ECM2415 Software Engineering
Continuous Assessment — PictureMouse Cinemas

Dr David Wakeling and Dr Yiming Ying

January 2014

1 Assessment

1.1 General

The assessment for this module involves developing a prototype computer system, as per the system outline given in Appendix A. Quite deliberately, this outline describes a large set of features — almost certainly larger than can be developed in time available — and part of the assessment is to choose a subset that delivers the most value to the stakeholders in the time available.

From the outset, three aspects are fixed: the diagrammatic language to be used for requirements analysis, specification and design will be the *Unified Modelling Language* (*UML*); the development language to be used for implementation and testing will be *Java*; and the site to be used to host the (open-source with restricted membership) project will be *Google Code*.

1.2 Hand-ins

The five hand-ins are set out below.

1.2.1 Backend: Requirements Analysis and Specification

Hand-in: Week 4, Term 2 (February 3rd, 2014). Duration: 20 hours. Value: 20%.

Product: a description of the *requirements* of the prototype computer system in the form of *use case diagrams* with accompanying *textual scenarios*.

Process: a record of the evolution of the requirements in the form of a printed copy of the discussion that took place on Google Code.

1.2.2 Backend: Design of the API

Hand-in: Week 6, Term 2 (February 17th, 2014). Duration: 5 hours. Value: 5%.

Product: a description of the *interface* between the back-end and front-end of the of the prototype computer system in the form of *class diagrams* with *method signatures*.

Process: a record of the evolution of the interface in the form of a printed copy of the discussion that took place on Google Code.

1.2.3 Backend: Design, Implementation and Testing

Hand-in: Week 8, Term 2 (March 3rd, 2014). Duration: 30 hours. Value: 30%.

Product: a description of the *static structure* of the prototype computer system in the form of *class diagrams*, and the *dynamic structure* in the form of *interaction diagrams*, along with an implementation of the class and interaction diagrams in *Java*.

Process: a record of the evolution of the interface in the form of a printed copy of the discussion that took place on Google Code, and the code that was developed there.

1.2.4 Frontend: Prototyping Design and Implementation

Hand-in: Week 10, Term 2 (March 20th, 2014). Duration: 25 hours. Value: 25%.

Product: an implementation of a graphical user interface for the prototype computer system using the Java Swing framework.

Process: a record of the decisions taken in the design of the graphical user interface in the form of a printed copy of the discussion that took place on Google Code.

1.2.5 Backend and Frontend: Integration

Hand-in: Week 11, Term 2 (March 25th, 2014). Duration: 20 hours. Value 20%.

Product: an integration of the backend and frontend.

Process: a record of the integration of the backend and frontend in the form of a printed copy of the discussion that took place on Google Code.

1.3 Marks

Marks for this assessment will be awarded according to a scheme devised by Jonathan Fieldsend. Two *ratings* will be given as follows:

1. an *overall rating* will be given by the project managers to each team, reflecting their product and process;
2. a *contribution rating* will be given anonymously by each team member to each other team member, reflecting their contribution to the product and process.

The *team assessment mark* awarded will be the overall rating.

The *individual assessment marks* awarded will be calculated as 75% of the overall rating + 25% of the overall rating, scaled according to the contribution rating.

An example marks calculation is shown in Appendix B.

Appendix A: System Outline

PictureMouse is a small company that has been set up to produce a system for managing networked self-service machines located in the foyers of cinemas.

The system should allow a customer to create an account using any networked self-service machine. The customer chooses a customer name that is not already taken, a password that is at least eight characters long, including at least one upper-case character and one digit, and supplies a sixteen-digit number of a credit-card to be used to pay for cinema tickets.

The system should allow a customer to book a cinema ticket using any networked self-service machine. The customer signs on with their customer name and password, if they are not already signed on. The customer then browses the films available for the next week, either reading the synopsis supplied by the film distributor, watching the trailer supplied by the film distributor, or reading the reviews written by other customers. The customer then chooses one of the films available, with the option to cancel and start again. The customer then browses the seating plans for the screenings available for the film. The customer then chooses a seat for one of the screenings of the film that is not already taken, with the option to cancel and start again. The customer then books a ticket for their chosen seat, screening and film.

The system should allow a customer to print a cinema ticket using any networked self-service machine. The customer signs on with their customer name and password, if they are not already signed on. The customer then browses the tickets that they have booked, with the option to cancel and start again. The customer then prints their chosen ticket, and their credit-card is charged with the price of the ticket.

The system should allow a customer to write a film review using any networked self-service machine. The customer signs on with their customer name and password, if they are not already signed on. The customer then browses the films available for the last week, reading the reviews written by other customers. The customer then chooses one of the films available, with the option to cancel and start again. The customer then writes a review of the chosen film and adds it those written by other customers.

The system should allow a customer to read a newsletter using any networked self-service machine. The customer signs on with their customer name and password, if they are not already signed on. The customer then reads the newsletter.

The system should allow the administrator to change the details of customer accounts using the networked administration machine. The administrator signs on with the name "Administrator" and the administrator password, if they are not already signed on. The administrator then browses the customer accounts. The administrator then chooses one of the customer accounts, with the option to cancel and start again. The administrator then changes the details of the chosen account.

The system should allow the administrator to change the details of the films available using the networked administration machine. The administrator signs on with the name "Administrator" and the administrator password, if they are not already signed on. The administrator then browses the films available. The administrator then chooses one of the films available, with the option to cancel and start again. The administrator then changes the details of the chosen film.

The system should allow the administrator to change the contents of film reviews using the networked administration machine. The administrator signs on with the name "Administrator" and the administrator password, if they are not already signed on. The administrator then browses the film reviews. The administrator then chooses one of the film reviews, with the option to cancel and start again. The administrator then changes the contents of the chosen review.

The system should allow the the administrator to distribute a newsletter to all customers using the networked administration machine. The administrator signs on with the name "Administrator" and the administrator password, if they are not already signed on. The administrator then creates a newsletter based on a template composed from the synopses supplied by film distributors. The administrator then distributes the newsletter.

Appendix B: An Example Marks Calculation

Suppose that an *overall rating* of 58% is given by the project managers to a Team X with four members, Member 1, Member 2, Member 3, and Member 4, reflecting the quality of their product and process.

Further suppose that a *contribution rating* is given anonymously by each team member to each of the others, reflecting their contribution to the product and process, as follows:

- Member 1 is given a contribution rating of 53 points by the others;
- Member 2 is given a contribution rating of 18 points by the others;
- Member 3 is given a contribution rating of 36 points by the others;
- Member 4 is given a contribution rating of 33 points by the others.

In total, there are 140 contribution rating points.

The *team assessment mark* awarded for Team X is then 58%.

The *individual assessment marks* awarded are then calculated as follows:

- for Member 1 is $58\% \times 0.75 + 58\% \times 4 \times \frac{53}{140} \times 0.25 = 43.5\% + 22.0\% = 65.5\%$
- for Member 2 is $58\% \times 0.75 + 58\% \times 4 \times \frac{18}{140} \times 0.25 = 43.5\% + 7.5\% = 51.0\%$
- for Member 3 is $58\% \times 0.75 + 58\% \times 4 \times \frac{36}{140} \times 0.25 = 43.5\% + 15.0\% = 58.5\%$
- for Member 4 is $58\% \times 0.75 + 58\% \times 4 \times \frac{33}{140} \times 0.25 = 43.5\% + 13.7\% = 57.2\%$

The average *individual assessment mark* can be seen to be 58%.