Software Documentation : Java Custom Cockpit Contest And Submission Entities 1.2

This page last changed on Jul 10, 2009 by mashannon168.

1. Scope

1.1 Overview

This component contains the entities for studio competitions. In version 1.2 of this component new entities will be added and others will be updated.

1.1.1 Version

1.2

1.2 Logic Requirements

1.2.1 Update and add new entities

In this new version of the component new entities will be added and other entities will be updated. See "Contest And Submission Entities 1.2 Class Diagram" from uml for new entities and updated entities. Entities are simple java beans with setters and getters and empty constructor.

1.2.2 Update diagrqams

The class diagrams of this component are not up-to-date and this will be resolved in this competition. The competing designers will check the current code and update theclass diagrams of this component. If necessary, the use case diagram and sequence diagrams also need to be updated.

1.3 Required Algorithms

No.

1.4 Example of the Software Usage

These entities will be used by the ejb services to manage studio competitions.

1.5 Future Component Direction

None.

2. Interface Requirements

2.1.1 Graphical User Interface Requirements

None.

2.1.2   External Interfaces

Design will follow "Contest And Submission Entities 1.2 Class Diagram" from uml

2.1.3 Environment Requirements

Java 1.5

2.1.4 Package Structure

- com.topcoder.service.studio.contest

- com.topcoder.service.studio.submission

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:

None
**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:

- 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 Poseidon.