ECSE 321 – Assignment #3

1. * 1. Repository architecture
     2. Pipe-and-filter architecture
     3. Client-server architecture
     4. MVC
     5. Layered architecture
     6. Advantage

Subsystem components are independent from one another. Changes can be made to one component without affecting the others.

Disadvantage

If there is an error with the central repository, the whole system fails.

* + 1. Advantage

Easy to understand. Flow resembles the structure of many business processes.

Disadvantage

A lot of data transfer between components. Data must be parsed so that the output data matches the correct format for the input of the receiving component every time data is transferred.

* + 1. Advantage

Servers can be distributed across a network of clients. Multiple clients can use the same server at the same time.

Disadvantage

If a server fails, all the clients who were using this server can no longer assess the system.

* + 1. Advantage

Allows for multiple ways to use and interact with data without changing the way the data is stored.

Disadvantage

Can lead to unnecessarily complex code for a system with a non-complex model and simple interactions.

* + 1. Advantage

Layers are separated from one another. Entire layers can be replaced as long as the interface is maintained.

Disadvantage

System can have poor performance since the data must pass through each layer every time a request is made.

* 1. 1. The pipe-and-filter architecture would be best suited. Processing insurance claims is a multi-step process that involves data flowing from one process to another. The pipe-and-filter architecture structure mimics this behavior.
     2. Layered architecture would be best suited. An OS like Linus should use layered architecture to maximize security. Users should only be able to interact with an interface that must pass through several layers in order to make changes to the system. The bottom layer would be the kernel of the OS.
     3. Client-server architecture would be best suited. Many clients wish to access Twitter at the same time. Thus, Twitter can use servers to allow these clients to all access Twitter.
     4. MVC would be best suited. The Event Registration System needed to perform on desktops, mobile devices and web browsers. Thus, by using MVC architecture, we can re-use much of the code since the view and model do not depend on one another.

**3)**

a) Over-engineering is the unnecessary over-use of design patterns.

It should be avoided because it makes the code more complex for

no reason. Thus, it unnecessarily makes the code harder to read

and debug. It also can decrease the performance of the system.

b) A god “blob” class is an example of an anti-design pattern. It is

undesirable because it will cause the system to lack cohesion. Most

of the processing of the system occurs in the blob class and other

classes are not used efficiently.