Candidate Name:

Graded Unit 2: Marking Scheme

Stage 2: Development

Production of application (up to 25 marks)

|  |  |  |
| --- | --- | --- |
| Topic | Method/approach | Mark |
| Coding of the problem domain (up to 5 marks for this topic). | Up to 5 marks for implementing the business model.  The mark given for this topic should reflect how well the learner has implemented the planned business model design. The criteria used to award the marks is likely to include the following:   * Does the code match the design? * Does the code make appropriate use of inheritance and/or associations? * Does the code make appropriate use of scope operators? * Does the code make appropriate use of setters and getters and/or properties as appropriate? * Does the code make appropriate use of constructors? * Are the methods and attributes coded correctly? |  |

|  |  |  |
| --- | --- | --- |
| Topic | Method/approach | Mark |
| Coding of the UI domain  (Up to 5 marks for this topic). | Up to 5 marks for implementing the view model.  The mark given for this topic should reflect how well the learner has implemented the planned view model design. The criteria used to award the marks is likely to include the following:   * Does the solution match the design? * Appropriate use of UI controls to ensure that the data entered is valid (e.g. validation routines and/or error prevention). * Coding of appropriate events. * Does the code link appropriately to the business model?   The code for the sample response is indicative of a response that would probably obtain at least 4 of the  5 marks available for this topic. A better response would include more user feedback, e.g. by coding appropriate mouse over events. |  |
| Use of unfamiliar libraries and/or constructs (up to 5 marks for this topic). | Up to 5 marks for the appropriate use of unfamiliar libraries and/or constructs which have been introduced through learners self-research.  When marking this topic, the assessor must be aware of the libraries and constructs that learners have been exposed to during the learning process. The mark awarded should reflect both the quantity and complexity of the libraries and/or constructs used. For example appropriate use of an unfamiliar simple UI object might be awarded with a single mark in contrast to the use of an unfamiliar data binding construct which might gain 2 marks. Another way in which learners could obtain marks here is to use some unfamiliar features of the development environment such as the use of the unit testing features in Visual Studio or Eclipse which might well be awarded up to 3 marks. The full 5 marks should only be awarded if the learner includes appropriate references to the sources used to investigate the library or construct. |  |

|  |  |  |
| --- | --- | --- |
| Topic | Method/approach | Mark |
| Error handling (up to 5 marks for this topic). | Up to 5 marks for coding error handling and/or error prevention, e.g. use of exceptions.  The mark for this topic should be based on the code provided for both the business and view model. The view model should ideally include the use of some selection UI controls (error prevention), validation routines (where appropriate) and some exception handling (error handling). The business model should ensure that invalid data is appropriately handled and ideally should include some appropriate use of exceptions.  In order to obtain the full 5 marks, the business model must throw some appropriate exceptions (or equivalent) that are appropriately handled by the view model. |  |
| Internal documentation (up to 5 marks for this topic). | Up to 5 marks for the internal documentation, standard documentation, naming conventions and appropriate use of indentation.  The mark for this topic should be based on code provided for both the business and view model. In order to get the full five marks, the learner must have consistently abided by appropriate naming conventions and indentation rules. In addition, the internal comments should be readable and abide by a recognised standard format such as Microsoft’s XML documentation schema or Oracle’s JavaDoc. |  |

Testing — up to 10 marks

|  |  |  |
| --- | --- | --- |
| Topic | Method/approach | Mark |
| Test plan (up to 5 marks for this topic). | Up to 5 marks for designing test plan and test cases.  This may include the development of test harnesses.  This should include the strategies used to test both the business model (e.g. talking about unit testing) and the view model (e.g. functionality testing and usability testing). The test cases produced should include evidence to illustrate that the learner has applied appropriate techniques to select test values, e.g. appropriate black box and/or white box techniques.  A learner may well decide to design a number of test harnesses for the unit testing of the business model. If this approach is taken, they should include some documentation (possibly in the internal documentation) to explain why the text values used were selected. All of the methods in the business model classes should really be tested.  The functionality testing of the view model should ensure that the UI behaves as expected. This should look at each of the events coded and ensure that the UI responds appropriately (e.g. validation events should display appropriate error messages) and in terms of updating the underlying business model appropriately. |  |
| Test runs (up to 5 marks for this topic). | Up to 5 marks for running, documenting and evaluating test runs.  The learner should provide evidence of having run and documented the test cases identified in the test plan. This may be in the form of test harness outputs for the unit testing. A learner should not be awarded the full 5 marks unless there is good evidence of evaluating the results of test runs. |  |

Documentation — up to 5 marks

|  |  |  |
| --- | --- | --- |
| Topic | Method/approach | Mark |
| User documentation (up to  5 marks for this topic). | Up to 5 marks for developing appropriate user documentation — this may also include online help features.  This may take the form of a traditional user manual and/or online help features. The view model should also provide appropriate feedback and visibility for the user. An installation guide might also be provided if appropriate for the prototype application developed. |  |

Phase 2 Production – Total: \_\_\_\_\_\_\_\_\_\_ out of 25

Phase 2 Testing – Total: \_\_\_\_\_\_\_\_\_\_ out of 10

Phase 2 Documentation – Total: \_\_\_\_\_\_\_\_\_\_ out of 5

Phase 2 Total: \_\_\_\_\_\_\_\_\_\_ out of 40