COS20007

Object-Oriented Programming

STUDENT NAME: Nguyen Duc Thang

STUDENT ID: 104776473

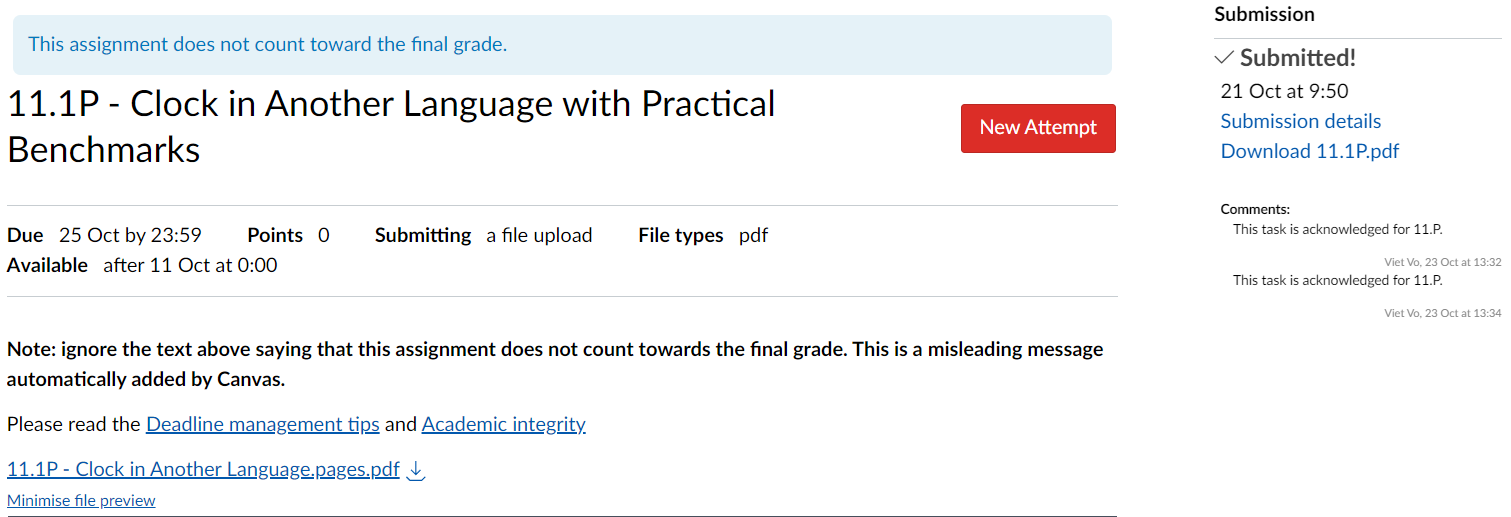
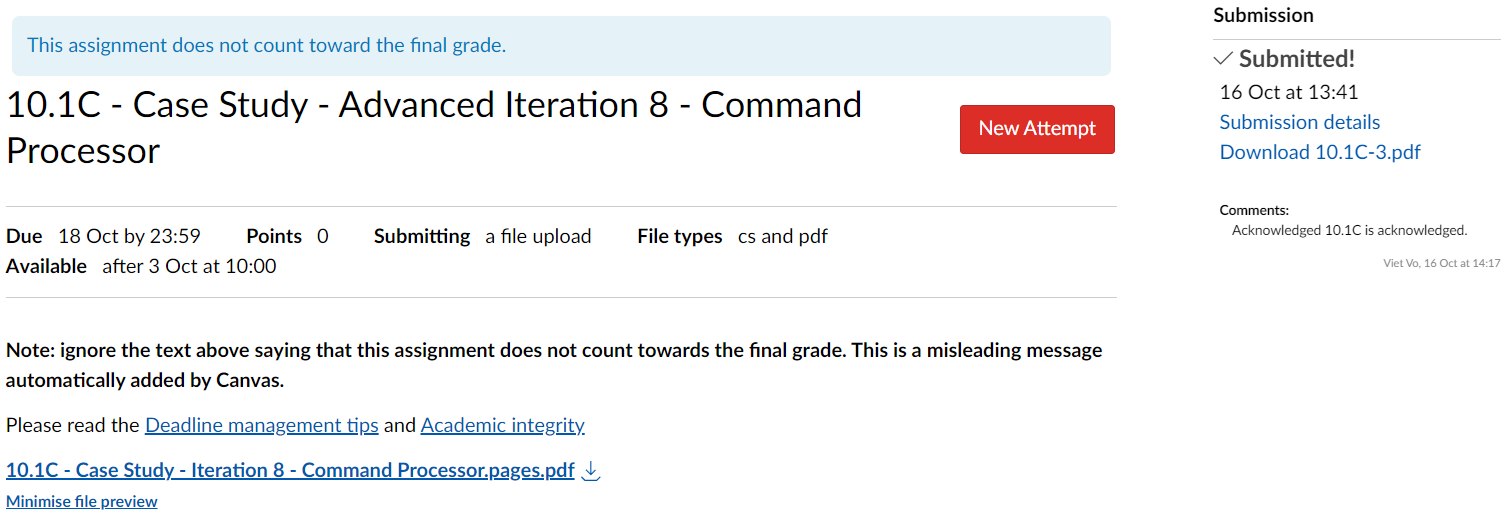
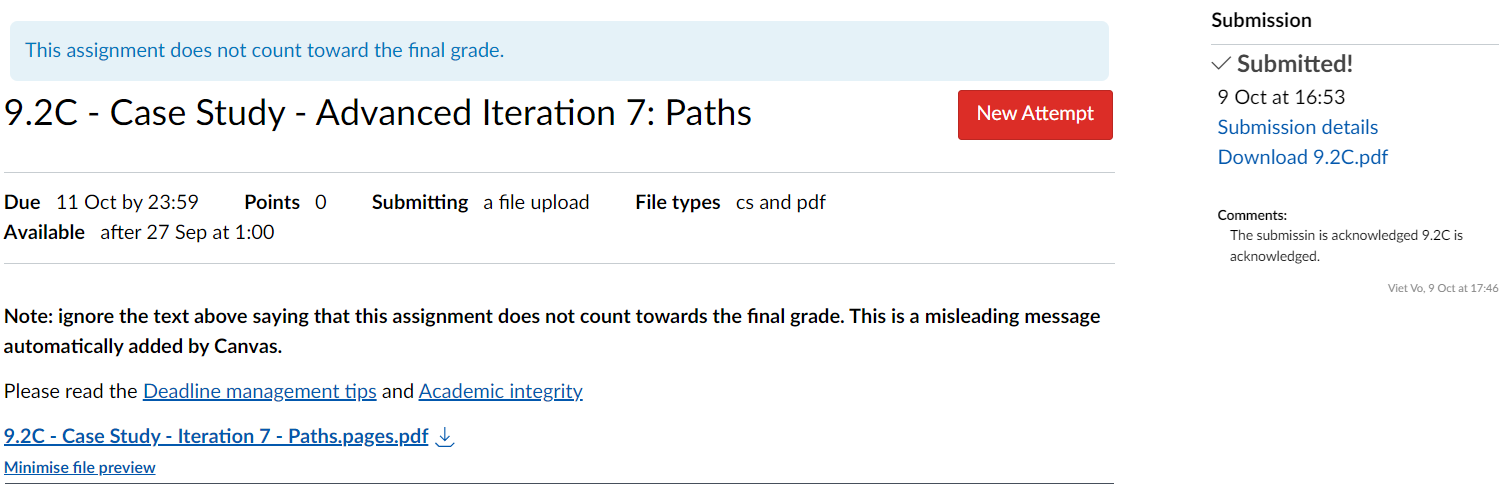
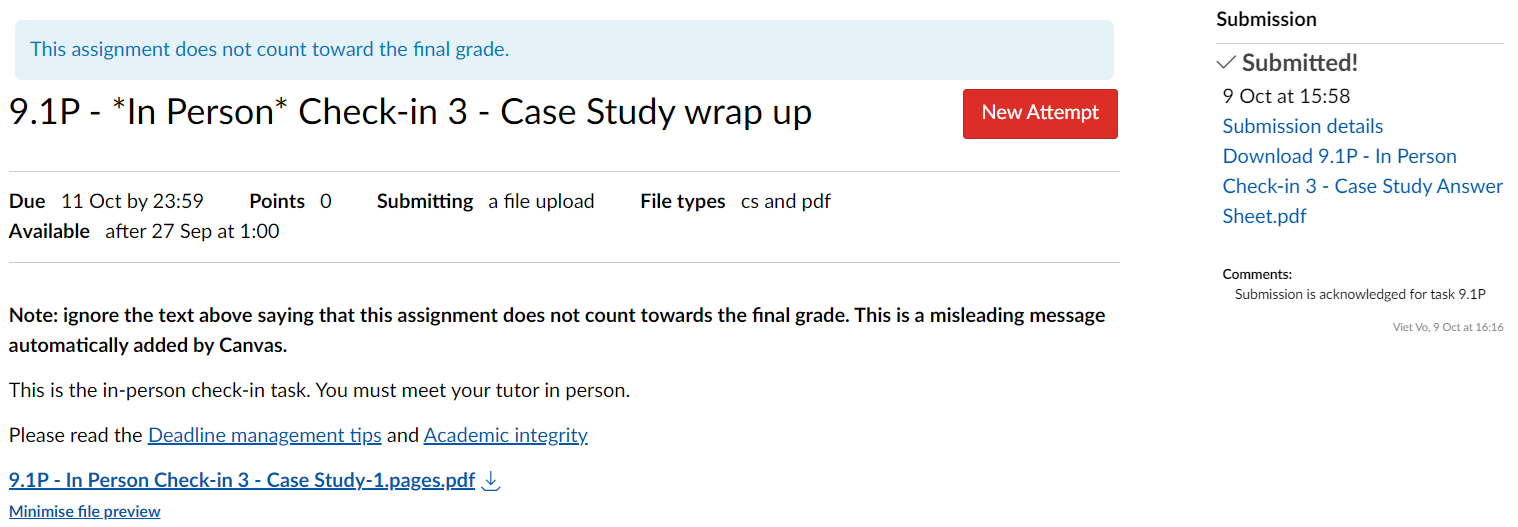
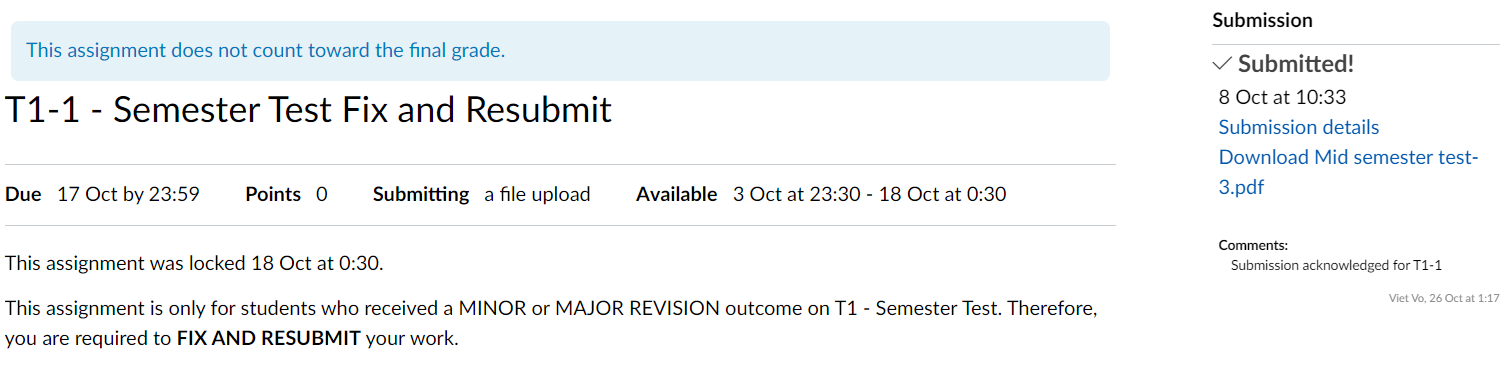
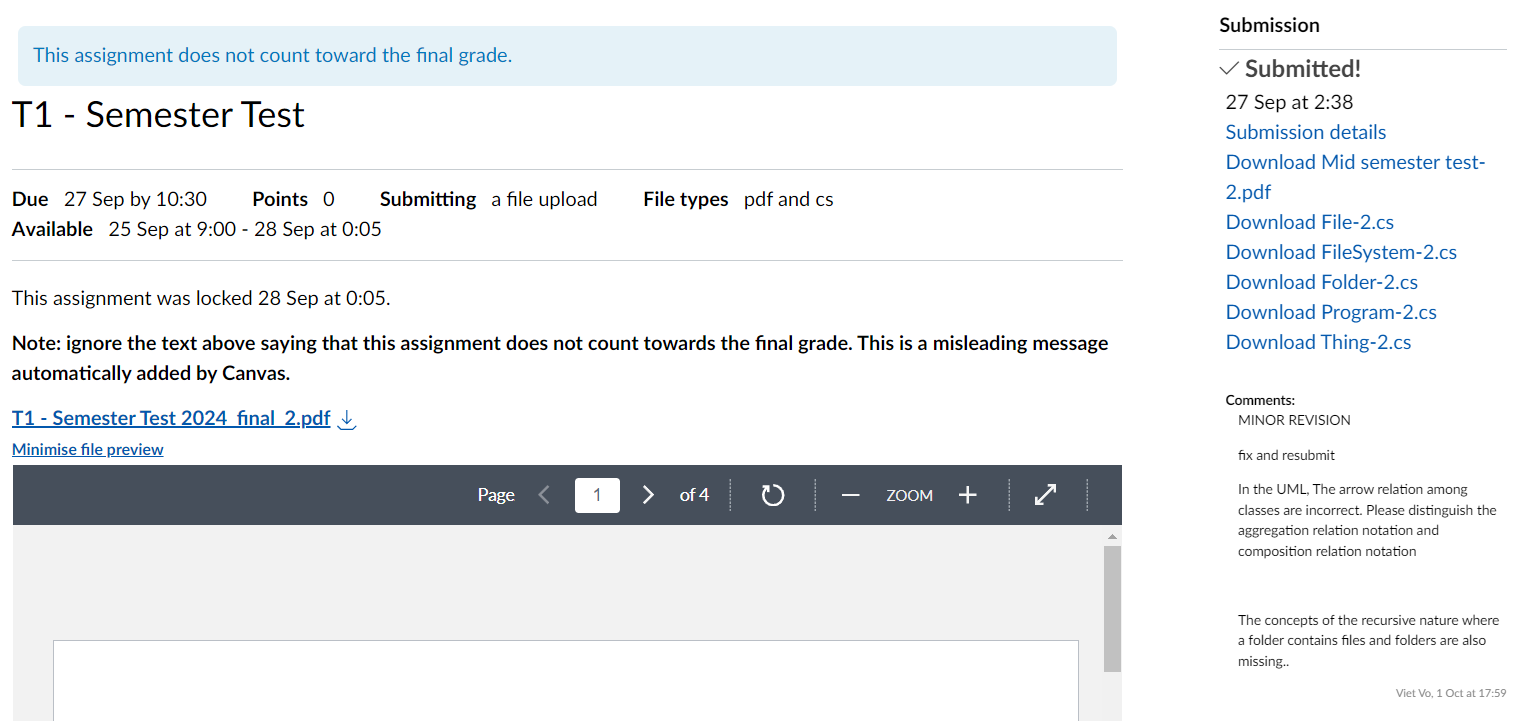
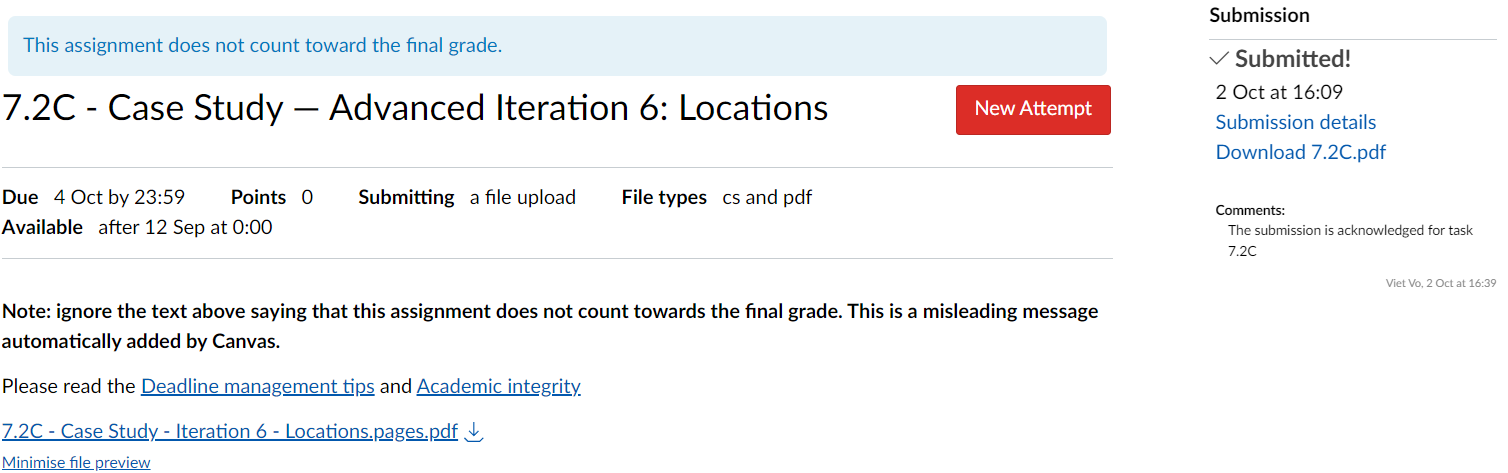
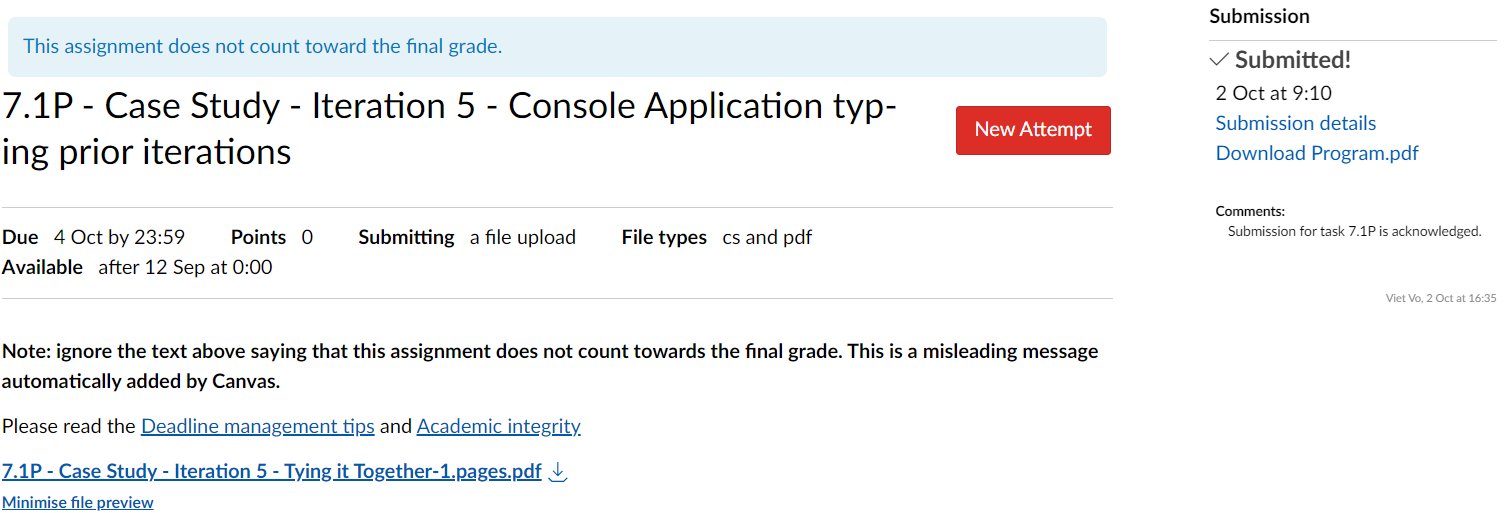
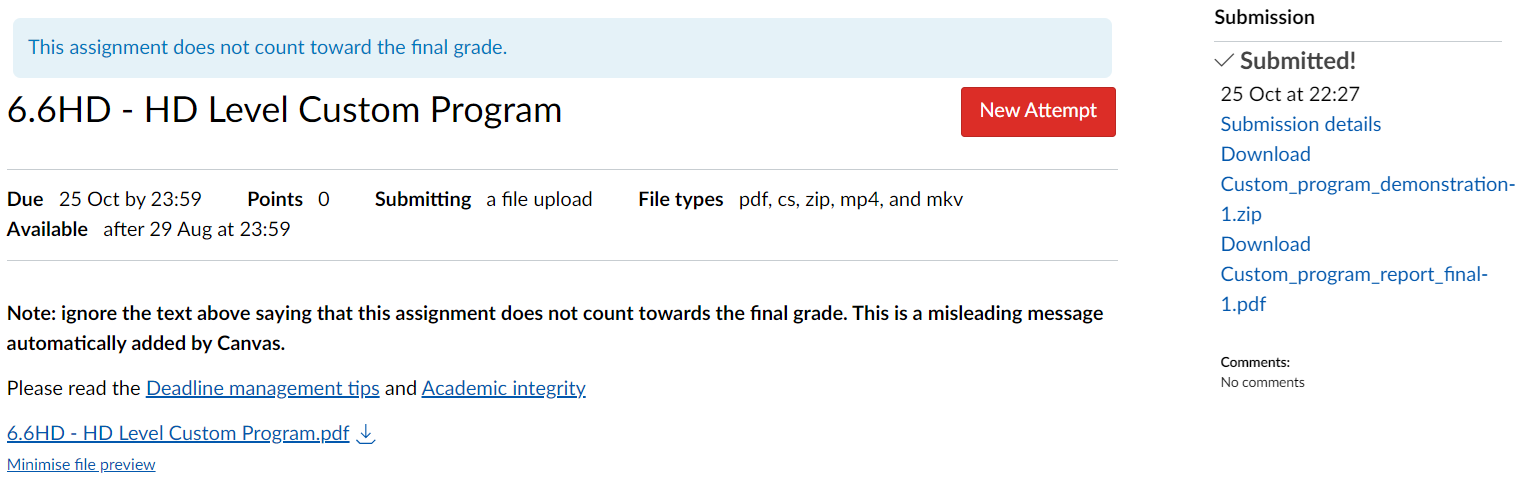
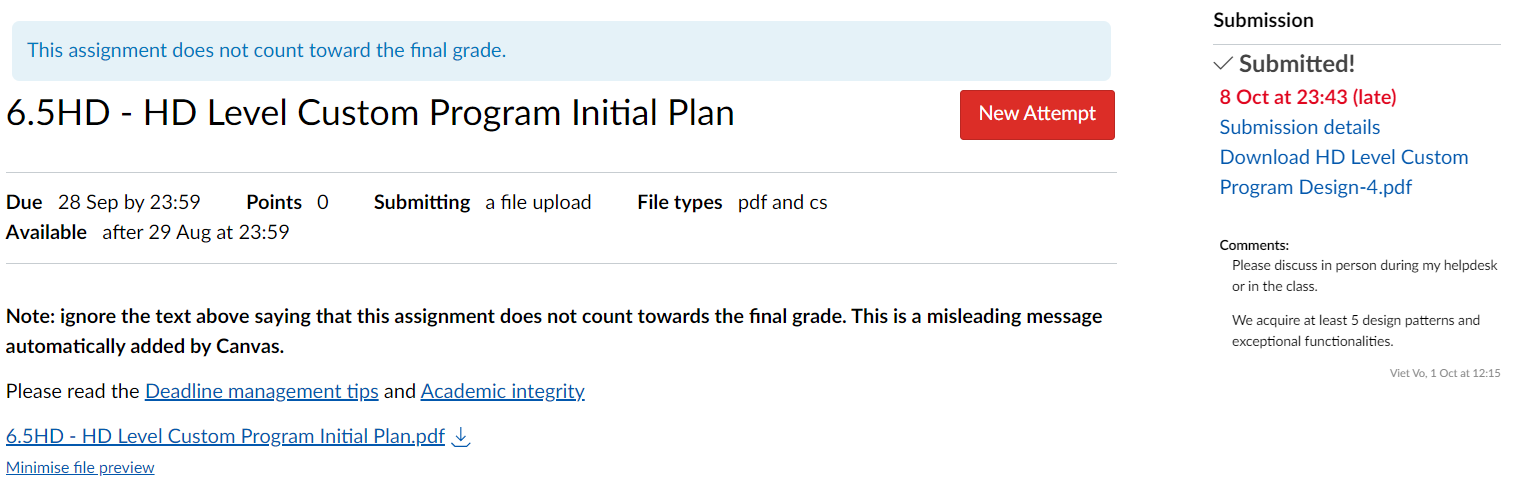
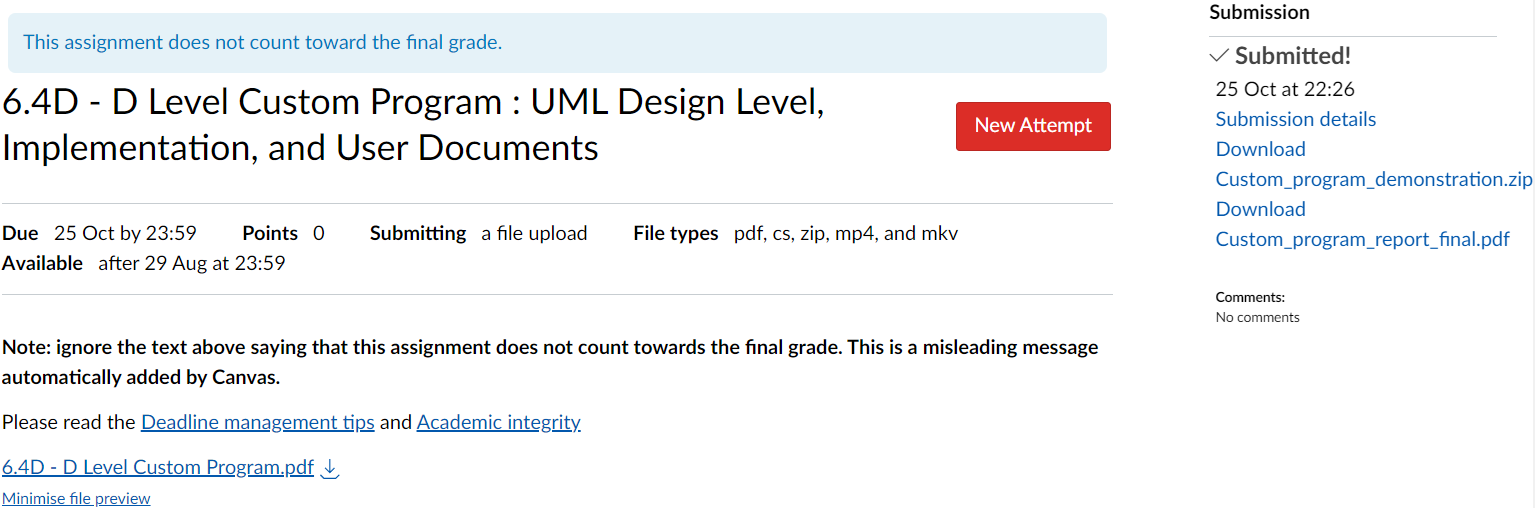
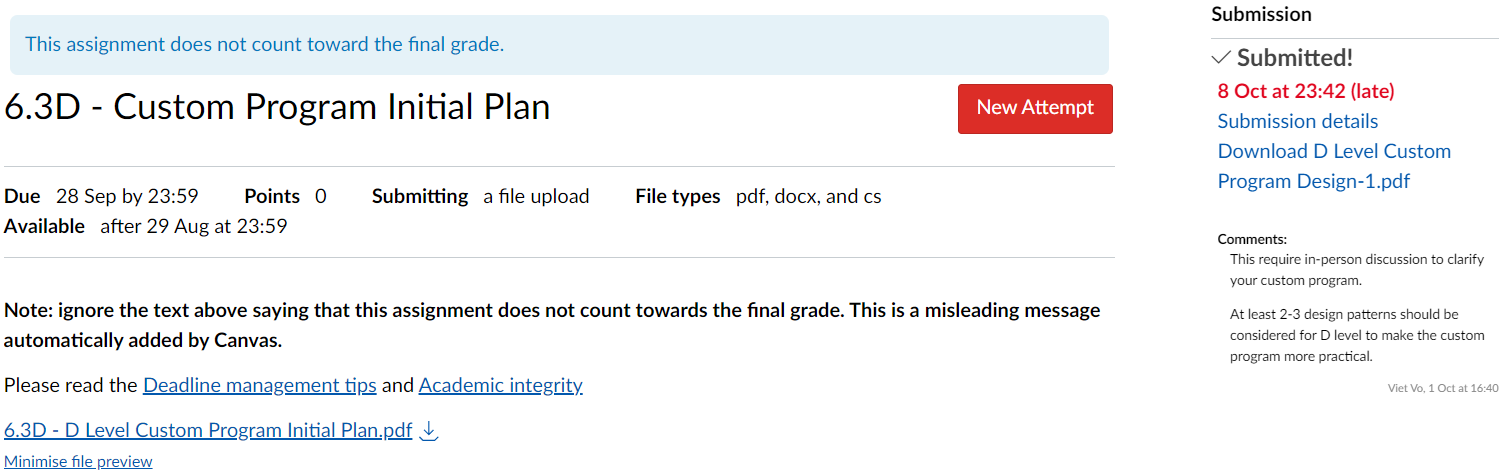
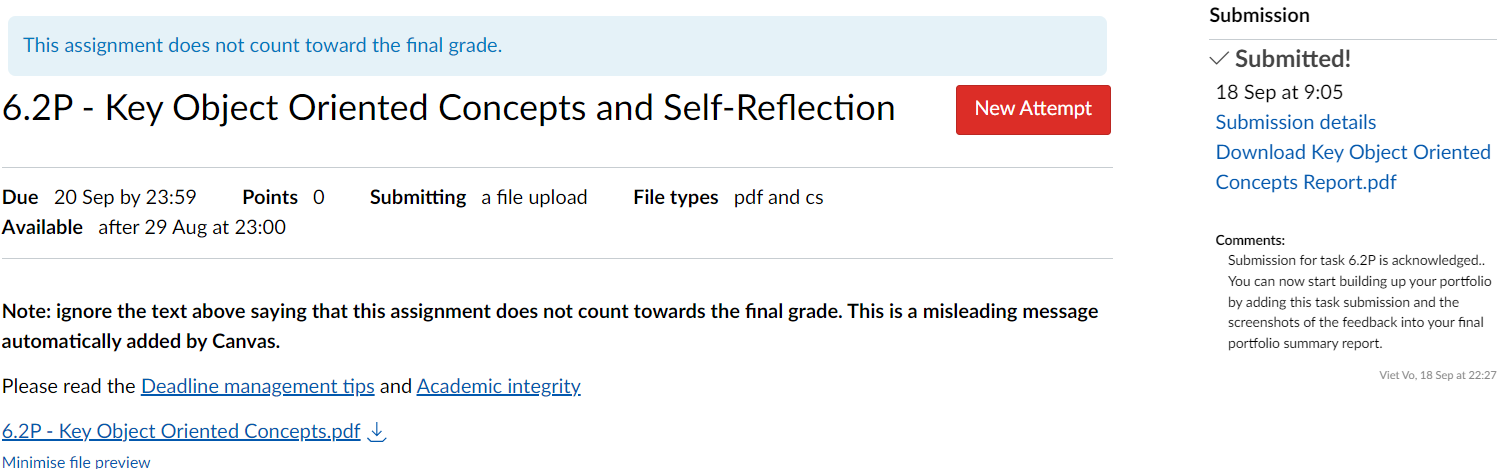
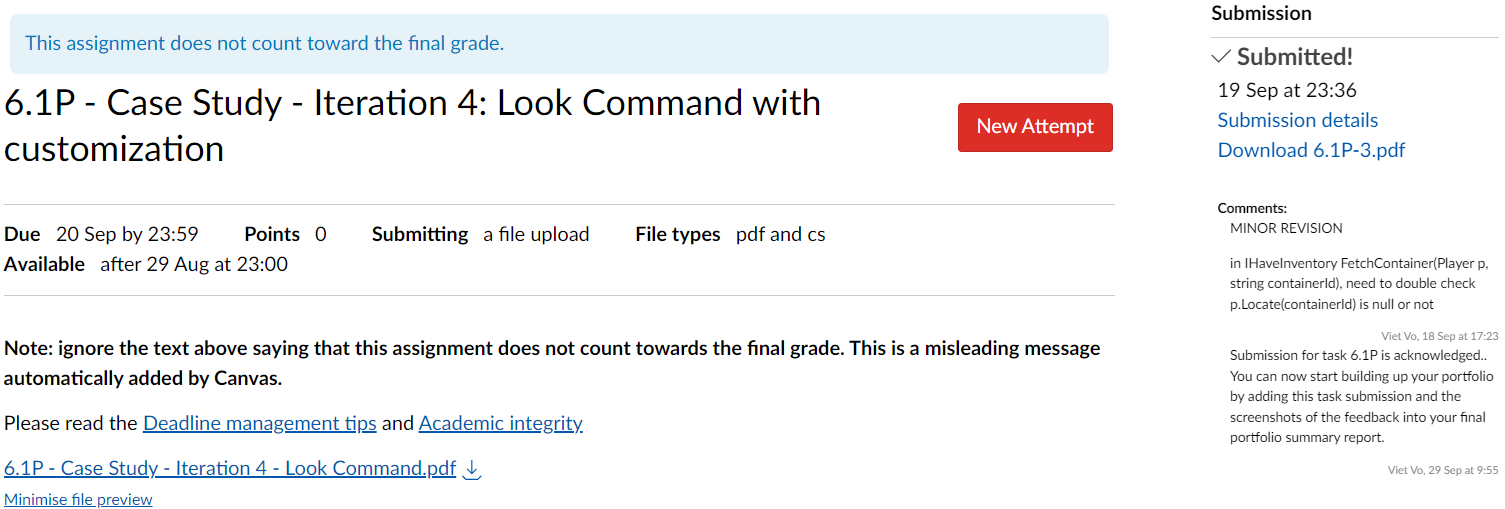
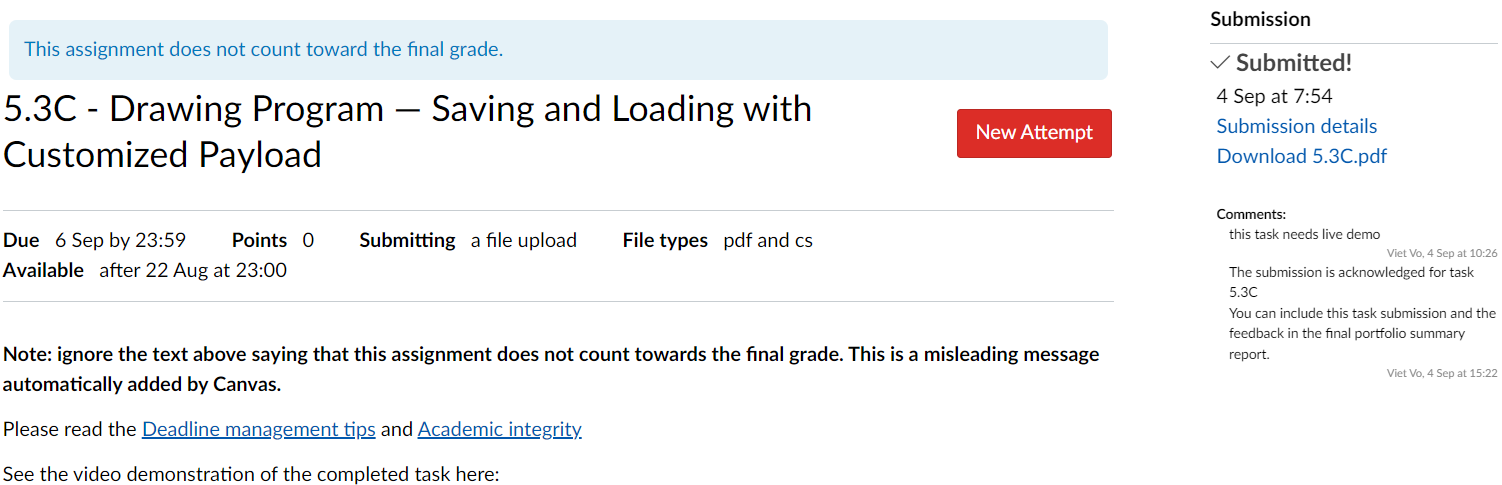
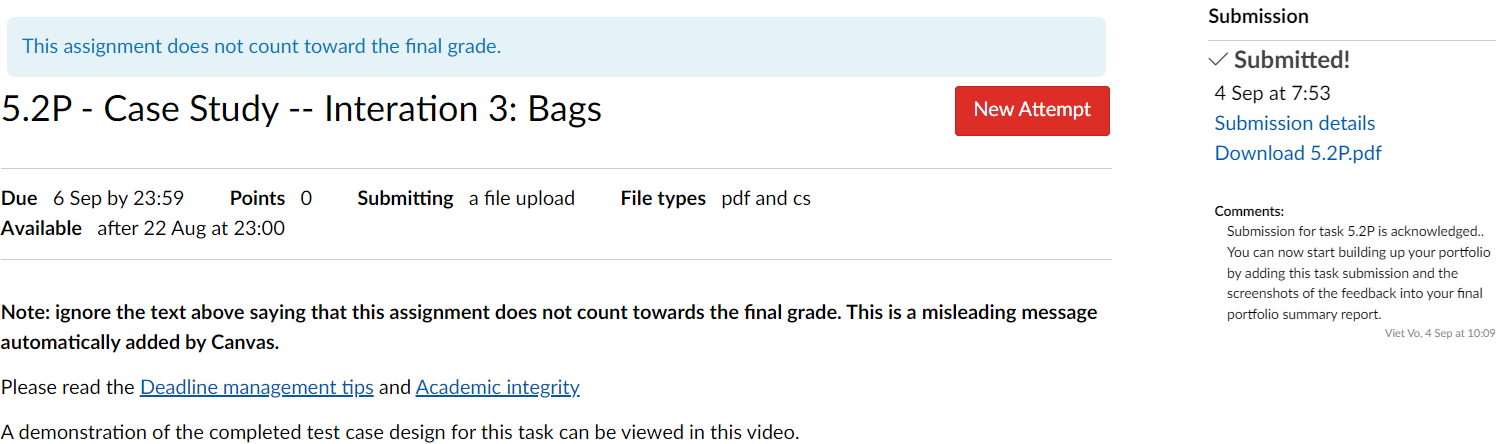
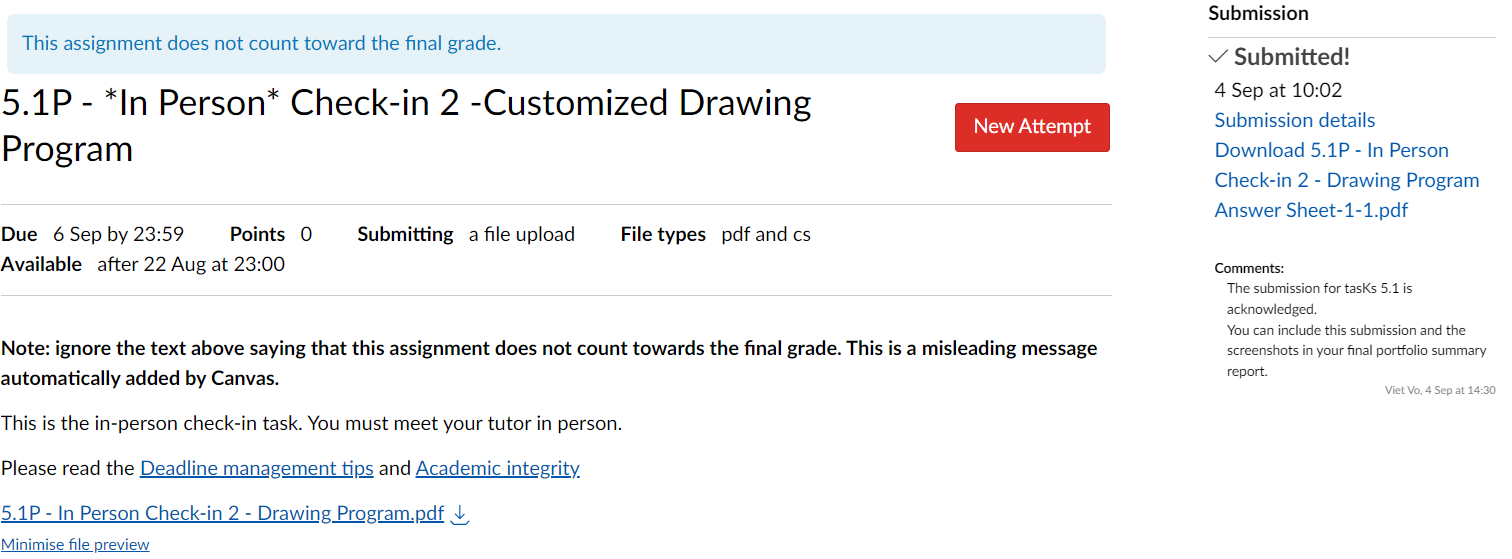
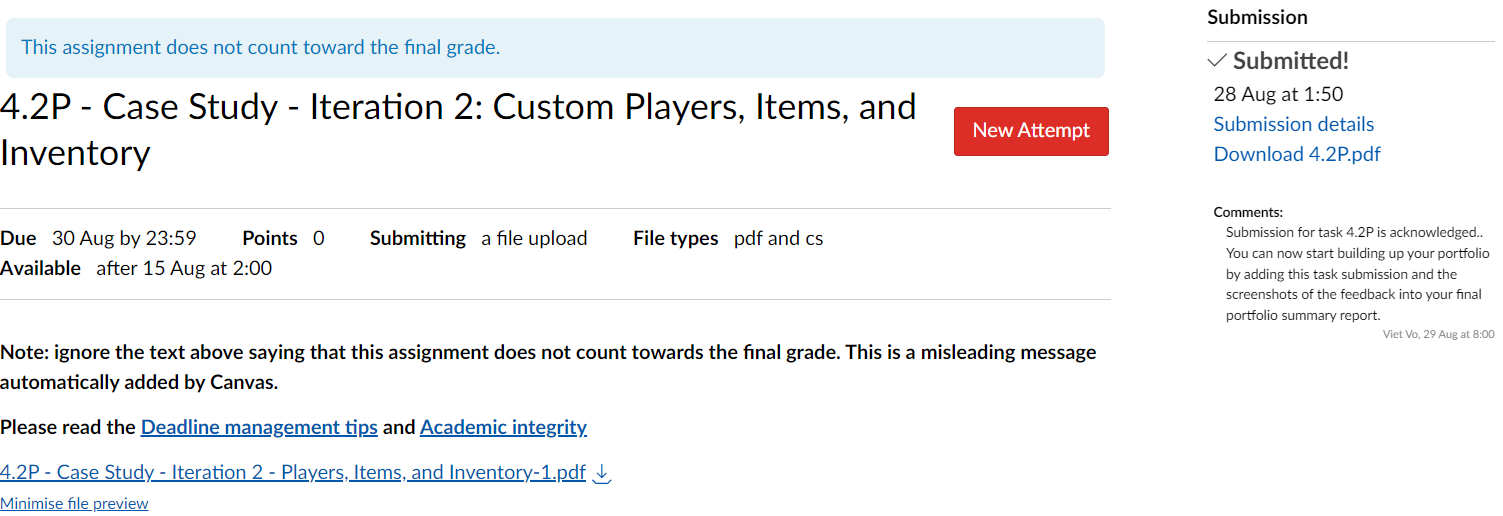
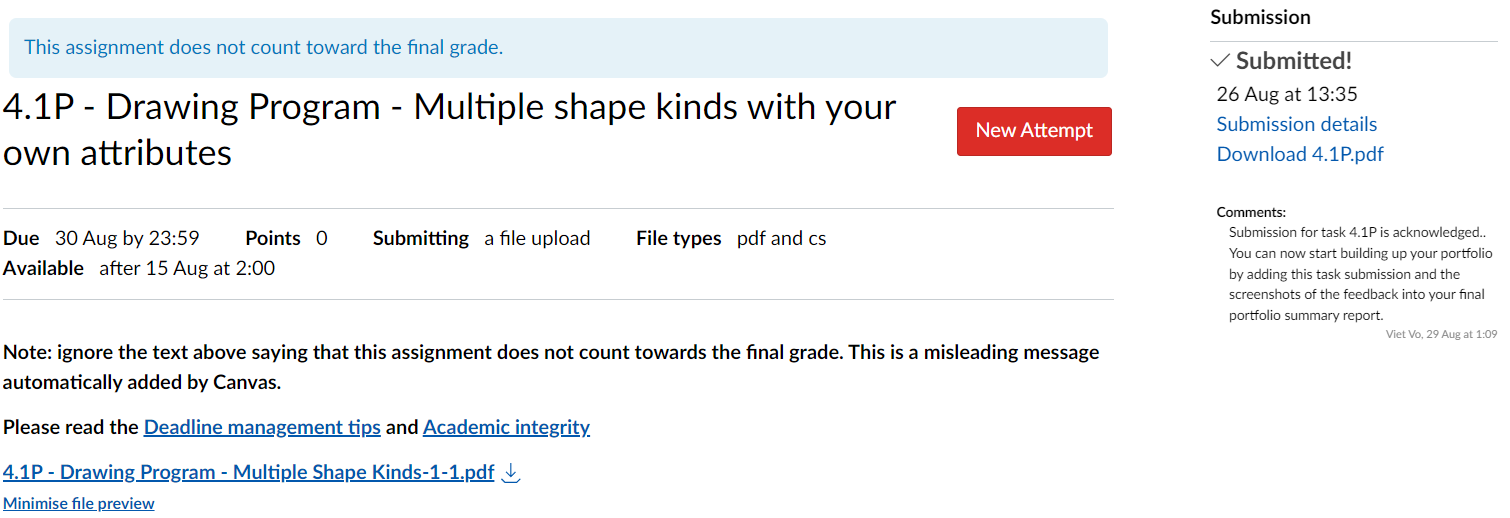
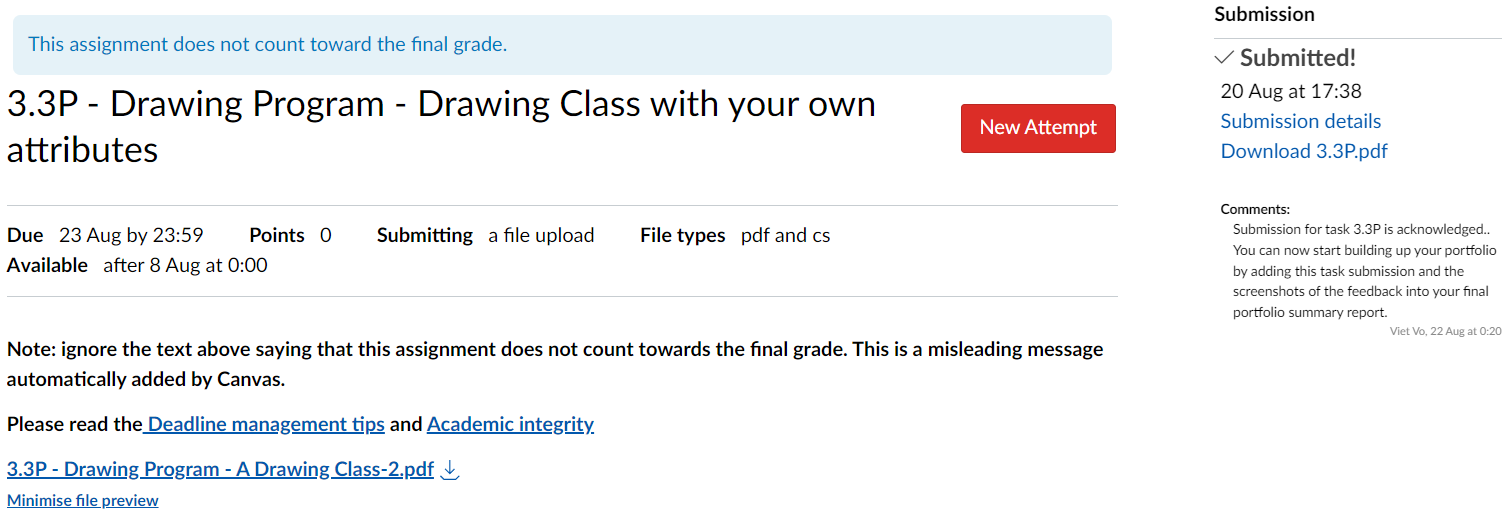
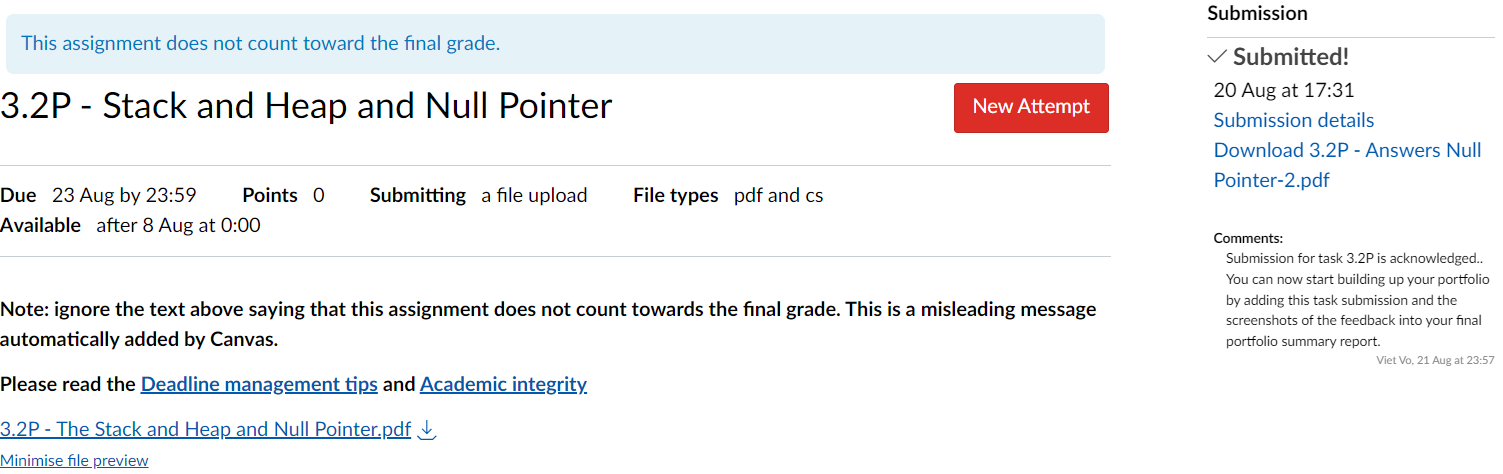
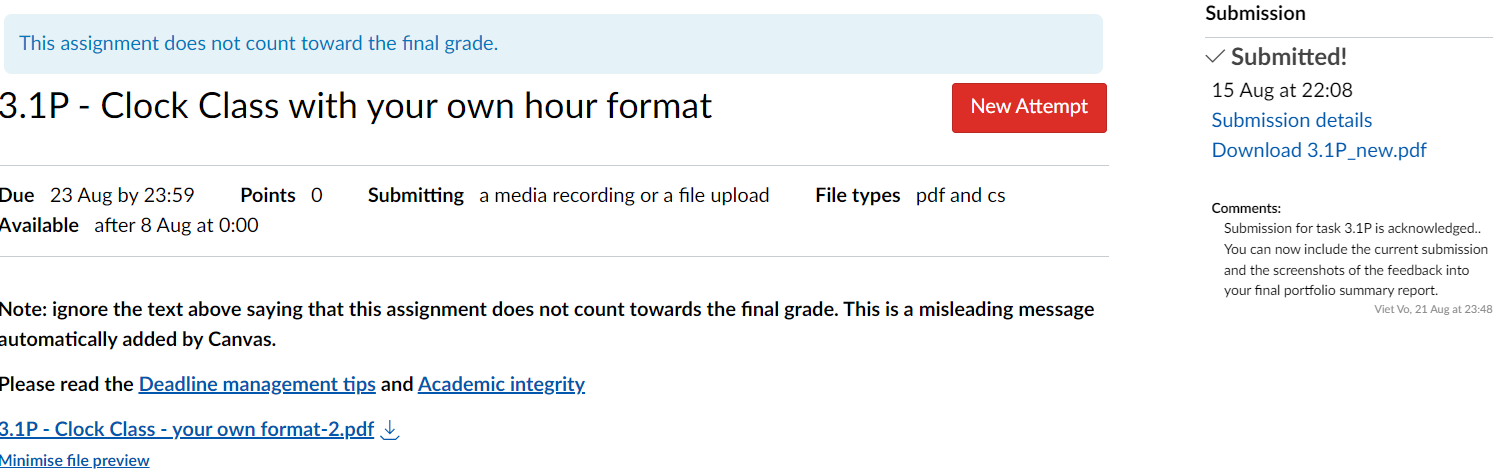
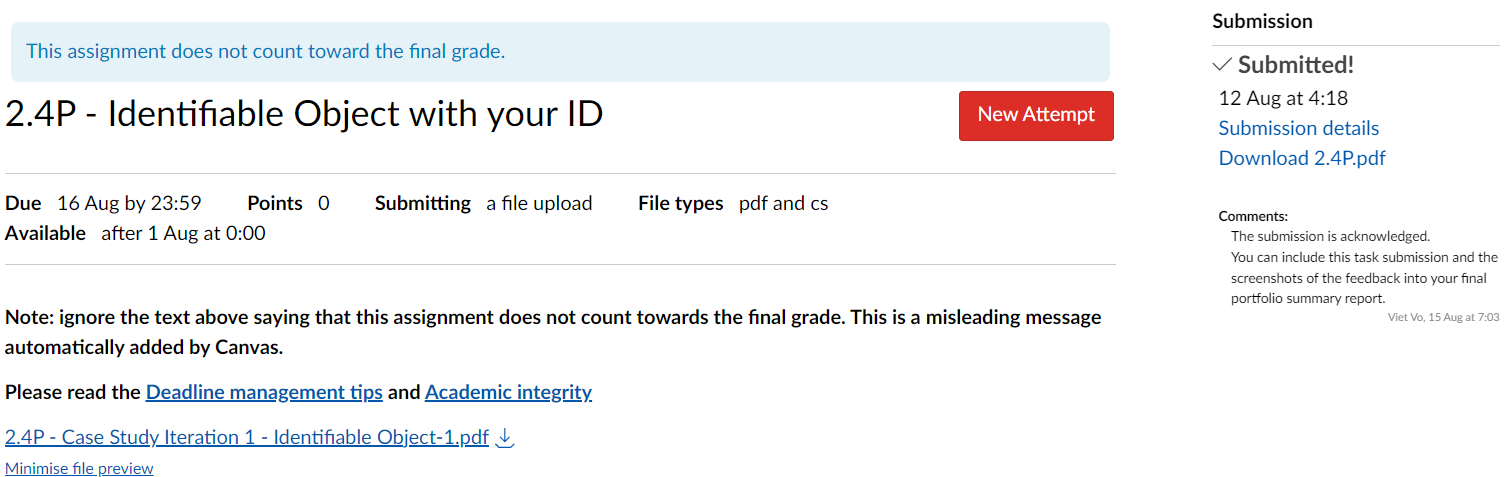
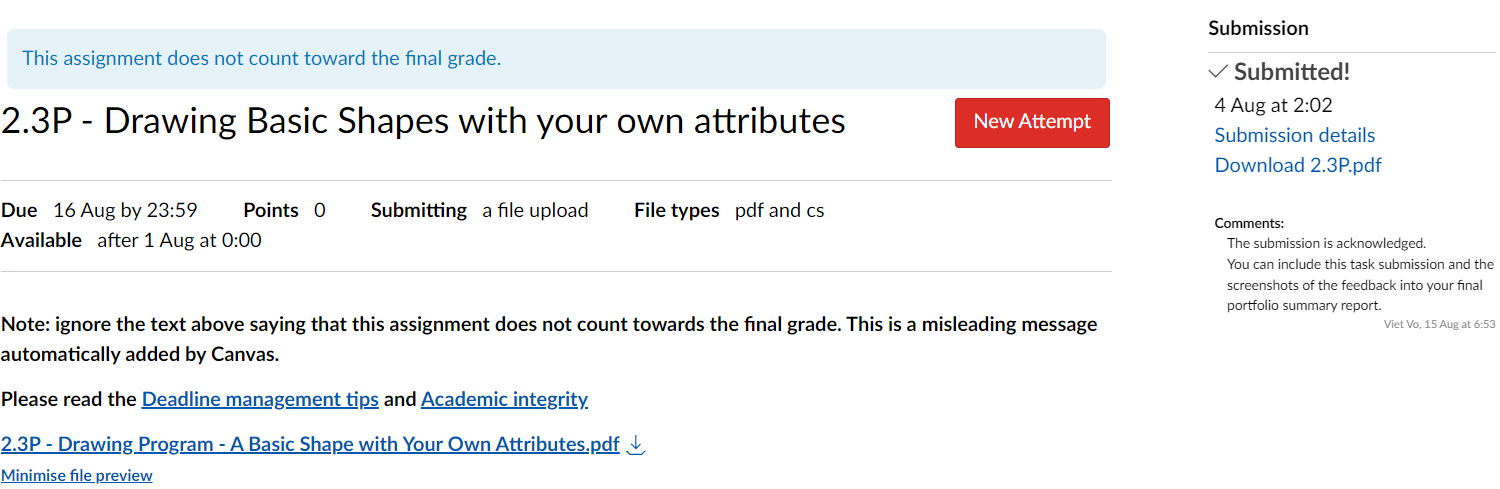
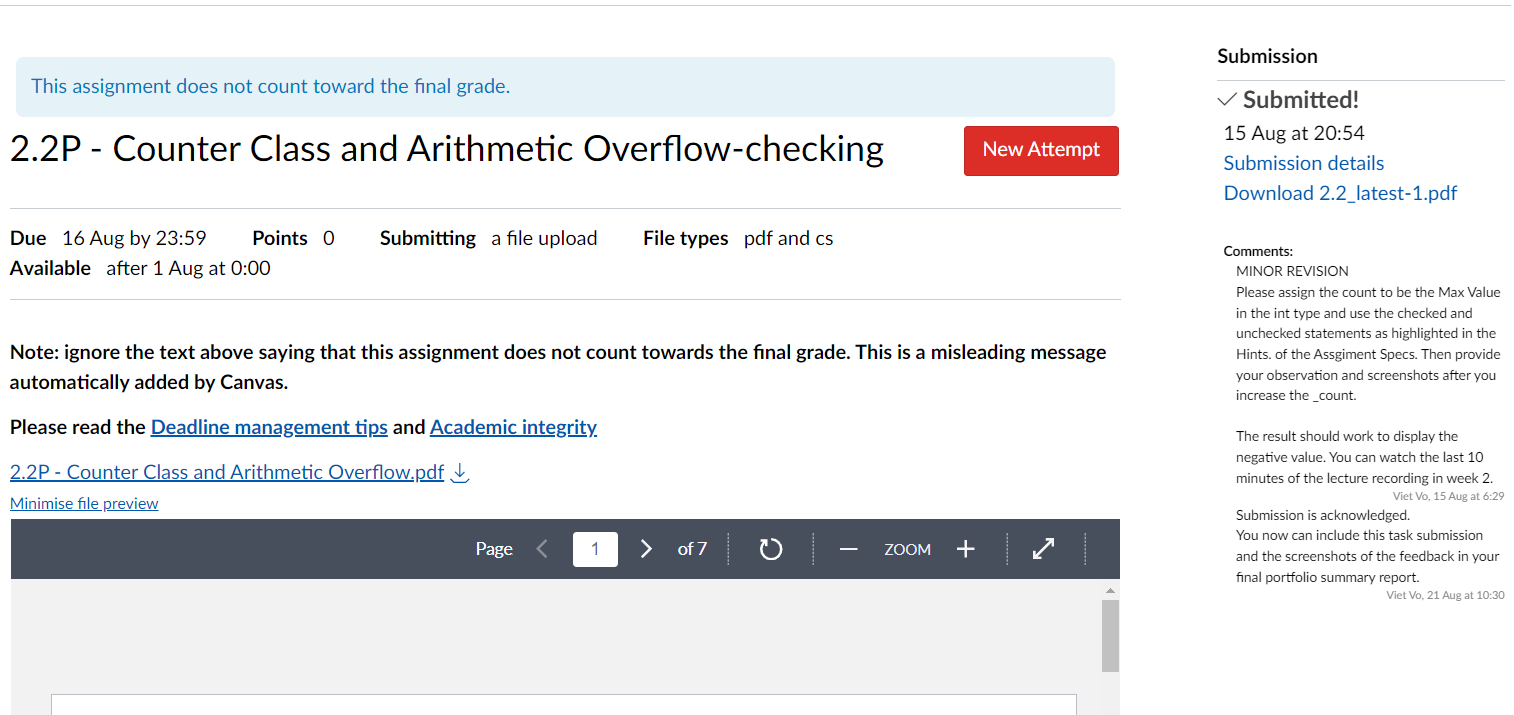
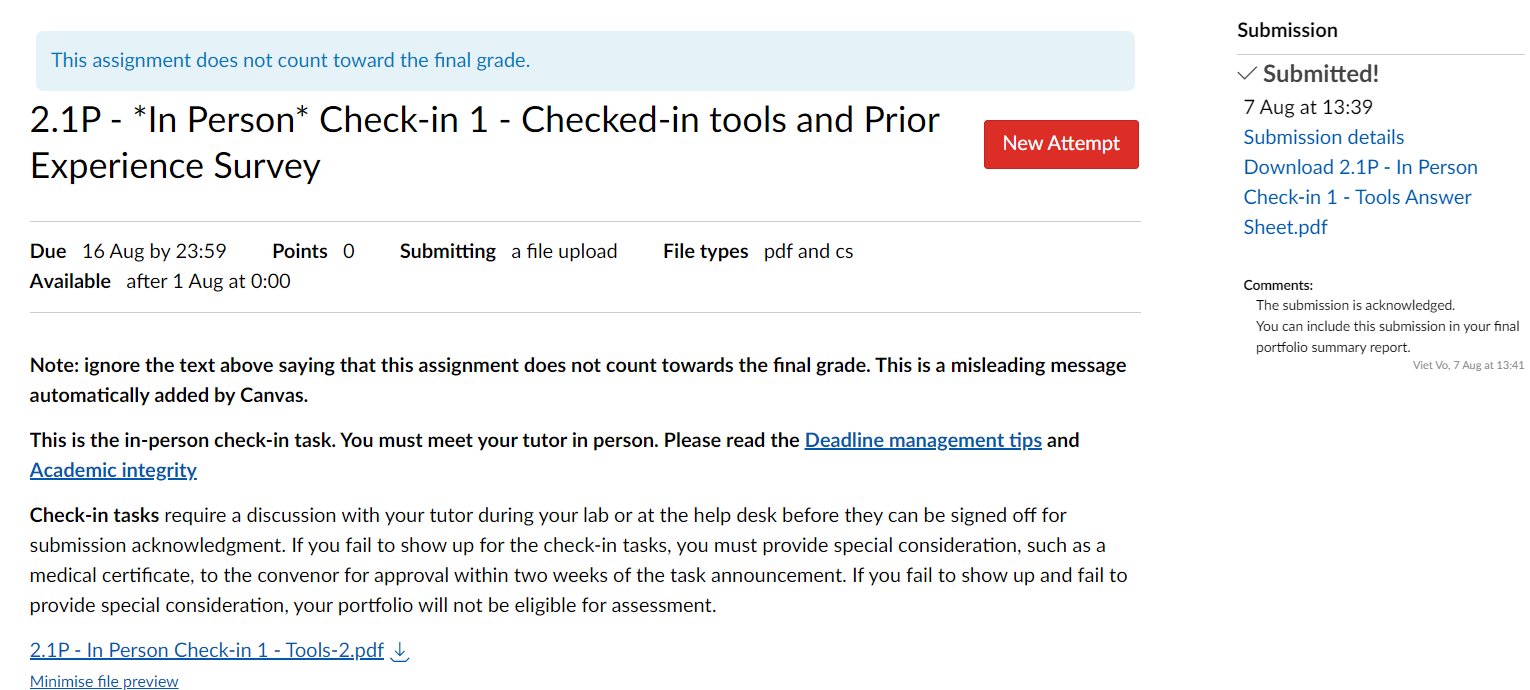
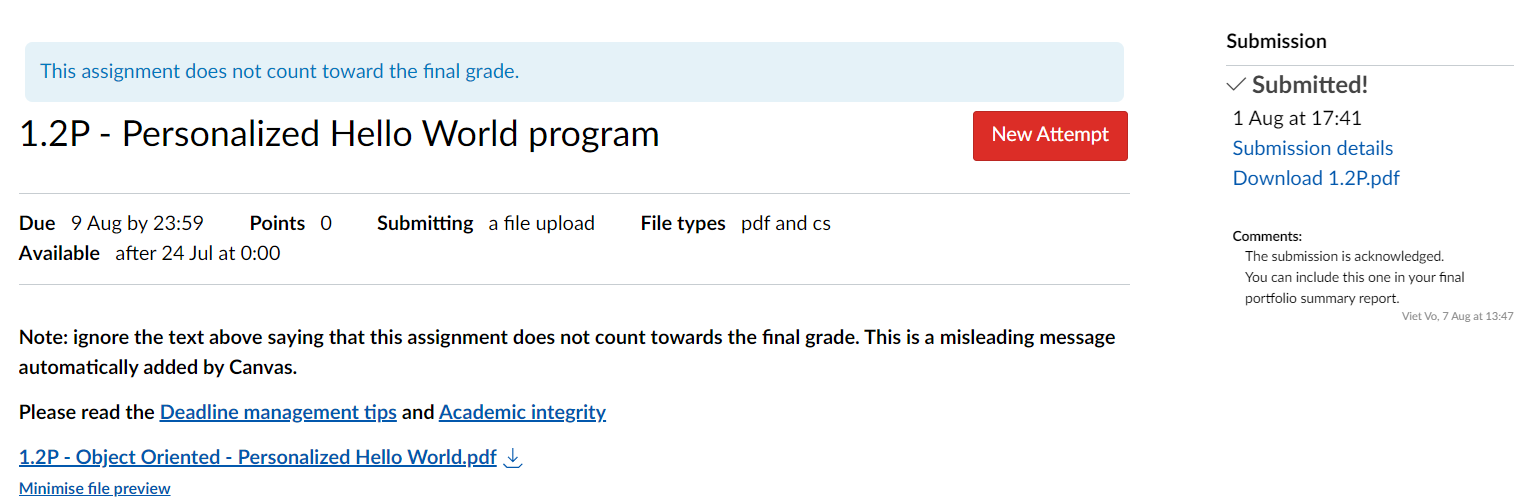
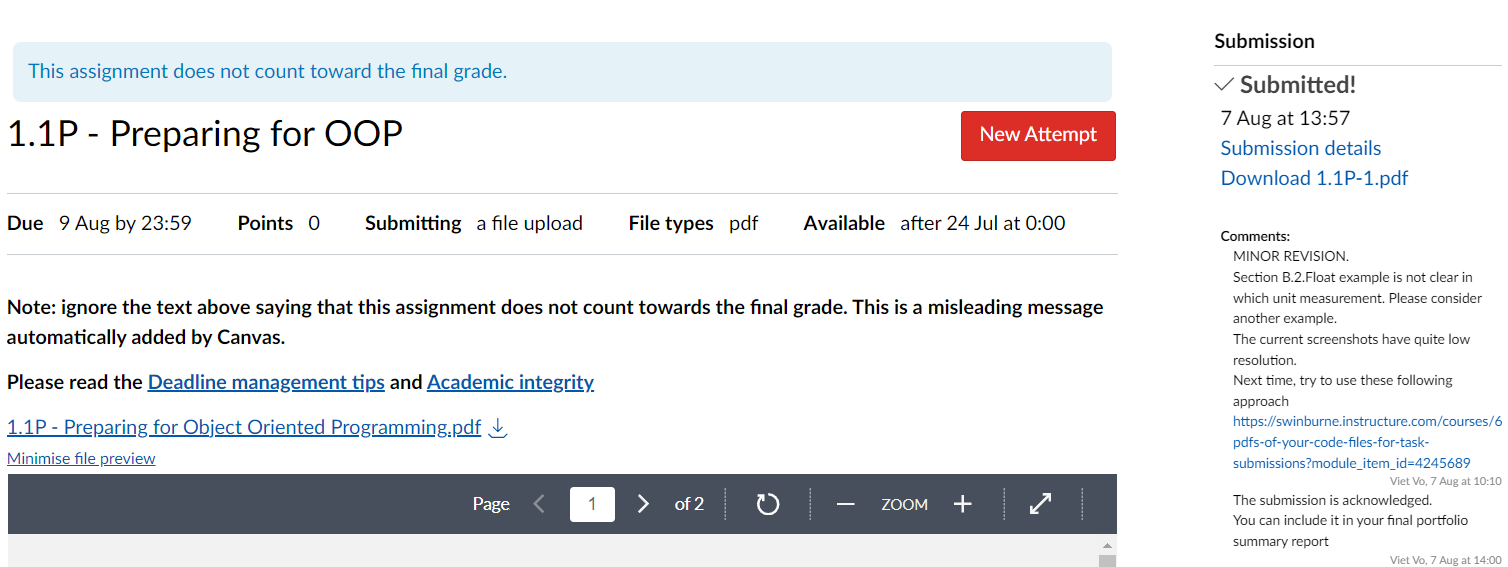
Learning Summary Report

# Self-Assessment Details

The following checklists provide an overview of my self-assessment for this unit.

In addition to the checklists, please append the following supporting appendices to your portfolio:

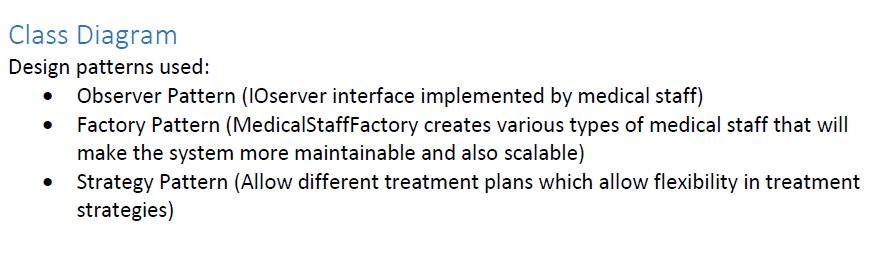
* **Appendix I:** A list of screenshots showing each task title and the feedback from your tutor upon submission.



* **Appendix II:** A summary of task corrections. For each task, if you have made any amendments or corrections that have not been previously assessed by your tutor, please summarize these changes. If there are no corrections or updates, indicate that the task submission from previous weeks (up to Week 12) remains unchanged.

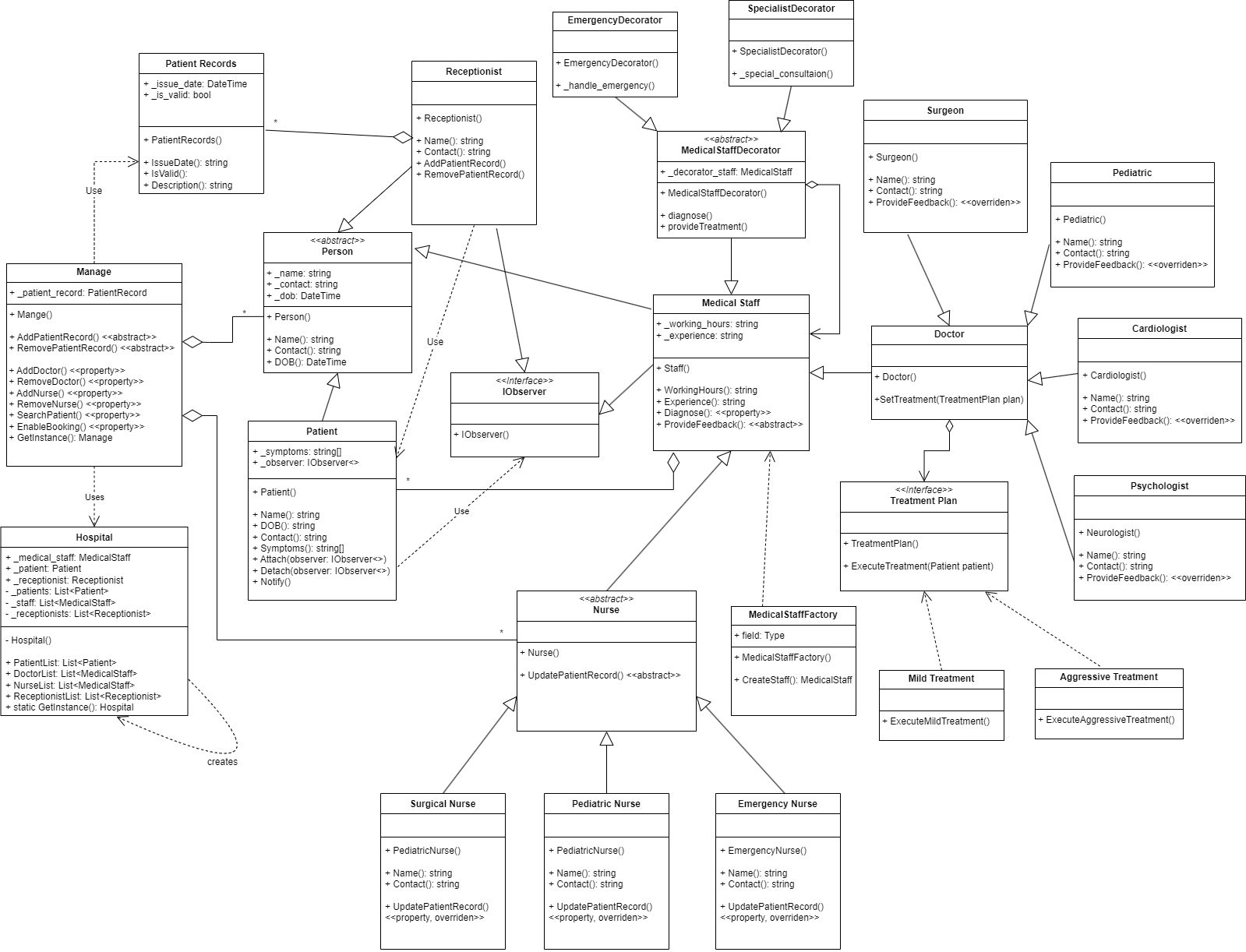
Task 6.3D – Custom Program Initial Plan:

* Added design patterns list and explanation where the design patterns used in the UML



Task 6.5HD – HD Level Custom Program Initial Plan:

* Edit the UML diagram so that it have 5 design patterns



* **Appendix III:** A list of your up-to-date **corrected** task submissions in PDF format. These should reflect any changes mentioned in Appendix II. If your submission has already been assessed by your tutor on a weekly basis and there have been no changes, you do not need to resubmit the PDFs, as we already have them.
* **Appendix IV:** The source code of all your previous task submissions in compressed zip format. Regardless of whether you made corrections or not, you must include all source code for your task submissions. Compress the source code into one or more zip files and submit them to Canvas along with your portfolio report. For example, the first .zip file can contain all the source code for the Hello-World program, The Counter, and Clock projects. The second .zip file should contain all the C# source code (.cs files) and test case implementations for your task submissions related to the Shape Drawing project. Finally, the third .zip file should include all the C# source code (.cs files) and test case implementations for your task submissions related to the Swin-Adventure case study.
* **Appendix V.** If you repeatedly receive minor/major revision feedback after the T1-1 - Semester Test Fix and Resubmit, you can still resubmit your corrections in this appendix. You need to summarize how you addressed the feedback and submit your full test solution again. However, this will result in a mark deduction from your final grade.
* Alternatively, if you failed to submit the T1-1 - Semester Test Fix and Resubmit by the deadline, you can still submit it in this appendix. However, this will result in a **FAIL** **grade** for the unit if your test resubmission is incorrect.

**Remarks:**

Failure to provide the source code for any task submission will result in that task not being assessed, even if the source code is printed and included in the PDF submission. The teaching team needs the source code to execute it and verify correctness. Additionally, the source code will be used for plagiarism detection and academic integrity checks.

Self-Assessment Statement

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Pass (D) | Credit (C) | Distinction (B) | High Distinction (A) |
| Self-Assessment |  |  |  | ✓ |

Minimum Pass Checklist

|  |  |
| --- | --- |
|  | Included |
| Learning Summary Report | ✓ |
| Test is Complete | ✓ |
| C# programs that demonstrate coverage of core concepts | ✓ |
| Explanation of OO principles | ✓ |
| All Pass Tasks are Complete | ✓ |

Minimum Credit Checklist (in addition to Pass Checklist)

|  |  |
| --- | --- |
|  | Included |
| All Credit Tasks are Complete | ✓ |

Minimum Distinction Checklist (in addition to Credit Checklist)

|  |  |
| --- | --- |
|  | Included |
| Custom program meets Distinction criteria & Interview booked | ✓ |
| Design report has UML diagrams and screenshots of program | ✓ |

Minimum Low-Band (80 – 89) High Distinction Checklist (in addition to Distinction Checklist)

|  |  |
| --- | --- |
|  | Included |
| Custom project meets HD requirements | ✓ |

Minimum High-Band (90 – 100) High Distinction Checklist (in addition to Low-Band High Distinction Checklist)

|  |  |
| --- | --- |
|  | Included |
| Research project meets requirements |  |

# Declaration

I declare that this portfolio is my individual work. I have not copied from any other student’s work or from any other source except where due acknowledgment is made explicitly in the text, nor has any part of this submission been written for me by another person. Failure to meet this requirement will result in a failing grade for the unit.

Failure to provide the source code for any task submission will result in that task not being assessed, even if the task is included in PDF format.

Signature: **Nguyen Duc Thang**

# Portfolio Overview

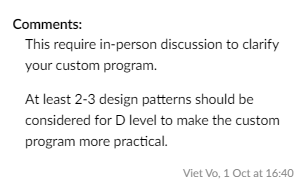
This portfolio includes work that demonstrates that I have achieve all Unit Learning Outcomes for COS20007 Unit Title to a **High Distinction** level.

I have completed all the task and received acknowledgement for the P and C marks, I also submitted the initial plans for my D and HD custom program and my custom programs contains 5 design patterns, also my file saving system for the patient records are able to save and update the saved data without deleting the whole file information and replace with the new data, this feature extend beyond from the C task of just saving and deleting the data and repeat the process which is inefficient.

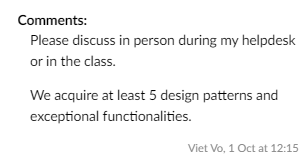
# Task Summary

To demonstrate my learning in this unit, I would like the following tasks to be considered part of my portfolio:

* 1.1P: Completed
* 1.2P: Completed
* 2.1P: Completed
* 2.2P: Completed
* 2.3P: Completed
* 2.4P: Completed
* 3.1P: Completed
* 3.2P: Completed
* 3.3P: Completed
* 4.1P: Completed
* 4.2P: Completed
* 5.1P: Completed
* 5.2P: Completed
* 5.3C: Completed
* 6.1P: Completed
* 6.2P: Completed
* 6.3D: Submitted and received comments and resubmitted



* 6.4D: Submitted
* 6.5HD: Submitted and received comments and resubmitted



* 6.6HD: Submitted
* 7.1P: Completed
* 7.2C: Completed
* T1 –Semester Test: Submitted and received minor revision
* T1 –Semester Test Fix and Resubmit: Completed
* 9.1P: Completed
* 9.2C: Completed
* 10.1C: Completed
* 11.1P: Completed

# Reflection

## The most important things I learnt:

In this unit I think I’ve learnt a lot of valuable knowledge about OOP such as the 4 basic concept in OOP (Polymorphism, inheritance, abstraction and encapsulation), how to develop a good coding project using that concepts and also extend them by using design patterns to make the solution more practical and scalable.

## The things that helped me most were:

I think the online lecture every week and the materials in Canvas module helps me a lot in the first few weeks when starting this unit because this is the first time I’ve never done my submission in Adobe Acrobat and also setting up and use Splashkit.

## I found the following topics particularly challenging:

When starting to learn about the OOP concepts I have some struggles in understanding the 4 basics concept in OOP (inheritance, polymorphism, abstraction and encapsulation) and interpret those concepts so that everyone can understand but with the help of my tutor Viet Vo, I started to do some deeper research about the concepts in different coding languages focused more in OOP such as Java and C++, now I think I have a better understanding about OOP and how it applies in making an optimal solution to solve real life problems in businesses.

## I found the following topics particularly interesting:

To me in this unit I think the design patterns is the topic that makes me interest the most because the design patterns help the coding solution more flexible and scalable in solving customer different requirements. For instance you can easily change your coding project depends on the customer criteria without much edit to your coding project.

## I feel I learnt these topics, concepts, and/or tools really well:

I think I learnt well about UML sequence diagram, UML Class Diagrams and Design Patterns because I was able to do iteration 7 and 8 where you need to implement your own UML diagram and sequence diagram to make the Path and Command processor for the game. Additionally I also create my own UML diagram and sequence diagram with 5 design patterns for my custom program.

## I still need to work on the following areas:

I think I still need some improvement in General Responsibility Assignment Software Patterns (GRASP) in OO design because my UML class connections in the Semester Test wasn’t optimal enough so I have to fix the connection arrows between the classes.

## My progress in this unit was …:

During the semester, I tried to submit my weekly task at the start of that week or before my class with the tutor so that I maintain the work flow throughout the semester and also ensure that if there any problems I need to fix then I still have time to adjust my errors. I also discuss and ask about the knowledge that I don’t understand in that week content and next week content with my tutor in class so that I can ensure about my task for that week and plan for next week tasks. I think scheduling your work helps me a lot in the unit and future units because it gives me more controls of my work and I can adjust if others unit need more time to study.

## This unit will help me in the future:

This unit will be the foundation for studying future units and also future works because it helps me understand how to create a coding projects using OOP and how are the criteria to make a good coding solution that will help solve business problems.

## If I did this unit again I would do the following things differently:

If there’s a chance to do this unit again, I might plan my work schedule better, start things as early as possible and seek as many helps as possible. During my study from week 6 and from week 9-12 I wasn’t able to focus fully focus on designing the custom program because I wasn’t predicted correctly the workload of other unit such as Network and Switching and Cloud Computing architecture so I have to split the work for others unit assignments.

## Other…:

[ Add any other reflections you think help you demonstrate your learning ]