



User Project Data Store Requirements Specification

1. Scope

1.1 Overview

The component provides API to access the TopCoder user and project database in order to integrate with the Online Review application.

1.2 Logic Requirements

1.2.1 User Database

Component will provide read only access to the TopCoder user database. Retrieval should be based on either external identifier or user handle. The following information will be retrieved: external identifier, user handle, first name, last name, emails, rating and reliability. Rating and reliability are optional and will be per phase (dev/design). A batch mode should be provided as well.

1.2.2 Project Database

Component will provide read only access to the TopCoder project database. Retrieval should be based on either external identifier, or project name/version. The following information will be retrieved: external identifier, project name, version, catalog type (Java/.NET), catalog reference identifier, forum reference identifier. A batch mode should be provided as well.

1.3 Required Algorithms

No specific algorithms are required.

1.4 Example of the Software Usage

After user is logged into the application, the handle of the logged user will be displayed in the header. The session contains the user's external identifier. The component can be used to retrieve the user's handle as well as rating, so that the handle can be rendered in the color corresponding to the user's rating.

1.5 Future Component Direction

None.

2. Interface Requirements

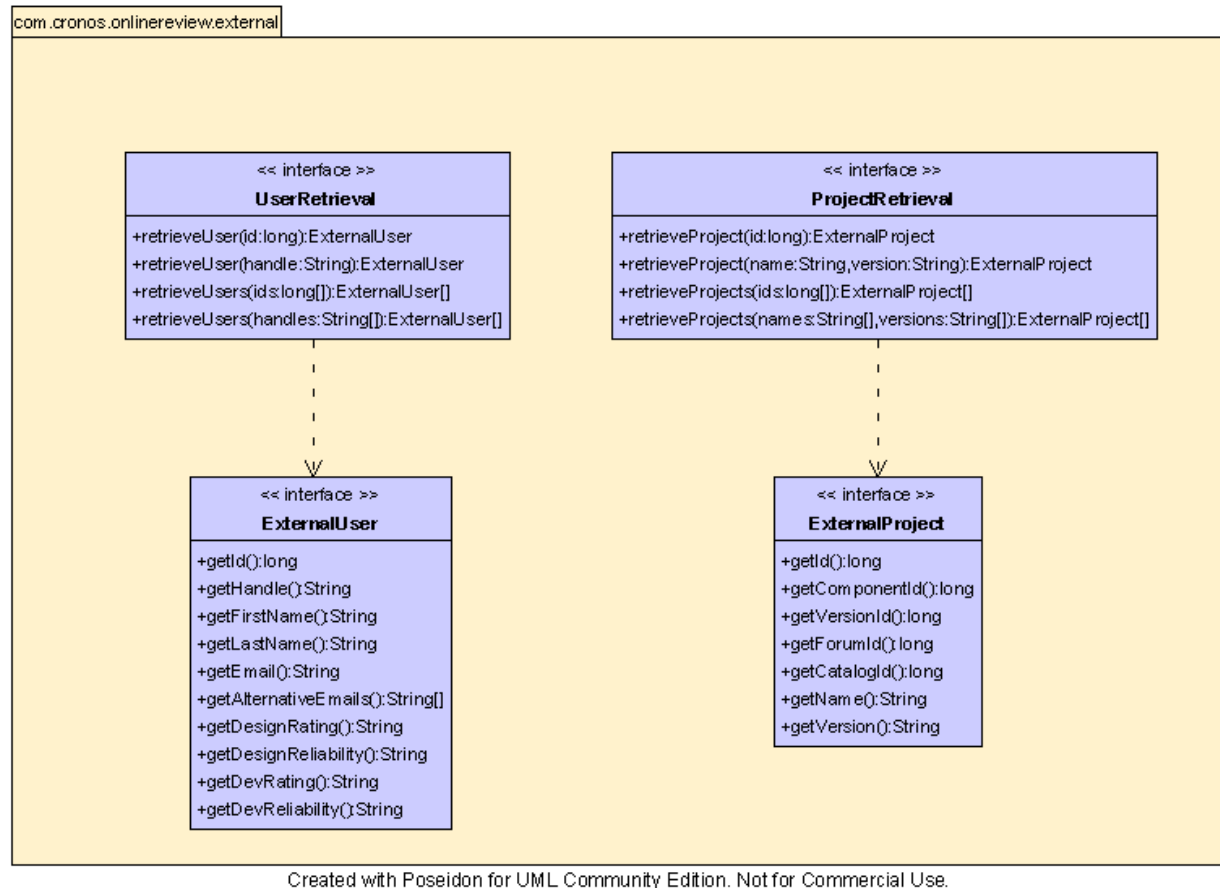
2.1.1 Graphical User Interface Requirements

None.

2.1.2 External Interfaces

Design must adhere to the interface diagram definition. Designer can choose to add more methods to the classes/interfaces, but must keep the ones defined on the diagram as a minimum. Source files can be found in the distribution.

User Project Data Store Interface Diagram



2.1.3 Environment Requirements

- Development language: Java1.4
- Compile target: Java1.4

2.1.4 Package Structure

com.cronos.onlinereview.external

3. Software Requirements

3.1 Administration Requirements

3.1.1 What elements of the application need to be configurable?

- Database connection

3.2 Technical Constraints

3.2.1 Are there particular frameworks or standards that are required?

JDBC

3.2.2 TopCoder Software Component Dependencies:

- Configuration Manager
- DB Connection Factory



****Please review the [TopCoder Software component catalog](#) for existing components that can be used in the design.**

3.2.3 *Third Party Component, Library, or Product Dependencies:*

None.

3.2.4 *QA Environment:*

- Solaris 7
- RedHat Linux 7.1
- Windows 2000
- Windows 2003
- Informix 10.0

3.3 Design Constraints

The component design and development solutions must adhere to the guidelines as outlined in the TopCoder Software Component Guidelines.

3.3.1 *Database Connections*

Database connections must not be cached within the component. Connections should be created for each operation and closed afterwards.

3.3.2 *Component Scalability*

The component needs to be scalable. Running multiple instances in the same JVM or in multiple JVM's concurrently should not cause any problem.

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 Poseidon.