Connor McGuire

Development Stage

Contents

[Production of Application 2](#_Toc41230117)

[Business Model Code Listings & Internal Documentation 2](#_Toc41230118)

[Get Members 2](#_Toc41230119)

[Person constructor 3](#_Toc41230120)

[Set Skills 3](#_Toc41230121)

[View Model Code Listings 4](#_Toc41230122)

[Use of unfamiliar libraries and/or constructs 4](#_Toc41230123)

[Use of sliders in WPF profile page 4](#_Toc41230124)

[Regex for username 5](#_Toc41230125)

[Regex for password 6](#_Toc41230126)

[Error handling 6](#_Toc41230127)

[Get Profile 6](#_Toc41230128)

[Add profile 7](#_Toc41230129)

[Delete Profile 7](#_Toc41230130)

[Testing 8](#_Toc41230131)

[Unit Testing 8](#_Toc41230132)

[Test Harness 1 – unit test members. 8](#_Toc41230133)

[Test Harness 2 – unit test add player 10](#_Toc41230134)

[Test Harness 3 – unit test Delete player 11](#_Toc41230135)

[Acceptance Testing 13](#_Toc41230136)

[Simply Football Acceptance test 13](#_Toc41230137)

[User Documentation 15](#_Toc41230138)

[Log In window 15](#_Toc41230139)

[Main Window 16](#_Toc41230140)

[Members Window 17](#_Toc41230141)

[Player Profile 18](#_Toc41230142)

[Bibliography 19](#_Toc41230143)

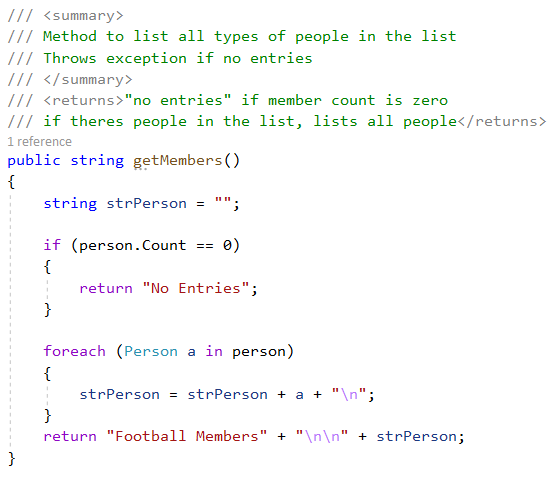
# Production of Application

## Business Model Code Listings & Internal Documentation

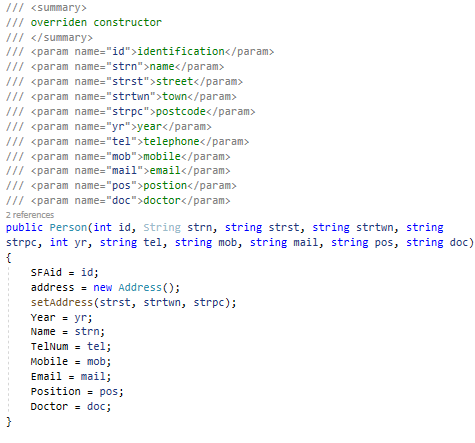
Please view attached files for more examples.

Internal documentation using XML comments, use of appropriate variable naming and use of white space/indentation.

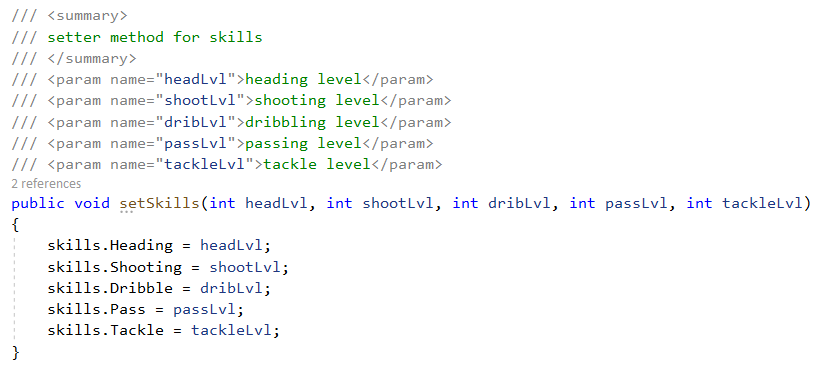
### Get Members



### Person constructor



### Set Skills



## View Model Code Listings

Unfortunately, due to the coronavirus pandemic, I didn’t have the timeframe to code the WPF files to the console application, I’ve coded the basic navigation between each window.

If you refer to the solution planning, you will see the layout of each page or alternatively, view the attached WPF file.

## Use of unfamiliar libraries and/or constructs

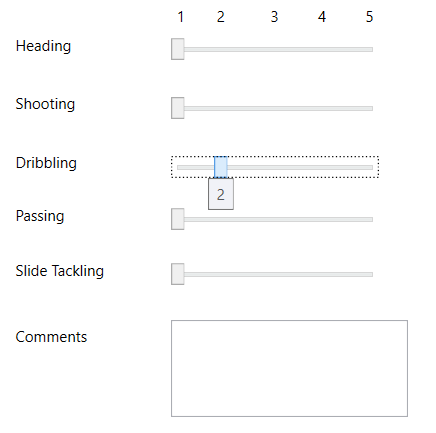
As I said, due to the coronavirus pandemic, I couldn’t fully implement the unfamiliar library’s/constructs to my project as I would’ve liked too. Below is just a preview of what I could get done under the time restrictions and how I would’ve gotten it to work

### Use of sliders in WPF profile page

The user would move the slider along to their chosen skill level, then that would get passed into the user’s skill level for that particular skill. I was able to modify the XAML for player profile WPF to have it show the users selected number on the screen and dynamically change as they moved it.(Sexton, 2014)

If I had time I would’ve used the system.windows.controls library.

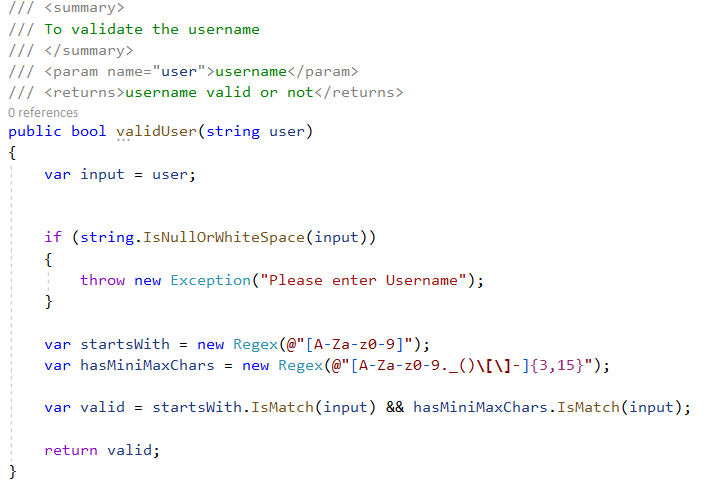




### Regex for username

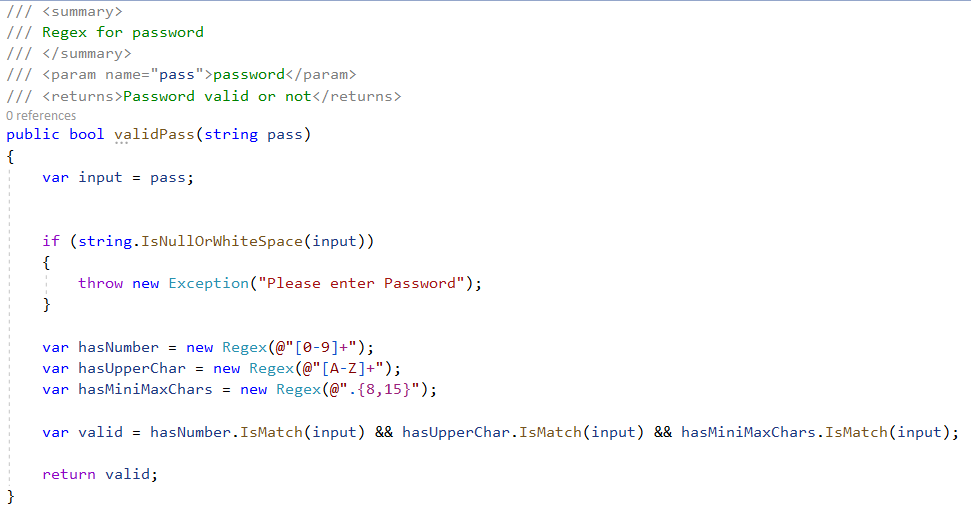
I was able to code the regex for the username input but again, never had the time scale to implement it or test this code. (Marlon, 2010)





### Regex for password

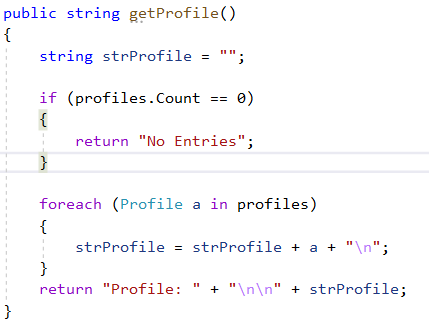
Just the same as username, never had timeframe to implement and test code.



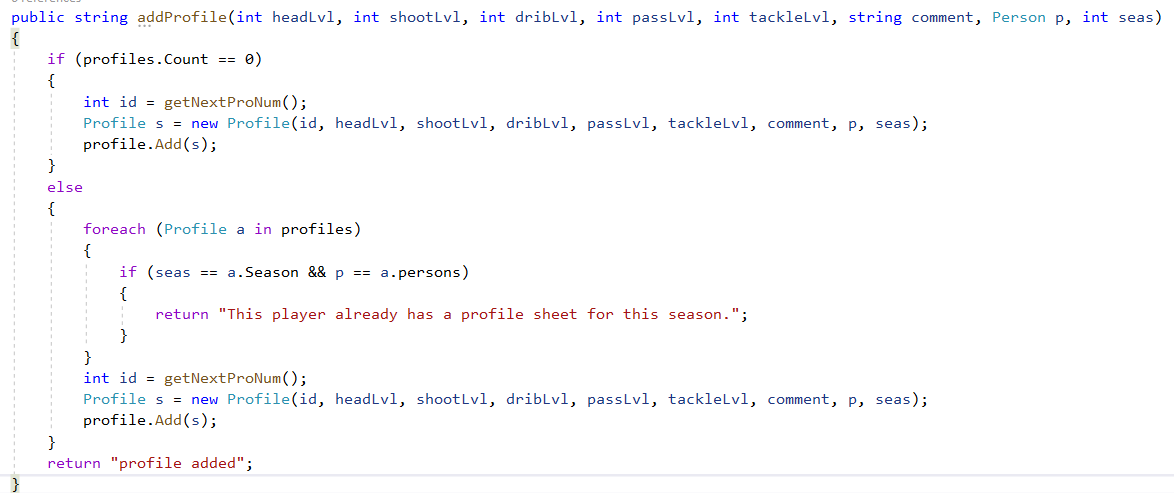
## Error handling

Please view attached files for more examples.

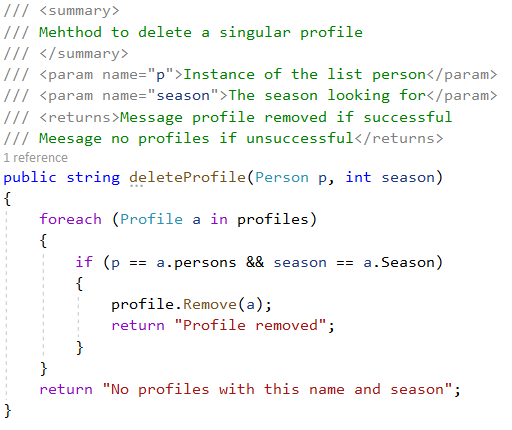
### Get Profile



### Add profile



### Delete Profile



# Testing

For the testing of the simple football application, I will be using two testing strategies, acceptance and unit testing.

## Unit Testing

Unit testing was a great way of testing each individual unit of code to ensure each unit was working as I expected it too. I created my own test harnesses to make sure the code was working to the specification before I moved onto another unit test. This testing was easy to conduct as It was simple and broken down, it allowed me to overall have less errors when the system was finally finished as most errors were already established and fixed.

I used testing harness’ to conduct the unit testing.

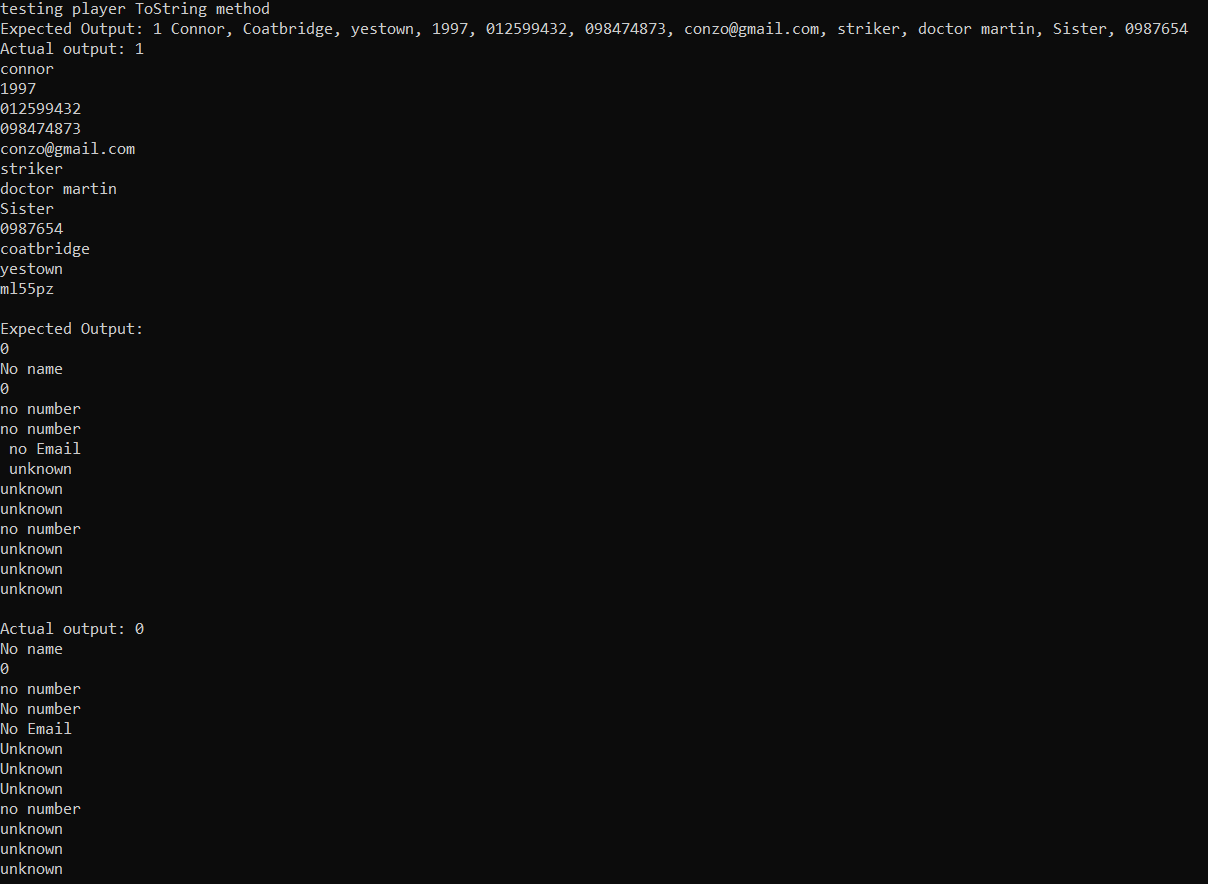
See attached file for more examples.

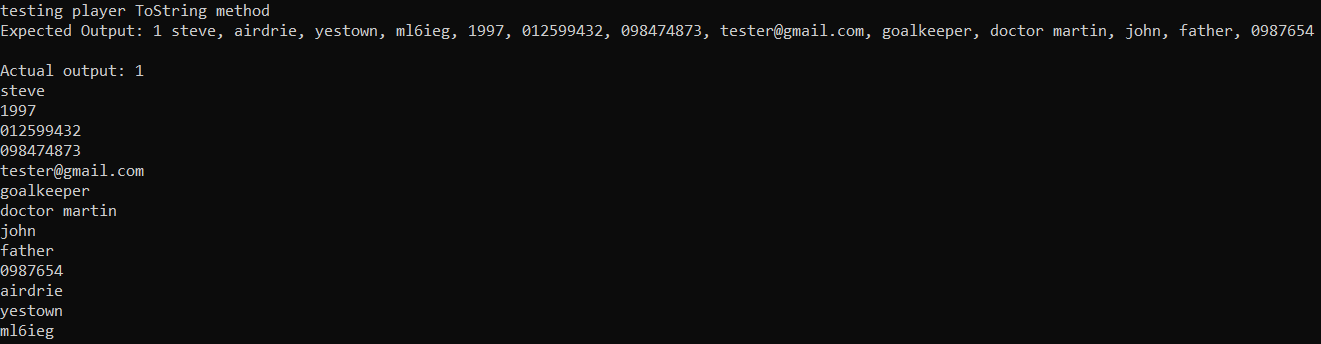
### Test Harness 1 – unit test members.

This test harness is to make sure that the players attributes are all being accepted and going into the appropriate variables. Same test for the youth.



Console Output:





Date Test Run \_\_\_\_\_\_3/05/2020\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Test Passed / Failed \_\_\_\_\_\_\_\_\_Passed\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

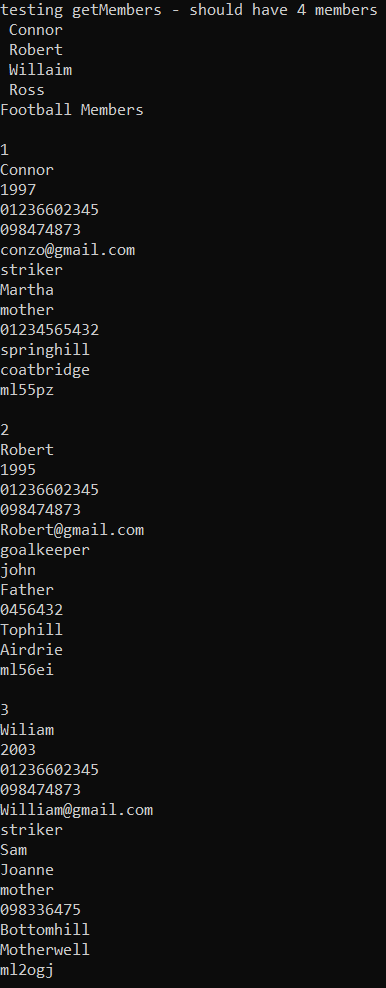
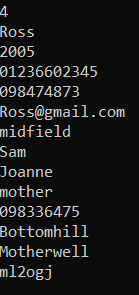
Name of tester \_\_\_\_\_\_\_Connor Mcguire\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### Test Harness 2 – unit test add player

This test harness ensures that’s the “add player” method takes in all parameters for player and youth then adds into to the person list. Will also check that getting all members works.



Console Output:

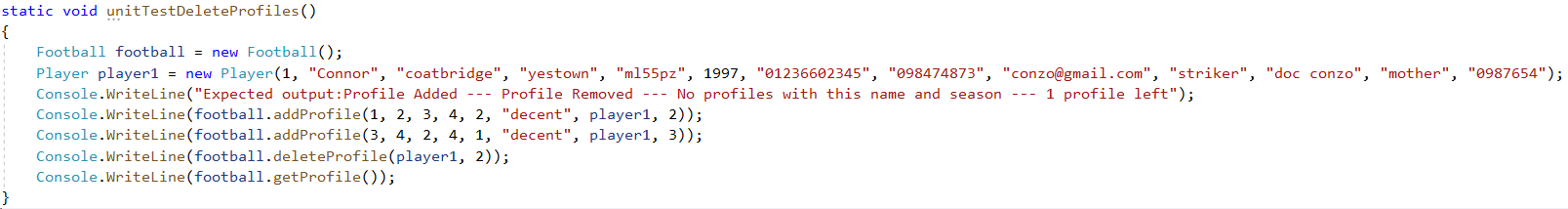
Date Test Run \_\_\_\_\_\_5/05/2020\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Test Passed / Failed \_\_\_\_\_\_\_\_\_Passed\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

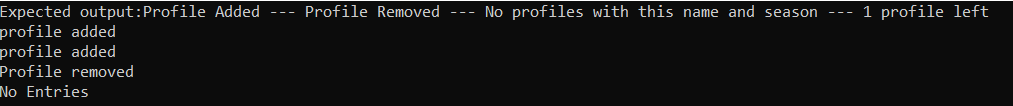
Name of tester \_\_\_\_\_\_\_Connor Mcguire\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Test Harness 3 – unit test Delete player

This test harness will add a player and two profiles, then try and delete one profile and show that there only one profile left in the list.



Console Output:

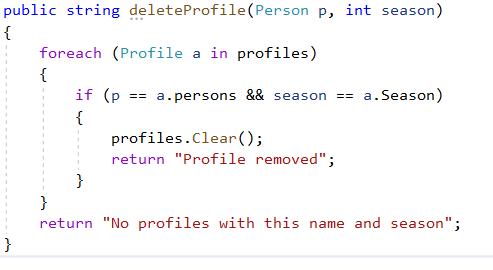


Date Test Run \_\_\_\_\_\_6/05/2020\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Test Passed / Failed \_\_\_\_\_\_\_\_\_Failed\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

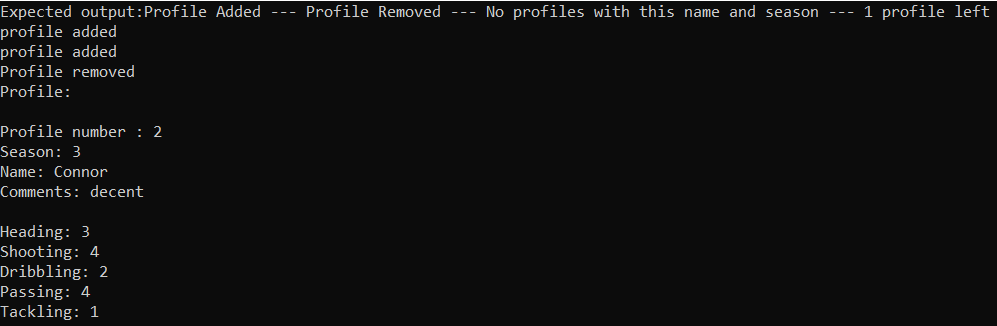
Name of tester \_\_\_\_\_\_\_Connor Mcguire\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

This unit testing failed as when one file was meant to get removed, it would remove all profiles inside the profile list. From the snippet bellow you can see that it states “profiles.Clear()” which was causing the problem. I changed this to “profile.Remove(a)” and the unit test passed.





Console Output:



Date Test Run \_\_\_\_\_\_6/05/2020\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Test Passed / Failed \_\_\_\_\_\_\_\_\_Passed\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name of tester \_\_\_\_\_\_\_Connor Mcguire\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Acceptance Testing

The other strategy I used was acceptance testing. This testing is a checklist with a pass or fail parameter. Test will have an “expected output” and “Actual output” fields. If all tests pass, then the full testing will be completed and passed but if one thing fails then the fixes will have to be made and retested from the beginning. This is a great way of using tests at the end of a programme to determine how well the development has panned out and how it functions together as one. Comments are also optional for each sub test.

The things I looked out for when planning this was too make sure that members/profiles were getting added into the appropriate place, that variables were getting inserted to where they were meant to go. I also testing the deletion of profiles and the listing of players/youth and profiles.

### Simply Football Acceptance test

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Description | Expected Result | Actual Result | Comments | Test Pass/Fail |
| 1 | Add Member  Add a Player with "Connor", "springhill", "coatbridge", "ml55pz", 1997, "01236602345", "098474873", "conzo@gmail.com", "striker", "joanne", "mother", "01234565432" | “Player added” output | “Member added “output | Appropriate message | Pass |
| 2 | Add Member  Add youth with "steve", "airdrie", "yestown", "ml6ieg", 1997, "012599432", "098474873", "tester@gmail.com", "goalkeeper", "doctor martin", "john", "father", "0987654" | “Youth added” output | “Youth added” output | Appropriate message | Pass |
| 3 | Request Get all members and print in a list. | Youth member shown, Player shown | Youth member shown, Player shown | Full person list was shown | Pass |
| 4 | Request Get all players and print in a list. | All Players shown | All Players shown | All players from person list shown | Pass |
| 5 | Request Get all youth and print in a list. | Youth members shown | Youth members shown | All youth players from person list shown | Pass |
| 6 | Request delete profile player1, season 2 | Player1, season 2 profile is removed | Player1, season 2 profile is removed | Worked as expected | Pass |
| 7 | Request delete profile player1, season 2 twice | Should delete first profile then error message “Profile doesn’t exist” | deleted first profile then error message “Profile doesn’t exist” | Appropriate message | Pass |
| 8 | GetMembers with no members in person list | Error message ”No players” | Error message ”No players | Appropriate message | Pass |
| 9 | GetProfiles with no profile | Error message ”No profiles” | Error message ”No profiles” | Appropriate message | Pass |
| 10 | Add profile with player 1, season 2 twice | Should add the first profile the error message  “Profile for this player and this season already exists” | Should add the first profile the error message  “Profile for this player and this season already exists” | Appropriate message | Pass |
| 11 | Find profile with no profiles | Text output “No entries” | Text output “No entries” | Appropriate message | Pass |
| 12 | Find profile with name “connor” season 2 | output Profile for “Connor” Season 2 | output Profile for “Connor” Season 2 | Appropriate message | Pass |

Date Test Run \_\_\_\_\_\_\_\_18/05/2020\_\_\_\_\_\_\_\_\_\_\_\_

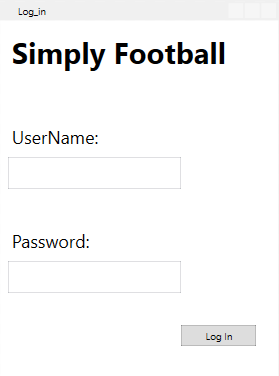
Test Passed / Failed \_\_\_\_\_\_\_Passed\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name of tester \_\_\_\_Connor McGuire\_\_\_\_\_\_\_\_\_\_\_\_

# User Documentation

This application is made to be an easy way to track and document user’s information and player profiles for the club simply football. After downloading and installing the app, you’ll need a username and password being a coach or parent. After this, you are free to roam the app as you please. (Kumar, n.d.)

## Log In window

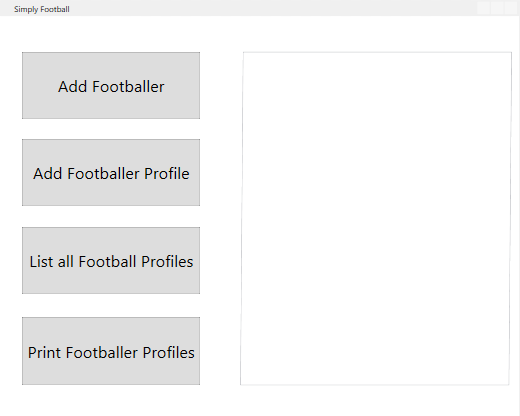


Username input

Password

Log in – Click this button when entering a correct username and password to be taking to the main simply football window. If your user and pass is wrong, you’ll be prompted to enter it again.

### Main Window



Output box

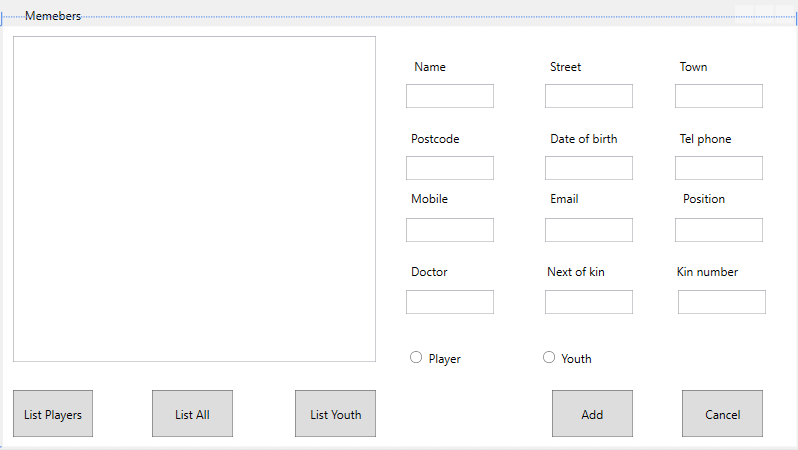
Add footballer – This will take the user to the football player input screen, where you can add, list all, list youth player or list regular players in the team.

Add football profile – This will take the user to the football profile creation screen where they can enter various skill levels for each skill, select a name from the list of players and season to create new profiles They can also delete or find a profile.

List all football profiles – This will list all profiles for simply football in the output box to the right.

Print football profiles – This will also list all profiles and will print these to a separate file outside of the app.

### Members Window

****

Radio buttons

Input for player

Radio buttons – Can switch between Regular player and youth player

Input for players – This is where you would add in each players info

List players – List all players in simply football

List all – Lists players and youth

List youth – lists all youth players in simply football

Add – Once Player info is filled in then this will add the player to either the players list or youth list depending on which radio button is selected.

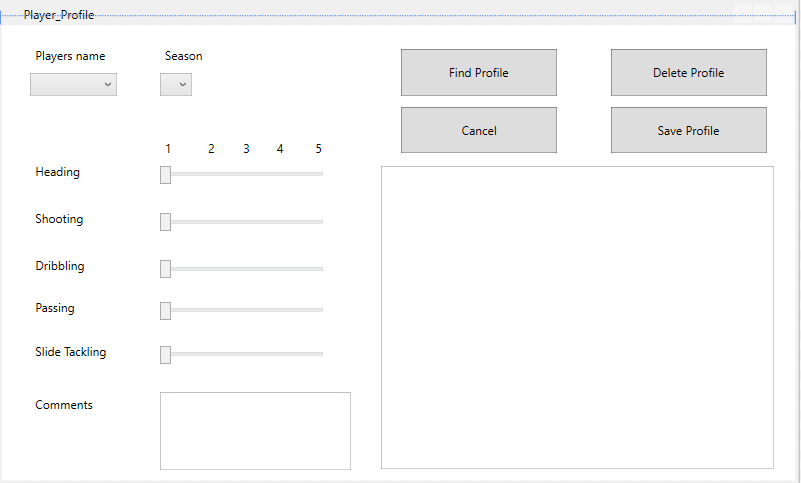
Cancel – takes back to main page.

### Player Profile

Season selection

Sliders

Player Selection buttons

****

Player selection – Where you can choose which player from the list at simply football, to add a profile sheet for.

Season selection – Where you can choose which season the profile is for.

Sliders – Can choose on a scale form 1-5 (need improving – Excellent) for each skill move for a player.

Comments – Can add any comments needed for the profile.

Find profile – Select a player and season, if there is one, then it will show in the text box, if not it will simply say not found.

Delete profile - Select a player and season, if there is one, it will delete that specific profile, if not it will simply say not found.

Cancel – goes back to main window.

Save profile – If all sliders and comments are filled out, it will add a profile for that specific name and season selected. If profile exists, then it will simply say “Profile already exists” in the text box.

# Bibliography

(Marlon, 2010) *–* StackOverFlow - <https://stackoverflow.com/questions/3588623/c-sharp-regex-for-a-username-with-a-few-restrictions?rq=1>

(Sexton, 2014) *-* <https://wpf.2000things.com/tag/slider/>

(Knot, n.d.)-- <https://www.techsmith.com/blog/user-documentation/>

(Kumar, n.d.) - <https://helpiewp.com/user-documentation/>