Projektkursus Systemudvikling 2015

Hjemmeside med bookingsystem til hostel

Delrapport 1 del 2

23/03/2015

Gruppen:

Rose Sofie Greve 250493

Jeppe Schönemann Skov 240992

Frederik Leed Henriksen 110394

Instruktor: Christoffer Wadum

Udvalgte afleveringsopgaver

1-6

The TicketDistributor must enable a traveler to buy weekly passes.
Er et funktionelt behov, da kunden skal kunne købe dette er der et behov for at systemet understøtter dette.

The TicketDistributor must be written in Java.
Er ikke et funktionelt behov, da det ikke direkte er relateret til systemets funktionalitet, det kunne ligeså godt være kodet i C++, uden at funktionaliteten havde være anderledes.

The TicketDistributor must be easy to use.
Dette er ikke et funktionelt behov, det ville være en rigtig god idé at systemet var nemt og enkelt at benytte, men kunden kan efter grundig gennemgang lære at benytte systemet.

The TicketDistributor must always be available.
Dette er et funktionelt behov, hvis en automat er i stykker skal der være mulighed for at få billetter på andenvis, dette kunne være igennem app, eller køb over nettet.

The TicketDistributor must provide a phone number to call when it fails.
Et funktionelt behov, da kunden skal kunne give besked hvis systemet fejler.

1-8

"Assume you are developing an online system for managing bank accounts for mobile customers. A major design issue is how to provide access to the accounts when the customer cannot establish an online connection. One proposal is that accounts are made available on the mobile computer, even if the server is not up. In this case, the accounts show the amounts from the last connected session."

Den første gang accounts bliver nævnt:

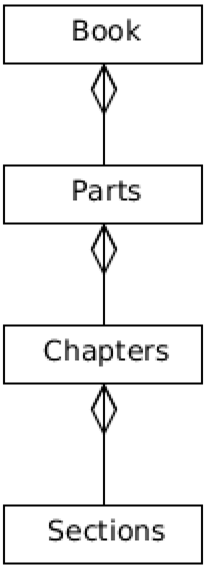
“Assume you are developing an online system for managing bank accounts for mobile customers."
Her er det en application domain concept, da vi modellere en abstrakt model af problemformuleringen.

Dette er også en application domain concept, da vi opstiller et problem.
"A major design issue is how to provide access to the accounts when the customer cannot establish an online connection."

De sidste to her er solution domain concept, da de opstiller nogle løsningsforslag.
"One proposal is that accounts are made available on the mobile computer, even if the server is not up. In this case, the accounts show the amounts from the last connected session.”

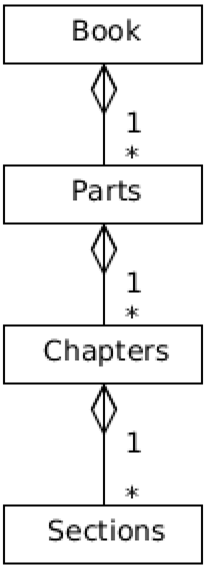
2-6

Draw a class diagram representing a book defined by the following statement: “A book is composed of a number of parts, which in turn are composed of a number of chapters. Chapters are composed of sections.” Focus only on classes and relationships.



2-7

Add multiplicity to the class diagram you produced in Exercise 2-6.



2-9

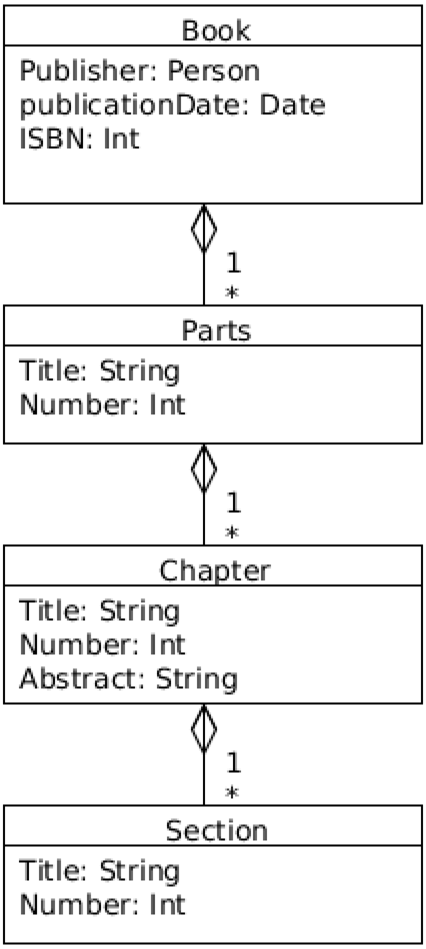
Extend the class diagram of Exercise 2-6 to include the following attributes:

• A book includes a publisher, publication date, and an ISBN

• A part includes a title and a number

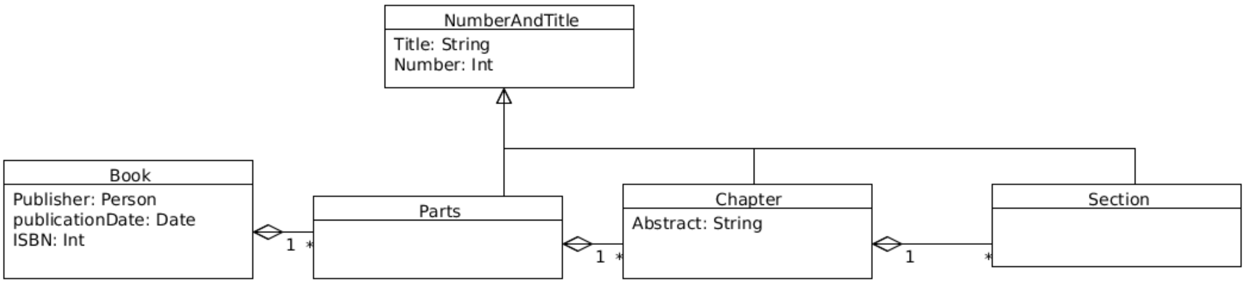
• A chapter includes a title, a number, and an abstract

• A section includes a title and a number.

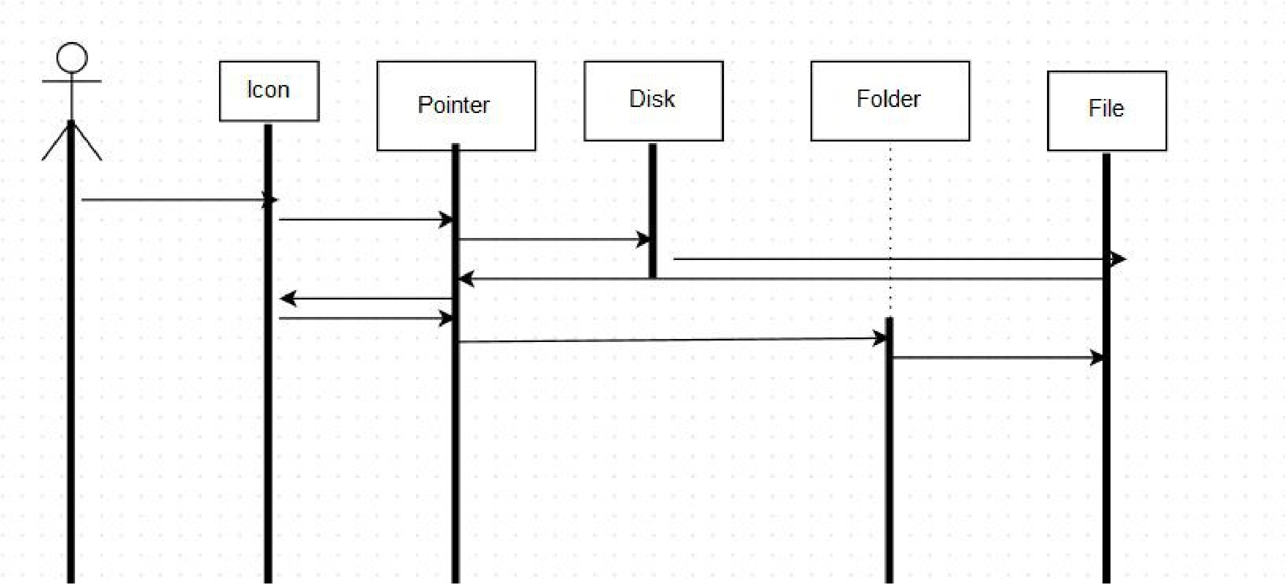


2-10

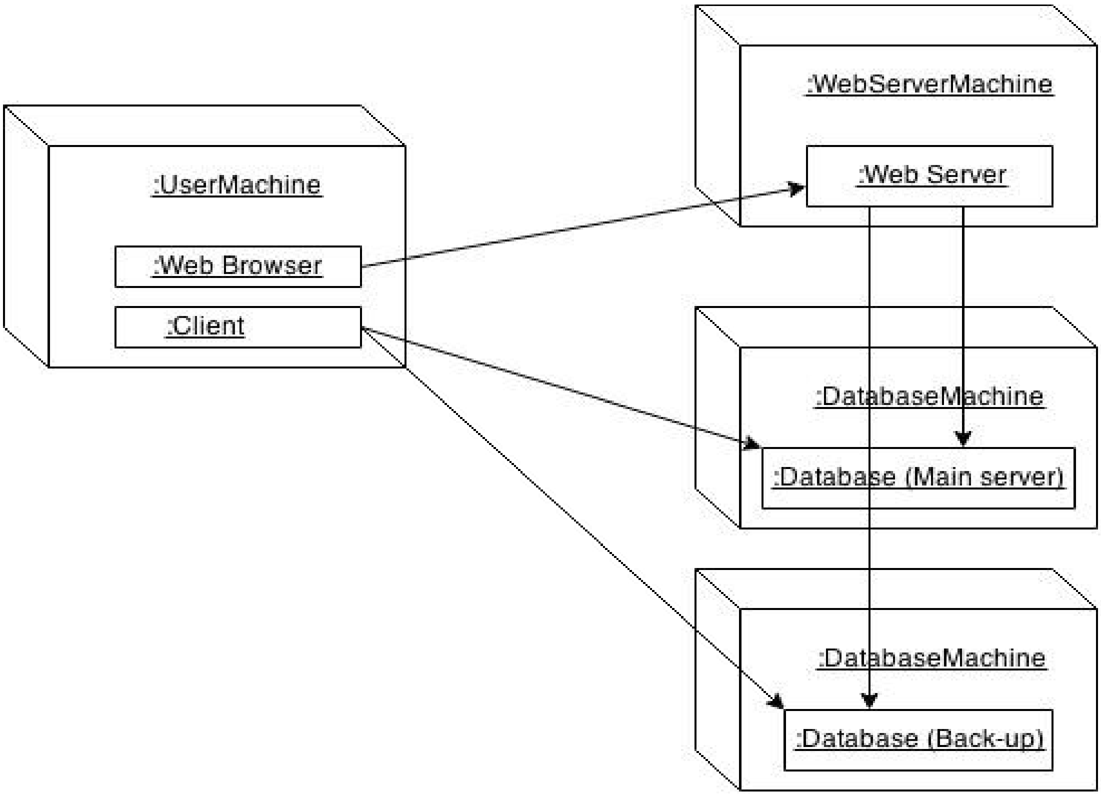
Consider the class diagram of Exercise 2-9. Note that the Part,Chapter, and Section classes all include title and number attributes. Add an abstract class and an inheritance relationship to factor out these two attributes into the abstract class.



5-3



7-1


:UserMachine
:Web Browser
:Client
:WebServerMachine
:Web Server
:DatabaseMachine
:Database (Main server)
:DatabaseMachine
:Database (Back-up)