EBU6304 Revision

Exam timing:

The exam is designed to be completed within 2 hours. You have in total 6 hours from the release of the exam paper to the completion of the submission on QM+.

Exam format:

Open book exam. You can use any resources available to you. If you quote text, or refer to a diagram or chart, you must cite the source.

Plagiarism:

Plagiarism is a serious academic offence. You must complete the exam by yourself. You must NOT discuss the answers with anyone else. The submission system will check the similarity of the submissions and any similar answers can be identified. You must TYPE your answer in the answer sheet and add word count. Any handwritten, scan, and picture of text will not be accepted and will not be marked.

Resources:

You have the following resources available:

- Lecture notes: total 15 sets, on QM+
- Lecture video recordings: total 34 videos, on Echo360 https://echo360.org.uk/section/77f7ebab-4742-40b3-a3c7-d0fa20423be0/home
- Tutorial exercises and solutions: at each teaching week
- Tutorial video recordings: available on Collaborate on QM+, each week
- Lab exercises
- Extra notes and explanations on the announcement and messageboard on QM+
-

Exam questions:

4 questions, 25 marks each.

The questions are of the "problem solving" types, which means you cannot find direct answers anywhere. You need to "apply" the knowledge you have learnt to solve problems.

Example of the questions are those discussed during the 4 tutorials, the lectures of exercises (videos 28,29,33,34), Lab exercises and some extra exercises in teaching week 3 and 4.

If you have any more questions, please feel free to use the messgeboard on QM+.

Key knowledge extracted from the lecture notes:

- Understand software types and features of good software
- Understand the need of software engineering
- Be able to choose suitable software process for a given project
- Use appropriate techniques to gather requirements
- Be able to write user stories, estimate and prioritise stories
- Understand fundamental design concepts, be able to identify classes and relationships in a given scenario
- Understand different software architectures
- Be able to use appropriate testing methods and techniques
- Understand software project management features and be able to use project management methods and tools, be able to identify and manage risks
- Understand the design principles and be able to use the design principles to solve problems
- Understand the design patterns and be able to apply the design patterns solve problems
- Understand open source software