Wenzhou Kean University

Advanced Software Engineering-CPS 5301

Assignment 1

Assignment Objectives

This assignment aims to deepen your understanding of object-oriented design and the practical application of UML in software development.

Assignment requirements

1. **Eclipse with Papyrus Designer installation**

   If you haven't installed Eclipse previously, download the Eclipse installer and install the Eclipse for Java developer, then add Papyrus Designer as a plugin to Eclipse. You can do that by going to help in Eclipse and going to the marketplace to search for a Papyrus designer. Use this link for more help

   https://www.ecourse.org/news.asp?which=5824

2. **UML Class Diagram**:
   - Create a UML class diagram using Papyrus Designer.
   - Your diagram must include:
     - **Classes**: Define at least three classes with appropriate attributes and methods.
     - **Interfaces**: Include at least one interface that is implemented by one of the classes.
     - **Relationships**:
       - **Inheritance**: Demonstrate inheritance by having at least one subclass that extends a superclass.
       - **Composition**: Illustrate composition by showing a strong relationship between a class and its components (i.e., one class is a part of another).
       - **Cardinality**: Indicate cardinality in your relationships:
         - One-to-One: Show a relationship where one instance of a class relates to exactly one instance of another.
         - One-to-Many: Illustrate a relationship where one instance of a class can relate to multiple instances of another class.
3. **UML Code Generation**:
   - After completing your UML class diagram, import it into Eclipse Java Developer using the Papyrus plugin.
   - Generate Java classes from your UML diagram:
     - Right-click on the model element in Papyrus and select "Generate Code".
     - Choose Java as the target language and ensure all classes, interfaces, and relationships are correctly translated into Java code.

- Review and refine the generated Java files as necessary. Ensure that all attributes and methods from your UML diagram are represented in your Java classes.

4. **Documentation**:
    - Write a brief report (1-2 pages) that includes:
        - A description of the system modeled by your UML class diagram.
        - An explanation of the relationships and cardinalities you chose to implement.
        - Any challenges you faced during the diagramming and coding process.
5. **Submission**:
    - Submit the following files:
        - The UML class diagram (in PDF or image format).
        - The generated Java class files.
        - Your documentation report.

Evaluation Criteria:

- Clarity and completeness of the UML class diagram.
- Correct implementation of UML concepts, including relationships and cardinalities.
- Quality and organization of the generated Java code.
- Thoroughness of the documentation report.