Software Documentation : Java Generic Registration Framework Validator Open Registration

This page last changed on May 27, 2008 by ghostar.

1. Scope

1.1 Overview

The Generic Registration Framework is a framework that will be used to perform registrations for any type of TopCoder competition, including Architecture, Design, Development, Assembly, Testing, Bug Races, as well as future types of competitions.

This component defines a registration validator that ensures that the window for registration for a given role in a contest is open at the time of the registration request.

1.1.1 Version

Version 1.0

1.2 Logic Requirements

1.2.1 OpenRegistrationValidator class

This validator validates both registration and unregistration to ensure that the window for registration for the given role is still open. This validator should make sure to use both the Contest.getRegistrationStart() and getRegistrationEnd() dates, applying any offsets from the contest role.

1.2.2 Date offsets

When comparing the dates, it is important to include the offsets of the role registration times. For instance, a reviewer position may open 24 hours after the contest's registration starts, so the start time is actually contest.registrationStart + role.registrationStartOffset. The same applies to the registration end offset.

1.2.3 Date comparison

The date comparison to use should be inclusive of the current time, to ensure registration is open and available at the exact moment specified in the contest and role information.

1.2.3 Insufficient Data

If there isn't sufficient data to determine if registration is allowed or not, due to missing or null date or offset values, the InsufficientDataException from the Registration Framework component should be thrown.

1.2.4 Registration failure message

If validation fails, the result returned should contain the message "Registration validation failed because registration is not open yet for the contest and role submitted".

1.2.4 Current implementation

Note that this validator is currently implemented as part of the Registration Framework component. It is being moved to its own component in the catalog, to be consistent with upcoming validator components.

1.3 Required Algorithms

None.

1.4 Example of the Software Usage

This component will be used when validating registrations in the registration framework.

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 API of the Registration Framework component.

class OpenRegistrationValidator

```
public class OpenRegistrationValidator implements RegistrationValidator
{
public ValidationResult validateRegistration(Contest contest, User user, ContestRole role);
public ValidationResult validateUnregistration(Contest contest, User user, ContestRole role);

}
```

2.1.3 Environment Requirements

- Development language: Java 1.5
- Compile target: Java 1.5 and Java 1.6

2.1.4 Package Structure

com.topcoder.registration.validators

3. Software Requirements

3.1 Administration Requirements

3.1.1 What elements of the application need to be configurable?

None.

3.2 Technical Constraints

3.2.1 Are there particular frameworks or standards that are required?

None

3.2.2 TopCoder Software Component Dependencies:

- Registration Framework 1.0

**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:

- Windows XP
- Windows 2003

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 Poseidon.