1 UML/RUP

Please mark the following statements as true or false (+1p for correct answer, no change for wrong answer):

A class diagram describes how classes and objects collaborate

- True
- False

Use cases are written after you have implemented the system to have something to test against

- True
- False

Interaction diagrams show the method calls that objects make on other objects

- False
- True

Communication diagrams show how users communicate with the system

- True
- False

In a class diagram you can see all the values of all attributes for all classes

- True
- False

In a sequence diagram you can see the objects that are required in order to solve a specific system event,

- False
- True

In a sequence diagram you can see the classes that the objects are instances of

- False
- True

A system sequence diagram is a special case of a class diagram

- False
- True

Methods in a class can be public, protected, or private.

- False
- True

A method declared as public is not allowed to use private attributes in the same class

- True
- False

If the object alpha of the type Greek should be able to call methods in the object aleph of the type Hebrew, then there must be at least be an association between the classes Greek and Hebrew

- False
- True

A system is not ready for delivery if not all use cases are fully implemented.

- False
- True

Design patterns describe how to solve common interactions between the users and the system

- False
- True

Maximum marks: 13

2 Interaction Diagrams

Please mark the following statements as true or false (+1p for correct answer, no change for wrong answer):

A collaboration diagram describes the same thing as a sequence diagram.

- False
- True

A sequence diagram describe the interaction between different classes

- False
- True

In a sequence diagram you list all the attributes and their current values under each object's life-line

- False
- True

You make one interaction diagram for each system event.

- True
- False

You extract system events from use cases by making a system sequence diagram

- False
- True

All objects in an interaction diagram must have a variable name and a class type. For example name:Type

- False
- True

You can always use state diagrams instead of interaction diagrams

- True
- False

You create one system sequence diagram for each use case

- True
- False

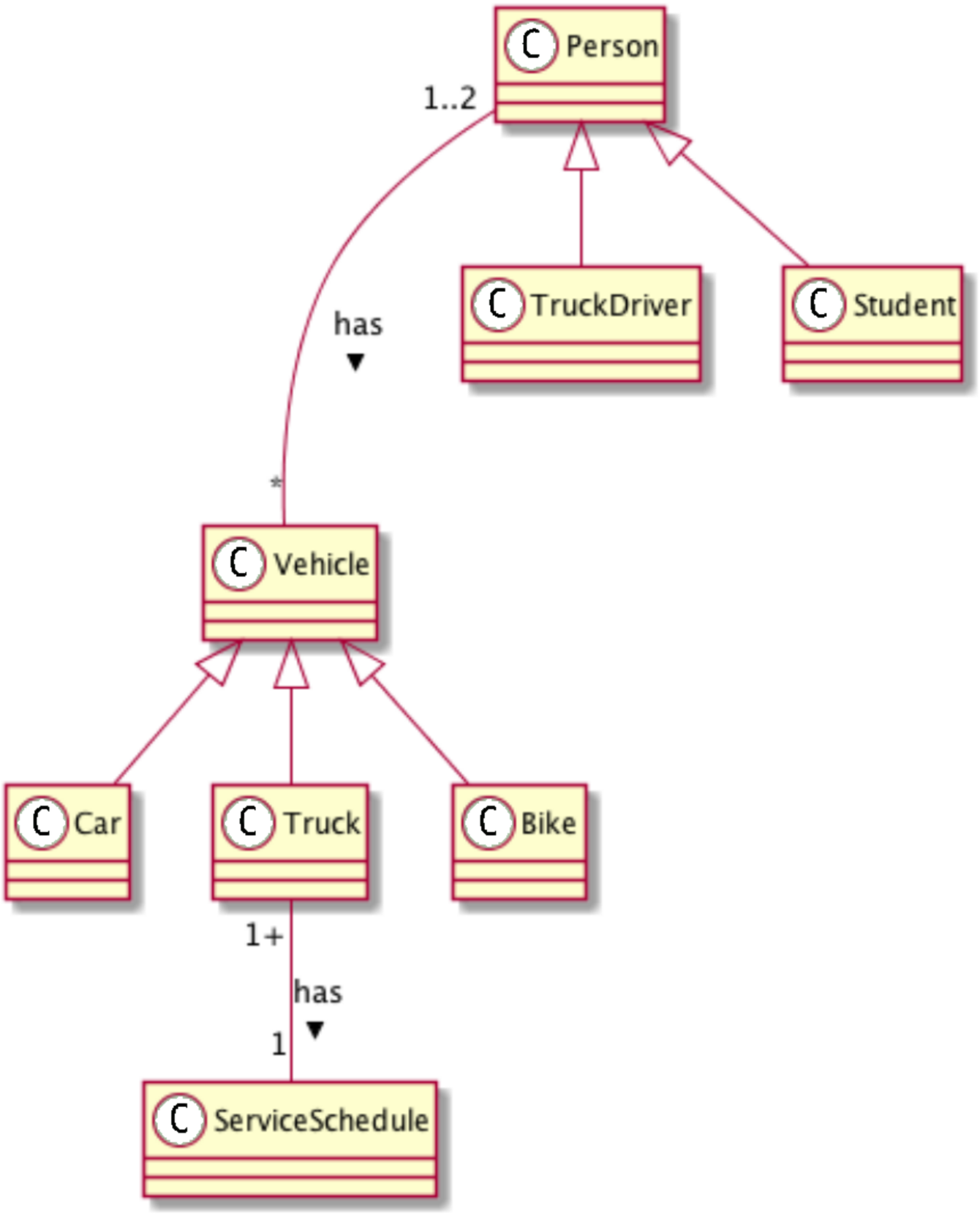
The purpose of a system sequence diagram is to identify how different actors interact with each other.

- False
- True

Maximum marks: 9

3 Class Diagram

Consider the following class diagram:



Then, please mark whether the following statements are supported (true) or not supported (false) by the diagram (+1p for correct answer, no change for wrong answer):

Tom, who is a Student, owns mater:Truck

- False
- True

Luigi, who is a Car, is owned jointly by Eve:Person and Ewan:TruckDriver

- False
- True

Every Truck has at least one ServiceSchedule

- True
- False

One ServiceSchedule is only applicable to one Truck at the time

- True
- False

A Vehicle must either be a Car, a Truck, or a Bike

- False
- True

Hudson, who is a Person, owns Ramone:Car, Flo:Car, and Sarge:Truck

- True
- False

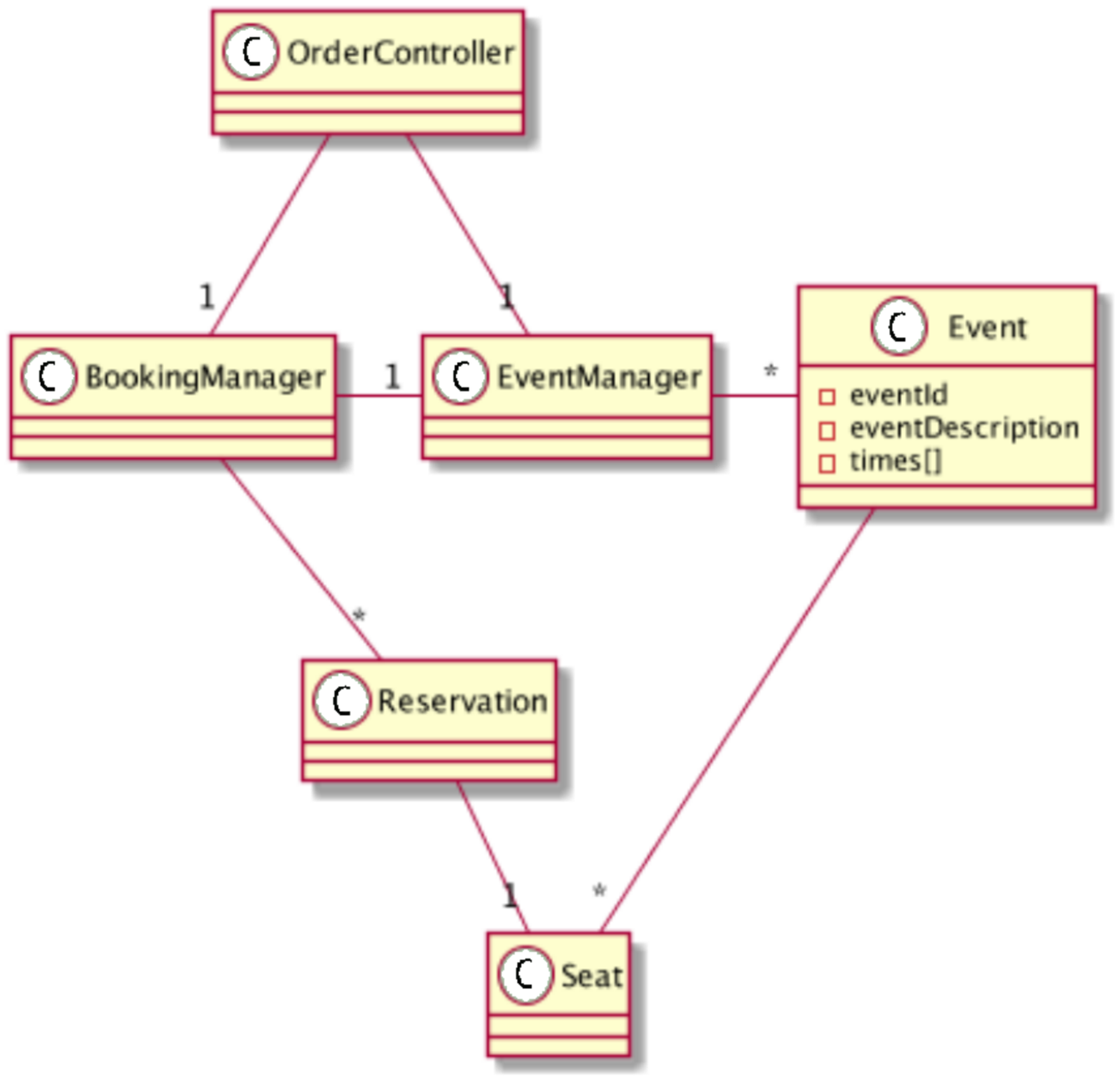
The Students Jack, Jill, and Jim own a Bike together

- False
- True

Maximum marks: 7

4 GRASP Patterns

Consider the following class diagram. Then, select the most suitable class for the described responsibility, and why.



The class Select alternative (OrderController, Event, BookingManager, Reservation, Seat, EventManager) is most suited to receive the system event **createBooking()** according to the

GRASP pattern Select alternative (Information Expert, Controller, High Cohesion, Creator, Low Coupling)

The class Select alternative (Reservation, Seat, OrderController, EventManager, Event, BookingManager) should contain the method **searchBooking()** according to the GRASP pattern

Select alternative (Controller, Information Expert, Creator)

Before creating a new Reservation, Select alternative (Reservation, Seat, OrderController, BookingManager) must find a Seat on an Event. This class is assigned this responsibility

according th the principle of Select alternative (Low Coupling, High Cohesion, Polymorhphism, Pure Fabrication)

Maximum marks: 6

5 Design Patterns

Please mark the following statements as true or false (+1p for correct answer, no change for wrong answer):

Singleton means you are only allowed to call the class once

- True
- False

Strategy pattern uses polymorphism

- False
- True

Abstract Factory is used to create the right type of object given a specific context, and where the rest of the system does not need to know exactly which type the object has.

- True
- False

When using the Layered architecture style, you must always have a GUI layer, a logic layer, and a data layer.

- False
- True

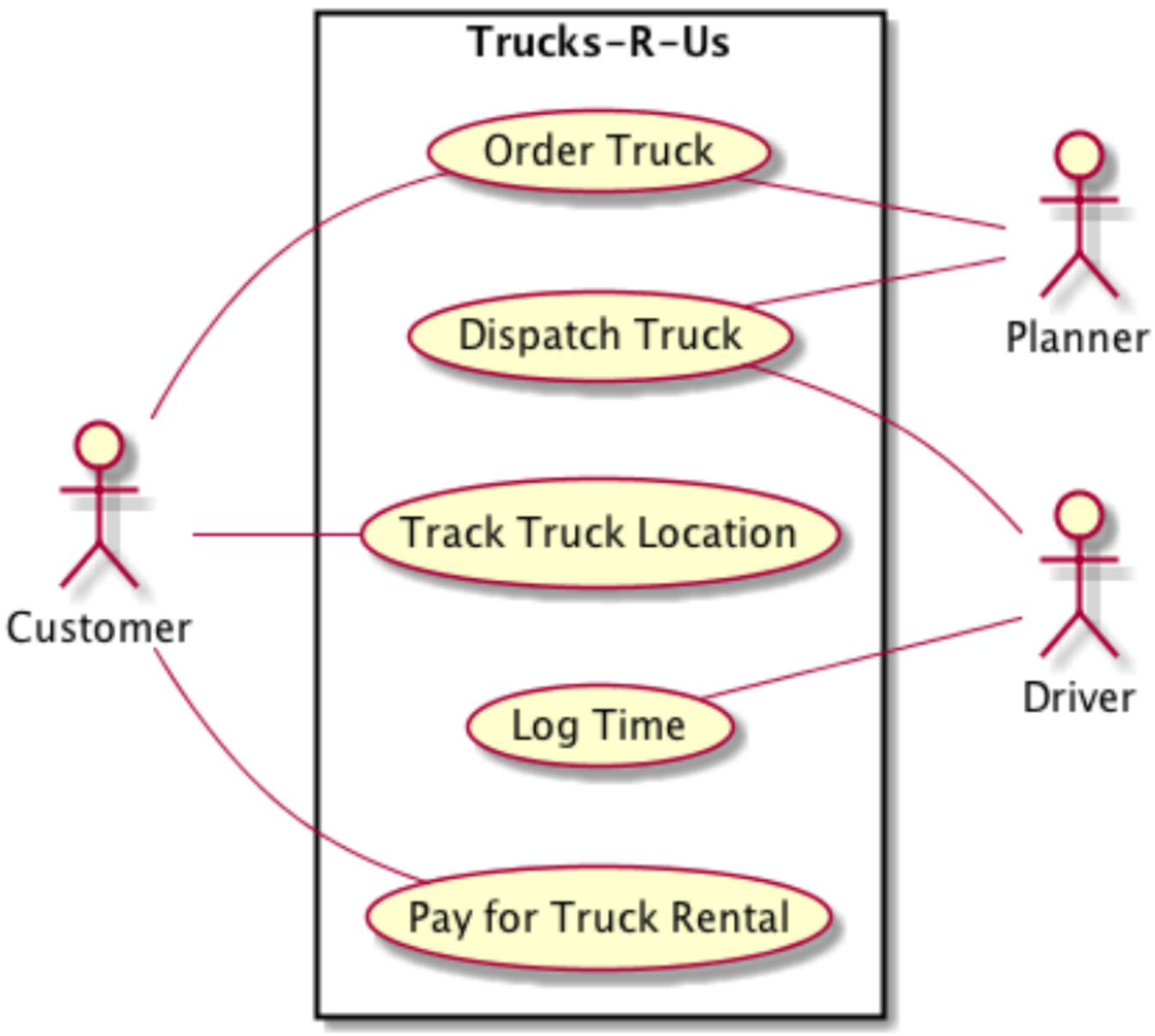
In Model-View-Controller you have several models that are presented in the same View.

- False
- True

Maximum marks: 5

6 Use Case Diagram

Consider the following class diagram:



Then, please mark whether the following statements are supported (true) or not supported (false) by the diagram (+1p for correct answer, no change for wrong answer):

The Customer never meets the Driver

- False
- True

The Planner interacts with the Driver to dispatch a truck

- False
- True

In order to track a truck's location you must first have ordered a truck

- False
- True

To pay for a truck rental the Driver must first have logged the time in the system.

- False
- True

Maximum marks: 4

Grade limits

Betygsgränserna för denna tenta är:

Betyg	Procent	Poäng
MAX	100%	44
A	90%	39
B	80%	35
C	70%	30
D	65%	28
E	60%	26

Lycka till!