

Frontend Web assignment (Individual study point exercises - 14 study points)

Hand-in: By Moodle.

Solution: Hand-in a zip file with documentation of a) source code and JUnit tests (NetBeans project as zip file); b) database (SQL scripts for database creation, insert of test data, possibly views and/or stored procedures).

Name the zip file "<user name>-<firstname_lastname>.zip"

Deadline: Sunday February 26th at 23.59

Introduction

In this assignment, you will be working individually on both functional and non-functional requirements for a small fictive library system.

Requirements to the development process

Quality must be built into the application, one story at time. Continued assurance of quality is achieved by continuous and automated execution of the aggregated testing. Test of the functionality (the user stories) can be done manually via the user interface, but data mappers must be tested via JUnit. It is important not to start implementing a new story before confidence in the existing user stories is assured.

Functional requirements

All students must implement user story 1 -2.
Yellow students must also implement user story 3.
Red students must also implement user story 4.
User story 5 is optional.

#1 As a borrower, I want to browse the library catalogue for a book by its ISBN

#2 As a borrower, I want to make a reservation for a book if I have a positive library status and haven't yet reached the maximum level of 10 reservations

#3 As a user, I want to register as a borrower.
Registration data: ssn, name, address, phone, mail and library status (default = true)

#4 As a librarian, I want to update the library catalogue with a new material. Material can be book, journal, DVD.
Registration data: isbn (books only), title, language, publisher, publish date

#5 As a librarian, I want to dispose of old journals and have them put into volumes at the end of each year.

Non-functional requirements

All students must implement story 6 -7.

Yellow students must also implement user story 8.

Red students must also implement user story 9.

6 Three layered architecture with a) MVC in UI layer (JSP + Servlet) and b) data mappers in data source layer must be implemented in order to obtain maintainability

7 Reliability must be high, i.e. invalid user input etc. must not be able to put the system in inconsistent state

#9 The user interface design must give high level of user satisfaction

8 Response time must be reasonable (response time above 10 seconds is not acceptable, and less than 4 seconds is preferred)