TopCoder Project Service v1.1 Requirements Specification

1. Scope

1.1 Overview

Several updates will be made to this component.

A new method will be added into the ProjectService interface to get the project by project name and user id.

The UserService will be updated to provide functionalities like: register a user, get user information, add user to groups, remove user from groups, add user term, and remove user term.

1.2 Logic Requirements

1.2.1 Existing Functionalities

The existing functionalities should be kept intact. No changes are made to them.

1.2.2 ProjectService Update

A new getProjectByName method is added to the ProjectService interface.
This same method will also be added into the ProjectPersistence interface.

The ProjectServiceBean class will be updated to implement this method, and it will simply delegate to the same method in the persistence interface.

The JPAProjectPersistence class will be updated to implement this method, and it will select Project whose userId and name are as specified. And only the first matched Project will be returned.

1.2.3 UserService Update

The UserService interface is updated with several new methods added. And the UserServiceBean class should also be updated to implement these new methods.

The user data will be saved into the db tables in the common_oltp database. The database schema will be provided.

The user information will be saved into the user db table.
The user group information will be saved into the user_group_xref db table.
The user term information will be saved into the user_terms_of_use_xref db table.
The user email information will be saved into the email db table.
The user address information will be saved into the address db table.

The newly added entities will be Java Beans with corresponding getters and setters.

Note that the user terms should be pre-configured in the terms_of_use db table.

1.2.3.1 Register User

It will use encryption component to encrypt the user password, and then save user information into the database.

1.2.3.2 Get User Information

It will retrieve user information from database by the given user id.

1.2.3.3 Add User to Groups

It will add the user to the specified groups. No exception occurs if the user is already in any of the specified group.

1.2.3.4 Remove User from Groups

Remove user from the specified groups. No exception occurs if the user is not in any of the specified group.

1.2.3.5 Add User Term

Add user to the specified term. This method will be called when user agrees to the given term.

1.2.3.6 Remove User Term

Remove user from the specified term.

1.2.4 Logging

The logging strategy from the existing component should be followed.

1.2.5 Transaction

The transaction strategy from the existing component should be followed.
If the EJB method changes the data in persistence, it should require a transaction, otherwise no transaction is necessary.

1.2.6 Thread Safety

This component should work thread-safely after being deployed in the container, as it’s implemented as a stateless session bean.

1.2.7 Database Information

ip: 174.129.64.230
Servername: informixoltp_tcp
Port: 2021
Username: informix
Password: 1nf0rm1x

You can connect it by http://www-01.ibm.com/software/data/informix/serverstudio/
or any tools which support Informix (11.5)

1.3 Required Algorithms

None

1.4 Example of the Software Usage

The newly added methods will be used in the Liquid Portal Service Component.

1.5 Future Component Direction

None

2. Interface Requirements

2.1.1 Graphical User Interface Requirements

None

2.1.2 External Interfaces

See Interface diagram.

2.1.3 Environment Requirements

- Development language: Java1.5
- Compile target: Java1.5, Java 1.6

2.1.4 Package Structure

com.topcoder.service.project
com.topcoder.service.user

3. Software Requirements

3.1 Administration Requirements

3.1.1 What elements of the application need to be configurable?

The encryption algorithm

3.2 Technical Constraints

3.2.1 Are there particular frameworks or standards that are required?

EJB 3.0

JPA 1.0

Hibernate 3.2 and higher

3.2.2 TopCoder Software Component Dependencies:

Base Exception 2.0
Logging Wrapper 2.0
JBoss Login Module 1.1 (use its class UserProfilePrincipal)
Encryption 2.0.1

3.2.3 Third Party Component, Library, or Product Dependencies:

None

3.2.4 QA Environment:

- RedHat Linux 9
- Informix 10
- JBoss 4.2

3.3 Design Constraints

The component design and development solutions must adhere to the guidelines as outlined in the TopCoder Software Component Guidelines. Modifications to these guidelines for this component should be detailed below.

3.4 Required Documentation

3.4.1 Design Documentation

- Use-Case Diagram
- Class Diagram
- Sequence Diagram
- Component Specification

3.4.2 Help / User Documentation

- Design documents must clearly define intended component usage in the ‘Documentation’ tab of the TopCoder UML Tool.