

UML Model Manager 1.0 Requirements Specification

1. Scope

1.1 Overview

The UML Model Manager component provides one place where the UML Model, the ActivityGraphs and the Diagrams are kept. It provides methods to add and remove diagrams and it keeps the project configuration instance.

1.2 Logic Requirements

1.2.1 *Pseudo Singleton*

The class will provide a pseudo singleton instance, for easy access inside the application, as the normal usage involves just one model.

1.2.2 *UML Model instance*

This class keeps a Model instance, accessible through a getter.

The model will be initialized in the constructor with a default empty Model.

1.2.3 *ActivityGraph instances*

This class keeps a list of ActivityGraph instances, accessible through a getter. There are also methods to add and remove ActivityGraph instances.

1.2.4 *Diagram instances*

The class keeps a list of Diagram instances, accessible through a getter. There are also methods to add and remove diagrams.

1.2.5 *ProjectConfigurationManager*

This class keeps a ProjectConfigurationManager instance, accessible through a getter.

1.3 Required Algorithms

None.

1.4 Example of the Software Usage

The component will be used in the TopCoder UML Tool as the central place for the model.

1.5 Future Component Direction

None.

2. Interface Requirements

2.1.1 *Graphical User Interface Requirements*

None.

2.1.2 *External Interfaces*

The design must follow the interface found in the class diagram with the component interfaces. The designer is encouraged to add to the existing interface, but not to remove anything.

2.1.3 *Environment Requirements*

- Development language: Java 1.5

- Compile target: Java 1.5

2.1.4 Package Structure

com.topcoder.uml.modelmanager

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:

- UML Model Core
- UML Model Activity Graphs
- Project Configuration Manager
- Configuration Manager 2.1.5 - recommended

**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

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.