KIAN GAULT

sTUDENT ID: 2019222 |

Evaluation Report – GRADED UNIT

HND: SOFTWARE dEVELOPMENT – WLC

Table of Contents

[Introduction 2](#_Toc136006785)

[The Original Project Plan 3](#_Toc136006786)

[Using Microsoft Project to create a project plan with a Gantt Chart: 3](#_Toc136006787)

[Analysis of The Client’s Original System 4](#_Toc136006788)

[Initial review of the client brief and supplied documentation: 4](#_Toc136006789)

[Review of the paper-based forms: 4](#_Toc136006790)

[Initial meeting with the client: 4](#_Toc136006791)

[Initial Design Phase of The Proposed System 5](#_Toc136006792)

[Creating use cases / use case diagram: 5](#_Toc136006793)

[Planning Phase 6](#_Toc136006794)

[Creating the Software Requirements Specification Document: 6](#_Toc136006795)

[Self-Research for the project: 7](#_Toc136006796)

[Design – CRC Cards: 8](#_Toc136006797)

[Design – Sequence Diagrams: 8](#_Toc136006798)

[Design – Activity Diagrams: 9](#_Toc136006799)

[Design – Class Diagram: 9](#_Toc136006800)

[Design – UI storyboards: 10](#_Toc136006801)

[Implementation Phase 11](#_Toc136006802)

[Implementation of the business model: 11](#_Toc136006803)

[Implementation of the view model 12](#_Toc136006804)

[Use of unfamiliar libraries 13](#_Toc136006805)

[Validation and Error Handling: 14](#_Toc136006806)

[Testing 15](#_Toc136006807)

[Test Plan: 15](#_Toc136006808)

[Test Logs: 15](#_Toc136006809)

[Documentation / Project Log 16](#_Toc136006810)

[Implementation report/user-manual 16](#_Toc136006811)

[Diary logs: 17](#_Toc136006812)

[Conclusion 18](#_Toc136006813)

# Introduction

This document is the official evaluation of the graded unit project for HND: Software Development. This document will highlight each section of this project and I will offer my subjective evaluation on my performance.   
  
The sections to be evaluated will be how I analysed and designed the initial project plan, how I actually followed this plan and what changed, evaluating how I analysed the proposed system from the client brief, how I planned the design with the use of diagrams and analysis, evaluating how the system turned out in the final result and finally how well the testing process and documentation went.   
  
I will also acknowledge any skills I gained be that from material learned over the course or new methods I learned through self-research/learning.   
  
Future evaluation will also be included on any areas I felt could be improved in time that I documented during the planning or implementation phase.

# The Original Project Plan

## Using Microsoft Project to create a project plan with a Gantt Chart:

For this project, we were tasked to design a full project plan that would cover each stage. This included a visualisation of the plan with the use of a Gantt Chart.

Subjective Evaluation:

* I believe I could have done a better job with this project plan. I didn’t design the plan and dates with the agile methodology which I had decided to use. I also believe this was due to my lack of knowledge on how this project would turn out, like how broken down each phase of the project could be so most of the milestones/tasks are more generalised.
* I do believe I made good use of the human resources system within MS Project which allowed me to assign fictional characters to be in place for the different roles that would be in a development team so a developer, tester, project manager. I then assigned the different tasks I had created then added these members to different tasks and made sure there was no conflicts.

Feedback received from the client:

* The client would have preferred if the different tasks on the project plan were separated into smaller sub tasks for more clarity with deadlines which I agree should have been included. This would of giving a clearer visualisation within the gantt chart on when tasks for each sub section would start and end.
* The lack of agile was also critiqued by the client which I also agree with. The plan submitted to the client was a waterfall methodology which wasn’t the desired plan. There should have been multiple check-up meetings with the client to constantly track the quality of the work.
* The client mentioned how there was little room for any slippage as each task started after the previous task ended. I can see how this is an issue because unforeseen errors might add delay to the current task and in the current state, the plan would fail even if one task went behind schedule so I should have included built in contingencies for this.

# Analysis of The Client’s Original System

## Initial review of the client brief and supplied documentation:

This task involved our first look at the client brief and what the client wanted. As expected, the brief was very lacking in detail so extensive analysis would be needed to have a full understanding of the proposed system.   
  
Right away my first idea was to create a list of all the important information that stood out from the brief. This included the two main features of storing personal information and updating a form of skills. This was crucial for my later analysis as I knew which areas needed to be further elaborated by the client.

## Review of the paper-based forms:

These documents contained the current paper version of the client’s management system. This included the different fields of data the club collected in their players and how they differed by age group. A list of the different skills that are tracked was also provided.  
  
This gave me a rough idea of what my actual user interface might look like. It also provided me that more clarity will be needed from the client if these fields would be the final version or if more would be desired.

## Initial meeting with the client:

This meeting served as the first encounter with our client. Here we got to ask any questions that might be needed, and the client would also delivery key information if not asked initially. From this we got information like the need to save data externally between sessions, and the size of the data.

Subjective Evaluation:

* I believe I carried out my initial analyse well. I was effective at taking notes on what the system would look like and extracting information from the brief we were giving. This proved extremely useful for my meeting with the client as I was able to ask for more information on areas that I deemed necessary.
* I do believe I missed out on a few crucial aspects that I should have asked. The main one being security on the system and hardware requirements. This became apparent to me when I started to write my Software Requirements Specification documentation that I noticed key areas had not been clarified by the client because I simply did not ask.

# Initial Design Phase of The Proposed System

## Creating use cases / use case diagram:

This task was our first stage of designing our system from our gathered analyse. This involved first creating user stories which then evolve into use cases which are user stories but more in-depth and focused. This is then all tied together with our use case diagram.

Subjective Evaluation:

* I believe my user stories could have been more in quantity. I only created user stories for the key aspects of the system like managing data and logins, but I never created stories for extra possible features that harmed my final product.  
  I do believe the user stories I created helped me easily create use cases from them as I had already a clear idea of what these cases would need from the information I gathered in my analyse.
* I believe my use cases were very good for the initial version. They had a clear understanding and guideline on how to successfully meet these cases during implementation.   
  Later on in the project these cases did however become outdated as the needs for the system changed over time which is fine because this was an agile project but with more planning the significance of these changes could have been less severe.
* I believe my initial use case diagram did what it was supposed to but could have been broken down into smaller sub use cases. For example, a use case should of have its own image and link to an actor than a more condensed version. My original idea was for simplicity, but the client was more confused than understanding of my plan.   
  I do think the way I grouped the use case diagram to admin use cases and coach use cases was great as it clearly visualised to the viewer what use cases was associated with what actor.

Feedback received from the client:

* Overall the client was very happy with my use cases/use case diagram. They did agree with my evaluation that it would have been preferred to have more use cases even if the task was really small.

# Planning Phase

## Creating the Software Requirements Specification Document:

This document was created after the first meeting with the client. Within this document held the main functional and non-functional requirements for the system so this file was extremely important throughout the whole project.   
  
Once finished the client would be presented with the document to ensure both the client and developer are satisfied with the listed requirements for the system. This ensures both the client and developer have a clear understanding of what the end result should be.   
  
Subjective evaluation:

* I do believe my SRS document was well written. It clearly listed each functional requirement alongside sub requirements to the main requirements. This is also reflected within the use cases for each of these requirements. This was extremely useful during development because I had already set conditions on what these features should implement.
* However I could have put the functional requirements in a table format to make it simpler when viewing. I could have also included more information for the sub sections of the SRS document like the non-functional requirements. This was also due to the fact I failed to gather all the information I needed from the client in our initial meeting, so I wasn’t able to give a fully detailed explanation on most non-functional requirements until I met the client again.

Feedback received from the client:

* The client enjoyed my SRS document when presented to in our second meeting. They did however want the font size changed to normal APA standards for reports and the size. This was useful feedback as this was only an initial draft at this stage.
* The client also believed I showed a clear understanding of the brief they presented initial within my requirements for the system. They did however wish the requirements were in a table format with unique IDs which I agreed with. This would have helped me later down the line with my testing.

## Self-Research for the project:

A main and crucial part of this project was the use of self-research. This was obvious after the first interview because the client wished for the data within the application to be saved across sessions.   
  
This idea of saving data was new to me so I knew I would need to do in-depth research on the subject. This resulted in me discovering the use of SQL imbedded within WPF applications. This system was called SQLite and I found great tutorials online explaining how this system worked with WPF directly and was crucial in creating the base design for my application.   
  
Subjective evaluation:

* I believe my research was extremely efficient as I found theory and practical tutorials for SQLite which helped me understand the concept’s knowledge and functionality. This research would be crucial on my projects final result which was as expected.
* I do think I could have perhaps researched other concepts more in-depth, like the use of SQL Server as I was more set-on SQLite fast and never really contemplated on if it was the best option.

Feedback received from the client:

* The client liked the use the integrating SQLite into the application. They enjoyed how extendable the possibilities are from using SQLite because it’s embedded into the program itself instead of being external like a server.

## Design – CRC Cards:

These designs were index cards that described the different classes that would be in the system. These would extend over to our class diagram that showed the relation between the classes.

Subjective evaluation:

* I believe I could have been more extensive with my CRC cards as some lacked great detail. This was due to my class diagram having more information than it needed so I felt like I was repeating myself, but the great detail should have been on these cards not the class diagram.

Feedback received from the client:

* The client wasn’t happy that no conceptual object modelling diagram was produced prior to the CRC cards creation. This would have helped with creating them much more which I agree with.
* The client did like the CRC cards produced which described the behaviours and contributions from other classes but would have preferred all relationship describes to be removed and instead let the class diagram explain these. I agree that this would have made both CRC cards and Class diagram have more individual value instead of both repeating one another.

## Design – Sequence Diagrams:

These diagrams show the sequence of events for our proposed system. This diagram is crucial for understanding the different functions/methods of our program and how they interact with the other classes of the system and the overall flow of the processes.

Subjective evaluation:

* I believe my sequence diagrams turned out good for the initial stage. I was able to use these sequence diagrams in my implementation and for the most part stayed accurate to these diagrams. These diagrams are what gave life to my idea of tracking the user’s current role through the system to set permission levels for the user which turned out great in the final build.

Feedback received from the client:

* The client agrees that the sequence diagrams result was clear and concise visualisations. They would have wanted more fragment notations within the diagrams to show the different alternative flows which I agree with.

## Design – Activity Diagrams:

These diagrams were created to show the flow of the different use cases created for the system. These are crucial to refer back to in development as they show the different paths a task might take.

Subjective evaluation:

* I believe my activity diagrams turned out the best out of all my diagrams. They are incredibly simple in design but extremely insightful visualisations.
* I created a good size of diagrams that helped show the flow of different tasks and I also included branching activities that lead into different activities which was great at showing how far a task could go.

Feedback received from the client:

* The client agreed that these activity diagrams were excellent and showed all the precision steps the user could take in a task. The client also enjoyed that the diagram featured all accurate UML notations.

## Design – Class Diagram:

This diagram was created to show the inner relationships of each class plus they also show the inner workings of the class. These are split into the attributes and behaviours.

Subjective evaluation:

* I believe I transitioned from the CRC card to the class diagram well. I designed the class diagram with an object-oriented programming mindset. This saw me include an interface with the use of inheritance and abstraction.
* I don’t believe I designed the class diagram with the idea in mind that this design could change dramatically later on. This saw my final application’s classes being extremely different from this original class diagram which wasn’t good. I should have returned to this diagram throughout the implementation phase and update the diagram to follow the agile methodology.

Feedback received from the client:

* The client liked the idea of inheritance within the system with the admin and coach classes inheriting from the member class.
* The client did not fully see the inheritance however through the visualisation. I agree a different method could be used to highlighting the methods which are classed as abstract and what methods actually override the method.

## Design – UI storyboards:

These diagrams were produced to provide an accurate visualization of the final user interface for each window on the program. These storyboards show the overall flow of the system and its functionality.

Subjective evaluation:

* I believe my storyboards were really effective in both explaining the functionality of each window and how the data would be presented to the user. The designs created in these storyboards heavily reflect the final product.

Feedback received from the client:

* The client though each storyboard was clear and concise. They appreciated the inclusion of descriptions for each UI element. I agree that this helps add clarity to what each UI element is responsible for.
* The client would have liked to see a prototype of these storyboards with added colours and buttons. I agree that this would have helped the final design more however this idea was not finished due to time constraint.

# Implementation Phase

## Implementation of the business model:

This section of implementation saw the development of the back-end program. The client wished for a management system to be created replacing their previous paper-based system so this section was converting that into software.

The main features that had to be included were, the ability to add a new player, edit a player, delete a player, and add skill ratings to players. The final build I created has these features and more with the ability to search for specific players.

Additionally the data entered has to be saved in between sessions which I was successful in achieving through the use of SQLite.

Finally the client wished for the system to follow the guidelines of GDPR which this final build achieved. This is done through the login system which identifies the type of user on the system giving them permission access to that role’s functions. This meant that admin and coach users had different experiences which was intended.

Subjective evaluation:

* I believe my final build of this application met the expectations of the client. This is because I met all the requirements originally set out in the planning phase of the project and even added more functions in.
* I do think I could have focused more on adding each pillar of Object-Oriented Programming into the system. My original class diagram had featured inheritance, but this idea was later scrapped due to being pressured by time. The use of inheritance could have reduced the need to repeat code which would have been much better.

Feedback received from the client:

* The client really liked the system overall, they liked the design of the database as it portrayed a professional manner which they would have wanted for their business which was my idea in mind when designing the layout.
* They were very happy that all requirements were met like adding, editing, removing, and seeing emails of all players. They would have liked the ability to copy the selected email from the list which I agree should have been featured here.
* The client would have preferred of the structure followed the use of inheritance originally stated in the class diagram which I agree with. This could have removed duplicated code in some areas and helped any future updates to the system.

## Implementation of the view model

This section saw the design and implementation of the user interfaces of the system. These designs were originally designed as storyboards in the planning phase, so the base skeleton design was already created.

Later in research I discovered the UI element ListView so this was later swapped into the user interface instead of regular lists. The ListView allowed the use of data binding which was one of the key features of the user interface.

Subjective evaluation:

* I believe my user interface was really strong and one of my driving positives in the system. This was due to the ListView element which drives the whole functionality of the database. The ability to select a record and have the text boxes be populated automatically allowed for seamless editing of existing records.
* I also believe my use of different UI elements made the entry of data a smoother experience. This is due to the use of datepicker for dates and combo boxes for a selection of pre-set values for skills.

Feedback received from the client:

* The client liked the use of a ListView to group the records together like most database applications. They would have liked the ability to sort the players alphabetically however which I agree should have been a feature.
* The client wished for the coaches to be the only user that edits the player skill profiles which I agree I missed in the brief discussion about these permission access levels. This does work that the coach can’t creating any player records only view their team’s information.

## Use of unfamiliar libraries

For this project, we had to use some kind of library in order to store data externally. Using my coding knowledge I knew SQL would be the resource used and with further research I discovered SQLite which is a package combined with entity framework that embeds a database file into a WPF application allowing the application to store and read data from it. This allowed me to use SQL commands to store the data from the application into this file which saves the data between sessions.

Subjective evaluation:

* I believe my use of the unfamiliar library was great in this application as it was something new to me and integrated it well into the system.
* I do think this could have been extended over to my login system so the data there was also stored with the use of SQLite instead of being hard coded.

Feedback received from the client:

* The client would have much preferred the login system have the SQLite integration as well which I agree should have been added but didn’t due to time constraints.
* The client really enjoyed the use of ListViews and SQLite integration in the application.

## Validation and Error Handling:

This section of the application was for creating a robust system with good exception/error handling in the system.

Subjective evaluation:

* I do believe this was my weakest point within the build. I could have put more focus on error handling with invalid data types as the base version only handles length exceptions.
* I do believe my use of combo boxes for skill profiles was good as this removed the need to add error handling code as it was virtually impossible for the user to select or enter invalid data.
* Additionally with the use of datepicker I was able to pre-set the range of dates to pick from determined by the current team level. This allowed me to validate that correct ages were selected for the correct levels because only the ages valid were visible to the user.

Feedback received from the client:

* The client agrees that this section could have been better, particularly with validating numerical values. The email entries should have been checked to see if they were valid emails which I agree with.
* The client would have liked to see the use of exception handling around database actions because these can sometimes present unseen errors so it’s better to catch these instead of the system crashing which I agree with should have been added.

# Testing

## Test Plan:

This document was created to serve as my overall testing process for this project. Within the document it would discuss the approaches I took while testing and what external tools I would use.

Subjective evaluation:

* I believe my test plan was effective and concise on clearly explaining my testing process for the application. This was present in my traceability matrix that I created which tracked each requirement from the software requirements specification document to make sure each requirement/feature worked correctly.

Feedback received from the client:

* The client liked the test plan also because the plan is clear in outlining my strategy and methods of testing. The client however would have liked to see sample test cases within the test plan itself as in most cases the person writing the test plan isn’t doing the test logs. I agree this would have been better to add into the test plan.

## Test Logs:

These documents are logs of different types of testing which have been carried out to test the application. These are data input validation logs, event validation logs, navigational logs, and functional acceptance logs.

Subjective evaluation:

* I believe my test logs were short and efficient in recording the results of my testing. They were organised into a table format which helps understand them easier.
* I could have perhaps added in more comments after each test to further understanding on how the test was a success or failure.

Feedback received from the client:

* The client liked most of my test logs but would have rather my functional acceptance testing log follow a more use case approach with testing each functionality like adding a member, editing a member, and deleting a member. I agree that this would have made the log better and make sure the functionality was matching the requirements.

# Documentation / Project Log

## Implementation report/user-manual

These two documents served as the official documentation for the application I created. The report focused on the more background aspects of the implementation phase and the user-manual served as a step-by-step guide on how to use the application.

Subjective evaluation:

* I believe my user-manual was written very well as I included screenshots alongside descriptions for each feature of the application.
* I think my report could have been more accurate to the final design of the program as some things are mentioned like abstraction within the program when this isn’t fully the case as its only partial abstraction.

Feedback received from the client:

* The client really enjoyed the user-manual especially with the use of screenshots alongside the descriptions which was my idea.

A black and white text on a black background

Description automatically generated with low confidence

## Diary logs:

This document served as my overall log of each session that I worked on this project. The format was sorted by date and is followed by a short description of what I did that day. This shows the overall time-frame that I worked over the course of the project.

Screenshot of table of contents. (Full document will be included within the submission for completeness):

# Conclusion

In conclusion I believe I did very well in this project. The project was extremely challenging at times, especially with new concepts like learning how to use SQL within C# programming but my use of self-researched made sure I managed to do this.

Planning Phase:

I believe my planning phase was really successful. I provided good analyse of the client brief accompanied by diagram designs for the system. I do think I should of went back to these designs later on in the project to update them as this would have followed agile methodology more.

Implementation Phase:

I again believe this phase was successful also as I provided a final build that matches the brief of the project with the ability to add a member, edit a member, and remove a member. Additionally I added in extra functionality like searching for a player by ID.