CRITERION D

Overall the program was considered a success seeing as all the success criteria were met. Criteria 1 and 2 from the success criteria are both met as when the program is run the program loads up a Menu GUI using the launcher class given the user the option to either Sign up or Log in. My client was very happy with this as he specifically asked for the login feature (Appendix 4). On top of this, the user has the option to go to an information page to find out more about the program, seeing as when the client was met for the penultimate meeting (Appendix 3) he suggested that some additional information would be nice therefore he was very pleased to see some guidance through the program. In addition to this, criterion 3 is met as on the GUI's there is an option to either go back a step or cancel the process. My client also liked this very much as he felt that users should be able to change their mind and go back if they wanted to (Appendix 4). Furthermore, criterion 4 is met as when the user completes a scorecard they are assigned a handicap. This was the most pleasing aspect for my client as his main need from the program was for it to generate a handicap for the user (Appendix 4). After the user logs in using their username and password, they can access this handicap and edit it which also meets criterion 5 and 6 as the user is easily able to log in through the login page and access their details. This was something my client was also happy about as ease of use was very important to him. He thought that the program flowed well and had a clear direction; this was vital because if the program did not flow as easily, users may not want to use the program at all (Appendix 4). Additionally, criterion 7 is met as on the Homepage as there is an option to change handicap, age, password or username. Knowing that a golfers handicap often changes whilst they continue to play, my client was very pleased that by entering scores, the user's handicap would change (Appendix 4). Finally, the program meets criterion 8 as the program works so that when the user decides to change their handicap, the program works it out from them by taking their score and the par of the course to calculate their new handicap. My client was particularly pleased that the system conformed to real-life CONGU guidelines for this part (Appendix 4). This means that all the criterion are met therefore the project was a success, in addition to this the client's needs were all met and he was very pleased with the final product that was produced. Despite this, there was some room for improvement in the program as realised by me or pointed out by the client in the final meeting.

Future Improvements:

Despite the program doing the job required there was scope for improvement. My client pointed out that although certain golfers had some specialisation in their account - younger male golfers had a blue GUI Homepage and younger female golfers had a pink GUI Homepage, there wasn't any extra functionality that came with that specialisation (Appendix 3 + Appendix 4). In the future, this could be improved by adding extra behaviours and attributes to the child classes BoyGolfer and GirlGolfer so that the user could have more options as a Boy or Girl golfer as currently, the function is almost identical. Additionally, there was a lack of encapsulation and abstraction in the program, meaning that every class was not completely independent from the other. This meant that by editing some parts of the code they affected other parts which would not be the case if these features were used more wisely in the future. In addition to this, some of the code was untidy and therefore it was hard to read. This is because, it consisted of large chunks of code, few methods and minimal commenting. In the future, it would be better to break up the chunks of code into smaller methods which would have been easier to understand and read. Furthermore, the GUI program was not great, despite the program flowing well, a new GUI was created each time a button was pressed. In the future, I would like to have one GUI that is able to change state therefore a new GUI object would not have to be created for every button that is pressed. Finally, the final product was presented in a pop-up form that could run on a computer however my client suggested that it may be better to design the program as an app (Appendix 4). Therefore in the future, it would be better to use android studio and an emulator in order to create an app for the client.

Despite these improvements listed above, the final product met all of the success criteria as well as the client's needs therefore the program was overall a success.

843 words

Final word count: 1,938