Please answer Question 1 and two other questions

The questions are based on the online assessment software case study used throughout the course. You should use your knowledge of the domain to make reasonable assumptions as necessary. **If in doubt, state those assumptions.** For full marks your answers should be concise as well as accurate.

*This paper will be taken on-line and this is the paper format which will be available as a back-up.*

The use of electronic calculators is **NOT** permitted.
Please answer Question 1 and two other questions.

**Question 1 – COMPULSORY**

1.1 State four reasons why design patterns are important in object-oriented software development. (4 marks)

1.2 Assume that there is a class `ExamServer`, and there should only ever be one instance of this, which must be created immediately when the server program runs. Show how the Singleton design pattern can be used to guarantee this (you can assume that the `ExamServer` constructor has no parameters). (4 marks)

1.3 Clearly the Observer pattern (in the form of the Java Delegation Event Model) would be used in the user interfaces of the project. Suggest a more fundamental role it could perform in the implementation of the Take Exam use case. (2 marks)

1.4 Draw a UML diagram to illustrate the Observer pattern as applied to this situation. (6 marks)

1.5 Suggest how two other design patterns might be used in implementing Take Exam. (4 marks)
Question 2

2.1. A year into the project, you give a successful demo to various users and stakeholders, including Simon Smartsuit. He then asks for a copy of all your documentation for his records, which you provide. Unfortunately he’s been reading up on the UP a bit and as usual has misunderstandings. You get the following email from him – how do you respond?

“Thanks for all the gen old boy, but surely there’s a lot of stuff missing. You seem to have gone to a lot of effort to produce a comprehensive glossary, but otherwise there’s not a lot there. There seem to be hardly any fully dressed use cases (Invigilate Exam and a few others). A lot of the class diagrams seem to be incomplete – some of them don’t even show any operations and none of them seem to document the code in full. Also they’re just photos of diagrams drawn on whiteboards - could I have the final versions, please? Oh and I’d particularly like to see your Vision document.”

Your answer should be concise, addressing each of the points raised in a single sentence. (8 marks)

2.2 Each of the following terms is ambiguous: suggest two or more alternative meanings for each, and note any other properties of these which should be included in a glossary.

i) Exam paper (2 marks)
ii) Set of answers (3 marks)
iii) Marks (3 marks)

2.3 Give two other examples from the domain of commonly used terms which need to be clarified in this way. (4 marks)

Question 3

3.1 Give three tests which can be applied to determine whether a potential Use Case is a good one. In each case give an example from the exam software domain of a potential use case which would pass the test and one which would not. (6 marks)

3.2 Write the main success scenario of the Mark Exam use case in casual format: no more than 150 words, in paragraphs, not bullet points. You should assume that the examiner gets the data from the exam server at the beginning and returns the marks there at the end, but that the process of marking itself is done offline, using a marking tool like that you’ve seen in the course. Also assume that an exam may contain a combination of multiple choice, text and diagram questions. (10 marks)

3.3 Suggest four important non-functional requirements related to this use case, in at least three “URPS+” categories. (4 marks)
Question 4

4.1 What is the purpose of domain modelling? Your answer should include the relationship between domain classes and design classes, and the factors which influence the amount of effort worth spending on domain modelling. (5 marks)

4.2 Consider the following description of the process of marking traditional paper exam scripts.

“The examiner is given a set of exam scripts and a mark sheet on which to record the marks. Each script contains the answers to the exam from one student, identified by a library card number on the front cover. The front cover also holds the real name of the student (hidden under a sticky flap), the title and course code of the exam, and a table for the examiner to enter the marks. The mark sheet has a row for each student, identified by library card number, with columns for the marks for each question and the total. The examiner marks all the answers for a complete question from all the students. Marks for each sub-question and the question total are written in the margins of the pages of the script. The question totals are transcribed onto the table on the front page of the script and onto the mark sheet. This process is repeated for each question, checking carefully that every page of the script has been inspected. The examiner adds up the marks on the mark sheet to give the totals. All the additions and transcriptions of marks are checked by clerical staff. Once checking is complete, they are entered into a central database.”

Draw a class diagram to represent the significant domain classes suggested by this description and their important relationships and attributes. (7 marks)

4.3 To what extent would these domain classes correspond to design classes? (4 marks)

4.4 Suggest ways in which the dynamic aspects of the process could be improved in the software implementation. (4 marks)
Question 5

In an early iteration you have decided to implement a cut-down version of Take Exam with two kinds of questions: text questions which are answered by typing text into a box, and composite questions which contain other questions (which themselves may be composite questions). You therefore have design classes `TextQuestion`, `CompositeQuestion` and correspondingly `TextAnswer` and `CompositeAnswer`. In addition you have classes to represent the complete documents, `QuestionPaper` and `AnswerPaper`.

5.1 According to the GRASP principles of Polymorphism and Protected Variations, two further classes should be added to this design. State what these classes are, why the resulting design conforms to these principles and what the advantages of doing this are. (6 marks)

5.2 Based on the appropriate GRASP patterns, suggest which of these classes should implement the following operations:

i) String GetQuestionText() – get the text for a question. (1 mark)

ii) String getAnswerText() – get the text of a particular student answer. (1 mark)

iii) QuestionPaper constructQuestionPaper(String storedRepresentation)

Construct a question paper from the String representation in which it is stored on Disk.

Suggest THREE options for this operation. (3 marks)

5.3 Explain the notion of a Controller, and the different types of controller, using the Client side of the application as an example. Suggest what sort of controller, if any, would be most appropriate here. (5 marks)

5.4 What additional classes, required to implement Take Exam are suggested by GRASP patterns? (4 marks)

END OF EXAMINATION