Software Engineering

Introduction

3 Ba INF 2018-2019

Stephen Pauwels
stephen.pauwels@uantwerpen.be

8/10/2018

1 Practical

During the exercise sessions we will study the different aspects of the development of high quality software. We will do this using a single software system that we use throughout the entire semester. This system will only be used as a guide through the semester, you will not be asked to implement the software system yourself.

Every week a new assignment will be given during the exercise sessions. These assignments have to be made in groups of 2 (except for the Formal Specifications assignment which has to be made individually), these groups are fixed for the entire semester. During the semester there are Report-Deadlines, you have to submit a scientific report about the previous assignments and send it in on Blackboard. These reports are not just the plain answers from the assignments. We expect you to write correct reports that are self-containing (ie. you should not need the assignments in order to understand the report and your proposed solutions).

After every report submission you will receive comments on your work. We expect you to use these comments to improve your future reports. In the end your overall progress and results will be graded. The only exception is the assignment about Formal Specifications, this assignment has to be made individually and only your results count.

We will study the following subjects during the exercise sessions:

Subject	Report	Groupsize	Deadline (23u55)
Project Management	1	2	4 November
Use Cases	1	2	4 November
Domain Models	1	2	4 November
Testing	2	2	18 November
Design By Contract	2	2	18 November
Formal Specifications	3	1	26 November
Software Architecture	4	2	2 December
Quality Control	4	2	2 December
Software Metrics	5	2	16 December
Refactoring	5	2	16 December

2 Case: Online Shop

The project we will consider an Online Shop. We will mostly focus our attention on the back-end of the system, which has to be implemented in Java. A detailed description of the software requirements are given below.

2.1 Requirements

2.1.1 Functional Requirements

- The system must have following entities in order to describe the current situation of orders and clients:
  - Items: items that can be sold. An item has a name, price and category (electronics, home & living, hifi, ...).
  - Cart: the current items a client has chosen but not yet ordered.
  - Order: a single order made by a client.
  - Clients: all registered and unregistered clients that already ordered something.
- Following categories must exists, but it should be possible to add new categories to the store:
  Phone, Tablet, Computer, Image, Sound, Home & Living, Kitchen, Travel, Fashion, Sport
- All items must be displayed in a catalog, ordered by category. Every category should be displayed on a separate page. A user can sort the items according to name, popularity or price.
- A user can add/remove items to/from the cart and change the desired quantity.

- A user has the ability to order the items in his cart. The user then has to fill in some standard information and afterwards select a payment method.
- When a payment has been received a confirmation mail has to be send to the user.
- An administrator must be able to make changes to placed (but not yet delivered) orders.
- An administrator should be able to get an overview of all open and delivered orders.
- A user is able to log in and get an overview of his personal information, his mailing preferences and his open and delivered orders.
- An overview of some of the most popular items should be displayed on the home page of the shop.
- An administrator must be able to get some statistics and analysis of previous orders. Such as: for every item the total amount sold, find items that are often sold together. Required is the Eclat algorithm for itemset mining. Also recommendations should be given when a user is looking for an item.
- A user should be able to cancel his order as long as it is not yet ready for delivery.

2.1.2 Non-Functional Requirements

- Programming Language: Java
- Web-based UI
- Adaptability: UI should be displayed nice on every device
- Availability
- Robustness
- Easy-to-use
- No inconsistent states should occur