**Evaluation-**

**If the program can enter an invalid password without causing a crash**  
this works correctly in the program, Evidence 1.3 shows that when the box is filled incorrectly the system catches any errors and outputs a correct error message

**Successful and Unsuccessful Login attempts for admin and users**  
this works correctly, Evidence 1.1 shows us that the software can successfully detect a successful attempt at a login and otherwise shows a message alerting the user of this

**If the user cannot be allowed to leave an empty box during registration**this works correctly, Evidence 1.3 shows us that when we leave an empty box at the registration form we are given an error message alerting us of this mistake

**If the user cannot be allowed to enter a string when entering Distance and time**This works correctly, Evidence 3.18, Evidence 3.19 and Evidence 3.20 shows that entering a string in the distance/time box is stopped by the software

**If the user is not allowed to enter a speed too fast/slow**this works correctly, Evidence 3.14, Evidence 3.15, Evidence 3.16 shows that when speed too fast/slow that it is erroneous will not be allowed to be logged since it is outside of extreme values

**Disallow athletes from editing each other’s personal details**This works correctly, Evidence 2.2 shows that the user has no access to account details of other users

**Large Text for Login/Registration to ensure no confusion**Evidence 1.1 can show that I have implemented large font in my design successfully

**Very simple text boxes for the user to interact with**

Evidence 1.1 can show that I have implemented simplistic text box design to aid the user

**The swimming activity log should use very easy to use drop down boxes**In evidence 3.13 you can see the drop down box design was implemented successfully

**The Data viewer should have columns labelled**In Evidence 4.1 it is clear that the columns of the data viewer and labelled correctly to aid user interaction

**The login should only involve a username and a password**I did not supply an image of my login form in my data test since I felt it did not test the software, in 3.2 User Interface though this design feature is shown to be implemented successfully

**Editing user details should require simply typing the details and hitting enter**This works correctly, some of this is evidenced in **Evidence 2.2** although it is hard to accurately show keystrokes through images.

**Allow administrators to sufficiently monitor athletes progress**there is no evidence for this section since it will not be witnessed on the user end of the product, although it does allow administrator to sufficiently monitor athletes progress

**The system needs to feel lightweight and not slow on any machine**all of the system files add up to only 21 MB of data this shows that the program is lightweight to download and when running the program is always at the bottom of processes organised by memory usage

T**he ability for the athletes to enter, store, retrieve and amend their personal details**This works correctly and is evidence with Evidence 1.1 and Evidence 2.2 that the software logs the data and stores it into the database and then that this information can be amended and retrieved on the application main page

**The ability for the athletes to enter, store and retrieve their training information**Evidence 3.12 and Evidence 3.13 and Evidence 4.1 successfully show the program can allow athletes to enter, store and then retrieve their training information

**The athletes should be able to compare their performance with the other runners**  
Evidence 5.2 shows that athletes have the ability to compare their performance with other runners

**The program should select a team**  
Evidence 6.1 shows that the software can determine and build a top 8 team of athletes

**The program should be secure against basic attempts to crack/crash/hack the program**The software is developed with SQL Injection protection, this involves not allowing the users to end the line of an SQL statement and inject their own code, I have stopped SQL injection by not making user input be injected into SQL script but instead only be referenced in SQL script