**08220 – Project Conduct Report.**

**Aim of project.**

The aim of this project was to, working as a group, create a piece of software that the business ‘Wynne’s tours’ could use to help run their business. The company run coach tours from their depot in Cardiff, to all around the country. A number of the company’s customers pay an annual fee to become gold members, meaning they receive discounts on tour tickets and newsletters informing them on upcoming tours. Previously the agents at the company had to record bookings for the tours using paper forms, which were then copied to give to the drivers so they knew how many passengers to take and return. Management information, such as finding what tours are popular or not, were also revised from these forms which was found to be slow and tedious work. Using this technique of managing bookings and information it was not uncommon that errors were made.

Due to these errors the company requested for our team to create a piece of software that is capable of recording all of their tour bookings for them. We were provided with a database beforehand on which we created a user interface that would be easily used by the staff at the company, who are possibly inexperienced computer users. The software needs to be able to add, remove and edit tours and customers in the database. Wynne’s tours must also be able to create passenger manifests and financials reports through the program.

**Work allocation.**

The first task for our group was to create use case and class diagrams for the software. The group decided the best way to create these would be to have the whole group work on both of these rather than splitting up responsibilities. This was decided as with more input on the task, less errors were likely to be made on the diagrams as there are more people to notice errors and more ideas are added.

After this stage, the project was split into smaller tasks for separate members of the group to work on. During this time the team made sure to keep in contact so that everyone knew how far through the project people had progressed, if anyone was struggling with their tasks and if any problems needed resolving.

A problem the team had to work through was that of a member of the group dropping out due to mitigating circumstances. This meant that their workload had to be taken up by the rest of the members, but this problem was quickly resolved as two of the team split the workload left over between them.

Here each team member have stated their rolls in the project:

**Sean Grady:**

Initially we all worked together to create use case and class diagrams. All of the team putting input to the task helped us to miss less out of the diagrams which would help make the production of the software run smoother and quicker.

My role in the project was to write the reports on the project conduct. This involved me making sure I knew what other people in the group’s role were and what development processes they were using to create our software. I also had to keep record of meeting we had and making minutes on said meetings. Keeping in contact with the rest of the team was important to my role as I had to report on the work they had done and the way they had done it. Also if any problems arose I had to take note of them and how they were dealt with, to add to the report.

I also helped present the presentation which required me to know how the whole project had been organised and required communication with the rest of the team to find out how they had produced the program.

I also made sure to communicate with the other members of the team via messaging and attending group meetings. This was important as when a team member dropped out of the project the, we had to decide as a team who would take up this responsibility. Everyone making sure they communicated with each other well helped to make the project run smoother and with less problems.

**Jack Hoyle:**

For my part of the group project I worked as the group leader and created the systems user interface and testing document.

As the group leader it was up to me to organise the group and make sure that people were working to adequate time frames and overall we as a group worked together pretty well keeping in contact and organising meetings through a group page on Facebook and even though we were missing a member due to mitigating circumstances the project as a whole has ran pretty smooth from a management perspective.

I was also the creator of the UI using visual studios built in tool box meaning it was relatively simple to create a decent user interface. When creating the UI I had to take into account the .dll ‘s built in methods as the UI would require a fixed number of data input points like text boxes and date/time boxes as the built in methods required certain data when creating and editing tours and customers etc.

As the main creator of the testing plan when writing it I looked at our programs UI as to make sure that I had factored in each event eg. button press so that we could check if the built in methods would work when needed and how needed I used a table format to make the testing plan easier to read and work through with both the expected result and actual result.

On the whole I think we worked well as a group as each person knew what they had to do and they did it coming together regularly and collaborating on any problems one person would encounter. There was no conflict of opinions and we all got the work done when required.

**Jake Godfrey:**

As part of a functional group, I have attended all the meetings, in which we had established a group hierarchy, and decided tasks for each colleague. Unfortunately for the whole team, one of the members has been unavailable due to unforeseen circumstances, and this has made our work-load just a bit more.

I have communicated with all of the group very regularly, and made contributions to their individual tasks, if it be commenting on their work, giving advice and my personal opinion on changes to it.

As a group we decided my personal task was to document the project, I have created the user guide which is deliverable to the customer. This contains how to set up the program, with graphical content to make it easier for the user. A walkthrough on how to use the program, giving specific details on how to use particular parts of the program, making the parts which might seem a bit more difficult easier to understand. The abilities of the program, so the user will fully understand what the program is capable of, giving details of how to add, edit and delete many different factors of the program. And finally any error reports the program might display to the user, helping them understand what the error is for and what they need to amend.

I have put the base of the presentation together, which includes the different slides we will be talking about, and how we can describe the project, to sell the program to a customer. I will be involving myself where needed when giving the presentation, to help project certain aspects of the program.

I have spent a lot of time with the group and on my own putting together the work to get the project finished. Most of my time was dedicated to the user guide, because as anything would change in the program, I would have to amend the user guide to suit it. The rest of the time would be spent on looking over my colleagues work, and giving them advice on certain aspects of it.

I have enjoyed working with my group, as they are all very motivational, and have come together well to help each other if anyone is having difficulties with certain parts of their tasks. We have time planned very well, and organised our group to make the most effectiveness of it. We have finished our project in good time, and pieced it all together well, giving the circumstances of a missing team member.

**Christopher Aldred:**

My main task was to put all the code behind the UI elements. I made the decision to use forms over xaml and thrash out any issues. My first job was to help out building the UI. This involved making sure that there were all the UI elements in place to link all the methods (e.g. buttons to execute the methods and textboxes to collect the data for the methods). This was a fairly quick and easy task because Jack had covered the majority of the requirements for the UI early on but nevertheless an essential task.

My next task was to hook up all the .dll files to the project so that I could use the methods within. As before this was a fairly simple task. I then went into the .dll and had a look at what methods and what information were available. This gave me a great starting point as to what information needed to be collected and what variables needed for the methods to execute properly. For example the price variable in the ticket class could have been a decimal but was represented as a double. Knowing these things made writing the code easier and cut the amount of mistakes significantly.

Implementing the methods behind the UI was for the most part just a case of implementing a try, catch and then inserting a method with the information from the corresponding UI elements and a listbox clear and listbox add methods to edit the database and update the listbox displaying the database to reflect that. Some methods were more demanding such as the manifest and financial report which I wrote from scratch and took slightly longer but nonetheless I am happy with the overall outcome.

My final task was to help out with testing and to remedy any mistakes that I had overlooked whilst writing the program. Jack found the problems and I fixed them as we came across them so that we could test them after the fix and enter them into the test document. Overall I am happy with the quality of software that has been produced and am confident of it performing exceptionally in its role.

**Development process.**

The team discussed the use of a variety of possible different development processes that could be suitable for the project at hand.

One technique that was considered was ‘test driven development’, which is an agile software development method that involves writing tests for the software and then developing the simplest code that will allow the software to pass said test. The code is then refactored and the process is repeated until the software can successfully function the way the tests require it to. The benefits of the method are that few errors reach the final code as there are so many tests to reveal them beforehand. It also encourage the developers to create small and simple methods which saves time and makes the project easier to develop further. However, we decided against this process as the tests require a lot of maintenance as any change in the design of the program can require many test to be changed or more tests to be added. Another downside is that because the developers are writing the tests means that they may overlook certain tests that need to be fulfilled.

Another technique considered was the ‘waterfall method’. This process is sequential moving through stages of development leading from requirements to design, implementation, verification and finally maintenance. A benefit of this method is that it is a highly disciplined process, as every stage must be completed before moving onward to the next, with each stage having defined beginning and end points. These distinct stages mean that the progress of the project can be identified easily by both the team and the customer. Also, because the requirements and design are completed before starting to write any code means that very little time and effort is wasted later on in the process. Disadvantages of the waterfall method are that customers rarely know straight away the exact product they want to end up with. In most situations, the customer figures out what they want during interactions between them and the developers throughout the project. Another problem often found with this process is that designs that seem reasonable on paper, can be hard or expensive to develop in actuality. If this becomes the case the project would require redesigning which causes breaks in the distinguished stages of the waterfall method.

We finally decided upon using ‘feature driven development’, another agile method that is driven by features that the clients value. This involves first developing an overall model for the software, then building a list of features and finally planning, designing and building by feature. This method was a good technique for this project as it was important that the software contained all the features that the company previously carried out by paper form and by creating a list of all the features before starting to create the program made the job easier. Making the user interface simple and efficient to use was a big priority too so by first creating a model the team could make sure this was carried out. This process also allows for more features to be added easily which is beneficial if any feature are overlooked at the beginning of the project.

**Conclusion**

In conclusion, the team have worked well together to complete the project that covers all the functions the customer required of it using appropriate software development processes and overcoming obstacles, making the project a success.

**Appendix.**

Meeting minutes:

|  |  |  |
| --- | --- | --- |
| Date | Members present | Notes |
| 20/10/14 | All | -Use case and class diagrams complete.  -Meet again next week. |
| 27/10/14 | All | -Allocation of work:   * Jack – User interface. * Christopher – Coding. * Jake – User documents + presentation. * Sean – Reports + presentation. * Conor – Testing.   -Meet next week. |
| 3/11/14 | All | -No issues.  -Continue work.  -Next meeting 2 weeks. |
| 17/11/14 | All | -Program close to ready for testing.  -Reports and user documents need finished program to progress further.  -Meet in 2 weeks |
| 1/12/14 | All | -Program ready for testing.  -Meet next week. |
| 5/12/14 | All – Conor | -Conor excused from project, mitigating circumstances.  -Jack + Christopher take up testing.  -Meet Monday. |
| 8/12/14 | All – Conor | -Testing complete.  -Coding finished.  -Report and user documentation ready to be completed.  -Presentation complete.  -Meet Friday. |
| 12/12/14 | All - Conor | -Report complete.  -User documentation complete.  -Presentation practised.  -Last meeting. |