



## **Software Design Patterns**

Year 2 (2024/25), Semester 4

**SCHOOL OF INFOCOMM TECHNOLOGY**  
Diploma in Information Technology

## **ASSIGNMENT**

Duration: 23 Jan – 16 Feb 2025

Weightage: 40%

Deadline: 16 Feb 2025, 2359 hours.

**Ngee Ann Polytechnic**  
**School of InfoComm Technology**

## **1. OBJECTIVE**

The aim of this assignment is to implement a console-based application based on a sample case description using appropriate design patterns. Students are required to implement skeleton code based on the design, produce class diagrams and a state machine diagram, explain the rationale behind the usage of their design patterns, and demonstrate and explain their code in a presentation.

## **2. SCOPE**

Students will work in teams of 4 – 5 members. The scope of the project pertains to the functions of a Document Workflow system. Although all students in the team should work together, each student will be in charge of one design pattern. **However, you should not be using the Singleton design pattern.** Students who implement more than one design pattern properly will be given additional credit.

Students may also implement additional features that are not explicitly stated in the case study in order to showcase their understanding of more design patterns. However, the additional features should be appropriate to the case study. **Only the basic features as described in the case study are required for maximum marks.**

## **3. INSTRUCTIONS**

Carefully study the case description in Appendix A to understand the requirements of the system before starting on the deliverables listed below.

In Weeks 15-17 (23 Jan – 9 Feb 2025), you will be producing your diagrams, write-ups and source code in class and online. Refer to Section 5 below for your individual responsibilities. You should clarify any doubts you may have about the case study with your lecturer.

In Week 18, each team will be assigned a 30-minute time slot for your presentation. The objective of this presentation is for your lecturer to assess your understanding of your design, including a demonstration of the code, the team components such as the class diagram and state machine diagram, and the design patterns used. **You are encouraged to use your lecturer's feedback from the presentation to amend and improve your final report and skeleton code prior to submission.**

## **4. DELIVERABLES**

The following are the deliverables, deadlines, and submission modes for this assignment. Only one copy of the deliverables is required – this should be topped with the assignment cover page shown in Appendix B, together with the Tasks Allocation List in Appendix C. A content page should also be included.

### **4.1 Report**

Each team is required to submit a report including the following:

- (i) Team components

**Ngee Ann Polytechnic**  
**School of InfoComm Technology**

- a. An overall class domain diagram for the system, including associations (with labels and multiplicities) and inheritances. Also provide all attributes (with data types) and operations (with parameter and return types), including whether each element is abstract (which should be written in *italics*).
  - b. A state machine diagram for the object: Document.
- (ii) Individual components
- a. A class diagram for the specific design pattern that you implemented. This can be a straightforward extraction from the overall class diagram.
  - b. A write-up explaining the following:
    - The name of the design pattern
    - Which classes map to each role in the design pattern
    - Why the design pattern is appropriate for this case study
    - The benefits of using the design pattern in this way, citing relevant design principles if any

## 4.2 Code

Each team is required to submit a C# console-based application based on the case study. The application should demonstrate the students' understanding of the proper application of design patterns. Also, appropriate text should be printed to the console where appropriate, e.g., when a method is called.

Table 1 – Deliverables

Item	Deliverable	Deadline	Submission Mode
1	Complete Assignment Report (softcopy) with: <ul style="list-style-type: none"> <li>• Cover page – Appendix B</li> <li>• Task allocation sheet – Appendix C</li> <li>• The overall Class Design Diagram of the Document Workflow System</li> <li>• A State Machine Diagram for the object: Document</li> <li>• (Individual) An explanation of the design pattern(s) used in your design</li> <li>• Any other relevant information as additional appendices</li> </ul>	16 Feb 2025, 23:59	Submit 1 copy per team through PoliteMall link: SOFTWARE DESIGN PATTERNS > ASSESSMENT > ASSIGNMENT – DOCUMENT WORKFLOW SYSTEM > Assignment Report
2	C# source code files (in a single zip file) implementing the console application based on the case study ( <i>must be able to compile</i> ):	16 Feb 2025, 23:59	Submit 1 copy per team through PoliteMall link: SOFTWARE DESIGN PATTERNS > ASSESSMENT > ASSIGNMENT – DOCUMENT WORKFLOW SYSTEM > C# Source Codes (Zipped Files)

**Ngee Ann Polytechnic**  
**School of InfoComm Technology**

## 5. ASSESSMENT

Each student in the team shall be graded according to the following components.

Table 2 – Assessment

Item	Component	Marks	
		Team	Individual
1	Completed Assignment Report: <ul style="list-style-type: none"> <li>Overall Class Diagram</li> <li>State Machine Diagram</li> <li>Class Design Diagram for specific design pattern(s)</li> <li>Write-up on specific design pattern(s)</li> </ul>	20 10	10 10
2	C# Source Codes <ul style="list-style-type: none"> <li>Design Pattern Implementation(s)</li> </ul>		30
3	Presentation <ul style="list-style-type: none"> <li>Demo</li> <li>Explanation of Design Pattern(s) Implemented</li> </ul>	10	10
<b>Sub-total</b>		<b>40</b>	<b>60</b>
<b>Total</b>		<b>100</b>	

Refer to Appendix D for more detailed marking rubrics.

## 6. PENALTY

### Submission Mode

Late submissions will be subjected to a penalty of 10% (of the total mark of the submitted component) for each day past the deadline (including Sunday and public holiday). The last day for late submission is 5 days after the submission deadline, after which the assignment will not be accepted.

### Plagiarism

If your submission was found to be plagiarised, you will fail the assignment and be put forward to further disciplinary actions.

**Ngee Ann Polytechnic  
School of InfoComm Technology**

**APPENDIX A – CASE DESCRIPTION – DOCUMENT WORKFLOW SYSTEM**

Your team is tasked with implementing a Document Workflow System that allows multiple users to work on the same document. The system can handle multiple types of documents, e.g., Technical Reports, Grant Proposals. All documents have a header and a footer, but some types of documents might have additional components.

When a user creates a document, they are assigned as the owner. Only the owner can add other users as collaborators for the document. All collaborators (including the owner) can edit the document. When the document is ready, any collaborator can submit it for approval, which requires nominating a user as the approver. However, the approver cannot also be a collaborator or the owner due to conflicts of interest. While the document is under review, no further edits can be made, although new collaborators can still be added.

When a document is under review, the approver has 3 options: (1) Approve the document; (2) Push back the document with a comment so that the collaborators can revise the document; (3) Reject the document. If a document is pushed back, the approver remains the same. If a document is rejected, collaborators can edit the document and submit it for approval again, whereupon they can choose a different approver.

Whenever a document is submitted for approval, pushed back, approved or rejected, all collaborators (including the owner) are notified.

The system is able to convert documents into various formats, such as Microsoft Word or PDF. The format that a document is converted to can be decided and changed during runtime.

Users that log onto the system can create new documents or work on existing documents. They can also view a list of all documents associated with them or filter this list to show only the documents they own.

**APPENDIX B – ASSIGNMENT SUBMISSION COVER PAGE**



**Software Design Patterns**

Year 2 (2024/25), Semester 4

**SCHOOL OF INFOCOMM TECHNOLOGY**

Diploma in Information Technology

**ASSIGNMENT**

Duration: 23 January – 16 February 2025

Weightage: 40% of overall grade.

Deadline: 16 February 2025 (Sunday), 2359 hours.

Tutorial Group:				
Student Name	Student No.	Individual Grade	Team Grade	Final Grade

**Ngee Ann Polytechnic  
School of InfoComm Technology**

**APPENDIX C – ALLOCATION OF INDIVIDUAL TASKS AMONG TEAM MEMBERS**

<b>Design Pattern (Individual)</b>	<b>Done By</b>
Design Pattern 1	
Design Pattern 2	
...	

Add more rows to the above table if necessary.

**Ngee Ann Polytechnic**  
**School of InfoComm Technology**

**APPENDIX D – MARKING RUBRICS**

**INDIVIDUAL REQUIREMENTS (60%)**

No.	Component Scoring	%	Criteria
1	<b>Class Diagram for Specific Design Pattern</b> <input type="checkbox"/> Mostly incorrect F <input type="checkbox"/> Partially correct D – C <input type="checkbox"/> Mostly correct C+ – B+ <input type="checkbox"/> Completely correct A – A+	10	<ul style="list-style-type: none"> <li>Attributes and operations are given to the correct classes, with parameter and return types</li> <li>Associations and inheritances and correct</li> </ul>
2	<b>Write-up on Specific Design Pattern</b> <input type="checkbox"/> Selected pattern is incorrect or inappropriate F <input type="checkbox"/> Selected pattern is ok, but reasons are not well explained D – C <input type="checkbox"/> Pattern is explained, but deeper understanding not shown C+ – B+ <input type="checkbox"/> Well-explained, showing good understanding of design principles A – A+	10	<ul style="list-style-type: none"> <li>The chosen design pattern must be appropriate</li> <li>The proper classes should be identified to fulfill the roles in the pattern</li> <li>The benefits of the pattern should be explained using design principles where appropriate</li> </ul>
3	<b>C# Source Codes</b> <input type="checkbox"/> Cannot compile, code does not make sense F <input type="checkbox"/> Feature is not working properly and shows little understanding of design pattern F – D <input type="checkbox"/> Feature is partially working showing basic understanding of design pattern D+ – C+ <input type="checkbox"/> Feature is mostly working, with the design pattern applied quite well, but edge cases are not handled well B – B+ <input type="checkbox"/> Feature works well and design pattern is well-implemented A – A+	30	<ul style="list-style-type: none"> <li>The code must compile and be well-integrated with the code from the rest of the team</li> <li>The roles of the design pattern should be well-realized (i.e., there should be some sequence of user actions to test each of the roles)</li> <li>Usual hallmarks of good programming are recognized here (e.g., meaningful variable names, good OO practices, comments, etc.)</li> </ul>
4	<b>Presentation</b> <input type="checkbox"/> Students shows very little understanding, or evidence of direct usage of gen AI without understanding F <input type="checkbox"/> Student shows a shallow understanding and is largely unable beyond basic questions D – C <input type="checkbox"/> Student shows moderate understanding and can answer some deeper questions C+ – B+ <input type="checkbox"/> Student shows full understanding, including the ability to answer advanced questions A – A+	10	<ul style="list-style-type: none"> <li>Student should explain what they did clearly, including reasons why the design pattern is appropriate and what the benefits are</li> <li>Student should be able to explain the code, and handle “what if” questions</li> </ul>



**Ngee Ann Polytechnic**  
**School of InfoComm Technology**

• **(A) TEAM REQUIREMENTS (40%)**

No.	Component Scoring	%	Criteria
1	<b>Overall Class Diagram</b> <input type="checkbox"/> Incorrect or incomplete F <input type="checkbox"/> Partially correct D – C <input type="checkbox"/> Mostly correct C+ – B+ <input type="checkbox"/> Entirely correct A – A+	20	<ul style="list-style-type: none"> <li>Attributes and operations are given to the correct classes, with parameter and return types</li> <li>Associations and inheritances and correct</li> <li>This class diagram should include parts of the system that are not actually part of any design pattern</li> </ul>
2	<b>State Machine Diagram</b> <input type="checkbox"/> Based on marking scheme 0 - 10	10	<ul style="list-style-type: none"> <li>Specific marking scheme based on expected state machine diagram</li> </ul>
3	<b>Presentation (demo)</b> <input type="checkbox"/> Code does not work, or frequently crashes F <input type="checkbox"/> Code implements some of the features D – C <input type="checkbox"/> Code implements many of the features, but not all the edge cases C+ – B+ <input type="checkbox"/> Code correctly implements all features, including edge cases A – A+	10	<ul style="list-style-type: none"> <li>The program should compile</li> <li>The student(s) conducting the demo should showcase all the different design patterns, preferably without prompting</li> </ul>
	<b>Total</b>	<b>100</b>	