# Assignment #1


Submitted by:

Najam Ali Abass

(p20-0471)

Submitted to:

Mrs. Sara Rehmat

(Instructor Introduction to

Software Engineering)

Session 2020-2024

# Model View Controller

Model View Controller is a predictable Software Design Pattern that can be used across many frameworks with many programming languages, commonly Python,Ruby, PHP, JavaScript, and more. It is popularly used to design web applications and mobile applications

Like Slader is web application which is made with the concept Model View Controller design, as slader is viewed in browser where internet is available and after that we can also pass information along the browser like we search for different questions of our exercise. And the answer for our question and other information is stored in the database and different servers.


# 2.Layered

Layered architecture patterns are n-tiered patterns where the components are organized in horizontal layers. This is the traditional method for designing most software and is meant to be self-independent. This means that all the components are interconnected but do not depend on each other. G-mail is divided into at least three layers, every one of them has a mission, and they exist separately to handle different processes at different levels. It is an excellent example of a layered architecture. Google has developed G-mail in different layers

a. there is an internal layer that does all the processing.

b. There is an external layer that is communicating with the users in their language.

c. There is also another that interacts with a database where user email messages are stored(million or may be billions).

# 3.REPOSITORY ARCHITECTURAL DESIGN

A repository architecture is a system that will allow several interfacing components to share the same data. Each component interfaces the same data set that is utilized system wide. Data manipulation taking place in one component will reflect an identical representation of data in another component. The best example an repository architectural design is Visual Studio which is an integrated development Environment for programming languages which is composed of many layers or components working independently on a central data structure. Like in VS code it allow us to write clean code with auto indentation and auto compilation as well as it auto complete. So all these components working on a central Data Base.

# 4. Client  Server Design

Applications are usually designed so that one computer act as a server(end system) , providing a service to other computers on a network. To access the server a program is run on users computer this is client-server program.
The most common example of this design is the web browser we are using like in my case I am using Google Chrome which is providing me information from a server which far far away from me , the data(information) is stored in server which we access through a network and it is displayed on our screens. So this is typical example of Client-Server Design.

# 5. Pipe and Filter Design

Pipe and Filter is a simple architectural style that connects a number of components that process a stream of data, each connected to the next component in the processing pipeline via a Pipe. The Pipe and Filter architecture is inspired by the Unix technique of connecting the output of an application to the input of another via pipes on the shell. A compiler performs language transformation: Input is in language A and output is in language B. In order to do that the input goes through various stages inside the compiler — these stages form the pipeline. The most commonly used division consists of 3 stages: front-end, middle-end, and back-end. The front-end is responsible for parsing the input language and performing syntax and semantic and then transforms it into an intermediate language. The middle-end takes the intermediate representation and usually performs several optimization steps on it, the resulting transformed program in is passed to the

back-end which transforms it into language B. Each level consists of several steps as well, and everything together forms the pipeline of the compiler.

# THE END