Tuesday 26th April 2022
14:00 BST
Duration: 2 hours
Additional time: 30 minutes
Timed exam - fixed start time

DEGREES of MSci, MEng, BEng, BSc, MA and MA (Social Sciences)

Professional Software Development H
COMPSCI 4015

(Answer all 4 questions)

This examination paper is an open book, online assessement and is worth a total of 80 marks.

1. Project Management

Consider the following scenario (provided ahead of the exam) and then answer the questions below.

> You have started work as an agile coach with a team who follow the Scrum process. The team comprises Lesley (product owner), Kerry (a senior developer) and four developers, Robert, Jody, Jamie and Alex. The team conduct a sprint planning meeting on Friday afternoon after a review meeting in the morning. You are invited to participate as an observer.
>
> Lesley begins the meeting by congratulating the team on the work they got done in the previous sprint and briefly summarises the five user stories that were successfully delivered, including one that she wasn't aware of as it wasn't included in the previous sprint plan. She then summarises the new requirements that were identified by the customer during the sprint review meeting that took place that morning. The team then work on turning these into user stories. This takes about 60 minutes of discussion resulting in eight new stories added to the backlog of work. A typical example of one of the stories proposed is "As a user I want to be able to edit photos I take directly on the app so that I can create content all in one place."
>
> Based on the product roadmap, there are already five user stories assigned to the coming sprint. Lesley chooses five of the new user stories for inclusion as well and labels them as Must Haves (giving 10 in total for the sprint). Lesley takes the remaining new user stories and divides them between the next three future sprints, according to their lower MoSCoW priority.
>
> The team then plays a round of Planning Poker for each story. As time is short and the team member's estimates vary considerably, Lesley just assigns each story an estimate of one week, so that all 10 can be completed within the coming sprint. Lesley allocates two stories to each of the developers. The team then spend a further 90 minutes independently identifying and recording the tasks that will need to be completed in order to implement the user stories. Lesley makes sure each team member makes a personal note of their tasks.
>
> Kerry reminds Lesley about their discussion in the retrospective that took place after the review meeting. The team had expressed unhappiness with the number of times commits to their mainline resulted in broken builds when the code was subsequently deployed. Lesley agrees and leads a discussion of the reason for this, which lasts about 30 minutes. The team decides that they need to configure a continuous integration service to build the project every time a commit is made to the mainline. Lesley wants to be supportive so agrees and adds this to the sprint backlog as well.

(a) Explain the purpose of a Scrum *sprint planning meeting*, contrasting the purpose with that of a *release planning meeting*.

[2]

(b) Clearly, it isn’t a good idea for estimates to be assigned based on the product roadmap. Instead, the details of the roadmap should be driven by the estimates for the issues.

Identify eight other aspects of the conduct of the sprint planning meeting that could be improved and explain in each case how it could be addressed.

[16]

(c) Halfway through the sprint the team report that none of their user stories have been completed. Lesley proposes that the team work overtime in the second week in order to catch up and maintain the project’s velocity. Explain whether you agree with this proposal.

[2]

2. Requirements Engineering

(a) Consider the following user stories, concerning the development of an augmented reality mobile app that allows gym users to make video recordings of their exercise routines using different type of gym equipment.

1. As a gym user I want to record a short video of myself using an item of gym equipment.
2. As a user I want to be able to delete a video so that I can prevent my account from becoming cluttered and remove recordings I don't like.
3. As a gym user I want to select other users as my friends so that I can create my personal social network of gym users.
4. As a gym user I want to share a recording of use of an item of gym equipment with the friends I select in the above user story so I can get constructive feedback.
5. As a gym user, I want to provide constructive comments on a videos that have been shared by my friends; my comments will be limited to 100 characters and I will only be able to comment for 24 hours. I will also be able to add a Like, Hate, Love or Laugh emoji reaction. This will allow my friends to benefit from my opinion.
6. As a gym user, I want to get free automated feedback on my use of an item of gym equipment so that I can exercise more effectively.
7. As a gym user, I want a qualified human instructor to be paid to review my use of an item of gym equipment so that I can exercise more effectively.
8. As a qualified gym instructor I want to register my qualifications so that I can provide paid for advice.
9. As a qualified gym instructor, I want to record videos of the correct use of gym equipment so that I can demonstrate and advertise my expertise.
10. As a gym user I wanted to set recording to 4k high resolution so that I am able to see the details of my exercise posture clearly.

Choose any **five** of the user stories and:

- Explain why the story violates one of the INVEST criteria.
- Propose a revised user story that complies with the INVEST criteria.

[15]

(b) Suppose that each of the user stories identified above required roughly equal effort, but that you only had budget to implement between 25% and 50% of them. The customer's key priority is to build a demonstration of a working app that will attract users, with the aim of building in revenue generation later. Propose a MoSCoW prioritisation of the user stories with the aim of ensuring that a minimum viable product is developed within the available budget. Justify your answer.

[5]

3. Software Architecture

Consider the following scenario (provided ahead of the exam) and then answer the questions below.

> A team is working on an application to automate the process of conducting a *systematic literature review* (SLR). An SLR is undertaken when a researcher wants to identify and analyse all the research papers published in peer reviewed conferences and journals about a particular topic. The review is called systematic because the process should (in principle) be described in a way such that the set of papers identified can be reproduced by independent researchers.
>
> An SLR typically follows several different stages. First an exploratory search is undertaken in a online paper database with associated API, such as the ACM Digital Library, to identify relevant keywords. For example, for an SLR on Behaviour Driven Development the keywords might be "Behaviour Driven Development", "BDD" and "Gherkin". Then the database is searched fully and all returned results are recorded, usually in a format like BibTeX. The PDF version of the files will also be retrieved. These results may then be filtered by date range, and publication type, since some databases will return non-peer reviewed documents, such as conference introductions. Any duplicate results will then be removed. Next, irrelevant papers are filtered, normally by identifying irrelevant venues. For example, a paper on the topic of gherkins in the Journal of Applied Ecology is unlikely to be relevant to a review of BDD. The filtering process is then repeated for relevance of the paper title and abstract.
>
> Once the set of of papers have been identified, a number of analyses may be performed. The researchers may want to produce a graph showing the number of publications per year identified for the topic. The researchers may also classify the papers according to research method such as case study or laboratory study. Other metadata may also be extracted, such as the number of participants involved in a laboratory study, or whether research questions are formally stated. Sometimes it is useful to reorder these steps, for example, by counting the number of papers per year before and after an exclusion criteria has been applied.

The team have developed some code that allows them to extract textual content directly from the PDF version of a paper and also use machine learning to automatically do a number of the analytical steps, such as determine the relevance of a paper based on title. They are now proceeding to develop an overall architecture.

(a) You are working with a colleague to decide on the overall architecture for the system. A colleague proposes that the overall architecture for the system should be based on the singleton design pattern. Explain the difference between an architectural and design pattern to your colleague.

[2]

(b) From those you have seen in the course, choose an appropriate architectural pattern for the SLR system. Explain why the pattern is a suitable solution for the design problem.

[3]

(c) Explain how you would *tailor* the pattern to fit the needs of the scenario.

[4]

(d) Sketch a component diagram to illustrate the architecture that you have chosen. Include the key components and interfaces.

[6]

(e) Later on the team propose being able to search across multiple databases such as DBLP and and IEEEXplore. Explain how the architecture would need to be altered to accommodate this new requirement.

[3]

(f) A colleague has proposed using the plugin architecture to allow third party tools to be included in the repository analysis system. State whether you agree with this proposal and explain your reasoning.

[2]

4. The following Java class sends notifications of changes to issues in an issue tracker.

```
public class NotificationMaker {
 private EmailClient emailClient;
 private String subject;

  // ...Constructor for initialising emailClient etc not shown....

 public void sendNewCommentNotification(Issue issue, String commenter,
   String commentContent, String commentURL, String commentDate){
  subject = "New Comment on Issue #" + issue.getID();

  for (User user : issue.getWatchers()){
   String body = "Dear " + user.getName() + ",\n\n";
   body += "There has been a new comment on issue #" +
     issue.getID() + ".\n\n";
   body += commenter + " said " +
    (commentContent==null)?"":commentContent + " on " +
     commentDate + "\n\n";
   body += "Issue owner: " + issue.getOwner().getName() + " \n\n";
   body += "See comment in context: " + commentURL;

   if (user.wantsEmailNotification()){
    String emailAddress = user.getEmailAddress();
    if (emailAddress != null)
     emailClient.sendEmail(emailAddress, subject, body);
   }
  }
 }

 public void sendTitleChangedNotification(Issue issue, String oldTitle,
  String newTitle){
  subject = "New Title for Issue #" + issue.getID();

  for (User user : issue.getWatchers()){
   String body = "Dear " + user.getName() + ",\n\n";
   body += "The title of issue " + issue.getID() +
     " has changed .\n\n";
   body += "The title changed from " + oldTitle +
     " to " + newTitle + ".\n\n";
   body += "Issue owner: " + issue.getOwner().getName() + " \n\n";
   body += "Last change date:" + issue.getLastChangeDate() + "\n\n";
   body += "See the whole issue: " + issue.getURL();

   if (user.wantsEmailNotification()){
    String emailAddress = user.getEmailAddress();
    if (emailAddress != null)
     emailClient.sendEmail(emailAddress, subject, body);
   }
  }
 }
 //... More methods for notifications about other properties
}
```

The comments indicate where additional code is present, but not shown for brevity.

Study the class then answer the questions below.

(a) Explain the steps taken in the refactoring process.

[3]

(b) Identify three bad smells in the code. For each bad smell:

- explain why it makes the code harder to maintain
- propose a refactoring to remedy the bad smell
- explain how the refactoring will improve the maintainability of the code.

[12]

(c) A colleague has developed a class called “SlackIntegration”. This has a similar method to the email client for sending notifications, but to a Slack Channel ID, rather than to an email address. Note that a slack channel might be a private channel of a user, or a shared channel between several team members. Your colleague proposes introducing a new if clause at the end of each notification method to handle this new feature. Explain why this may be unsuitable for future maintenance and instead how the existing code could be refactored to accommodate multiple notification mechanisms.

[5]