4 Time Management Account

4.1 Changes in Estimates

Initial, interim time estimation per activity per member (hour)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Activity Name | Estimation | Actual Spent | Edward | Kristian | Brian | Oscar | Kelvin |
| 1 | Project Plan | 18 | 19.5 | 5 | 5 | 3.5 | 3 | 3 |
| 7 | Analysis, Design and Testing | 38 | 39.25 | 16 | 4.5 | 8 | 7.5 | 3.25 |
| 16 | Programming | 180 | 104.75 | 33 | 23 | 18 | 17 | 17.25 |
| 21 | Testing | 40 | 8 |  |  |  | 2 | 6 |
| 24 | First Presentation | 30 | 30 | 6 | 6 | 6 | 6 | 6 |
| 32 | Final Report | 50 | 29.75 | 5.75 |  | 5.5 | 7.5 | 13.25 |
|  | Formal & informal meeting | Missed | 70 | 14 | 14 | 14 | 14 | 14 |

At the beginning we have spend a little bit more than expect for the project plan because we want to make sure our direction and our planning is more realistic and easier to achieve so we spend 1.5 hours more on reviewing our project plan especially on the Gantt chart and work breakdown.

For the Analysis, Design and Testing document again we have spent a bit more than expected because we did not agree on some decisions on the implementations of some process flow and some of the interface design. That is why we consume more effort and time on refining our requirements and design UML diagrams, especially for class diagram due to some recommendation from our supervisor in order to make the structure become more efficient and tends to a MVC structure, and Human Computer Interaction related tasks.

On the other hand, the estimation of time needed for programming is much overestimated, from 180 hours of initial estimate to 110 hours of actually implementation. I would say our implementation rate is better than our expectation and we try to finish the core part of the model classes for less than 20 hours right before the first demonstration in order to get enough concrete progress to be demonstrated. Before the Easter we are expecting to finish implementation within 100 hours but then we faced the problems of misunderstanding and misinterpretation of the codes within the group. We need more effort than the expectation after Easter in order to clear our doubts and misunderstanding and proceed to testing, final reports and preparation of the final presentation and demonstration.

In order to solve the problem of misunderstanding of the codes we have delayed the testing process that is why we do not have much time to conduct a fully test on our system but at least test it against the UAT and some normal, error, boundary cases which does not require too much time so our actual time spent for testing is far less than our expectation.

After finishing most of the implementation, progressing in testing we started to review the whole project and writing for this final report. Most of us spend decent time and effort on this final report in order to get a higher mark so the difference between the actual time spent and the initial estimation are not really significant.

There is one thing that we have forgotten to add into our initial time estimation which is our formal meeting with our supervisor and informal meetings within the group. Even though we were just reporting our progress to our supervisor during the formal meeting but we still have some time to get some suggestions from our supervisor or to discuss our direction and work distribution for the coming week which should be also counted in our time expenditure for this project.

**4.2 Analysis**

Graphs as floating figures, discuss what they mean in the narrative.

Analyses (graphic and narrative) of time spend by week/ activity/ member. Discussion of any time management decisions requiring departure from the 100 hours budget, e.g. choice to overspend to add value to the product, or the choice to spend the "ghost" budget of an absent (ill or dropped out) group member.

Throughout the whole project we have spent xxx hours in total which is less the original estimation 356 hours and the followings are the graphs which give a clearer image of how we spend our time on different part of the project and individual efforts.

[CHART 4.2.1 – Distribution of work per task per member]

From chart 4.2.1, you can see the time spent for most of the task are fairly distributed among the group members except for Edward who have paid a bit more than the others that is why he deserved to get 24% of mark distribution on this project. While Kelvin pay slightly less than the others in some tasks as he is the time manager while the time for managing the time expenditure form did not shown in the graph but you can see the time he paid in the final report to summarise the time expenditure and time management accounting. For testing as Kelvin and Oscar are responsible for the model classes which require more unit testing than the other classes so they paid most of the effort on testing. For Kristian, he paid a lot of effort on visualization of the system so he did not have much time to write for the final report and we decided to distribute most of the task in final project to the other group members. Brian provided some critical comments and worked for most of the task as he responsible for the controller classes which requires him to understand both model and view classes of how they works and integrate both of them into a single system.

[CHART 4.2.2 – Number of commits per week per member]

[CHART 4.2.3 – Working hours per week per member]

By comparing the charts above, chart 4.2.2 and chart 4.2.3, the number of commits to the SVN is in proportion with working hours as the shape of the charts are quite similar which is quite delightful that the whole group pay around the same range of time before each commits. From the first few weeks we work on a average time of five hours and do six commits per week and we reach to a small peak on the fourth and fifth week as we are going to do the first demonstration so we make frequent changes and more effort within week four and five. After the motivation for the first demonstration we fail to keep it as Easter comes the work rate of the whole group drops gradually until the Easter finishes. After Easter finishes we try to get back on track with a slow start then a blast to the maximum of work rate in the last two weeks of the project schedule. But then the project deadline is postponed so we can still make some fine tunes on both our systems and documents so the work rate drops gradually in the last week of the schedule.

Again, obviously Edward has paid more effort and time on the project that is why his lines on both graphs are on top of other member’s line for most of the time. In the chart 4.2.2 shows Brian and Kelvin do not commit as much as the other members but when compare to chart 4.2.3 they actually pay decent time on each tasks which shows they try to do a bit more work before doing a commit to the SVN.

[CHART 4.2.3 – Actual time spent against original estimation]

4.3 Reflection

Reflection on the effectiveness of time usage and management.