

1. Client Project Entities DAO v1.2 Requirements Specification

1.Scope

1.1Overview

Several updates will be made to this component.

The ProjectDAO will provide functionalities to add user to billing projects, remove user from billing projects, and get projects by client id.

The Project entity needs to be updated with a new budget attribute, and this attribute needs to be added into corresponding db table too.

A new method should be added, to retrieve all users who have access to the specified project

The ProjectDAO needs to be updated accordingly.

1.2Logic Requirements

1.2.1Existing Functionalities

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

1.2.2Existing ProjectDAO Update

New methods are added to the ProjectDAO interface, and the ProjectDAOBean class should be updated to implement these new methods.

The time_oltp database schema will be provided, and all entities of this component will be saved to this database.

1.2.2.1Add User to Billing Projects

The specified billing project ids are all from the project table (maps to the Project entity), and to add user to the billing projects, we will need to insert the user to billing project associations into the project_manager db table.

Note that if the user doesn't exist in user_account db table, a new record will be created into the user_account db table.

1.2.2.2Remove User from Billing Projects

Remove the user to billing project associations from the project_manager db table. Nothing happens if the user doesn't exist in the user_account db table.

1.2.2.3Get Projects by Client Id

Get all projects associated with the given client id.

1.2.3Billing Project Budget

The Project entity needs to be updated with a new budget:double attribute, and this attribute needs to be added into corresponding project table too.



The existing ProjectDAO's methods need to be updated to retrieve or update newly added budget attribute.

One extra method needs to be added into ProjectDAO:

updateProjectBudget(billingProjectId:long, updatedAmount:double):double

It needs to update the project's budget to current budget + updatedAmount. The updatedAmount can be negative or positive. Modified budget should be returned.

Whenever the budget is updated, the budget change needs to be audited into the project_budget_audit table.

The ERD will be provided.

1.2.4 Get users by project id

One extra method needs to be added into ProjectDAO:

getUsersByProject(billingProjectId:long):List<String>

It gets all users' name who have access of the billing project.

The following SQL statement can be used.

```
SELECT distinct user_name FROM project_manager p,  
user_account u WHERE p.user_account_id = u.user_account_id and p.active = 1 and  
p.project_id = PROJECT_ID union  
SELECT distinct project_id FROM project_worker p,  
user_account u WHERE p.start_date <= current and current <= p.end_date and p.active =1 and  
p.user_account_id = u.user_account_id and p.project_id = PROJECT_ID
```

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

Servename: 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)

1.3Required Algorithms

None

1.4Example of the Software Usage

The newly added methods will be used in the Launch Contest Assembly.

1.5Future Component Direction

None

2.Interface Requirements

2.1.1Graphical User Interface Requirements

None

2.1.2External Interfaces

See Interface diagram.

2.1.3Environment Requirements

Development language: Java1.5

Compile target: Java1.5, Java 1.6

2.1.4Package Structure

com.topcoder.clients.model

com.topcoder.clients.dao

com.topcoder.clients.dao.ejb3

3.Software Requirements

3.1Administration Requirements

3.1.1What elements of the application need to be configurable?

None.

3.2Technical Constraints

3.2.1Are 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

Search Builder 1.4

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.