

Object-oriented system design with UML

Homework exercises

1 Functional modelling with use cases

1.1 Training request management system

The following text describes the process for managing employee training requests in the WorkHard Company, Ltd.

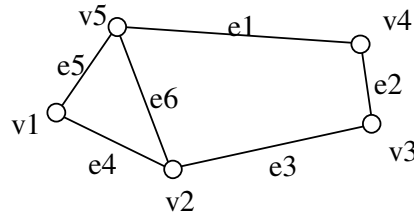
1. The training process is initiated when the Training Manager receives a request for training from an employee.
2. The request for training specifies the employee's name, position and corporate extension number, as well a description (as detailed as possible) of the desired training.
3. Based on budget constraints and the employee's position and previous training history, the Training Manager approves or refuses the request, and informs the employee.
4. If the request is approved, the Training Manager selects a suitable course from an approved catalogue, and provides the employee with detailed course information and session dates.
5. The employee selects a session and informs the Training Manager, who contacts the training company and enrolls the employee in the course.
6. If the employee cannot attend an enrolled session, he/she must inform the Training Manager as soon as possible to cancel the inscription.
7. At the end of the training session, the employee must supply an attendance certificate (signed by the training company) and an evaluation of the course to the Training Manager.

Your job is to prepare a functional model for a new intranet system to replace this offline process. The request management system must present available courses to the employees, and enable the training manager to manage them more easily.

2 Structural modelling with class and object diagrams

2.1 Undirected graph

The following is an example of an undirected graph.



2.1.1 Prepare a class diagram to describe the general structure of any undirected graph. An undirected graph consists of a set of vertices and a set of arcs. Each vertex and arc is named. Each arc links a pair of vertices. Your model does not have to include the graphical layout of the graph.

2.1.2 Prepare an object diagram to describe the specific structure of the example graph. Your object diagram must be coherent with your class diagram.

2.2 Airline reservation system

The following statements describe the simplified data structures of an airline reservation system. Prepare a class diagram and carefully motivate your choices of data structures.

1. Different airline companies propose different flights
2. An airline company may open and close a flight for reservation.
3. A client can reserve multiple flights for different passengers.
4. Each reservation concerns one flight and one passenger.
5. A reservation can be cancelled or confirmed.
6. A flight has a departure airport and an arrival airport.
7. A flight has a date and time of departure, and a date and time of arrival.
8. A flight may include stopovers in different airports.
9. A stopover has an arrival time and a departure time.
10. An airport serves one or more cities.

2.3 File system

A simplified file system can be described as follows:

1. A file may be an ordinary file or a directory.
2. An ordinary file contains a sequence of bytes.
3. A directory contains other directories and/or ordinary files.
4. Each file has a name.
5. Each file has an owner, which is a user of the system.
6. Users are arranged into groups; each user belongs to exactly one group.
7. Each file has a set of permissions that determine the rights of its owner and the other members of the group. The system supports read and write permissions.

3 Dynamic modelling with sequence diagrams

3.1 File system

Refer to the structural model developed above.

3.1.1 Prepare a sequence diagram showing the creation of a new file. Assume that you have a reference to the current working directory, the user creating the file, and a mask of default file permissions.

3.1.2 Prepare a sequence diagram showing the deletion of a file by a user having the necessary permissions to do so.

3.2 Access control system

A Parisian engineering school decides to protect sensitive zones of its premises (for example the computer rooms) with an access control system. The system consists of self-closing doors that can be opened by sliding an ID badge through an associated wall-mounted card reader.

Doors are organised in groups (e.g.: lecture halls, computer rooms, staff rooms). Each door belongs to exactly one group. Users of the system are also arranged into groups (e.g.: staff, students), and a user may belong to multiple groups. Access rights are assigned between groups of doors and groups of users. Each group of users may be granted rights to multiple door groups, and vice versa.

The system maintains a log of both failed and successful access attempts by users.

3.2.1 Prepare a sequence diagram showing legitimate entry into a room protected by this access control system.

You may find it useful to prepare a class and/or object diagram of the data structures as well.

4 Modeling behaviour with statecharts

4.1 Wristwatch

The wristwatch illustrated below provides two functions: Time and stopwatch.



The watch operates as follows:

1. The STOPWATCH button allows the user to change between the time and stopwatch displays.
2. When the time is displayed, the MODE button is used to set the current time. Pressed once, it allows you to set the hours; pressed twice it allows you to set the minutes; pressed a third time it saves the new settings and returns to the time display.
3. While setting the hours and minutes, the SET button cycles incrementally through valid possibilities.
4. When the stopwatch is displayed, the MODE button starts and stops the stopwatch. When the stopwatch is stopped, the SET button resets the counter to zero.
5. The stopwatch remains active while the time is displayed.
6. When the watch is activated for the first time, the hours, minutes and seconds are all initialised to zero.

4.1.1 Prepare a sequence diagram showing the messages sent between the user and the watch in order to set the current time to 03h02.

4.1.2 Prepare a statechart showing all possible behaviours of the wristwatch.