Janajan Jeyabalan - 100698148
Faraaz Mohsin - 100659110
Rohan Radadiya - 100704614
Abdullah Malik - 100698109

Question 4

MVC Pattern

The MVC Pattern is used to separate any type of application concerns and it stands for Model-View-Controller Pattern. The Model aspect of MVC represents an object that carries data and it utilizes logic to update the controller if data changes. The View aspect of MVC represents the visual representations of the data that the model will contain. The Controller aspect of the MVC Pattern keeps the view and model aspects separate and it controls the data flow into the model object and updates the view when data changes.

The advantages of the MVC Pattern are that the development of the application becomes faster as it is easy for multiple developers to collaborate and work together. Since the development is fast it is also easier to update the application.

The disadvantages of the MVC Pattern is that it is hard to understand the pattern itself as it must have strict rules on methods. This pattern is also not scalable as the time it will take an individual to make it more scalable the whole pattern is not considered as an MVC pattern at that point.

Observer Pattern

The observer design pattern is a behavioural pattern that has a mechanism that notifies multiple objects about events that occur on an object for which it is being observed.

The advantages of the Observer Pattern is that without any change in the Subject or Observer classes it allows sending data to other objects efficiently. In addition, another advantage is that observers can be added and/or removed at any point in the process.

The disadvantage of the Observer Pattern is that the observer can add complexity and lead to inadvertent performance issues if not correctly implemented. Moreover there is no option for composition, as the observer interface can be embodied.

Dependency Inversion Principle

The dependency inversion principle is also known as DIP and it is considered a behavioural design principle that should depend on the abstraction as the high-level modules should not depend on the low-level modules.

The advantages of the dependency inversion principle is the loose coupling and the fact that it beneficially has an interface-based design.

The disadvantages of the dependency inversion principle is it requires more upfront development effort and it demands that the configuration details must be supplied in the construction code.