**Software Development Simulation**

**D10126532 John Warde DT230B**

# 1. General Information

TODO:

The SimSE ***Incremental Model*** was chosen for this part of the assessment.

The Incremental Model divides a project into logical small parts; each part will go through requirements, design and implementation phases – a sort of mini-waterfall. Each module versions is delivered to the customer for feedback, this feedback is used for refining further versions of the module and possible insight into other modules within the project. Extra analysis tasks (such as risk/difficulty analysis) are employed to prioritise modules based on value to the customer, difficulty and changeability. Results are demonstrated earlier to give customer confidence and feedback from the customer allows the project team to increase accuracy to requirements and customer satisfaction. Customers are involved throughout the project and sometimes sitting alongside the project team members.

The Incremental model is popular today, especially web related projects, and there are many incremental/iterative software development methods including Agile Unified Process (AUP), Dynamic Systems Development Method (DSDM), Extreme Programming (XP) and SCRUM.

I have been involved in many projects where Waterfall Model was used and have researched the incremental models for possible use in projects that I managed, however circumstances beyond my control prevented this. I want to get involved in projects using an Agile/iterative method and choosing this model is an opportunity to further my knowledge in this area. I can see the benefits of this type software development and I can see how other sections of this Masters can be used in unison (in particular design patterns and refactoring, from Programming Paradigms: Principles & Practices) to build better quality and more robust software.

*[Software Lifecycle model chosen]*

*[Brief overview of model chosen]*

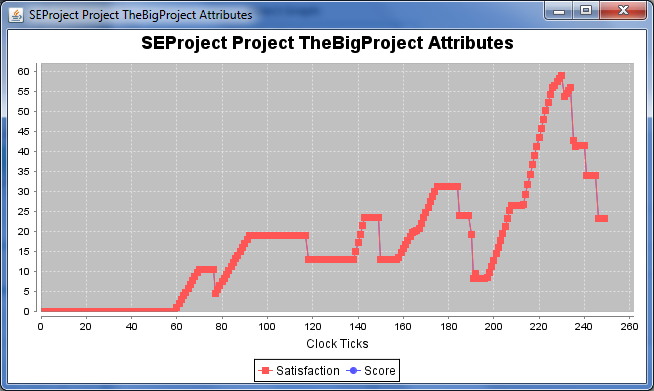
*[Justification of choice]*

# 2. Simulation Output

Below is a point form summary of my attempts using the SimSE Incremental Model.

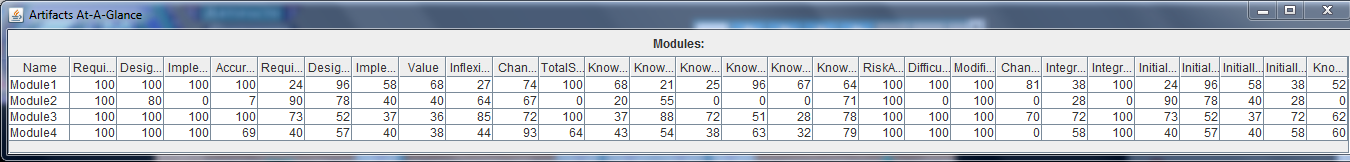
## Attempt 1

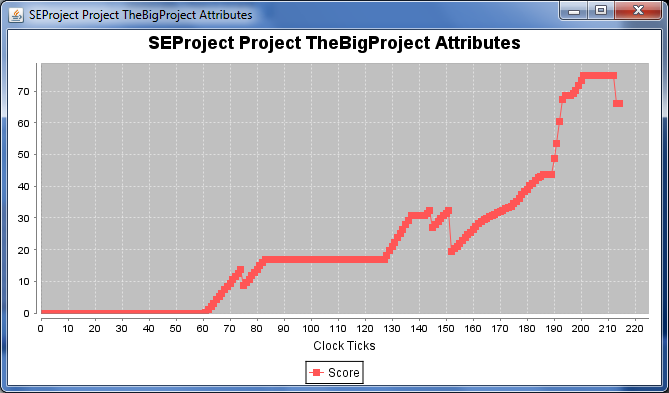
* For each module, concentrated on getting the Requirements done, then Design before starting implementation.
* Probably need to concentrate on a few more of the simulation variables.
* Set team members’ one task at a time whereas they might be able to do some tasks in parallel more of the time.
* Simulation aim was to complete project in/around 200 ticks. Probably needed to keep an eye on the score in real time toward the end.
* Noted that score was directly related to (customer) satisfaction.
* Noted that I should have been doing Risk Analysis after Requirements



## Attempt 2

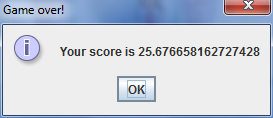
* Planned a more structured approach, get requirements and risk analysis done on all modules before deciding which modules to focus on first, i.e. with the highest value
* Kept an eye the table below, it offered the most/concise feedback.
* At clock tick 202 I had 100 for all of Requirements, Design, Implementation, Accuracy, Total Satisfaction, Risk Analysis, Difficulty Analysis, Integration for Module 1,3 & 4 and a score of 75 approx. Then kept advancing one tick at a time (while concentrating the team on design of the last module), deciding as soon as the score started to go down I would submit the final product, however I did not think the satisfaction/score would drop as much as it did when another user requirement came in at clock tick 214. Lesson learned.

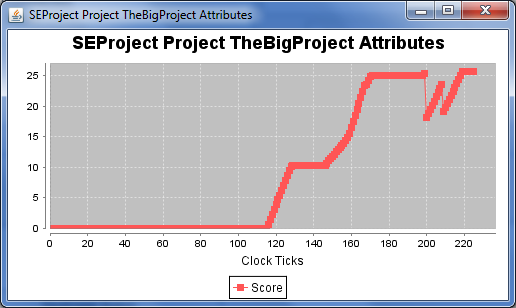


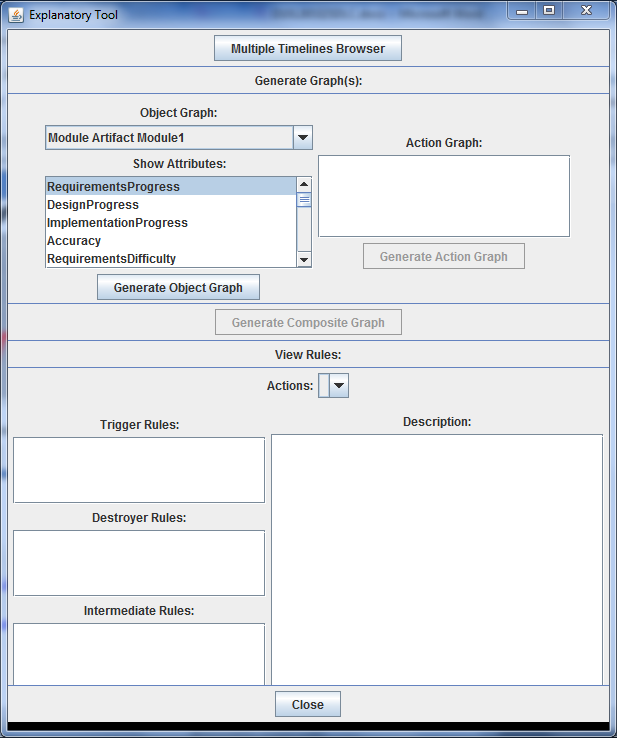


## Attempt 3

* Took the same approach plus took better notes of who was doing what, don’t know why I didn’t score as well.
* In this model, you can’t view any of the rules/triggers, can’t view any action graph or composite graphs (should I be able to do this in this model? See screenshot below) to analyse what caused value changes in the attributes of the project.
* The Known Value of the modules (after risk analysis) were low for all modules, not sure whether this was a factor in the low score.

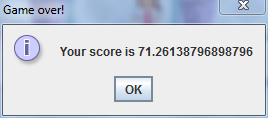


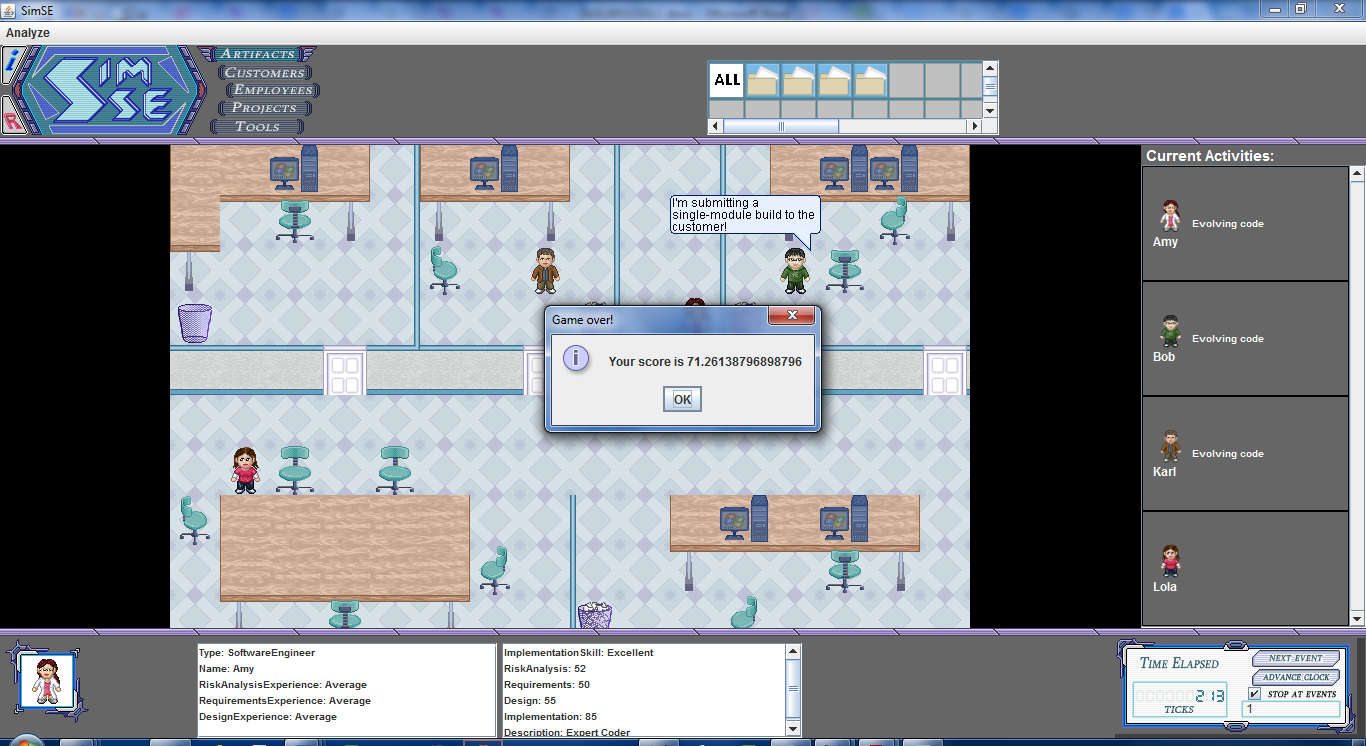


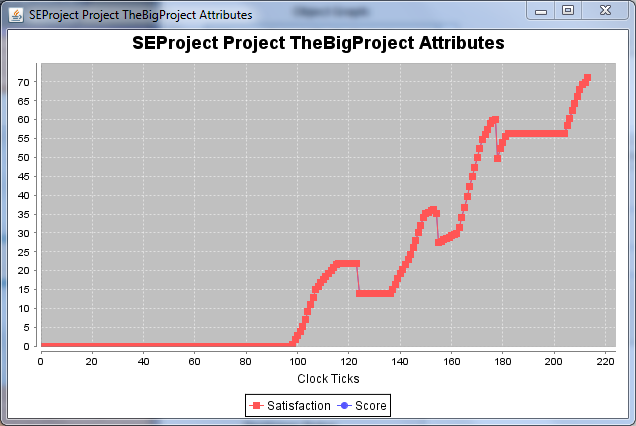


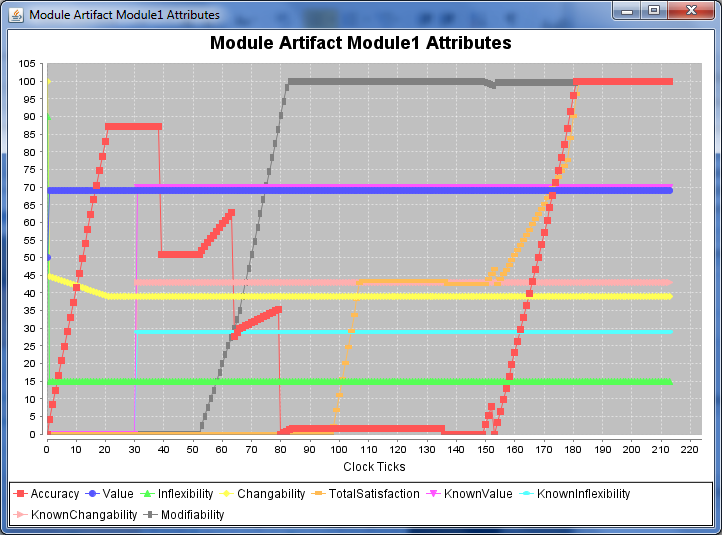
## Attempt 4

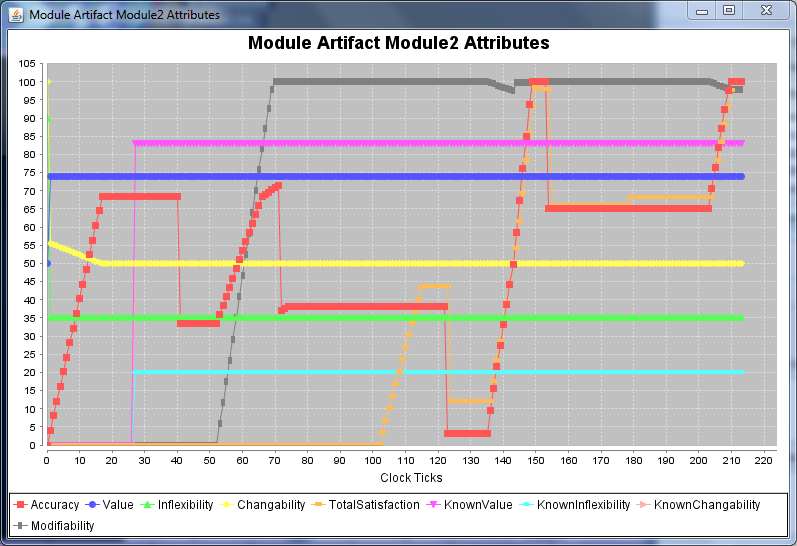
* In this attempt, a sharper focus was put on accuracy which is linked to satisfaction (and score).
* However, in this attempt/project accuracy fluctuated greatly and “evolve code” tasks were used to improve the accuracy value which has a positive effect on satisfaction. After reviewing the graphs for all 4 modules (below), evolve code could perhaps have been employed even earlier in the timeline.
* Found it frustrating when Implementation Integration for a module was less than 100% and the Integrate task could not be assigned to anyone on the team – is this right?
* However, achieved a targeted 70 plus score of 71.26 in this attempt.

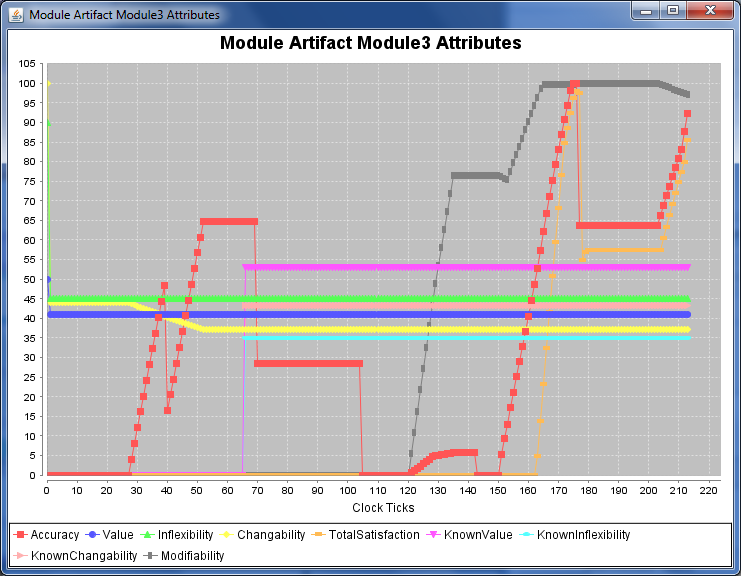


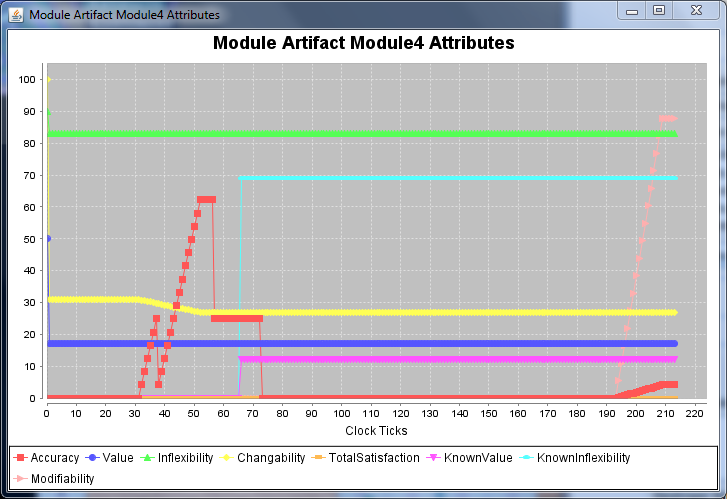












*[Detail the number of attempts]*

*[For each attempt outline key points of difficulty, error, challenge or success]*

*[Include detail of your most successful attempt and include evidence of your completion of the game]*

# 3. Reflections

The simulation focuses the student on the key aspects of incremental project management success, such as accuracy to customer requirements and adjustment through customer feedback.

The approach I took was to complete requirements analysis and risk analysis on all modules, risk analysis reveals the value the customer puts on each module plus the inflexibility (adherence to customer requirements) and changeability (the likelihood of changes by customer) of the modules.

Implementation was done in order of customer value, with the acceptance that very little of the last module completed with the timeframe of 200 clock ticks. Where modules were rated of a similar value, inflexibility and changeability shaped the final decision.

For the majority of the task assignments, I assigned two people, one with strong experience in that task and the other with less experience, having 2 people assigned at a time should lead to better quality output of the task.

The methods learned/applied here not only helped in the execution of a project would also help in quality control of a project – monitoring and responding to key performance indicators will help produce a better quality product for the customer.

The model I chose did not have any attributes such as budget, time tools, interaction with the customer and impact of workload on resources - incremental/iterative methods have an aim not to overwork resources and keep to a 39 hour week.

*[Reflect on your experience of managing a virtual software project][What key points did you learn that relate to project management as discussed in lectures?]*

*[You should consider aspects such as:*

* 1. *Correct and timely use of resources*
  2. *Impact of workload on resources*
  3. *Changes to requirements*
  4. *Changes to budget and time*
  5. *Scheduling of activities*
  6. *Use of incentives*
  7. *Use of tools*
  8. *Monitoring progress*
  9. *Interaction with customer*

*This is list is not intended to be exhaustive and the points you consider may vary with the lifecycle model you choose.]*