26bais Log Out

## 6.1.6 Search Donors Screen

SWIFT Search Customers Add Customer Record Donation Search Donors Edit Inventory Volunteer Accounts

### Search Donors

**By ID:**

**By First Name:**

**By Last Name:**

**By Phone:**

**By Email:**

© SMARK SOLUTIONS 2015 Logged in as sa26bais

## 6.1.7 Manage Volunteers Screen

SWIFT Search Customers Add Customer Record Donation Search Donors Edit Inventory Volunteer Accounts

### Manage Volunteer Accounts

#### Create New Account

**First Name:**

**Last Name:**

**Username:**

**Temporary Password:**

#### Edit Existing Account

**Edit First Name:**

**Edit Last Name:**

**New Password:**

Director Permissions  
 Volunteer Permissions

#### Activate/Deactivate Account

**Activate:**

© SMARK SOLUTIONS 2015 Logged in as sa26bais

## 6.1.8 Edit Inventory Screen

SWIFT Search Customers Add Customer Record Donation Search Donors Edit Inventory Volunteer Accounts

### Edit Inventory

**Add New Item Type:**

**Rename Item:**  **to:**

**Edit Value of:**

**Update Quantity:**   **to:**

ID	Item Name	Category	Value	Gender	Quantity
118	Clock/Radio	Home Goods	\$8		0
119	Coffee Pot	Home Goods	\$12		0
121	Comforters/Bed Spreads	Home Goods	\$12		0
122	Dish Set	Home Goods	\$10		0
14	Dress	Tops	\$9	Women's	0
95	Dress	Tops	\$9	Children's	56
78	Dress Pants	Bottoms	\$9	Men's	0
21	Dress Slacks	Bottoms	\$9	Women's	0
108	Duffie Bag	Accessories	\$14		0
120	Electric Can Opener	Home Goods	\$12		0

© SMARK SOLUTIONS 2015 Logged in as sa26bais

## 6.2 Logical Data Dictionary

This logical data dictionary is used to describe the metadata that we will use in our database for S.W.I.F.T. The information we will keep track of are data name, synonym, what the data is applicable to, data type, data size, description, acceptable input, an example, and note.

Data Name	Synonym	Applicable to	Data Type	Data Size	Description	Acceptable Input	Example	Notes
affiliation	customers affiliation with agencies	adding a customer to the system, viewing a customer in the system	Varchar	1-30 Characters	Agencies the customers are affiliated with	ASCII char 32 (space), -, A-Z, a-z, ', `', ASCII char 128 to ASCII char 165	AA	
box_count	Bag/box count of donation	adding donation to the inventory	Integer	2 Digits	Count of bags/boxes donated by the donor	0 through 9	2	Min:0 Max:99
cust_address	Customer's current address	Adding Customer to system, viewing customer in system	Varchar	1-70 Characters	Customer's current address	ASCII char 32 (space), -, A-Z, a-z, ', `', ASCII char 128 to ASCII char 165	555 Anystreet Drive, Albany, NY, 12206	
cust_firstname	Customer First Name	Adding Customer to system, searching for customer within system	Varchar	1-30 characters	Customers first name	ASCII char 32 (space), -, A-Z, a-z, ', `', ASCII char 128 to ASCII char 165	Joe	
cust_lastname	Customer Last Name	Adding Customer to system, searching for customer within system	Varchar	1-30 Characters	Customer's last name	ASCII char 32 (space), -, A-Z, a-z, ', `', ASCII char 128 to ASCII char 165	Smith	
dep_firstname	Dependent's first name	adding customer information to the system, viewing	Varchar	1-30 Characters	Dependent's first name	ASCII char 32 (space), -, A-Z, a-z, ', `', ASCII char 128 to ASCII char 165	Bob	

**S.W.I.F.T. Acceptance Test**

		customer in the system				128 to ASCII char 165		
--	--	------------------------	--	--	--	-----------------------	--	--

Data Name	Synonym	Applicable to	Data Type	Data Size	Description	Acceptable Input	Example	Notes
dep_gender	Dependent Gender	Adding Customer to system, searching for customer within system	Char	1 Character	Dependent's Gender	M for Male item, F for Female item	M	
dep_lastname	dependent's last name	adding customer information to the system, viewing customer in the system	Varchar	1-30 Characters	Dependent's last name	ASCII char 32 (space), -, A-Z, a-z, ' , ` , ASCII char 128 to ASCII char 165	Smith	
dep_relation	relationship of dependent to customer	adding customer information to the system, viewing customer in the system	Varchar	1-30 Characters	The relationship of the dependent to the customer	ASCII char 32 (space), -, A-Z, a-z, ' , ` , ASCII char 128 to ASCII char 165	Son	
director_firstname	director's first name	name stamp on entered information	Varchar	1-30 Characters	The director's first name	ASCII char 32 (space), -, A-Z, a-z, ' , ` , ASCII char 128 to ASCII char 165	Store	
director_lastname	director's last name	name stamp on entered information	Varchar	1-30 Characters	The director's last name	ASCII char 32 (space), -, A-Z, a-z, ' , ` , ASCII char 128 to ASCII char 165	Director	
director_password	director's password	login and logout	Varchar	6-12 Characters	The director's password so they can log into S.W.I.F.T.	Password must be 6-12 characters long, must contain a number, and capital letter	donationS2	
director_username	director's username	login and logout	Varchar	1-30 Characters	The director's user name so they can log into S.W.I.F.T.		director	

S.W.I.F.T. Acceptance Test

Data Name	Synonym	Applicable to	Data Type	Data Size	Description	Acceptable Input	Example	Notes
donation_value	Monetary value of donation	adding donation to the inventory	Float	5 Digits	The monetary value of donation	0 through 9 and '.'	123.45	Value in Dollars and cents, must be proper decimal format
donation_weight	Weight of bag/boxes from donation	adding donation to the inventory	Float	5 Digits	The amount the bags/boxes donated weigh	0 through 9 and '.'	12.55	Weight in pounds
donor_address	Donor's address	adding donor to the system, viewing donor in the system, creating receipts	Varchar	1-70 Characters	Donor's address	ASCII char 32 (space), -, A-Z, a-z, ', ` , ASCII char 128 to ASCII char 165	555 Anystreet Drive, Albany, NY, 12206	
donor_firstname	Donor's first name	adding donor to the system, viewing donor in the system, creating receipts	Varchar	1-30 Characters	Donor's first name	ASCII char 32 (space), -, A-Z, a-z, ', ` , ASCII char 128 to ASCII char 165	Fred	
donor_lastname	Donor's last name	adding donor to the system, viewing donor in the system, creating receipts	Varchar	1-30 Characters	Donor's last name	ASCII char 32 (space), -, A-Z, a-z, ', ` , ASCII char 128 to ASCII char 165	Johnson	
donor_phone	Donor's phone number	adding donor to the system, viewing donor in the system, creating receipts	Varchar	1-15 Characters	Donor's phone number	0 through 9 and '.'	518-555-5555	Standard US 10 digit phone number

**S.W.I.F.T. Acceptance Test**

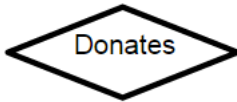
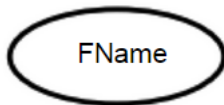
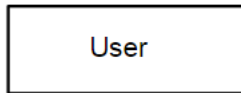
<b>Data Name</b>	<b>Synonym</b>	<b>Applicable to</b>	<b>Data Type</b>	<b>Data Size</b>	<b>Description</b>	<b>Acceptable Input</b>	<b>Example</b>	<b>Notes</b>
item_desc	desription of item	adding donation to the inventory	Varchar	1-60 Characters	Describes the item	ASCII char 32 (space), -, A-Z, a-z, ', ` , ASCII char 128 to ASCII char 165	black men's suit jacket	Color, Gender, Article of clothing
item_gender	Keeps track of the gender the item is for	Adding donation to the inventory	Char	1 Character	Keeps track of the gender the item is for	M for Male item, F for Female item	Women's pants, men's shirt	
item_quant	number of items	adding donation to the inventory	Integer	2 digits	Count of the number of items	0 through 9	4	Min:0 Max:99
timestamp	timestamp for entered information	checking out the customer, creating customers and donors	Varchar	16 Characters	The time and date the information was input	0 through 9, '/', and ':'	14:24 11/09/2014	
vol_firstname	volunteer's first name	creating volunteer accounts, name stamp on entered information	Varchar	1-30 Characters	Volunteer's first name	ASCII char 32 (space), -, A-Z, a-z, ', ` , ASCII char 128 to ASCII char 165	Ed	
vol_lastname	volunteer's last name	creating volunteer accounts, name stamp on entered information	Varchar	1-30 Characters	Volunteer's last name	ASCII char 32 (space), -, A-Z, a-z, ', ` , ASCII char 128 to ASCII char 165	Jones	
vol_password	volunteer's password	creating volunteer accounts, login and logout	Varchar	6-12 Characters	The password for a volunteer so they can log into S.W.I.F.T.	Password must be 6-12 characters long, must contain a number, and capital letter	Donations 8	
vol_username	volunteer's username	creating volunteer accounts, login and logout	Varchar	1-30 Characters	The user name for a volunteer so they can log into S.W.I.F.T.	Volunteer's first initial, followed by their last name	ejones	Will be first initial and last name





## 6.3 ER Diagram

### 6.3.1 ER Diagram Legend



**Entity:** An object which we wish to model within a database.

**Attribute:** A characteristic of an entity which we wish to store. If the attribute is underlined, that attribute is a primary key.

**Relationship:** Connects two or more entities, showing how the entities are related.

**Single Line:** Signifies a connection between a relationship and an entity.

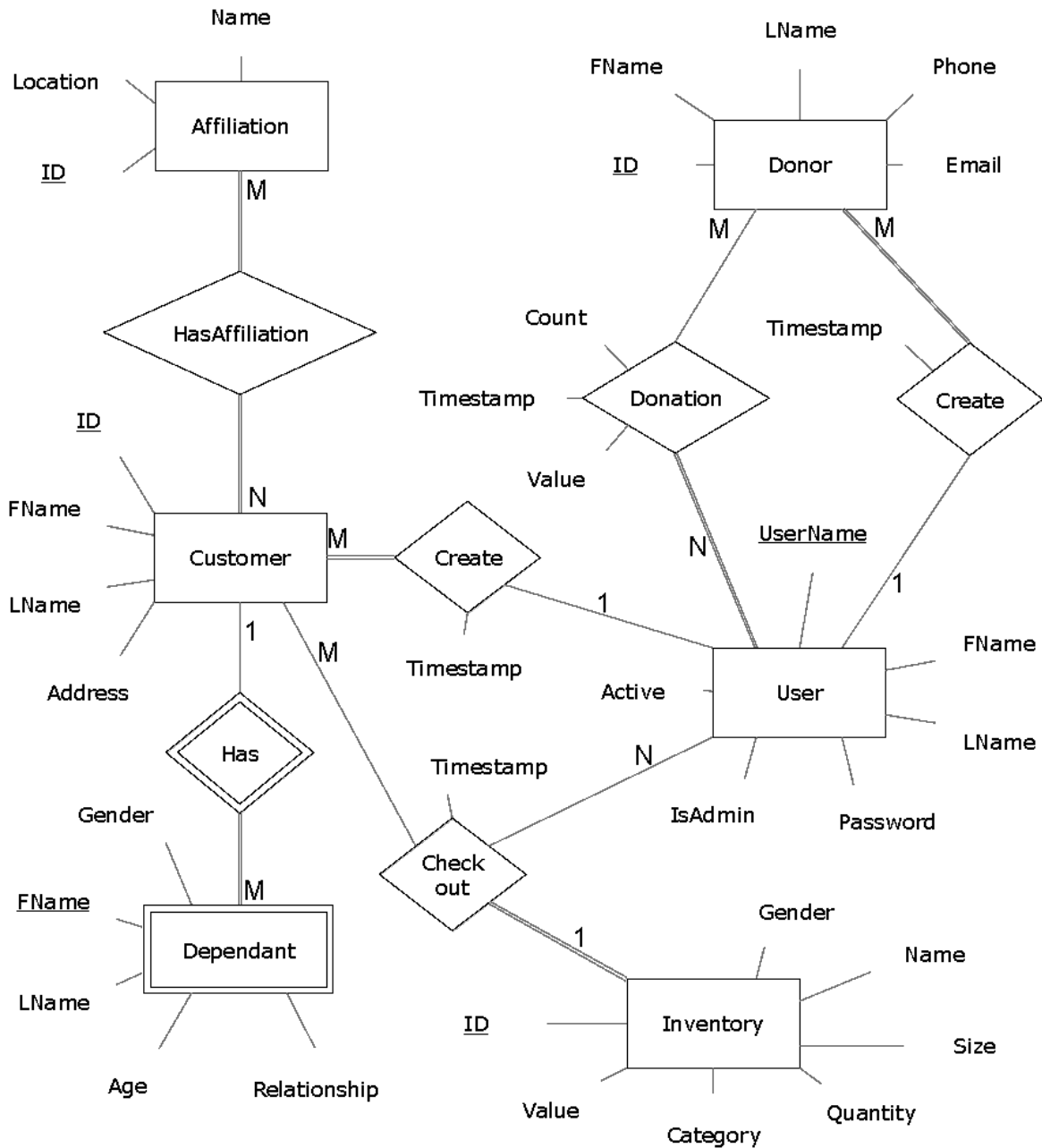
**Double Line:** There must be at least one relation for each individual tuple of the entity attached to the line.

**M/N Relationship:** There can be more than one relation where an individual tuple occurs.

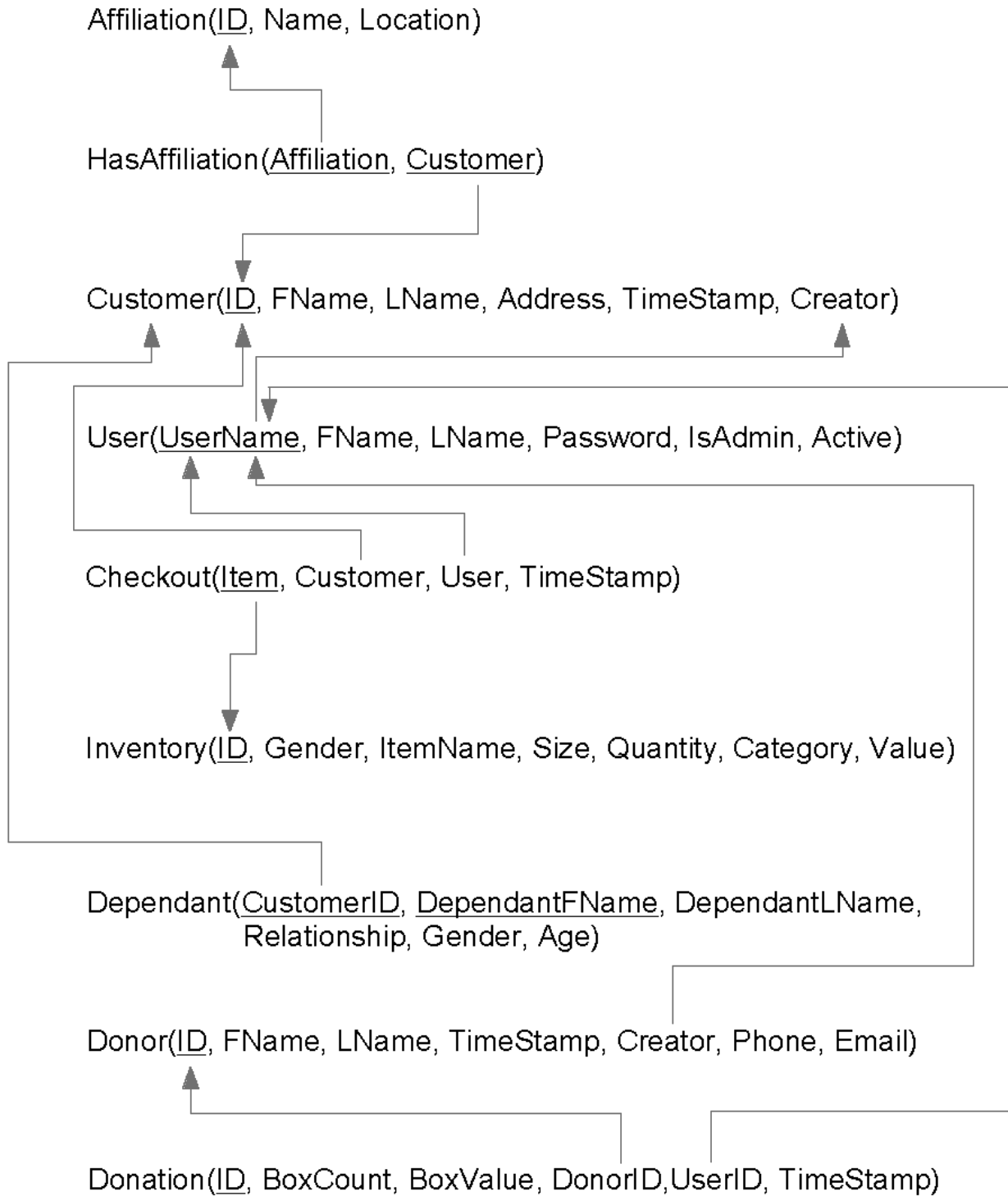
**M/1 Relationship:** The entity on the M side of the relation can only have a max of 1 instance of each individual tuple. The other side has no maximum.

**1/1 Relationship:** The individual tuples from each connected entity can appear a max of once.

### 6.3.2 ER Diagram



## 6.4 Relational Schema





## 6.5 Database Tables

## Affiliation

Column Name	Type	Not Null	Foreign Key
<u>ID</u> (auto increment)	Int(11)	Yes	
Name	Varchar(45)	No	
Location	Varchar(145)	No	

## Checkout

Column Name	Type	Not Null	Foreign Key
<u>Item</u>	Int(11)	Yes	Inventory
Customer	Int(11)	Yes	Customer
User	Varchar(20)	Yes	User
TimeStamp	Datetime	Yes	

## Customer

Column Name	Type	Not Null	Foreign Key
<u>ID</u> (auto increment)	Int(11)	Yes	
FName	Varchar(45)	No	
LName	Varchar(45)	No	
Address	Varchar(145)	No	
TimeStamp	Datetime	Yes	
Creator	Varchar(20)	Yes	User

## Dependant

Column Name	Type	Not Null	Foreign Key
<u>CustomerID</u>	Int(11)	Yes	Customer
<u>DependantFName</u>	Varchar(45)	Yes	
DependantLName	Varchar(45)	No	
Relationship	Varchar(45)	No	
Gender	Varchar(45)	No	
Age	Int(11)	No	

## Donation

Column Name	Type	Not Null	Foreign Key
<u>ID</u> (auto increment)	Int(11)	Yes	
BoxCount	Int(11)	No	
BoxValue	Double	No	
DonorID	Int(11)	No	Donor
UserID	Varchar(45)	Yes	Customer
TimeStamp	Datetime	No	

## Donor

Column Name	Type	Not Null	Foreign Key
<u>ID</u> (auto increment)	Int(11)	Yes	
FName	Varchar(45)	No	
LName	Varchar(45)	No	
TimeStamp	Datetime	Yes	
Creator	Varchar(20)	Yes	User
Phone	Varchar(45)	No	
Email	Varchar(45)	No	



## HasAffiliation

Column Name	Type	Not Null	Foreign Key
<u>Customer</u>	Int(11)	Yes	Customer
<u>Affiliation</u>	Int(11)	Yes	Affiliation

## Inventory

Column Name	Type	Not Null	Foreign Key
<u>ID</u> (auto increment)	Int(11)	Yes	
Gender	Varchar(45)	No	
ItemName	Varchar(45)	No	
Size	Varchar(45)	No	
Quantity	Int(11)	Yes	
Category	Varchar(45)	No	
Value	Varchar(45)	Yes	

## User

Column Name	Type	Not Null	Foreign Key
<u>UserName</u>	Varchar(45)	Yes	
FName	Varchar(45)	No	
LName	Varchar(45)	No	
Password	Varchar(45)	Yes	
IsAdmin	Bit(1)	Yes	
Active	Bit(1)	Yes	

## 7. Testing Plan

### 7.1 Overview and Strategy

SMARK Solutions is working to create an application that goes above and beyond the expectations of the clients, Dr. Fryling, and Dr. Lim. S.W.I.F.T. is a web-based application, therefore it will be tested to ensure compatibility with all major web browsers. These browsers include Apple Safari, Mozilla Firefox, Google Chrome, and Internet Explorer. Along with this capability, we will test our application by using a number of test cases designed to ensure our application correctly accepts information, and then does not accept the information that is not designed to enter the database. In this section, you will find our test cases, along with our acceptance test. Actual results are not included in this document, as they will be entered in our Detailed Design document, which will be written during the spring semester.

The best and most efficient solutions possible will be created to help solve our client's problems. Together we can be intuitive, be efficient, be SMARK.

### 7.2 Acceptance Test

Each of the following major functional processes will have test cases created designed to test all of the individual aspects our S.W.I.F.T. to make sure that S.W.I.F.T. works the way it is designed to. Based on the results of these test cases, SMARK Solutions will come together to decide whether or not the program is acceptable to deploy for client use.

### 7.3 Unit Tests

The following unit tests are tests that will be run to ensure that our application is working properly. Each test will test an individual aspect of one of the major processes to show what inputs will work and what ones will not/should not work. Following these individual unit tests, a final test will be run to ensure each of the processes will integrate together correctly, allowing our application to do its designed job.

## 7.4 Test Cases

Each of the Unit Tests is made up of a series of test cases. In each case, there are specific guidelines in how to properly test the functionality of the process being tested. Each case also contains information about how the application should respond when each input is entered both before and after the input is processed. If the system responds how it is supposed to according to the case, then the system will be considered to be functional.

## 7.5 System Test

System testing will be conducted on S.W.I.F.T. to ensure our application meets all of the set requirements, both functional and non-functional. We will use black box testing to make sure the application behaves as it should. This means that when various inputs are entered in, such as new users and new inventory items, certain outputs should or should not be seen, ensuring our application works properly.

## 7.6 Integration Test

Integration testing will be conducted on S.W.I.F.T. to ensure each of the different components of the application interact as they should with all of the other components that make up S.W.I.F.T. . This will be completed through tests built into our unit and system testing to make sure everything works and cooperates properly.

## 7.7 Regression Test

Regression testing conducted on S.W.I.F.T. will take place after completion and in the future as updates and changes are made. This testing will be carried out to make sure any of the new changes or updates do not harm the functionality of the application. For this, both integration tests and the unit tests should be re-run to ensure the application is fully functional.

## 8. Development and Production Environments

### 8.1 Development Environment

#### *Windows Computer*

Operating System: Windows 7 Enterprise (x64) Service Pack 1  
Processor: Intel Core i5-3470 @ 3.20 GHz  
Ram: 6GB  
HDD Capacity: 499 GB

#### *Macintosh Computer*

Operating System: OS X Lion 10.7.5  
Processor: Intel Core i5 @ 2.5 GHz  
Ram: 4GB  
HDD Capacity: 378 GB

### 8.2 Operating Environment

This information has yet to be determined by the client. This application will be web-based, so it will operate from an off-site server. The application is designed to be as simple and easy to operate as possible, to allow anyone to easily use it.

### 8.3 Maintenance

Maintaining this application involves ensuring that the information is correct and up-to-date, and making sure that any updates to the server are remain compatible with S.W.I.F.T. as server maintenance is completed by the third party that houses the servers.

## 9. Deliverables

The following items will be included in the deliverables due April 27<sup>th</sup>:

1. A thumb drive or CD containing:
  - a. All of the files from our team directory, including all of the files that are required for the team website. This includes all of the past presentations and documents that the team has produced in the past.
  - b. All of the files that are associated with the team's project.
  - c. A file which explain where each of the files is located.
  - d. All of the usernames and passwords for the software as well as the necessary credentials to connect to the database.
  - e. The lyrics to the team song, the music video created by the team, and an audio file of the team song being performed.
2. A printed out copy of this document.
3. A printed out copy of the presentation prepared for the client.
4. The documents that the team prepares for Academic Celebration and at least one team member present at Academic Celebration.

### 9.1 Source Code

As stated above, the source code for the project will be delivered to the client in the form of a thumb drive at the end of the year software party.

## 10. Future Enhancements

Currently, SWIFT is designed to only add information and edit information in the master database. In the future, SWIFT will be modified to feature a printable receipt function, which will take the newly inserted donation/donor information and create a printable receipt that the donor or customer may keep for their use. Along with the receipt feature, a credit system where workers earn store credit through volunteer hours will be added in.

## 11. Appendices

### 11.1 Appendix A: Test Results

SMARK Solutions tested and approved the functionality of the SWIFT application. The finished product reflects the predetermined goals to track information within Grassroot Giver's Community Store. SWIFT has the ability to save donor and customer information including the donation and purchases respectfully. The application has a fully functional inventory system that is user-friendly and intuitive, as desired.

### 11.2 Appendix B: Sources of Information

SMARK Solutions utilized several outside sources of information in the development of SWIFT. The assistance of Siena College Computer Science professors was instrumental in the creation of the website and the database. Notably, Dr. Robert Yoder, Dr. Scott Vandenberg, Dr. Eric Breimer were resources beyond the professors of the course, Dr. Meg Fryling and Dr. Darren Lim. Additionally, Twitter's Bootstrap Framework was employed for the front-end of the application and Amazon's Relational Database was the home for the database.

### 11.3 Appendix C: Glossary of Terms

**Gantt Chart** - Bar chart typically used to project scheduling

**Data Flow Diagram** - A visual representation of how data moves throughout a system.

**Database** - An organized collection of data.

**ER Diagram** - (Entity-Relationship Diagram) Is a data model of the information of the business domain or process requirements to show how they will be implemented in a database.

**Functional Requirements** - Defines what the system will be able to do and what is testable about the system.

**Non-Functional Requirements** - Requirements that are not necessarily specific features that exist in a system, but what the system is intended to do.

**Processor** - The part of the computer that handles and executes operations.

**Prototype** - An early sample, model or release of a product built to test a concept.

**Pseudocode** - Is an informal high-level description of the operating principle of a computer program or other algorithm.

**Random Access Memory (RAM)** - a memory unit that allows any specific byte to be used randomly at any time.

**Relational Schema** - Is a model that shows how the database logically groups objects such as tables, views, stored procedures.

**Server** - a computer or program that manages access to a resource or service in a network.

**S.W.I.F.T.** - Simple Web Inventory For Tracking

**UML Use Case Diagram** - A visual representation of the users interaction with the system in a specific instance.

**Use Case Narrative** - a written explanation of the course of events a user will encounter when interacting with the system.

### 11.4 Appendix D: Timeline

Grassroot Givers		Start Date:		September 2, 2014					
SMARK Solutions									
Task	Start Date	End Date	Duration (days)	Percent Complete	SEP	OCT	NOV	DEC	
1.0 Software Plan	2014-09-08	2014-09-18	11	100.00%	█				
1.1 Software Plan Due	2014-09-19	2014-09-19	1	100.00%		█			
1.2 Software Plan Presentation	2014-09-20	2014-09-23	4	100.00%		█			
2.0 Requirements Specifications	2014-09-24	2014-10-23	30	100.00%		█			
2.1 Requirements Specifications Due	2014-10-24	2014-10-24	1	100.00%			█		
2.2 Requirements Specifications Presentation	2014-10-25	2014-10-30	6	100.00%			█		
3.0 Preliminary Design	2014-10-31	2014-11-25	26	100.00%			█		
3.1 Preliminary Design Due	2014-11-26	2014-11-26	1	100.00%				█	
3.2 Preliminary Design Presentation	2014-11-27	2014-12-04	8	100.00%				█	
4.0 Detailed Design	2014-12-05	2015-03-05	91	100.00%				█	
4.1 Detailed Design Due	2015-03-06	2015-03-06	1	100.00%				█	
4.2 Detailed Design Presentation	2015-03-07	2015-03-09	3	100.00%				█	
5.0 Acceptance Testing	2015-03-10	2015-04-20	42	100.00%				█	
5.1 Acceptance Testing Due	2015-04-21	2015-04-21	1	100.00%				█	
5.2 Acceptance Testing Presentation	2015-04-22	2015-04-22	1	0.00%					