[TopCoder]

Jive Forum Services 1.0 Requirements Specification

1. Scope

1.1 Overview

TopCoder uses Jive Forums for its site, which provides an API for programmatically managing them.

This component will provide some higher level services that will help setting up the structure of forums, groups and permissions used in TopCoder Software forums.

TopCoder will provide Jive Forums API to developers that sign up for developing the component.

1.2 Logic Requirements

1.2.1 Forum user

In order to manage forums, an authentication token is used. The authentication factory must not be used for this purpose. Instead, the authentication token must be built from a user id specified in configuration.

For this purpose, a simple class implementing `AuthToken` must be created. It will just store a `userId` to be returned in method `getUserId`. Then, an instance of this class can be used to get a forum factory, for example:

```
ForumFactory forumFactory = ForumFactory.getInstance(new MyAuthToken(userId));
```

1.2.2 Create a watch for a user in a category, forum or thread.

Given a user id and either a category, forum or thread id, if the user was not previously watching it, add the watch

1.2.3 Get whether a user is watching a category, forum or thread

Given a user id and either a category, forum or thread id, return whether the user is already watching it.

1.2.4 Set User Role in Category

Given a user id and a category id, one of these roles can be assigned:

- No access: the user must not belong to moderators or users groups.

- Contributor: the user must belong just to users group

- Moderator: the user must belong just to moderators group

The users group name for a category is: *GROUP_SOFTWARE_USERS_PREFIX* + categoryId.

The moderators group name for a category is: *GROUP_SOFTWARE_MODERATORS_PREFIX* + categoryId.

1.2.5 Get User Role for a Category

Given a user id and a category id, return whether the user has no access, is contributor or moderator, depending on the groups he belong, according to 1.2.4.

1.2.6 Create Software Forums

It creates a new category, and if a template category is specified, it will also copy all the contents.

Those parameters are used:

- Category Name

- Category Description

- Root category id

- Component Id

- Version Text

- Version Id

- Whether the forum is public or not

- Optional: Template Category Id

- Optional: Moderator user id.

Create the category and set its properties (see algorithm 1.3.1) and then create groups and set permissions (see algorithm 1.3.2).

If a Template Category Id is specified, all the forums, threads and announcements must be copied from that category to the created category, maintaining the structure.

If a Moderator User Id is specified, that user id must be assigned to the moderators group for the category, and a watch for the user must be created for the category.

The category id for the created category must be returned.

1.2.7 Create Software Forums for an entity type

The user can specify one of the following types: Application, Component, Assembly Competition or Testing Competition.

Each of those types has a template category id configured (see 3.1.1). This id will be used to create a forum by copying from a template category (see 1.2.6).

The category id for the created category must be returned.

1.2.8 Session Bean

This component must implement stateless session bean(s) (with both local and remote interfaces) providing operations 1.2.2 to 0.

Each operation must run in a container managed transaction of REQUIRED type.

1.3 Required Algorithms

See the appendix for constant's values.

1.3.1 Create a category and set properties for Software Forums

```
ForumCategory newCategory = forumFactory.getForumCategory(rootCategoryId)
                      .createCategory(categoryName, categoryDescription);

newCategory.setProperty(PROPERTY_ARCHIVAL_STATUS, PROPERTY_ARCHIVAL_STATUS_ACTIVE);
newCategory.setProperty(PROPERTY_COMPONENT_ID, componentId);
newCategory.setProperty(PROPERTY_COMPONENT_VERSION_ID, versionId);
newCategory.setProperty(PROPERTY_COMPONENT_VERSION_TEXT, versionText);
newCategory.setProperty(PROPERTY_MODIFY_FORUMS, "true");
```

1.3.2 Create groups and permissions for Software Forums

```
GroupManager groupManager = forumFactory.getGroupManager();
```

```
Group swAdminGroup = groupManager.getGroup(GROUP_SOFTWARE_ADMINS);
Group moderatorGroup = groupManager.createGroup(GROUP_SOFTWARE_MODERATORS_PREFIX +
        category.getID());
Group userGroup = groupManager.createGroup(GROUP_SOFTWARE_USERS_PREFIX +
        category.getID());
moderatorGroup.setDescription(category.getName());
userGroup.setDescription(category.getName());
PermissionsManager categoryPermissionsManager = category.getPermissionsManager();
for (int i = 0; i < MODERATOR_PERMS.length; i++) {
    categoryPermissionsManager.addGroupPermission(moderatorGroup,
                    PermissionType.ADDITIVE, MODERATOR_PERMS[i]);
}
for (int i = 0; i < REGISTERED_PERMS.length; i++) {
    categoryPermissionsManager.addGroupPermission(userGroup, PermissionType.ADDITIVE,
                REGISTERED_PERMS[i]);
}
for (int i = 0; i < ADMIN_PERMS.length; i++) {
    categoryPermissionsManager.addGroupPermission(swAdminGroup, PermissionType.ADDITIVE,
                ADMIN_PERMS[i]);
}
if (!isPublic) {
    for (int i = 0; i < SW_BLOCK_PERMS.length; i++) {
        categoryPermissionsManager.addAnonymousUserPermission(PermissionType.NEGATIVE,
                SW_BLOCK_PERMS[i]);
        categoryPermissionsManager.addRegisteredUserPermission(PermissionType.NEGATIVE,
                SW_BLOCK_PERMS[i]);
    }
}
```

1.4 Example of the Software Usage

An application will use this component to create categories for the different types of competitions.

1.5 Future Component Direction

More services can be provided to meet additional needs.

2.   Interface Requirements

2.1.1 Graphical User Interface Requirements
None

2.1.2 External Interfaces
None

2.1.3 Environment Requirements
- Development language: Java 5.0
- Compile target: Java 5.0

2.1.4 Package Structure
com.topcoder.forum.service

3.   Software Requirements

3.1 Administration Requirements

3.1.1 What elements of the application need to be configurable?
Configure those parameters using environment entries or another EJB mechanism.

- User id to perform forum operations

- Template category_id for entity types:

  - o Application

  - o Component

  - o Assembly Competition

  - o Testing Competition

3.2 Technical Constraints

3.2.1 Are there particular frameworks or standards that are required?
  None

3.2.2 TopCoder Software Component Dependencies:

  **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:

- Jive Forums 4.2.5
  (http://www.jivesoftware.com/builds/docs/prof/4.2.5/documentation/javadoc/api/index.html
)

- Informix

- EJB 3.0

3.2.4 QA Environment:

- RedHat Enterprise Linux 4
- Java 1.5
- Jive Forums 4.2.5
- Informix 10
- JBoss 4.2.1 GA with EJB 3.0

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.

3.3.1 Publication of services

  Since the access to the forums is done through an administrator account, those services must not be made public without securing them first. The designer and developer must warn about that in the component specification and source code.

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.

4. Appendix

4.1 Constants

Those are the constants used in the algorithms. It's not required to define them exactly as they are shown here.

```java
    public final static String PROPERTY_COMPONENT_ID = "componentId";
    public final static String PROPERTY_COMPONENT_VERSION_ID = "compVersionId";
    public final static String PROPERTY_COMPONENT_VERSION_TEXT = "versionText";
    public final static String PROPERTY_COMPONENT_TECH_TYPES = "compTechTypes";

    public final static String PROPERTY_ARCHIVAL_STATUS = "archivalStatus";
    public final static String PROPERTY_ARCHIVAL_STATUS_ACTIVE = "1";
    public final static String PROPERTY_ARCHIVAL_STATUS_CLOSED = "2";
    public final static String PROPERTY_MODIFY_FORUMS = "modifyForums";

    public final static String GROUP_SOFTWARE_MODERATORS_PREFIX = "Software_Moderators_";
    public final static String GROUP_SOFTWARE_USERS_PREFIX = "Software_Users_";
    public final static String GROUP_SOFTWARE_ADMINS = "Software Admins";


    public final static long[] REGISTERED_PERMS = {
                ForumPermissions.READ_FORUM, ForumPermissions.CREATE_THREAD,
ForumPermissions.CREATE_MESSAGE,
                ForumPermissions.RATE_MESSAGE, ForumPermissions.CREATE_POLL,
ForumPermissions.VOTE_IN_POLL};

    public final static long[] ADMIN_PERMS = {
        ForumPermissions.READ_FORUM, ForumPermissions.CREATE_THREAD,
ForumPermissions.CREATE_MESSAGE,
        ForumPermissions.RATE_MESSAGE, ForumPermissions.CREATE_MESSAGE_ATTACHMENT,
ForumPermissions.CREATE_POLL,
        ForumPermissions.VOTE_IN_POLL, ForumPermissions.ANNOUNCEMENT_ADMIN,
ForumPermissions.FORUM_CATEGORY_ADMIN};

    public final static long[] MODERATOR_PERMS = {
        ForumPermissions.READ_FORUM, ForumPermissions.CREATE_THREAD,
ForumPermissions.CREATE_MESSAGE,
        ForumPermissions.RATE_MESSAGE, ForumPermissions.FORUM_CATEGORY_ADMIN,
        ForumPermissions.CREATE_MESSAGE_ATTACHMENT, ForumPermissions.CREATE_POLL,
ForumPermissions.VOTE_IN_POLL};

    public final static long[] SW_BLOCK_PERMS = {
        ForumPermissions.READ_FORUM, ForumPermissions.CREATE_THREAD,
ForumPermissions.CREATE_MESSAGE,
        ForumPermissions.RATE_MESSAGE, ForumPermissions.CREATE_MESSAGE_ATTACHMENT,
ForumPermissions.CREATE_POLL,
        ForumPermissions.VOTE_IN_POLL, ForumPermissions.ANNOUNCEMENT_ADMIN};
```