Edinburgh Napier UNIVERSITY

Assessment Brief Proforma

1. Module number	*SET10111*
2. Module title	*Multi-Agent Systems*
3. Module leader	*Simon Powers*
4. Tutor with responsibility for this Assessment Student's first point of contact	*Simon Powers* *S.Powers@napier.ac.uk*
5. Assessment	*Report*
6. Weighting	*60% of module assessment.*
7. Size and/or time limits for assessment	*The main body of the report, from the title through to the end of the conclusion, must be no longer than 8 pages as specified in the coursework description.*
8. Deadline of submission Your attention is drawn to the penalties for late submissions	*The report and code must be submitted to Moodle **by the Friday of Week 13** (deadline 15:00, 6th December).* *Each student will be required to make a demonstration of their implementation during Week 14 or Week 15. No demonstration may result in a mark of 0.*
9. Arrangements for submission	*The report should be submitted to Moodle as a .PDF document.* *Your code should be uploaded as a.zip archive. Please ensure that your code is contained within an Eclipse project. Add a readme.txt file with the name of the project and any other relevant information required to run the code.* *You are advised to keep your own copy of the assessment.*
10. Assessment Regulations All assessments are subject to the University Regulations	*No exemptions.*

11. The requirements for the assessment	*See below.*
12. Special instructions	*Your multi-agent system must be implemented in JADE.*
13. Return of work and feedback	*One to one verbal feedback will be offered during the demonstration session. Written feedback will be made available within 3 working weeks of submission.*
14. Assessment criteria	*See attached. Please note that checks for plagiarism will be made on electronic submissions. Students may be required to attend a further demonstration if there exists doubts as to the authorship of work.*

1. Introduction

Manufacturing companies are typically part of a supply chain, in which they must take customer orders, procure the materials to fulfil the order, assemble the materials into the final product, and then ship the product to the customer. Multi-agent systems are increasingly being used to automate this process by handling negotiations with customers and suppliers, and scheduling production to meet demand.

This coursework has two tasks. Your first task is to design and implement a multi-agent system in JADE to model the smartphone manufacturing supply chain scenario specified below. Your design should include an ontology suitable for the problem domain, and appropriate use of the FIPA communication language for agent interactions. Your second task is to design and experimentally evaluate a control strategy for the manufacturer agent in your model, with the goal of maximising the monetary profit that the manufacturer receives.

Your deliverable will be a report detailing the design and implementation of your multi-agent system model, and the experimental evaluation of your manufacturer agent control strategy. You will also submit as a separate .zip file the source code of your multi-agent system.

2. Problem specification

A smartphone manufacturer has to perform the following tasks each day:

1. A number of customers, *c*, each submit a single order request (Section 2.2 below). Each order request is for a stated quantity of smartphones, all of an identical specification. The possible specifications are stated in Section 2.1. Each order has a price, set by the customer, which the customer will pay upon delivery of the order. Each order also has a due date, and a per-day penalty for late delivery, both of which are set by the customer. For each order request, the manufacturer must choose whether to accept it or not, with the aim of maximising its profit. Any order not accepted is discarded.
2. Decide which components to purchase from which suppliers (Section 2.3), with the aim of maximising its profit.
3. Decide which orders to assemble and ship, based on the current stock of components in the warehouse and the order due dates and late penalties (Section 2.4), with the aim of maximising its profit.
4. Calculate their current profit (Section 2.5).

You should design and implement a multi-agent simulation of this process in JADE. **You must include separate agents to represent at least the customers, manufacturer, and suppliers**. You may include other agents in your design as you see fit. A simulation should run for 100 simulated days, where each day consists of the four events described above. At the end of the simulation, your manufacturer agent should print its total profit (the sum of profit across the 100 days).

2.1 Smartphone specifications

A smartphone can be of two types, either “small” or “phablet”. A “small” smartphone must contain a 5” screen and a 2000mAh battery, while a “phablet” smartphone must contain a 7” screen and a 3000mAh battery. Both types of smartphone also require RAM and storage (these components are interchangeable between small and phablet smartphones).

Each component is available in the following specifications:

- Screen: 5” or 7”
- Storage: 64Gb or 256Gb
- RAM: 4Gb or 8Gb
- Battery: 2000mAh or 3000mAh

2.2 Customer orders

Each customer requests one order per day, for a number of smartphones **of an identical, randomly generated, specification**. The specification for each order should be generated according to the following algorithm:

```
if(rand < 0.5)
    //small smartphone
    Screen = 5”
    Battery = 2000mAh
else
    //phablet
    Screen = 7”
    Battery = 3000mAh
if(rand < 0.5)
    RAM = 4Gb
else
    RAM = 8Gb
if(rand < 0.5)
    Storage = 64Gb
else
    Storage = 256Gb
```

where **rand** is a uniformly distributed randomly generated floating point number greater than or equal to 0.0 and less than 1.0 (you can use `Math.random()` for this in Java).

In addition to the specification, an order consists of a quantity, a unit price for one smartphone (in whole £s), the number of days in which delivery of the order is due, and a per-day penalty (in whole £s) for late delivery. You should generate these properties of the order using the following formulas:

Quantity of smartphones = **floor**(1 + 50 * **rand**)

Unit price for one smartphone = **floor**(100 + 500 * **rand**)
Number of days until order due = **floor**(1 + 10***rand**)
Per-day penalty for late delivery = Quantity of smartphones * **floor**(1 + 50 * **rand**),

where **floor** is a function that returns its argument rounded down to the nearest integer, and **rand** is again a uniformly distributed randomly generated floating point number greater than or equal to 0.0 and less than 1.0.

2.3 Purchasing components from suppliers

Each component can be purchased from one of two suppliers who sell at different prices and deliver at different speeds. *Supplier 1* has all components available and delivers these to your warehouse at the start of the next day (so that the ordered components may be used during the next day or thereafter). *Supplier 2* offers storage and RAM at a cheaper price but at a slower delivery speed -- it delivers any components that you order to your warehouse at the start of the fourth day after the order is placed (so that the ordered components may be used on the fourth day or thereafter). *Supplier 2* does not sell screens or batteries. The price of each component from each supplier is as follows:

Component	**Price from *Supplier 1* (delivery next day after order placed)**	**Price from *Supplier 2* (delivery on 4th day after order placed)**
5" Screen	£100	N/A
7" Screen	£150	N/A
64Gb Storage	£25	£15
256Gb Storage	£50	£40
4Gb RAM	£30	£20
8Gb RAM	£60	£35
2000mAh Battery	£70	N/A
3000mAh Battery	£100	N/A

Each supplier has an unlimited stock of components available for orders on each day. Payment is made to the supplier on the day that the component order is placed. For each component not assembled into a smartphone on the day that it arrives, there is a per-component per-day charge of *w* pounds for warehouse storage.

2.4 Order assembly and shipping

An order can only be shipped to the customer if sufficient quantities of the required components are available in the warehouse to assemble all of the smartphones in that order. Given the components currently available in the warehouse, the manufacturer agent must decide which orders to assemble and ship on that day. **A maximum of 50 smartphones can be assembled on each day**. Assembly and shipping happen on the same day that the manufacturer requests them. Delivery

to the customer also occurs on this same day, at which point the manufacturer immediately receives payment.

2.5 Profit calculation

The profit a manufacturer agent receives on a single day, *d*, is given by:

TotalValueOfOrdersShipped(*d*) – PenaltyForLateOrders(*d*) – WarehouseStorage(*d*) – SuppliesPurchased(*d*),

where TotalValueOfOrdersShipped(*d*) is the price of all orders shipped to customers on day *d*, PenaltyForLateOrders(*d*) is the penalty for any accepted orders where the due date has passed and they have not been shipped as of the end of day *d* (Section 2.4), WarehouseStorage(*d*) is the cost of all items currently in the warehouse that did not arrive on day *d* (Section 2.3), and SuppliesPurchased(*d*) is the cost of all components purchased on day *d* (Section 2.3).

2.6 Default parameter settings

For the initial evaluation of your system you may assume that the number of customers is 3 (c=3), and the cost of warehouse storage per-day per-component is £5 (w=5).

3. Summary of constraints to be implemented

You should ensure that your implementation enforces the following constraints:

1. A “small” smartphone must contain a 5” screen and a 2000mAh battery, while a “phablet” smartphone must contain a 7” screen and a 3000mAh battery. Both “small” and “phablet” smartphones must also contain one amount of RAM and one amount of storage. This should be enforced in the ontology.
2. The component delivery times from the two suppliers (Section 2.3).
3. The per-component per-day warehouse storage cost (Section 2.3).
4. An order can only be shipped if there are sufficient components in the warehouse to build all of the smartphones in the order (Section 2.4).
5. A maximum of 50 smartphones can be assembled and shipped on one day (Section 2.4).
6. Penalties for late delivery.
7. Correct calculation of profit at the end of each day (Section 2.5).

4. Design of manufacturer agent control strategy

You should design a control strategy for the manufacturer agent that on each day decides:

1. which customer orders to accept and which to reject,
2. how many components to order from each supplier,
3. which components received from suppliers to place in warehouse storage as opposed to using immediately,
4. which orders to assemble and ship.

It should make these decisions with the aim of maximising its expected profit.

5. Experimental evaluation

You should evaluate the performance of your manufacturer agent with respect to the profit earned after 100 simulated days. To evaluate the performance of your manufacturer agent control strategy, you may wish to vary some or all of these parameters:

1. The per-day per-component cost of warehouse storage, *w*.
2. The number of customers, *c*.
3. The range of per-day penalties for late delivery that can be generated.

Because of the stochasticity in the problem specification, you will need to repeat each experiment for a number of trials and present results in the form of appropriate statistics (e.g. mean and standard error of profit across 30 replicates).

You should proceed with experimentation even if you did not successfully implement all of the constraints in Section 3.

6. Report specification and mark scheme

Your report **must** be formatted as follows, 12 pt Arial text should be used throughout. The main body of the document from the abstract to the end of the conclusion must be no more than eight A4 sides in length (including any tables or diagrams). **You must include the three appendices stated below, which do not count towards the page limit. Additional appendices are not permitted**. You may not place additional information in the three appendices -- the appendices must only contain the information detailed below. **Marks will be deducted for exceeding the page limit or placing additional information in the appendices other than that specified**. Where you make use of the work of others you should use citations in your text and include a list of references at the end of your report. You **must** include all of the sections in the table below using the headings given. You should pay attention to the presentation of your report. Marks may be deducted for spelling, grammar and any other presentational issues.

Note that the guidelines for the length of each section are approximate and you should use your own judgement. However, the report must be no longer than eight pages from the title until the end of the conclusion.

The maximum mark for this coursework is 60.

The submission must be your own work, written by you for this module. Collusion is not permitted. The university regulations define collusion as ``Conspiring or working together with (an)other(s) on a piece of work that you are expected to produce independently.''

Section	Description of content	Suggested length	Mark	Suggested time spent on task
Introduction	A brief description of the smartphone supply chain problem and its importance, and a justification of why multi-agent systems are suitable for addressing it. For higher marks you should reference the academic literature.	1/2 page	/5	2 hours
Model design	Identify each agent type and explain its role in your system. Justify the design of your ontology, with reference to Appendix 1. Justify the communication protocols chosen for each type of conversation, with reference to Appendix 2.	2 pages	/12 Decomposition into agents with appropriate responsibilities: 2 marks Design of ontology (including diagram in Appendix 1): 5 marks Design of agent communication protocols (including sequence diagram in Appendix 2): 5 marks	10 hours
Model implementation	Give a description of the behaviours that each agent in your implementation executes, and state their type, e.g. one-shot, cyclic, multi-step. For each behaviour, explain where approporiate how it uses the	1 1/2 pages	/21 Implementation of agent communication protocols: 7 marks Implementation of ontology: 7 marks	20 hours

	communication protocols and ontology, with reference to example source code in Appendix 3. State how your implementation in JADE meets each of the constraints listed in Section 3 above. For each constraint include a reference to source code in Appendix 3 that implements it. Only constraints that are supported by a reference will be credited. The referenced code should be easy to read and commented as appropriate. Marks will be awarded for partial implementations, i.e. only implementing some of the constraints.		Implementation of constraints in Section 3: 7 marks	
Design of manufacturer agent control strategy	Explain and justify the control strategy for your manufacturer agent, explaining how it makes decisions 1-4 described in Section 4 to attempt to maximise profit. For higher marks your strategy should be justified with reference to the academic literature.	1 page	/7	5 hours
Experimental results	Evaluate the performance of your manufacturer agent control strategy with respect to the	2 pages (including tables and graphs)	/9 Experimental plan with hypotheses to be	8 hours

	parameters in Section 2.6 (you may also vary other aspects of the simulation as you see fit). Justify your choice of experiments, i.e. which features of your agent control strategy you are evaluating, with hypotheses about expected results**. You can be credited for this justification even if you are not able to actually run the experiments.** Results from the experiments should be presented as graphs or tables. Higher marks will be awarded for experimenting with two or more parameters. Marks will also be awarded for rigour, e.g. performing the experiments multiple times and looking at average performance.		tested: 3 marks Experimental analysis with appropriate statistics: 6 marks	
Conclusions	Suggest how the design of your multi-agent system could be expanded to handle a more realistic supply-chain scenario. Suggest how your manufacturer agent control strategy could be improved to achieve a greater profit.	1 page	/6	3 hours
References	Reference any websites, blogs or	No limit		

	published papers that you refer to in the report.			
Appendix 1: ontology	Identify all of the AgentAction, Concept, and Predictate subclasses that you have created, and show the hierarchical relations between each class on a diagram. For each class you have created, list the slots, their types, and any restrictions. Show an example instance of each non-abstract class.	No limit	Marked as part of design	
Appendix 2: communication protocols	For each type of conversation between agents, show in a diagram the communication protocol. The protocol must include the sequence of messages sent and received in the conversation, including their FIPA performatives and an example of the message content.	No limit	Marked as part of design	
Appendix 3: source code	Include the sections of source code that are referenced from the Implementation section.	No limit	Marked as part of implementation	