INTRODUCTION

The problem domain : People want to find out how well their friends did , just by entering their friend’s name and bib number. However there are many races happening all around the world. Therefore we need an application that helps us to organise the races in terms of their location and enable people to find their friends using their friend’s bib number.

This report presents a design analysis which consists of the domain model of the task and the task analysis of the user who will be using our application. The objective of this design is to be as efficient as possible when the user wants to find his/her friends and also to provide a good user interface which gives a "user-friendly" experience, allowing the user to interact with the software in a natural and intuitive way.

The current assumptions we have in the this model is that , a racer can run in more than one location organised by the same club therefore one racer has many “data” due to the different locations that they ran at. another assumption that we make is that in a location there are different races as well , due to different clubs organising different races but all within the same location. and a racer can be a member of more than one club.

Design Analysis

From figure 1.1 , we can see the domain of the problem , because there are many organisations and clubs holding different races. it is a many to many relationship to races. The only way to distinguish between each race is the location of where it is held and who it is held by. Each race has many racers , and a racer could run in many races because they can be a member of different clubs at the same time. Thus we can see it can get really messy.

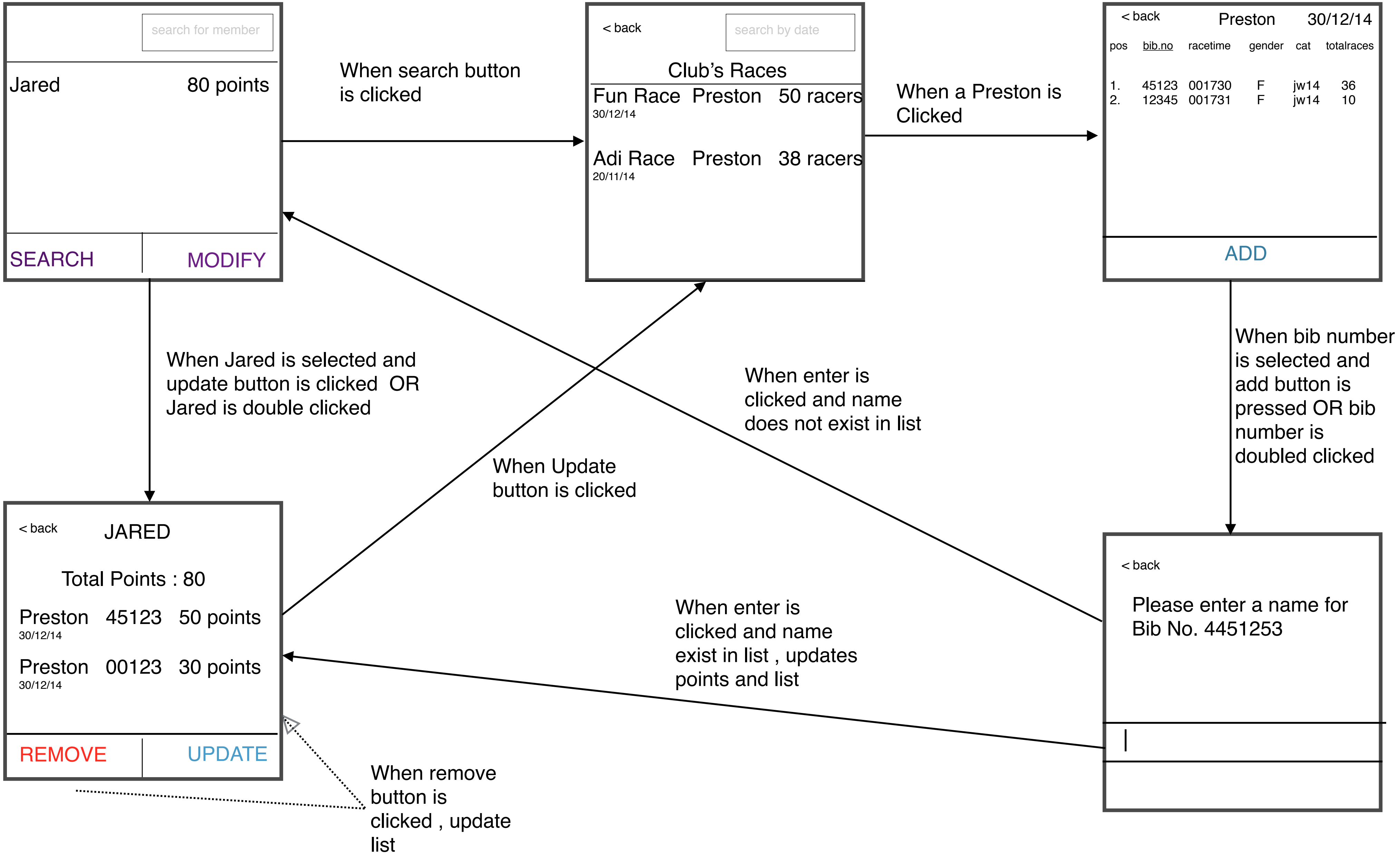
Referring to 1.2 , we can see that the main task for the User of our application is to manage the club championship and the User's goal is basically just to keep track of the of the points system of all the members. The User can basically do two tasks when using our application which is adding members to the member's list to know who is leading the points table to win the Club's championship , and by adding a friend , the application would automatically calculate the points (by checking his gender , category , total number of races and the position that the member achieved in each race) , that user has initially and add/update his points and name to the points table.

the second thing that the user has to do is to update the record that the user has. So basically the user has to modify the data. this encompasses a lot of different scenarios , for example the member got disqualified in a race , thus the user has to modify the data and remove points from the member , or another scenario could be that the member participated in a race that the user did not know about. and when the User finds out , she can still update the points table with no problem.

These 2 tasks are the Goals for the user and what the user wants to use the application for.

Figure 1.0

Virtual Windows



search for member

Jared 80 points

SEARCH MODIFY

When search button is clicked

< back search by date

Club's Races

Fun Race Preston 50 racers
30/12/14

Adi Race Preston 38 racers
20/11/14

When a Preston is Clicked

< back Preston 30/12/14

pos	bib.no	racetime	gender	cat	totalraces
1.	45123	001730	F	jw14	36
2.	12345	001731	F	jw14	10

ADD

When bib number is selected and add button is pressed OR bib number is doubled clicked

When Jared is selected and update button is clicked OR Jared is double clicked

When enter is clicked and name does not exist in list

When Update button is clicked

< back JARED

Total Points : 80

Preston 45123 50 points
30/12/14

Preston 00123 30 points
30/12/14

REMOVE UPDATE

When enter is clicked and name exist in list , updates points and list

< back

Please enter a name for Bib No. 4451253

When remove button is clicked , update list

As from the above figure , we can see that the aim of the design of this application is to provide user friendliness to the user and to enable the user do the task she wants to easily. Having a back button, enables to user to retract what she did , if she accidentally press anything wrong. The purpose of assigning colour to the buttons is because it attracts the attention of the user of where the buttons are and using different colours creates different emotion for the user. For example , red which is usually related to danger or a warning , means you must be careful when you are around the sign. just like this case , once a race is removed , it cannot be retrieved and you must go back to search for it and if the member has no more race , he is out of the competition. thus the remove button is being set as red. Whereas for the other buttons for example, add , update . it is blue and this instills a calm feeling , thus the user would know even if i click this button nothing major is going to happen and it can in-fact help me in the process of what the user is going to do.

(Refer to figure 1.0)
Scenerio 1: User opens application and see that his member list is empty and thus User wants to add a friend.

User clicks the search button , it brings us to a page with all the races the club had ever organise and it is organised by dates and name . And when a race is being clicked , The results are being tabled and put onto a new window that is being displayed. And from that window we can assign the name to a bib number. and it will update the member list automatically showing the members name and points.

Scenario 2: User wants to modify members's races.

User see member list and saw a member that he wants to update , User clicks on the member once , and click on the update button. a new window containing the member records appear , his/her races that he/she had ever participated in. In the new window , user sees two buttons one to remove and one to update, remove button removes the selected race (which leads to deducting his/her points as well). the update button adds a new race to the member's records(which leads to adding more points to the total score as well).

Scenario 2.1: User wants to remove one of the member’s races.

User would be able to see the remove button and When user removes the members’s race , the total points get updated(deducted) but still remains on the same window in case the user wants to remove more races from the member.

Scenario 2.2 : User wants to add more races to the member.

User would be able to see the update button , which will in turn (by pressing it) open up a window with all the races that the club has ever held. and the User would be able to select a race and assign a name to the bib number therefore adding a race to the member. thus window would return to the member’s records and show the updated points and race list.

Referring back to figure 1.2 , by using the design , we are able to complete all the plans that was stated in 1.2 with minimal room for error. Thus the User would be able to accomplish his/her task easily as we provided the most efficient solution to the problem in the domain (refer to figure 1.1).

figure 1.1

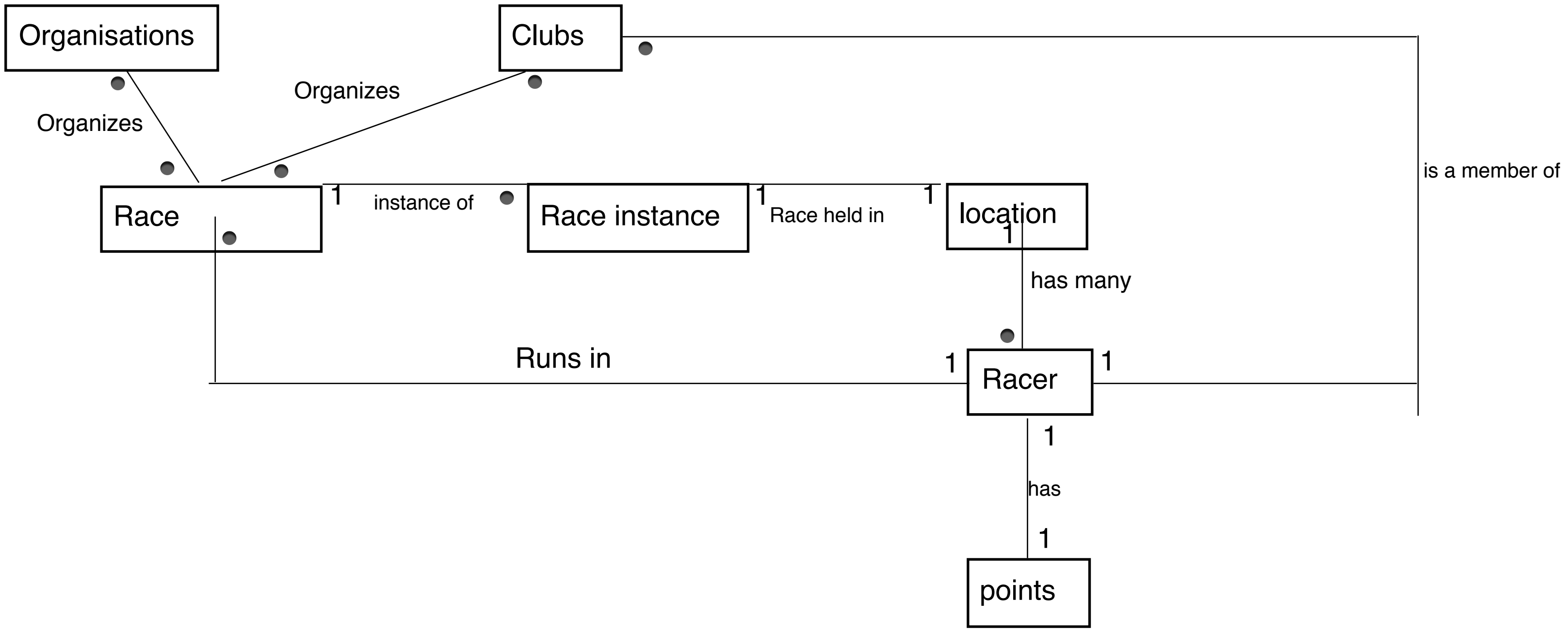
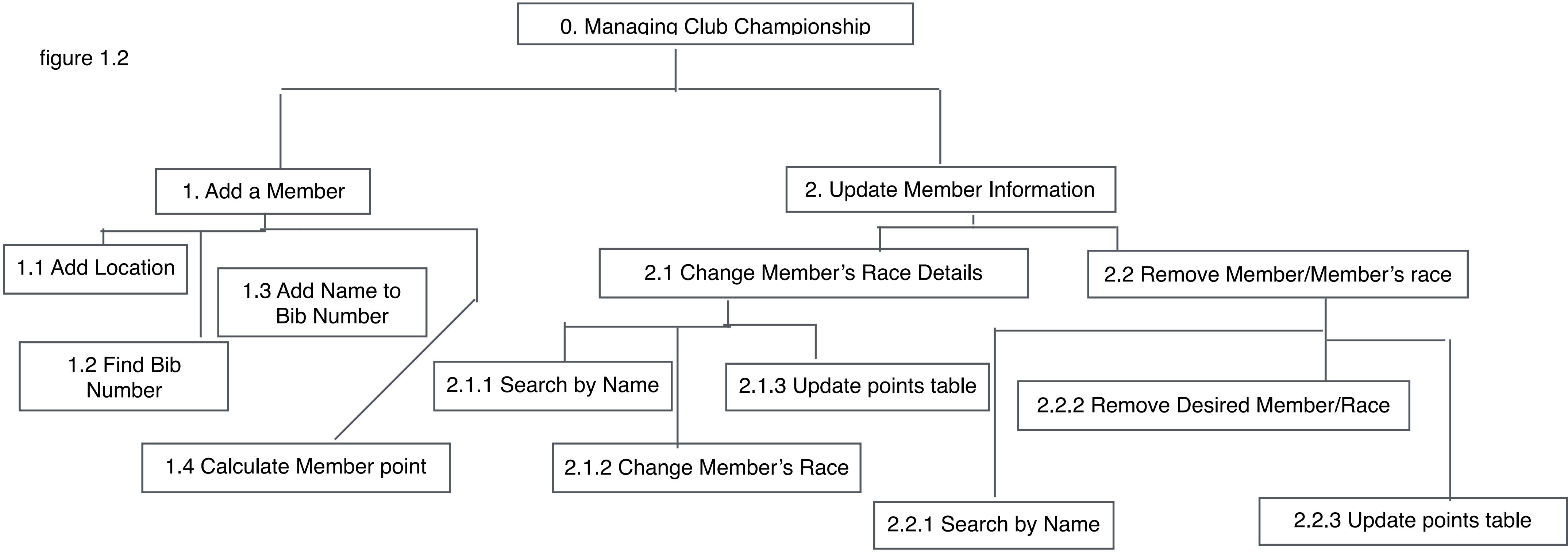

Organisations
Clubs
Organizes
Organizes
is a member of
Race
1
instance of
Race instance
1
Race held in
1
location
1
has many
Runs in
1
Racer
1
1
has
1
points

figure 1.2


0. Managing Club Championship
1. Add a Member
2. Update Member Information
1.1 Add Location
1.2 Find Bib Number
1.3 Add Name to Bib Number
1.4 Calculate Member point
2.1 Change Member's Race Details
2.2 Remove Member/Member's race
2.1.1 Search by Name
2.1.2 Change Member's Race
2.1.3 Update points table
2.2.1 Search by Name
2.2.2 Remove Desired Member/Race
2.2.3 Update points table

PLAN 1 : Do 1 repeatedly , then Do 2 as often as needed.
PLAN 2 : Do 1.1, 1,2 , 1,3 then 1.4.
PLAN 3 : Do 2.1 and/or 2.2 as often as needed.
PLAN 4 : Do 2.1.1 then 2.1.2 then 2.1.3.
PLAN 5 : Do 2.2.1 then 2.2.2 then 2.2.3.

Conclusion

This report has discussed the development of an application design that is used to help organise, display and manage a point system that is held for a club championship competition. The objective of this report is to provide a design that is feasible to implement to an application so that the user of the application could interact with it in a natural way. After analysing the domain and the task i would say that the design that we have discussed would be the most effective in accomplishing the user’s goals.