Architecture patterns

An architectural pattern expresses a fundamental structural organization schema for software systems.

It provides a set of predefined subsystems, specifies their responsibilities, and includes rules and guidelines for organizing the relationships between them.

Five architecture Patterns Used In Software Development.

We will talk about five architecture patterns that use in software development and their applications.

There are other patterns are used but we will just mention five of them.

- Layered pattern
- Client-server pattern
- Model view controller pattern
- Event bus pattern
- Microservices architecture pattern

Layered pattern

This is the most common architecture pattern in software development, here the software is divided into layers:-

- Presentation layer
- Application layer
- Business logic layer
- Data access layer

Each one has a specific role and responsibilities.

This pattern performs poorly in the high performance applications because it is not efficient to go through multiple layers to fulfill the business request.

Pattern applications

The layered pattern is usually used in building general desktop applications.

Client-server pattern

In client-server architecture there are two main components.

The client which is the service requester and the server which is the server provider, the client and the server may be located in the same system, they often communicate over a network on separate hardware.

Pattern applications

Typical real world use cases of the client server pattern include online applications such as email, document sharing and banking.

Model view controller pattern

The user interface is typically the most frequently modified portion of an interactive application.

How can user interface functionality be kept from application functionality and yet still be responsive to user input or to changes in the underlying applications data and how can multiple views of the user interface be created, maintained and coordinated when the underlying application changes.

Using MVC pattern will be a good solution which separate application functionality into three components

Model:- which contains the core functionality and data.

View:- which display the information to the user.

Controller:- which handles the input from the user.

Pattern applications

MVC is commonly used in web frame works such as Django.

Event bus pattern

The modern enterprise software is often built as a distributed system that can service asynchronously arriving messages associated with high volume of event.

The event bus pattern has four major components:-

- Event source
- Event listener
- Channel
- Event bus

Sources publish messages to particular channels on the event bus.

Listeners subscribe to particular channel, listener are notified of messages that are published to a channel to which they have subscribed before.

The advantage of this pattern that the new publishers, subscribers and connections can be added easily however, the scalability might be problem for this pattern as all messages travel through the same bus.

Pattern applications

Event bus pattern is often use in android development, e-commerce application and notifications services.

Microservices architecture pattern

The modern enterprise apps are required to support a variety of browsers and native mobile clients these days, the apps usually handle client request by executing business logic, accessing a databases, exchanging messages with other systems and returning response.

In this regard monolithic apps can become too large and complex for efficient support and deployment, the solution is build apps as microservices.

Each service is independently deployable and scalable and has its own API boundary, different services can be written in different programming languages, manage their own database and developed by different teams.

Pattern applications

Many use cases are applicable for a microservice architecture especially those that involve an extensive data pipeline.