Software Engineering and Project Management - Final Exam

2017-12-22. **Duration:** 8:00 - 11:00

Start by reading all the questions, to see if anything is unclear. One of the teachers will visit the exam around 8:30 to clarify questions.

Answers must be written in English. Dictionaries may be used.

Start each answer on a new page. Please hand in the pages in the correct order.

For each question, I give its maximum points of the answer.

The format and extension of your answer should adapt to the type of questions:

- **Specific questions:** a concept, a list… the *optimal* answer is usually around a single paragraph (This does not include drawing).

- **Short questions:** explanations, reviews, comparisons… the *optimal* answer is usually a few (2 or 3) paragraphs (This does not include drawing).

- **Questions for reasoning:** your arguments are presented in a concise, well-reasoned way,… and written with your own words. The *optimal* answer is usually less than 2 pages. (This does not include drawing).

A checklist of common mistakes that cost points:

- Answer all **9** questions. A bad answer never gives less points than no answer.

- Read the question again after you have written the answer. Verify that you have actually answered the question. Verify that you answered *all* parts. Verify that you have *not* hidden the answer between many other irrelevant comments about the topic. *Cursives are added to highlight the key elements of the questions.*

- In particular, don't forget to give an example if that is requested, and make it a concrete one.

- When a question asks you to compare two things A and B, make sure to highlight the contrasts: their differences. I do *not* want a full description of A and a full description of B, leaving it to me to find the differences. It is better to use tables than paragraphs of text to compare.

CONTINUES NEXT PAGE

Specific questions.

Question 1. [2 points].
Define what is a layered architecture.

Question 2. [2 points].
Compare over-the-shoulder and pre-checking code review processes: Indicate one (1) *advantage* and one (1) *difference* between them.

Question 3. [2 points].
Compare validation and verification: Indicate one (1) *similarity* and two (2) *differences* between them. Three (3) in total.

Question 4. [2 points].
Describe what are the criterias to assess legacy systems: Enumerate the *criterias* and how *they are used* to make a decision.

Short questions.

Question 5. [5 points].
5.a (3/5 points) *Describe* the V-Model using a software project as an *example*.

5.b (2/5 points) *Give* a different *example* than 5.a where it would be reasonable to use a waterfall model instead of a V-Model. *Explain* your answer.

Question 6. [5 points].
The diagram 1 (page 4) is a partial *class diagram* of an online BookingManager for a car service company. It also includes the code for the function bookDriver of the DriversManager interface and the addAddress function for the Client class.

6.a (2/5 points) *Indicate* three (3) design patterns from the diagram.

6.b (3/5 points) For each one of them, *identify* the class/es involved and *explain* how the pattern is used in the context of the example.

CONTINUES NEXT PAGE

Question 7. [5 points].
7.a (2/5 points) *What* are the components that define every test case? For each component, provide an *example* based on a performance test.

7.b (1/5 points) *Compare* black-box testing with glass-box testing. *Indicate* (1) difference between them.

7.c (2/5 points) A performance test *is* a black-box test, a glass-box text or both? *Explain* your answer with *examples*.

Questions for reasoning.

Question 8. [9 points].
Your company have been tasked with the development of a new online portal to substitute StudentPortalen. The minimum functionality of the platform must include a scheduling module to automatically assign rooms to courses (e.g., based on the number of students registered) and teachers to courses (e.g., based on teachers' expertise and availability).

8.a (4/9 points) Which aspects of the project planning are highlighted by an activity network? Which are hidden? *Give* one (1) *example* of an activity network for the project.

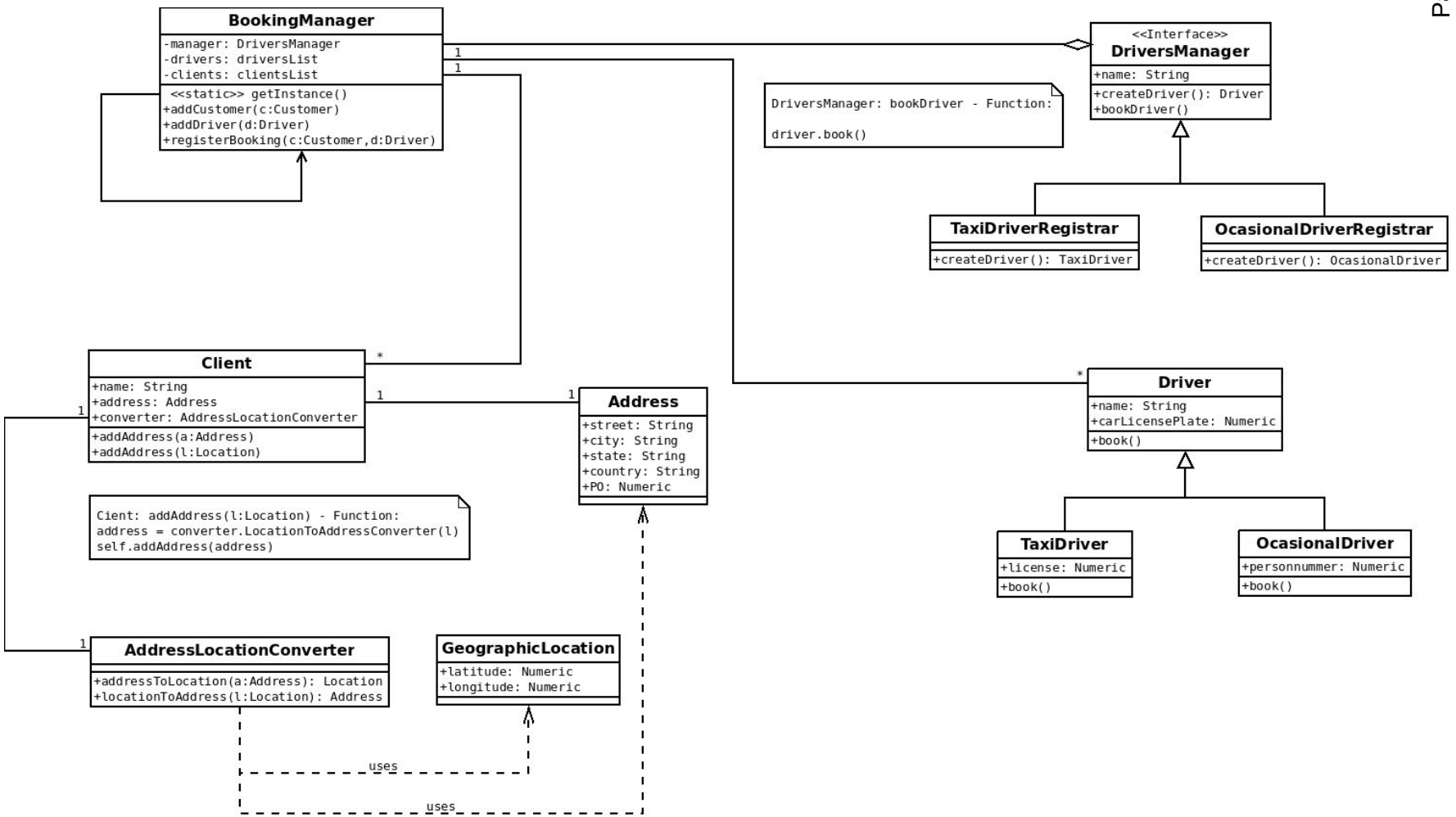
8.b (5/9 points) *Describe* with an activity bar chart and a staff allocation chart the differences between release planning and iteration planning.

Question 9. [9 points].
9.a (3/9 points) *Explain step by step* the process of requirements engineering, with special emphasis on *which* actors and/or stakeholders are responsible of each step of the process and *what* are their responsibilities.

9.b (3/9 points) *Compare* functional and non-functional requirements. Indicate at least two (2) *differences* between them.

9.c (3/9 points) Consider the construction of a software for controlling a snowplow machine (A snowplow is a device intended for mounting on a vehicle, used for removing snow and ice from outdoor surfaces, typically those serving transportation purposes.). *Describe* one (1) safety requirement and *motivate why* it is functional or non-functional.

CONTINUES NEXT PAGE


BookingManager
-manager: DriversManager
-drivers: driversList
-clients: clientsList
<<static>> getInstance()
+addCustomer(c:Customer)
+addDriver(d:Driver)
+registerBooking(c:Customer,d:Driver)
<<Interface>>
DriversManager
+name: String
+createDriver(): Driver
+bookDriver()
DriversManager: bookDriver - Function:
driver.book()
TaxiDriverRegistrar
+createDriver(): TaxiDriver
OcasionalDriverRegistrar
+createDriver(): OcasionalDriver
Client
+name: String
+address: Address
+converter: AddressLocationConverter
+addAddress(a:Address)
+addAddress(l:Location)
Address
+street: String
+city: String
+state: String
+country: String
+PO: Numeric
Cient: addAddress(l:Location) - Function:
address = converter.LocationToAddressConverter(l)
self.addAddress(address)
Driver
+name: String
+carLicensePlate: Numeric
+book()
TaxiDriver
+license: Numeric
+book()
OcasionalDriver
+personnummer: Numeric
+book()
AddressLocationConverter
+addressToLocation(a:Address): Location
+locationToAddress(l:Location): Address
GeographicLocation
+latitude: Numeric
+longitude: Numeric
uses
uses
1
*