



BLEKINGE INSTITUTE OF TECHNOLOGY

Ludwik Kuzniarz
Blekinge Institute of Technology
School of Computing
Karlskrona

August the 28th, 2014

Course PA1415 Software Design Second Exam

Points

Question 1	Question 2.1	Question 2.2	Total

Grade

BTH	ECTS

----- Explanations -----

Questions.

For the multiple choice questions your task is to indicate the following statements as *true* T or *false* F by placing the appropriate letter indicator in the [].

For instance

[T] John is English

indicates that the statement is true, or more precisely you think it is true.

John is

[F] Swedish

[T] English

[F] 5 years old

indicates the John is not Swedish, he is English and he is not 5 years old,

If you know that John is German and 20 years old, you should made the following indications:

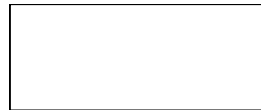
John is

[F] Swedish

[F] English

[F] 5 years old

For the problem questions your answers should be written in the predefined marked places



either labelled boxes

or along labelled lines

Well structured answers will be appreciated.

All questions should be answered in the context of the course, i.e. the terms and phrases used in the text have the meaning defined and used in the course materials provided in Itslearning.

Marking

Every question, after the question number, has a number of points allocated for that question.

If all entries for the question are filled correctly you obtain that number of points.

For any wrong answer for the multiple choice question, one point is subtracted from the number of allocated points but no negative points are generated. It means that if a question has 2 points allocated and has three places to mark T or F then when you make one error you get 1 point for that question, when you make two errors you get 0 points and when all the answers are wrong you also get 0 points.

The exam is worth 60 points, from 31 points is E (passed), from 48 – A.

Allowed books

English – Swedish dictionary

A remark on drawings

In the case of tasks that require producing drawings – conceptual models, state diagram, class diagram – please *draw first your draft solutions on a spare paper* and then *redraw them* on the marked area on the examination paper trying to arrange the elements (and especially connecting lines) of the picture so that the models were easy readable.

So, good luck!

1. Knowledge

25 p

1. 1 p
A Development Process describes
 - order of activities the have to be performed during the development
 - language used for the implementation of the design artefacts
 - life cycle (how the activities are ordered and performed)

2. 1 p
Unified Development Process can be characterized that:
 - is based on the Water-Fall model
 - is iterative
 - is incremental

3. 2 p
The tasks during producing conceptual model are supposed to
 - to identify and model concepts,
 - to identify and model attributes allocated to the concepts,
 - to identify and model operations allocated to the concepts,
 - to identify and model methods allocated to the concepts.

4. 2 p
The recommended post-condition categories for a System Operation include:
 - deleted objects,
 - created objects,
 - established links between objects,
 - modified attributes of objects .

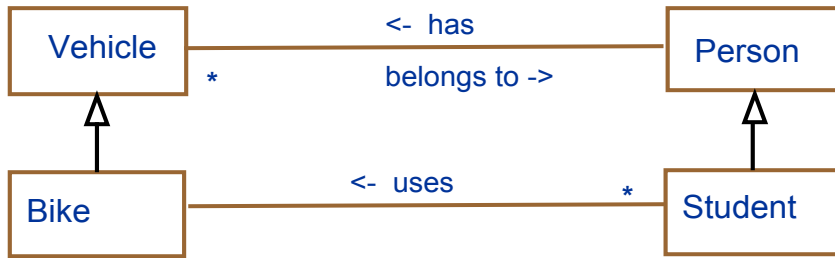
5. 2 p
The tasks performed during the Requirements Analysis include
 - drawing UseCase Diagrams,
 - finding Actors interacting with the system,
 - describing UseCases
 - identification of messages sent within the system

6. 2 p
Interaction Diagrams include:
- collaboration diagrams
 - state diagrams
 - class diagram
 - sequence diagrams
7. 2 p
Purpose of producing System Sequence Diagrams is
- to identify system operations,
 - to illustrate realisation of the system operations,
 - to illustrate collaboration between actors interacting with the system
 - to illustrate interactions between actors and the system.
8. 2 p
Collaboration Diagram:
- is prepared for a system operation to illustrate messages sent between objects inside the system,
 - starts with a message representing a system operation,
 - can be replaced by a sequence diagram,
 - can be replaced by the state diagram.
9. 2 p
Controller object is responsible for
- dispatching (properly redirecting) messages coming to the system
 - ensuring proper interaction with the system by actors using the system
 - allocating resources to collaborating objects
 - controlling the values computing during the execution of system operations

10.

9 p

Consider a domain described by the model



Evaluate as true or false the following statements about the domain described by the above model:

- Tom, who is a Person, does not have a vehicle.
- John, who is a Person has two vehicles: "Volvo v70" and "BMW 325".
- Vehicle "Ford Mondeo" does not belong to anyone
- Bike "Torpedo" belongs to Paul who is a person
- Mary who is a student has a vehicle "Toyota Corolla"
- Jane who is a student has a bike "Speedo"

- Every person must have a vehicle
- Every vehicle must belong to some person
- Every bike must belong to a student
- A student may use a bike
- A student must use a bike
- A bike may be used by two students

- Number of vehicles that can exist must be greater than the number of persons.

2. Skills

35 p

2.1. Modelling

2.2. Design

2.1. Modelling

10 p

2.1.1. Domain Description.

Draw a conceptual model for the following fragment of the problem domain.

- In ACampus lives a community of persons – teachers and students. Every person has a name and address.
- A teacher can work at a university.
A student studies at university.
- Each university consists of from 3 to 8 departments.
- Student has a document – “legitimation” - indicating the faculty and the year she/he is studying.
- Teacher has a salary, may be different in different university.
- Teacher can work at many universities but student can study only in one university.
- At a given moment teacher can be unemployed.
- There are two sorts of vehicles – bicycles and motorbikes.
- Every vehicle has two wheels. A motorbike has also an engine.
- A teacher can have any number of vehicles but student can have only one bicycle.

2.1.2. Your task

Draw conceptual model for the above fragment of the problem domain

In your model use advanced modelling features.

Domain Model

2.2. Design

25 p

Super Banking

In the Super bank there is a number of clients.

To become a client a person has to be registered by a bank manager, who introduce the client data to the system and gives him an id and password.

Every client has identification.

And may have a number of accounts in the bank.

Accounts are open by a bank manager but can be closed directly by a client.

There could be normal accounts and saving accounts.

Client of the bank can perform a number of transactions- pay the money to the bank, withdraw money from an account, transfer money from one account to another account, apply for a loan to a client.

Banking system keeps record of all the transactions done by its clients.

Bank keeps two special list of the clients – *rich*: those who have savings for more than 1 M SEK and *good* : those who own more than 100 000 SEK to the bank.

Every client can use a banking system to perform the transactions.

At the beginning of each transaction the client has to log in to the system by providing an id and a password.

To withdraw money, the client has to use an ATM machine. He has to identify herself using her bank card and providing relevant password(code) stored on the card. She also has to indicate the amount of money to be delivered. All those information are verified by the banking system. After a successful withdraw, the bank account is updated accordingly. In case of three incorrect identification attempts the card is blocked.

2.2.1. Your tasks

A	Identify Actors	1 p
B	Identify Use Cases	1 p
C	Draw Use Case Diagram	1 p
D	Evaluate Use Cases and provide Development Plan	1 p
E	Write Extended version of <i>WithdrawMoney Use Case</i>	3 p
F	Draw Conceptual Model	4 p
G	Identify System Operations based on <i>WithdrawMoney Use Case</i>	3 p
H	Draw Design Class Diagram including elements relevant for <i>WithdrawMoney Use Case</i>	4 p
I	Write a Signature and Contract (text) for a <i>GetMoney</i> System Operation from <i>WithdrawMoney</i> usecase	3 p
J	Write Contract for the <i>GetMoney</i> operation (from p.I) using Stage Metaphor (picture)	4 p

C. Use Case Diagram

2 p

D. Evaluation of Use Cases

1 p

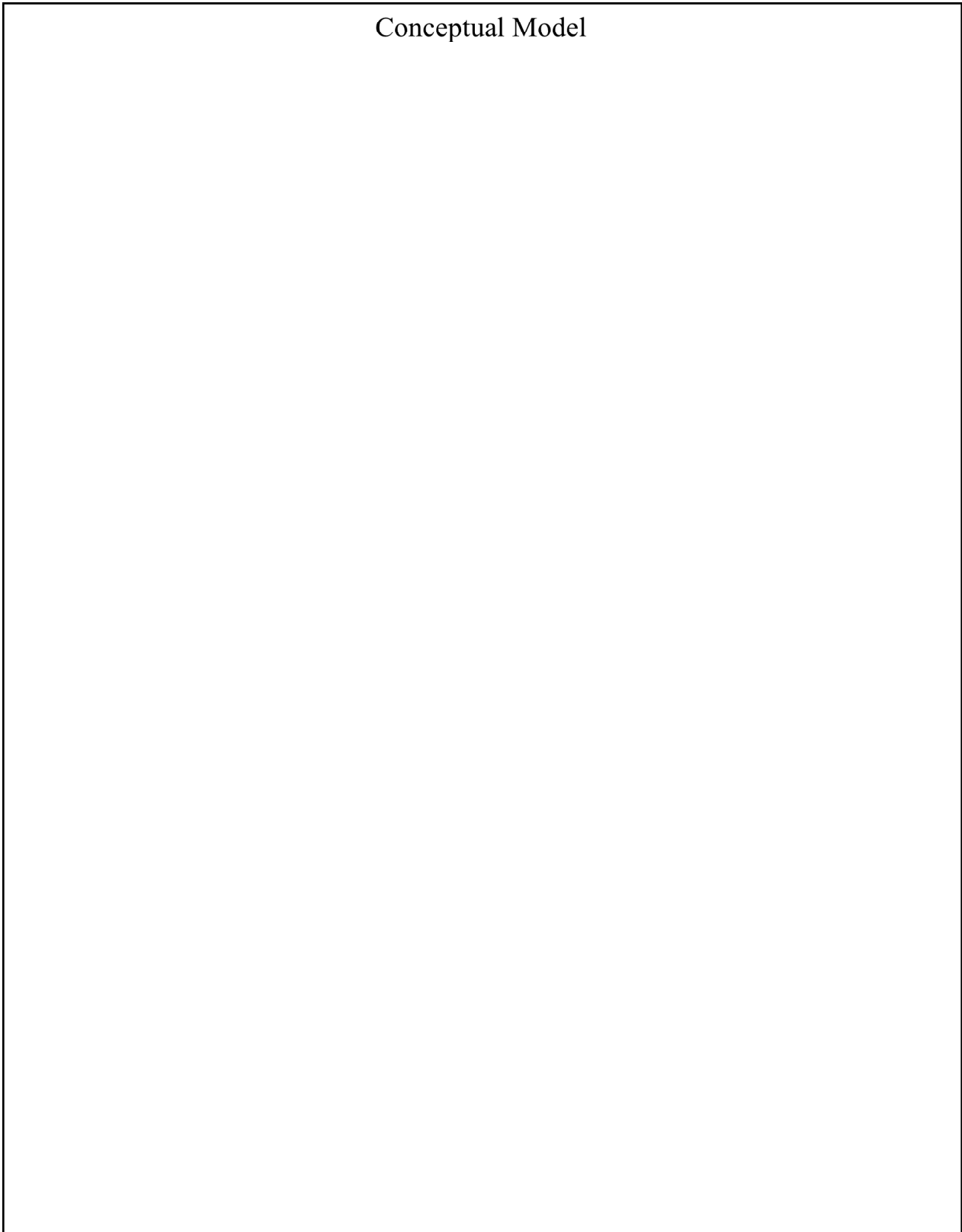
.....
.....
.....
.....
.....
.....
.....
.....
.....

Development process

F. Conceptual Model = Domain Model from the previous question

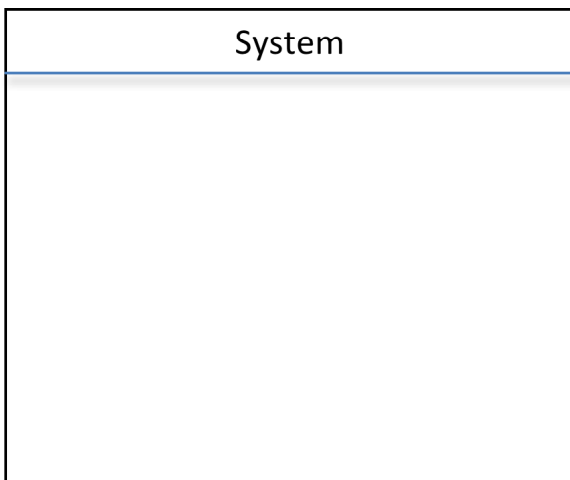
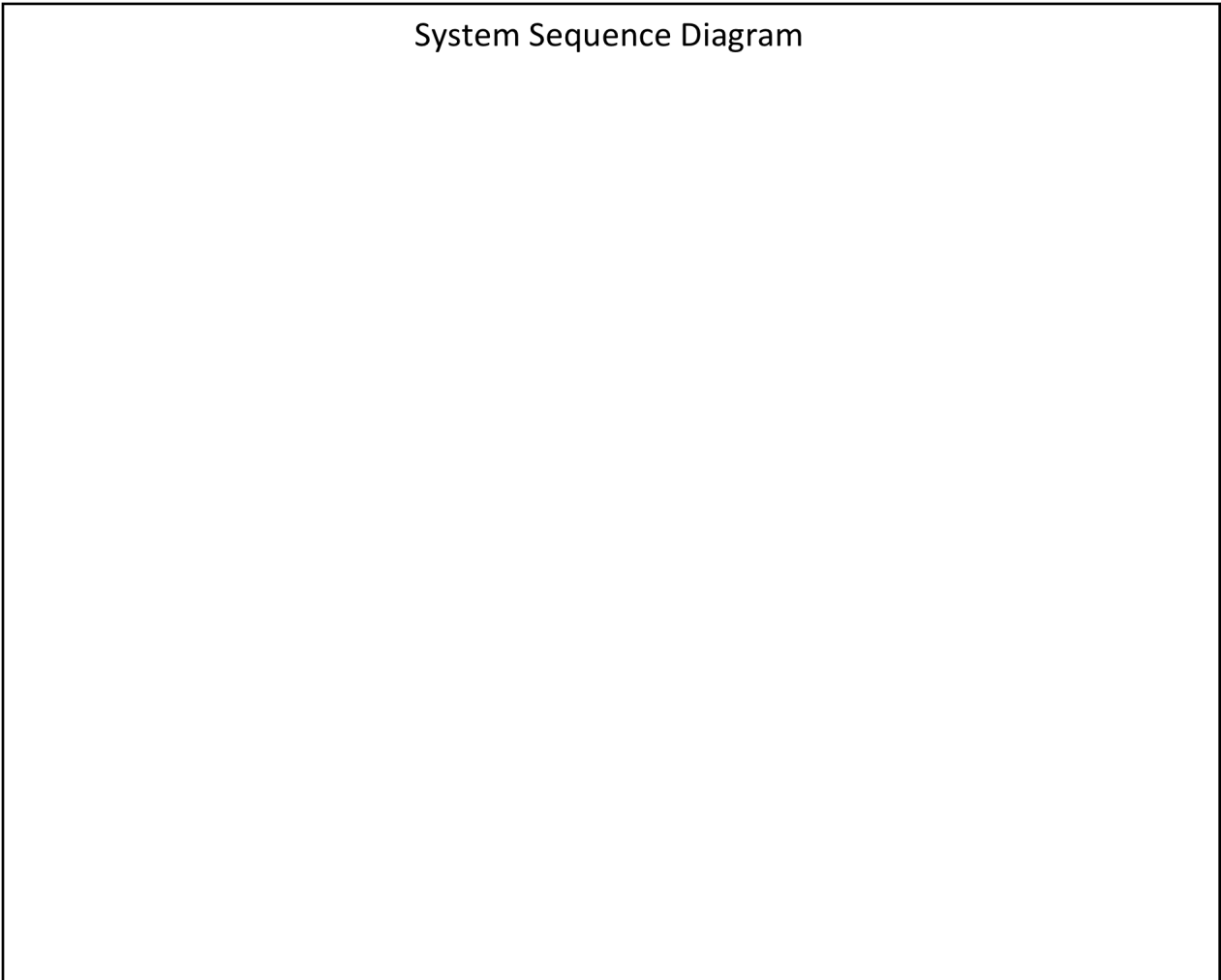
4 p

Conceptual Model

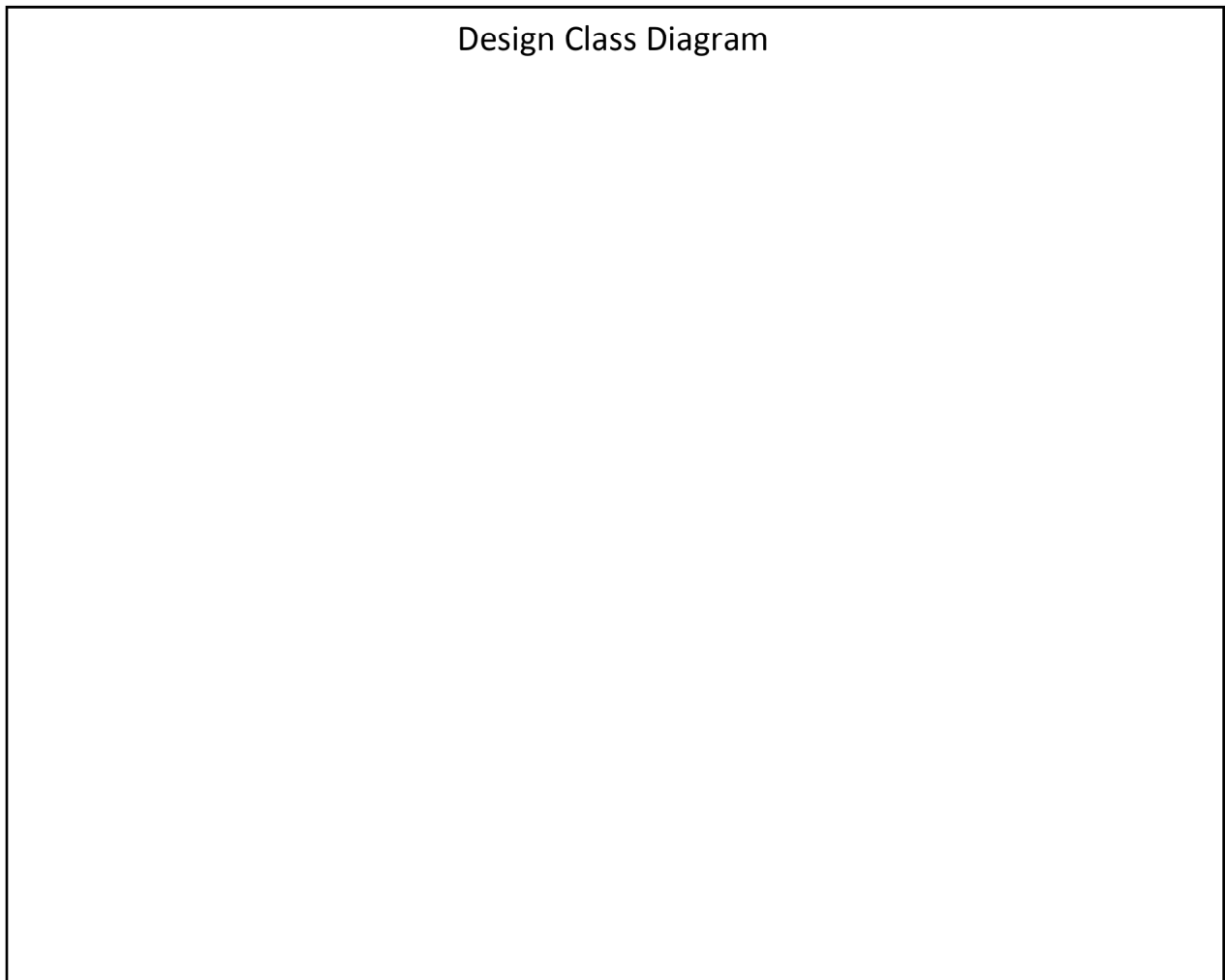


G. System operations
System Sequence Diagram for *WithdrawMoney usecase*

3 p



- H. Design Class Diagram including elements relevant for WithdrawMoney use case 4 p



- I. Signature for the getMoney operation 3 p

Hint. Signature for an operation specifies the name of the operation, arguments and returned value

.....
.....

Contract for the getMoney operation - text

.....
.....
.....
.....
.....
.....

J. Contract for the *Chosen* operation
using *Stage Metaphor*

3 p

