Entry One*:* Agile Exercise - The Planning Game

Student Name: Gregory Sloggett

Student Number: 14522247

Work Done:

Part one of the exercise saw the customers write out a story on the flashcards, followed by the developer estimating how long it would take to draw it. Shane and I acted as developers first, with Paddy and Brian taking on the roles of customers. Paddy and Brian came up with some varied stories for the fridge, and Shane and I worked on making estimates on how long it would take to draw them.

In part two Shane and I felt rather competent about our drawing ability and selected a maximum of 10 effort. As such, the customers picked the ten cards that appealed most and that did not surpass ten minutes worth of time.

The construction section in part 3 saw Shane and I attempt to finish the drawing of the refrigerator in the ten-minute period, but to little avail. We organised the story cards from Paddy and Brian, in sequence, and attempted to implement them accordingly.

Part four saw the first iteration end, and as such we swapped the job of customers & developers. Shane and I now took on the task of adding additional stories, but we added a lot of stories to the degree that we had enough to give a completely new set to the developers and discard the old set. Paddy and Brian gave a lower effort than the maximum we provided, and had higher time estimations for their drawings, so we were limited to only selecting five cards.

Part five saw Brian and Paddy draw the fridge according to our constraints. They had to converse with us for clarification on some of the stories, as they put together their drawing.

Reflection:

I feel that there was a lot to be learned and taken from this planning game exercise. I think such a learning curve was evident even between the two iterations of the iterative development process of Extreme Programming. In the first iteration, Shane and I were naïve in our predictions of our times to draw the fridge. We were also too ambitious in our promised effort number. The combination of these too factors made the drawing of the fridge in the ten-minute time bracket an impossible task. I think Brian and Paddy witnessed our how flustered and panicked we got over the drawing having given such poor estimations and used this to do a better job of producing a more accurate and in time drawing.

This hit me as the prime example of how crucial the element of planning is in Extreme Programming. We were over-ambitious and foolish in our predictions, and no customer would have been content with our finished product in the timeframe we had, given that it was incomplete. In hindsight, we should have aired far more on the side of caution, just as the second pair of developers (Paddy & Brian) did. As a customer, it felt pleasant to see that our developers were in control of their work, and we were getting done what we had asked and expected, in the allocated time.

Another important point of note was our lack of attention to detail when it came to the planning game. By this I mean our attention to the specific steps in the process. For example, one of the important steps that we failed to pick up on and exercise properly was seeking classification from the customers on the details of the fridge. One of the key elements in the Extreme Programming

methodology is that it allows for customer requirements and feedback to be used to update and improve the product, which in our case was a fridge drawing. As we went in sequence through our drawings which were being added to the fridge, we had the opportunity to fine tune the later stories based on our current implementation by asking the customers. This would have been a useful step as our perception of the tasks entailed in the story cards were different after we started drawing, compared to what they were when we first read them.

Similarly, as developers, we didn't ever execute our right to tell the customers what an improbable or impossible task was. When Shane and I were reading the story cards, we scoffed at many of the ideas thrown at us, yet we naively didn't return them to the customer to say that this was out of proportion, or simply a ridiculous idea. In this sense, we did a poor job of following the procedure accordingly, but it helped me to realise that the developer should oversee this aspect, and not just simply make a time prediction for the task without the knowledge that they can complete it at all. The developer is the one who knows the constraints involved in a task, and they must be the one to decide whether a task should or should not be undertaken.

The planning game exercise really helped me to understand just how important the challenge of planning a project and the entailed tasks are, and how much care and attention must be placed on this element of a project, particularly when customer satisfaction is at the heart of it. Several elements struck me to be over great importance.

The developer needs to be honest. A developer who gives a customer false hope over product design or over-estimates his ability to get a job done in a certain amount of time, will likely either not meet the project deadline, or have a faulty, incomplete product, or both.

Constant communication between developer and customer is paramount in producing a product that is up to the customers satisfaction, and to prevent the developer from straying from the customers product specification.

Experience is important when predicting a suitable time to produce a product within, and up to a certain level of quality. We lacked experience, and it showed.

In conclusion, this has been an informative introduction into Extreme Programming, with planning being a critical aspect. It has also been an insight into a piece of the agile development world, and a step away from the waterfall process, which I find to be evident in my coding methodology.