Task 1

## Introduction

A small local volunteering community has set up a club for local residents, so that the residents may be able to go out since they are mostly immobile and cannot make it out by themselves without help from either the volunteers or their families. There are 15 elderly residents ranging in the ages from 80’s to 90’s. To manage this there are a total of 20 volunteers, who are coordinated by a lead volunteer each week. This responsibility is changed each week to a different volunteer.

## Background - Investigation into the current system

The current system is a manual production of paper based rotas (created using Microsoft word with the help of a clerical assistant) which are distributed among the volunteers by post and email. Along with this, people then contact the each other by telephone or email in the event of emergency or unexpected changes to availability.

There are 20 volunteers to the club, and 15 members. Some of the members drive members to the events and meetings, whilst others only volunteer at the meetings. Some members do both.

The system requires at least 3 volunteers be present at events and club meetings, as well as a leader who changes weekly. Although there are about 20 volunteers, only some choose to be leaders, and most are only available once a month to be a leader.

The volunteers often told 2-3 months ahead of the meeting whether they are expected to volunteer.

The current system faces several issues. For example, the system provides poor distribution of personal details, and does information regarding equipment and supplies. As well as this, the system does not provide security for member and volunteer details, and does not allow weekly updates to the club events.

## New system

The new system shall be an IT based solution. It will be able to handle a rosta for each volunteer, for each week.

Programs/solutions that could possibly be used are :

- Web interface: The new system could use a web interface, which could either be developed specially, or use a pre-created package. This will make the roster system easily accessible by the web for all volunteers to see. It will also be easier to make changes to the roster if the lead volunteer for the week is away on other business.

Excel / spreadsheet software : this will make just as easy to make changes and make new roster’s for the volunteers however it will be stored locally on one machine meaning only one person shall have this roster. however it could be made available through a web interface.

Access / database softwares : again this will make it easy to change and make rosta’s for the volunteers however this software will be harder to modify and if anything is to happen to the system, i.e the database breaks, they will need to hire an IT professional to fix it.

External roster software : Simple softwares such as ABC rosta will help provide a clear roster

Based on the information that is provided, the conclusion to a decision is that the system will have to be as easy to use, navigate and to make changes as possible, this is only because the people that are using this system are not computer literate, so a simple system that is for the layman will be of great advantage to the volunteers.

## Feasibility

To determine that the new system will be at all possible to implement, a feasibility study needs to take place, to see if:

1. The cost of the new system will be affordable for the members. Since this club is run by volunteers, it could be assumed that this group is non-for profit. So how do they get the funding to run the group ?
2. Free resources are hard to find, meaning that some features of this new system will cost the group. This is identified in point 1.
3. Is this new system technically possible with any known software free or otherwise?
4. Has this new system got a time that it must be finished by, if the system must be built and working within a specific time, a schedule will need to be designed to help the development of the new system.
5. How many persons will be involved with building this new system, will they each have their own tasks?

## System Requirement Specification

### Functional requirements

Managing volunteers

- The system shall allow addition of new volunteers to the roster system.

- The system shall allow the editing of the volunteers details on the roster system.

- The system shall allow the deletion of a volunteer.

- The system shall be able to create and centralise all letters.

- The system shall be secure in order to provide information on a need to know basis.

- The system shall allow times to be allocated by request of the volunteer for a specific time slot and date.

- The system shall alert volunteers to changes in the rostering system.

- The system shall allow an individual volunteer to log in.

- The system shall allow the volunteers to change their password on the system.

Managing residential members  
- The system shall allow addition of new residential members.

- The system shall allow the deletion of current residential members.

- The system shall allow editing of residential members details.

- The system shall allow residential members to see upcoming events, and sign up for them.

- The system shall be able to produce a roster.

Pick-up System

- The system shall show who needs to be picked up on given days

- The system shall allow a route id to be given to each residential member

- The system shall allow a routes to be seen and chosen by volunteers

Rostering system

- The system shall allow volunteers times to be changed any any point during a working period.

- The system shall alert any volunteers of changes made to their roster if they have not made it themselves, by a prefered means of contact.

- The system shall allow extra shifts to be created for days when more volunteers are needed.

- The system shall allow consideration for illness of members and volunteers.

- The system shall take into consideration any private appointments that the members have, so they should not be disturbed.

## 

## Non Functional requirements.

- The system shall be available most hours of the day. This is to ensure that last minute changes can be made if a volunteer has an emergency to attend, however this could cause abuse of the system.

- The system shall have a very minimal downtime. Downtime shall be limited to accident and system restarts during the night ( 3AM ).

- The system shall be user friendly.

- The system shall be easy to navigate.

- The system shall be able to hold a large number of volunteers.

- The system shall be able to hold a large number of members.

- The system shall have a look and feel that is professional.

- The system shall manage all logins for volunteers.

- The system shall manage all passwords for the volunteers.

- The system shall allow allow for multiple users to use it at the same time.

- The system shall have a secure connection using encryption and/ or secured connections.

- The system shall take into consideration the Data Protection Act 1998. To ensure that all volunteers and members details are kept safe within the system.

- The systems database shall encrypted on a secure server.

## Technical system options.

There are many options that are available to us to make a system that is successful. One such option is a rostering system on a single computer. This by itself is very secure and information is not likely to be taken from it. However the problem with this is, that only one person shall be able to use it, the computer may be available for many hours during the day, in the central office of the volunteers. However again this poses more problems, the computer is no longer available during ‘ most hours of the day’. One more way to get around this is that you could host the rostering system on a separate dedicated server, that any person may be able to log into. There will be no downtime, since restarts on these servers take only a minute and the roster is back up and running.

Another option is spreadsheet software, again free softwares are available from website such as sourceforge, or premium software such as microsoft excel would also work, however for central convenience is google docs, With GD you can create a spreadsheet that is available for everyone with an available google account. There is also no down time on this as the document is held in the cloud. However, with sensitive data being kept within this system, it is liable to breaking the data protection act 1998.  
  
The final option and most likely to be the most secure is to create a website that shows the rostering system on a log in screen, and the database where all of the information is to be kept is on a secure server with only the designer of the system knowing where the information is. This way the only way to enter and delete data in the database is to use the online forms and rostering system. This shall also be in line with the data protection act 1998.

## Recommendation

The final Recommendation is a web based system which allows accounts for each of the members, each of the volunteers, and an admin (the clerical assistant). The web based system should store the rotas, as well as the upcoming events, and allow members to RSVP to events. The system should also allow volunteers to show their availability.

The system will store all contact details of the volunteers, members and the members assisting family/next of kin, but allow allow the admin account access to these details.

The recommended system would allow the admin account to generate rotas, create and make changes to user accounts, and make and edit event. The system would send out the rotas, and lists of events by email, but also allow the clerical assistant or other admin to print information out to allow it to be sent by mail.

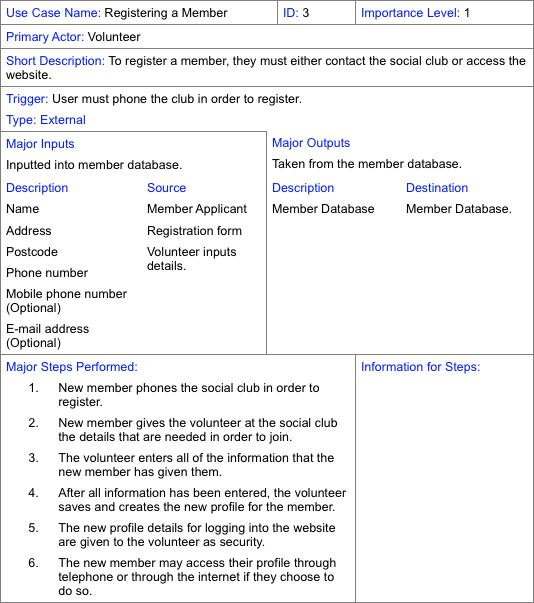
The volunteers would be able to see their rotas online, and request changes to specific rotas directly to the admin.

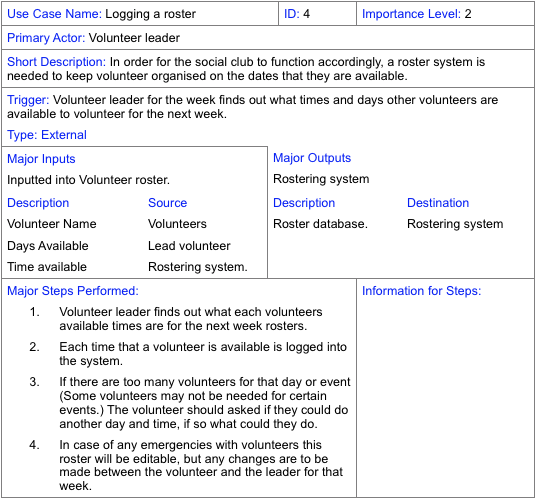
## Logical Design.

### Use Case narratives

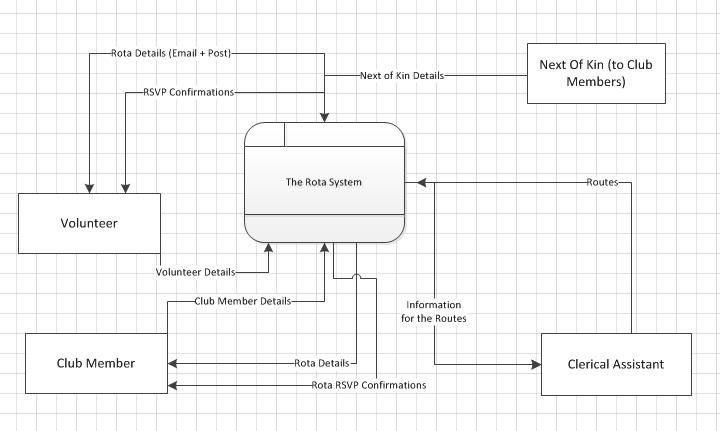
### 

### 





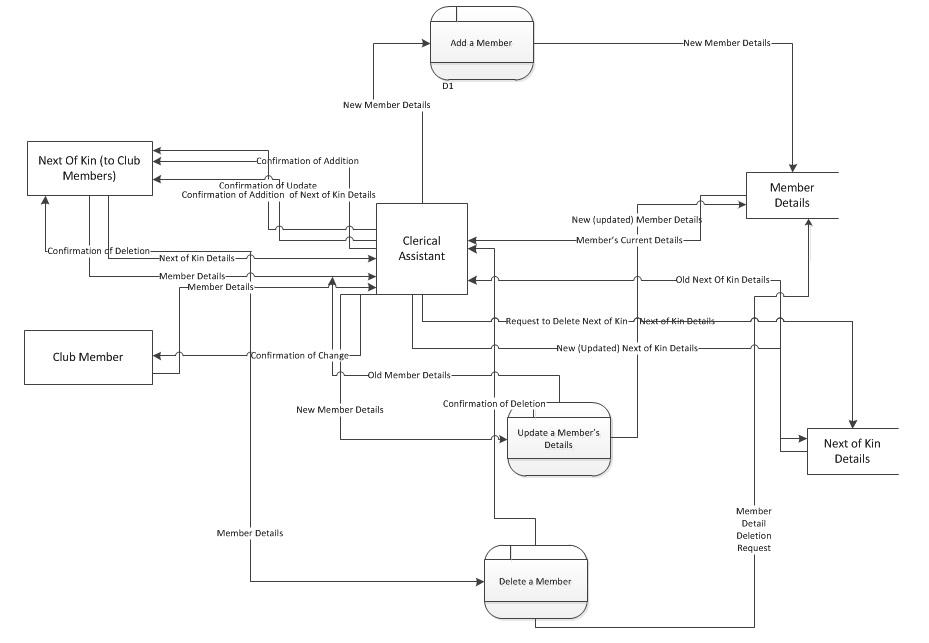
## Data Flow Diagrams



Context Diagram which shows the external entities which add contribute to the Rota system, and what data they give or take from the system.

### Level 1.jpg

Level 1 Data Flow Diagram for the System



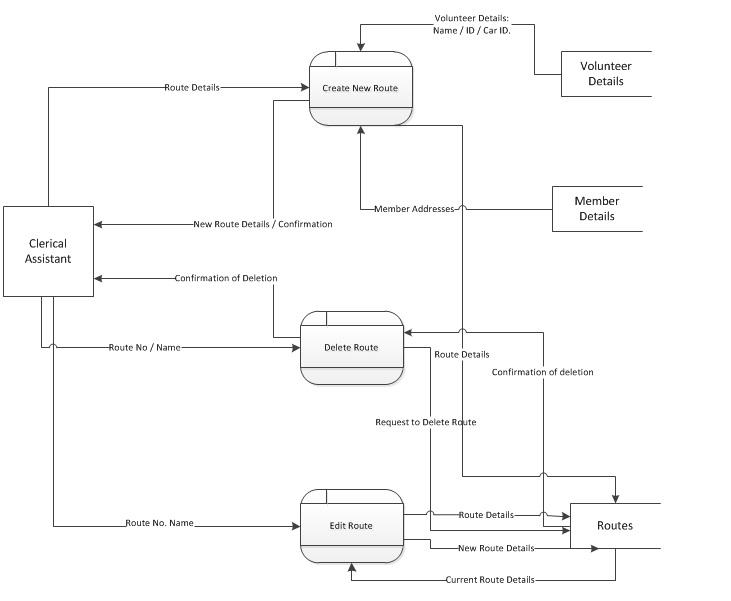
Managing changes to the Member and Next of Kin Details Data Flow Diagram

### Managing Volunteer Details.jpg

Managing Volunteer Details Data Flow Diagram

### Managing Events.jpg

Managing Events DFD



Managing Routes Data Flow Diagram

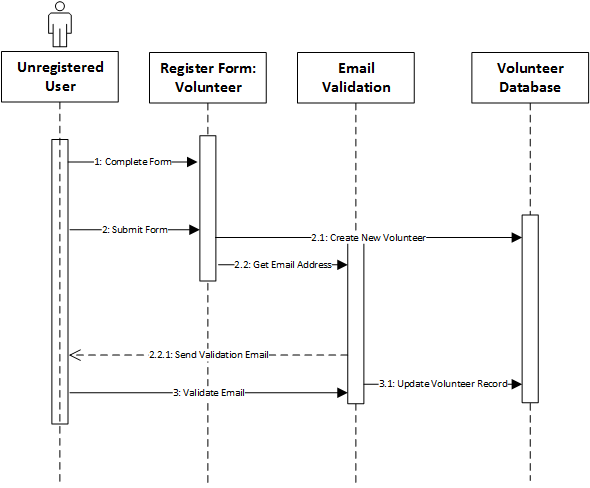
### Managing Rotas.jpg

Managing Rotas Data Flow Diagram

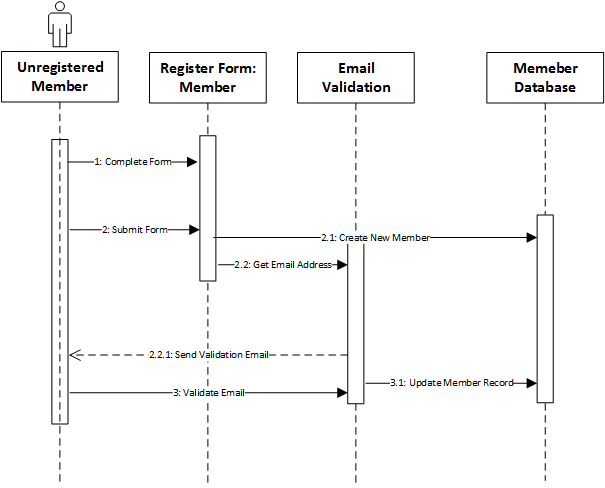
### Entity Relationship DiagramsERD.png

### Sequence Diagrams

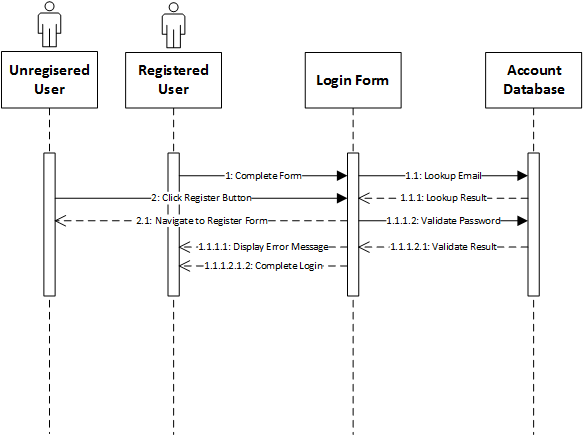
Volunteer Registration

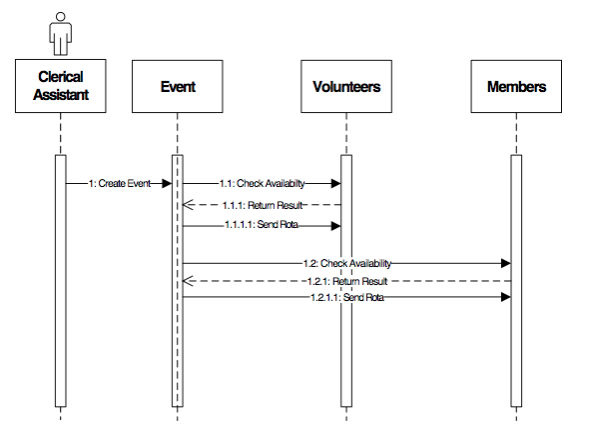


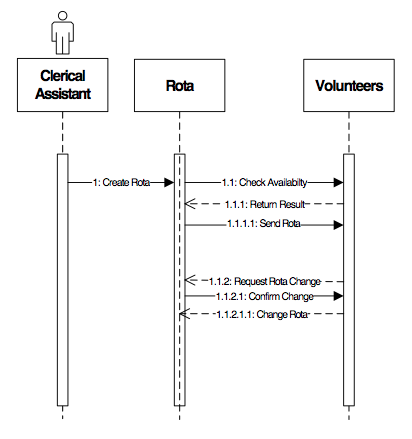
Member Registration

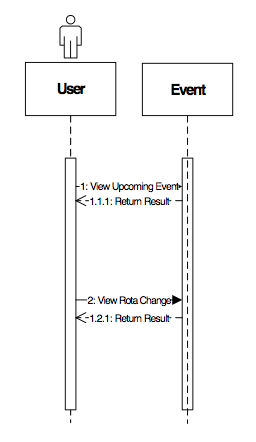


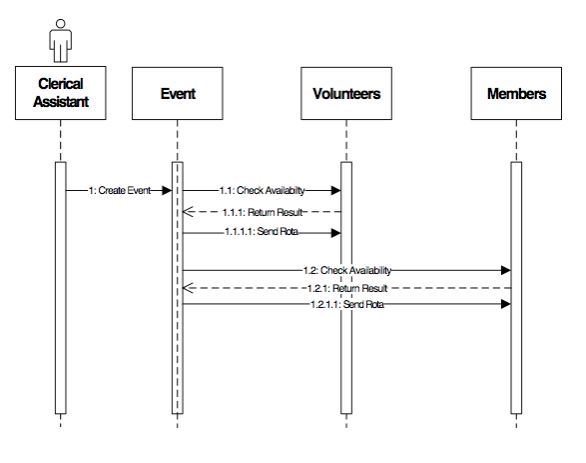
User Login



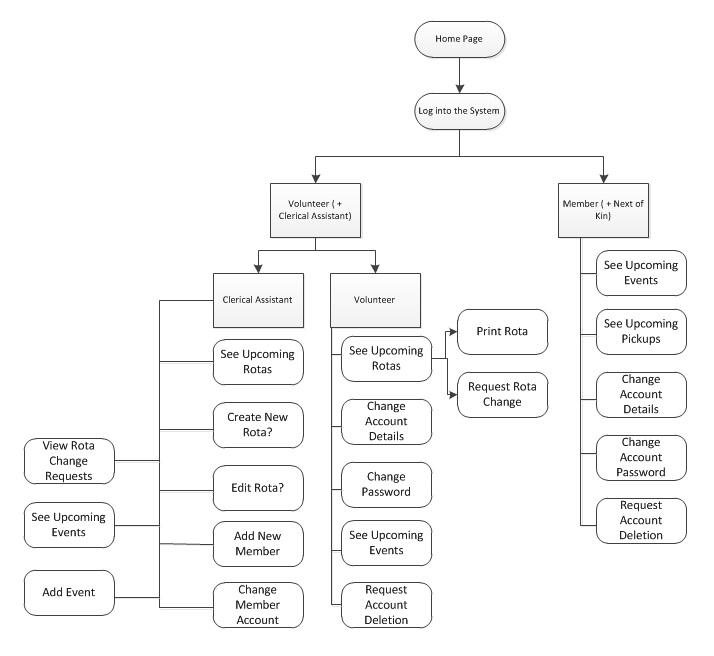


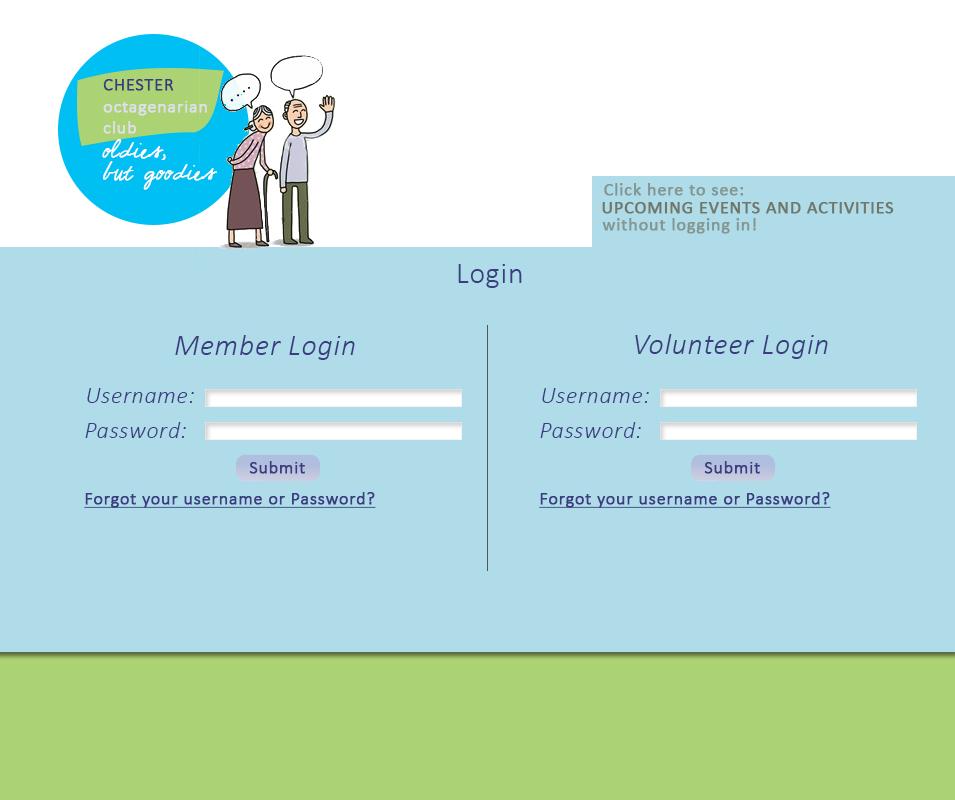




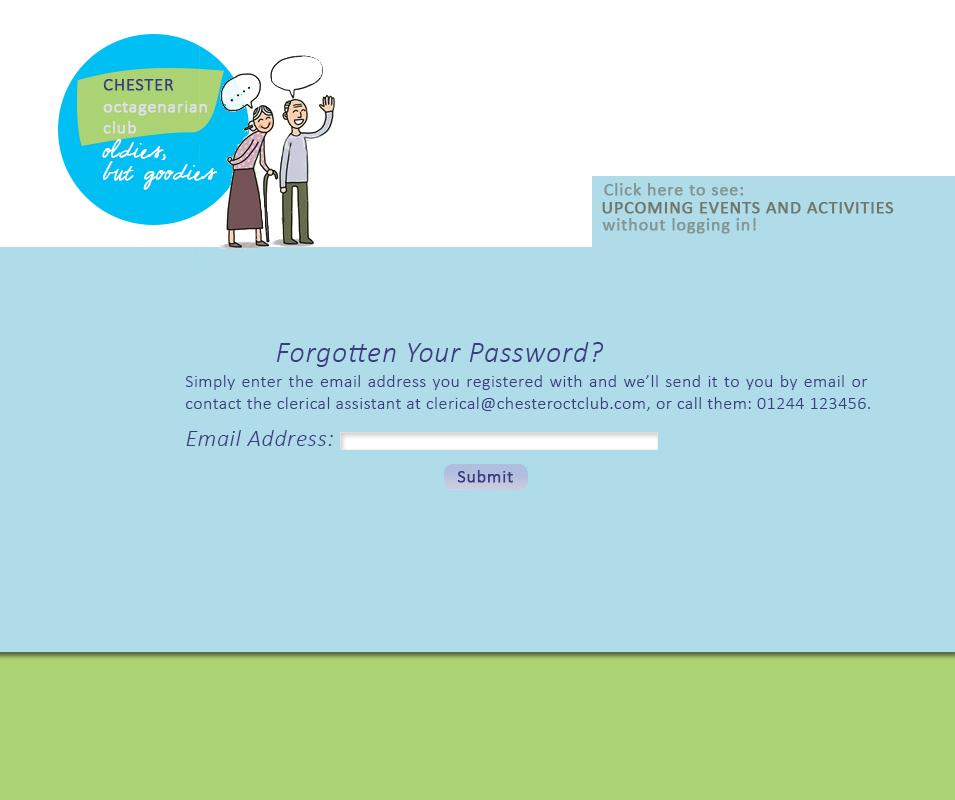


### User Interface Diagrams

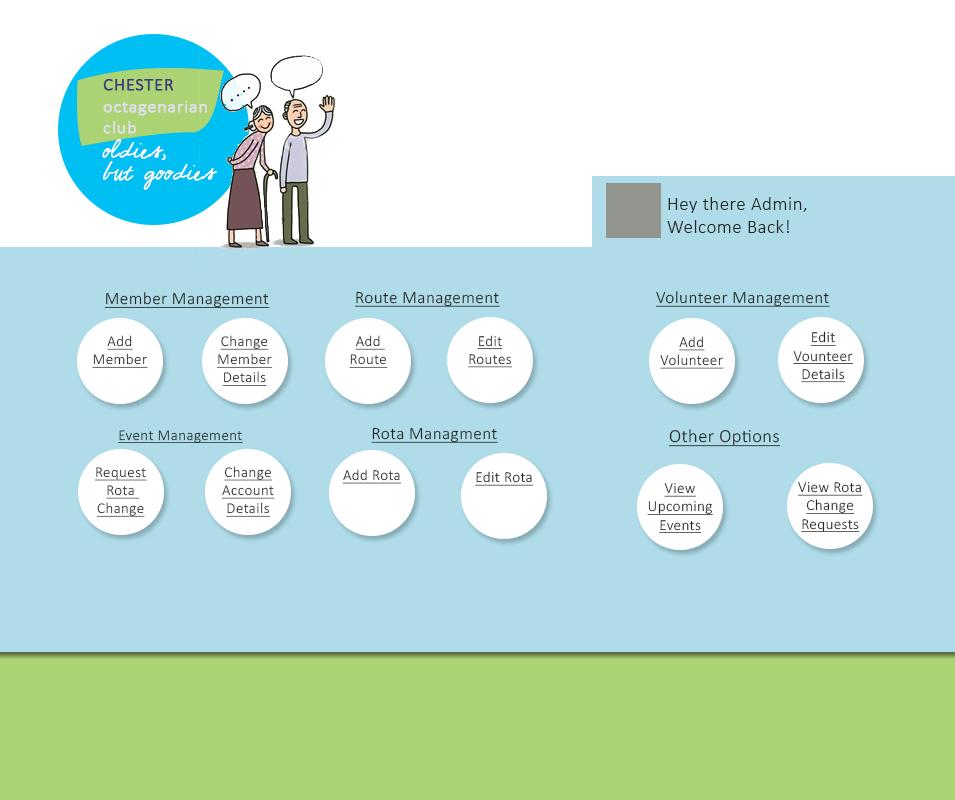
 User Interface Map



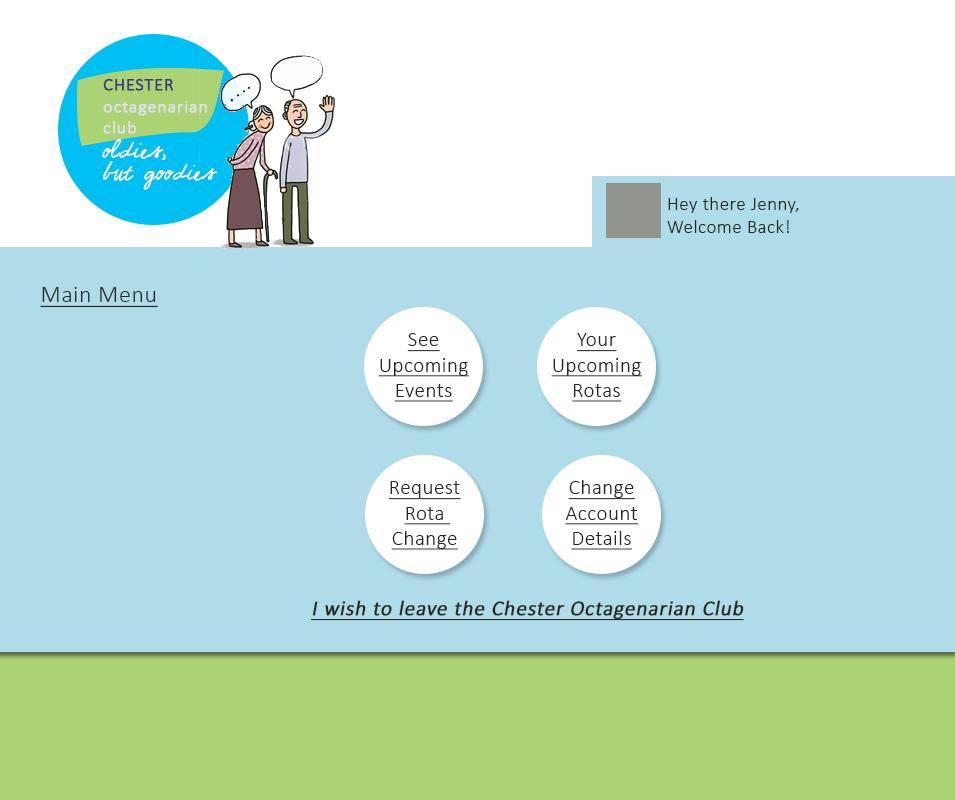
Example Login Page



Example Forgotten Password page



Example Admin (clerical assistant) menu



Example Main menu for volunteers



Example Upcoming Events Page

## Testing Design

|  |  |  |  |
| --- | --- | --- | --- |
| Test Name | Expected Result | Actual Result | Action Taken |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |