**HW 7 – CS 4321, Fall 2015**

|  |  |
| --- | --- |
| Name: | Tyler Cabutto |

**Questions – Lesson 8: A Tale of Analysis & Design (37 minutes) & Lesson 9: Design Patterns (23 minutes)**

Watch videos, type answers (leave questions), print before class and turn-in in class.

1. Write a paragraph about the Lesson 8 video focusing on what you learned.

* Lesson 8 was a nice role play on how to work with the people who are going to be using the software that we are designing. It allowed them to build the UML based off what was suggested by the professor, but then getting the librarians input which helped made more since of what was going on, and assisted with avoiding any issues later on when you go to implement the software. It shows that you need to be a good communicator and be open to criticism of what you have built so far to avoid any excessive issues later on.

1. Define the five types of patterns.

* Fundamental patterns: The basic patterns.
* Creational patters: Support object creation
* Structural patterns: Help compose and put objects together.
* Behavioral patterns: focuses on realizing interactions among different objects.
* Concurrency patters: Support concurrency.

1. Why is the Factory Method pattern appropriate for the ImageReaderFactory example?

* Because you start off not knowing what type of image that you are going to create until the code is actually ran.

1. Explain an example of the Strategy pattern not found in the video lecture.

* A good example would be how a character in a video game can switch between a few different modes of movement dependent upon the input given by the user such a flying, teleporting, walking, etc.

Source: **https://www.youtube.com/watch?v=MOEsKHqLiBM**

1. Explain an example of the Decorator pattern.

* A example could be a class that makes coffee. You can have a regular cup of coffee with no milk, or you can add the decorator class that can mix milk in, or sprinkles and it does not affect the original class.

1. Explain the intent of the Observer pattern in terms of push and pull.

* It is there to notify a user of a change that they would like to be aware of. For example, you may have a program that is constantly changing in the background without notifying you. If there is something you want to be notified of, it will push a notification to you. The same can be if something is being pulled it will let you know.

1. What is *abusing patterns*? What is a consequence?

* Using too many patterns and ending up with a design that is more complicated rather than less complicated.