Laboratory in Software Engineering -61143

**1. Introduction :**

The requirement analysis stage of project developing is consists of three phases:

* Elicitation : *Understand the problem*
* Specification : *Describe the problem*
* Validation :*Test the solution*

On this lab we will talk in details about the third phase of requirement analysis process.

When you have identified the ways in which a use case may fail, you have to write the acceptance tests (it is recommended to write them together with the customer).

Conceptually, an acceptance test case is very simple. It defines correct operation of the system by stating:

* The initial state of the application
* The various inputs from the users
* The final state of the application
* The various outputs from the application

Let’s see on the sample use-case of the Training System:

**Use Case UC1**

**Name:** Add a customer to the SFS

**Actors:** Customer(principal).

**Goal Description:** Add a new customer to the SFS.

**Pre-Conditions:**

1. Customer is not already in the SFS

**Success story (Happy path)**

1. Customer requests to register to the SFS
2. SFS presents a form for entering the customer details: name, e-mail, desired account name and desired password.
3. The customer enters the required details and calls for saving.
4. SFS validates the supplied information, checks that there is no existing customer with the supplied account name.
5. SFS prompts for confirmation
6. Customer confirms
7. SFS creates new customer account and updates its database accordingly.

**Variations:**

4a. There is already a customer with the same account name in the SFS.

4a1. SFS offers the customer to enter another account name.

4a2. Continue as specified in description from 3

6a. Customer does not confirm

6a1. SFS continues to item 2 without clearing the details entered by customer

**Exceptions:**

3a. Customer cancels the operation

3a1. No changes made to the SFS

To make acceptance tests based on the given Use Case we look at its

**Pre-Conditions:**

Customer is not in the SFS

To check its validity we use the following customers:

|  |  |
| --- | --- |
| Customer name | Does the customer exist in DB? |
| "Sara" | No |
| "Abraham" | Yes |

The corresponding test cases are shown in the following table:

|  |  |  |  |
| --- | --- | --- | --- |
| Test ID | Description | Expected results | Comments |
| AddCustumer1 | Enter: menu option “Register”  Name: Sara Cohen  e-mail: sara@gmail.com  customer name: Sara  password: 12345  Enter: confirmation | System creates new customer account and updates its database accordingly | customer “Sara” is not in the System database |
| AddCustumer2 | Enter: menu option “Register”  Name: Abraham Cohen  e-mail: abraham@gmail.com  customer name: Abraham  password: 54321 | System offers the customer to enter another account name | customer “Abraham” exists in DB |

You can see acceptance tests for the CoffeeMaker Example here: <http://agile.csc.ncsu.edu/SEMaterials/tutorials/coffee_maker/blackboxtests.html>

Recommended site for acceptance tests: <http://www.informit.com/articles/article.asp?p=26652&redir=1>

**Task to do:**

Based on the Use Cases of the Training System (from Lab1), write acceptance tests for two first use cases (**Log in** and **Lecturer sends a request**) using flow of events description given in Lab1\_solution.doc. Consider all paths and exceptions presented in the use case description.

Prepare tables of tests and of data in the format presented in this file.

**Attachment**

**User story of the Training System**

Implement a qualification/training automatic system for the lecturers in Braude College.

The system must work as specified:

1. a lecturer can ask for a course/conference he wants to attend. He must fill a form with the required fields: his name, department, the requested topic he wants to attend. Optionally he may mention the specific course, place and time if he knows it.
2. The request is then sent (by fax or e-mail) to the dept. chief. If approved it is submitted to the training manager of the college who may find an appropriate course based on worker's request. The training manager does not receive a request directly from the lecturer, only from the chief of department.
3. The training manager sends either a confirmation –if the lecturer requested a specific course and the course is available , a list of possible courses if the lecturer filled only a topic ,or a negative answer in case his request is rejected
4. The lecturer must confirm the course –in this case an invoice is sent to the finance department to pay for the course.
5. In parallel a confirmation is sent to the training body to enroll the lecturer to the course
6. The training manager must maintain a catalog with available courses (he can add courses, remove courses )