The Role of Model-View Controller in Object Oriented Software Development

Ram Naresh Thakur[1] and U S Pandey[2]
[1]PhD Scholar, Mewar University, Chittorgarh, Rajasthan, India
[2]Professor, University of Delhi, India.

Corresponding Author
Ram Naresh Thakur
Email: aryanthakur@gmail.com

Abstract
Object Oriented Software Development (OOSD) is a design technique that is used before the development and design of a software. This design method makes the system appears as a collection of objects to communicate with other objects by passing messages. The Model-View-Controller (MVC) has been inherited from Object-Oriented Programming (OOP) with the integration of Graphical User Interface (GUI) and interactive program execution. The MVC is very useful for developing Interactive and Dynamic Web Applications and iOS. With MVC, developers can trust on design patterns that are widely accepted as solutions for recurring problems. MVC can be used to develop flexible, reusable and modular Software. Applying the MVC design pattern in object-oriented Software development a flexible, reliable, modular and scalable website can be built. So, it's necessary for every developer to have the knowledge of software development using MVC design pattern.

Keywords*: OOSD, MVC, OOP, GUI, Dynamic, Web Application, iOS, Design Pattern.*

1   Introduction
In today's era, it is very essential to develop and design a working software in an effective and efficient manner. As we know that Object Oriented software development is a designing technique that is used before the development and design of a software. Through this method the system appears as a collection of objects which communicate with other objects by passing messages. Basically, Object oriented concept is used for maintaining complexity of software systems. MVC is inherited from Object Oriented Programming with the integration of GUI and interactive program execution. [1] Similarly, Model-View-Controller is the design pattern for developing a system software efficiently. This MVC framework is more reliable for designing and developing Interactive and Dynamic Web Applications as well it is more common in an iOS development. It has become the most powerful and dominant Programming Paradigm for developing large scale and Dynamic Web Applications and iOS. Moreover, it helps in classifying objects in accordance to their role and encourage for the separation of code in their respective role. Especially, MVC is a three-layered architectural design which divides the characteristics of particular app.

1.1   MVC Architecture
MVC is a software design pattern built around the interconnection of three main component types: Model, View, and Controller, often with a strong focus on Object-Oriented Programming (OOP) software paradigms. MVC is a framework for building web applications using an MVC Design. It is the most important architecture of development the software now a day. This

architecture automatically managed the code and help the programmer to develop the well-managed Web Applications.

How these Model-View-Controller [2] acts in the software development is given below:
1. A Model is an object representing data or a process such as database, and process of machines. For instance, the application logic is a model.
2. A view is some form of visualization of the state of the model. The user interface component is view.
3. A controller offers facilities to change state of the model. The control functions as controller.

Before the development of software, it goes through a lot of phases, they are: logical conceptual designing, developing, testing and many more. It is very crucial to plan a good architectural design in order to make sure that the software is modular, stable and scalable before coding it. MVC plays a great role in Object Oriented Software Development because it supports rapid and parallel development as well as makes the development and maintenance simpler. Additionally, two different programmers can work on two different areas such as one can work on the view while others can work on controller section simultaneously. This helps in to generating business logic for a particular object-oriented software development. Furthermore, With MVC, developers can trust on design patterns that are widely accepted as solutions for recurring problems and used to develop flexible, reusable and modular Software. [3] Because of its faster development process and it offers multiple views without any problem. Since MVC has plethora of advantages in object-oriented software development and hence it is often used in dynamic web development and iOS development.
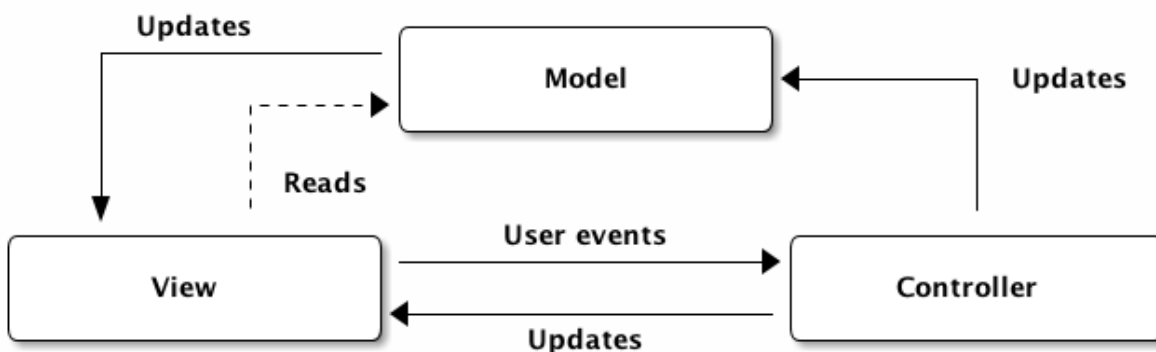

Figure 1: MVC Architecture

In recent years, most of the large dynamic website and iOS has followed MVC design patterns in Object oriented software development because without implementing Object Oriented Concept the code is not understandable by the developer itself and it is difficult to understand the flow of the code. In the programming world, the View manages the rendering of output; while the Controller interprets device input events and triggers appropriate changes in Model and View; the Model manipulates the behavior and data structure. The Model responds to requests for its state information from the View and instructions to change states from the Controller. The View and the Controller pair are associated with each.

1.2 Advantages of MVC architecture

- Multiple Views of the same Model can be used simultaneously. New data Views can also be introduced at any point of time.
- MVC prevents tight coupling between the objects. Since the dependency of a class with so many other classes are reduced, the class can easily be re used.
- The application's look and feel can be altered without affecting the Business Logic or the data.
- Different interfaces or user levels can be maintained by the same application.
- The entire application can be built or managed independently by Business Logic developers, flow of control developers and web page designers.
- Analysis and Impact of Using Model/View/Controller (MVC) Pattern for Web Applications
- The MVC is ideal to maintain an environment comprising of different technologies across different locations.
- MVC promotes scalability and maintainability.
- Patterns, particularly MVC is an extension of object-oriented programming and hence promote reusability.
- MVC structures include more of interfaces than implementations.
- Therefore, it can be plugged into any application to address a specific issue without affecting the remaining part of code / functionality.

2 Literature Review

In case if any changes are required then the whole system has to change. That's the reason for using MVC pattern widely in object-oriented software development. The facilities have given by different platform to use MVC design pattern in object-oriented programming in several programming languages like JAVA, .Net, PHP and so on. There are some built in features in MVC framework such as authentication, session management, transactional business logic, web application security, object relational mapping, localization, membership and roles and URL authorization etc. there has been development of many e-commerce site, online shopping sites, news portal, movie ticketing, colleges and several mobile applications using MVC in object-oriented software development. Various interfaces can be maintained by the same application [1].

Customer Relationship Management (CRM) has been as important to the Banking sector at the start of the 21st century as it has been to any other sector. Banks have used E-CRM tools to acquire more customers and to improve relationships with them. MVC pattern is a software design pattern that is suitable for interactive system. This paper analyzes the CRM characteristics: interactive, expansibility, the complexity of data and integration. To these characteristics of CRM, it provides a system design scheme on MVC pattern, and describes the advantages of function and structure in the system [3].

Biometric Authentication Systems are now widely recognized as the strongest authentication technologies available on the market. To incorporate this mechanism as a web service, an alternative to the classical approach constituted of a classifier and a vector of facial biometric features is proposed to be used. In order to make it possible, the adopted methodology based on

Model–View–Controller (MVC) architecture is used. This accredited architectural strategy in the proposed system, allowed us by the end to do online authentication based on the facial recognition with good obtained results of experimentation on ORL database [4].

3 MVC Architecture using ASP.NET

In the given MVC Architecture which was used to develop the application in ASP.NET. The View was developed using an Active Server Pages (ASP) and the Controller is in C# code. The Views were developed using ASP.NET web pages. There are two engines for developing the Views, namely Razor Page (*.cshtml) and ASP.NET Page (*.aspx). The razor engine was used to develop all the Views, the Controllers were developed using C# classes (*.cs) and the Models were developed using Entity Classes which are also C# classes (*.cs).
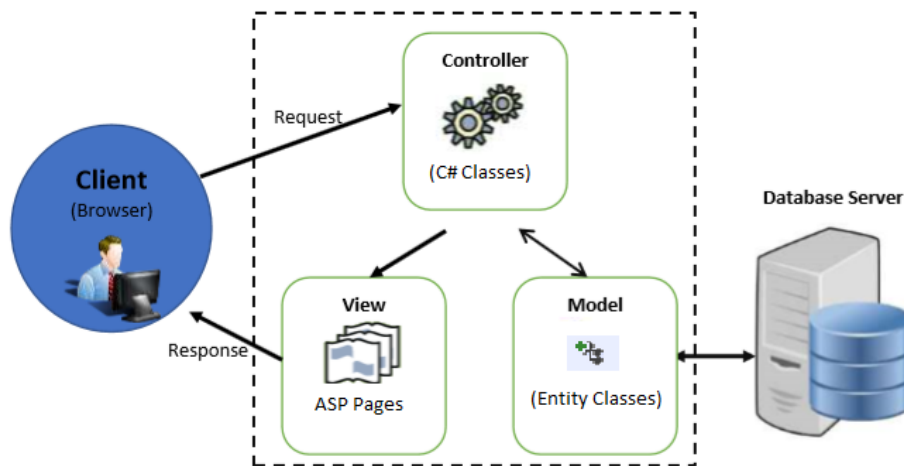

Figure 2: MVC Architecture using ASP.NET

3.1 Implementation in ASP.NET MVC

The Application was implemented in ASP.NET using ASP.NET MVC 5 Framework. ASP.NET MVC is a Framework for developing Web Applications that applies Model-View-Controller Design Pattern to the ASP.NET. The Razor View Engine was used for Views, C# Controller Classes for the Controllers and C# Entity Classes for the Models. The Razor View Engine combines the C# codes and HTML 5 tags to generate Dynamic Web Pages. The Controllers are written in C# Language to process incoming requests, perform an operation on the model, and select views to render to the User. The Entity Classes represent Database Object and they are responsible for implementing Business Logic [2].

3.1.1 Implementation Tools

The following tools were used for Implementation in ASP.NET
- Visual Studio 2019 Community Edition
- C# Programming Language
- ASP.NET MVC Razor View Engine
- Microsoft SQL Server 2014 Express Edition
- IIS Express (Internet Information Services)

4   MVC Architecture using JSP

In this MVC Architecture which was used to develop the application in JSP. The View was developed using a Java Server Page (JSP) and the Controller is in Java code. The Views were developed using JSP web pages. There are two engines for developing the Views, namely HTML (*.htm or html) and JSP Page (*.jsp). The razor engine was used to develop all the Views, the Controllers were developed using Java Servlet classes (*.java) and the Models were developed using EJB (Enterprise Java Bean) or Java Classes (*.java).
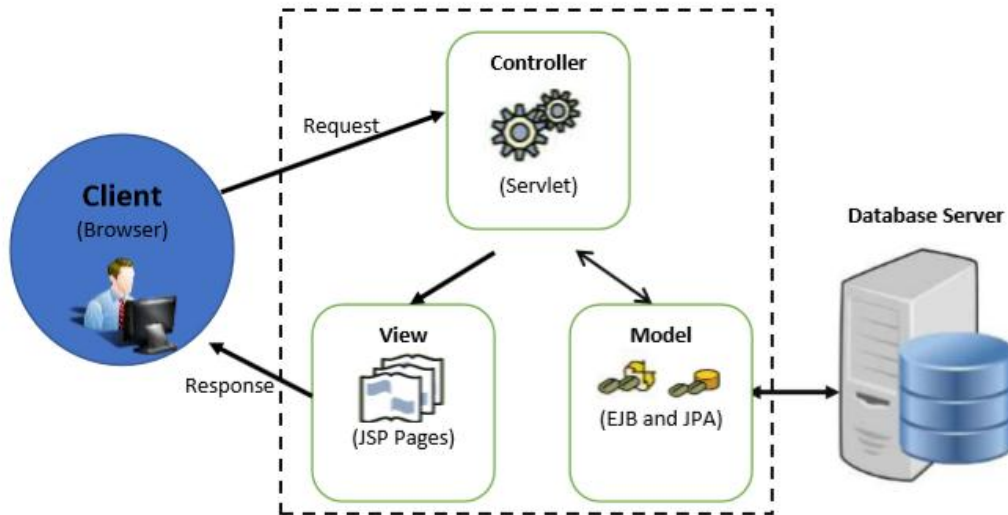

Figure 3: MVC Architecture using JSP

4.1   Implementation in JSP MVC

Developing Web Application in Java, The Models were developed using Enterprise Java Beans (EJB) and Java API(JPA), the View where developed using Java Server Page (JSP), The Controllers were developed using Java Servlets(.java). The given figure shows the MVC design pattern using Java.

4.1.1   Implementation Tools

The following tools were used for Implementation in ASP.NET
- NetBeans 8.0.2
- Java Programming Language
- HTML 5
- MySQL 7
- Glassfish 4.1

5   Results and Discussion

This research paper was based on the Role of Model-View-Controller in Object Oriented Software development. We have discussed how MVC is essential and how a developer can produce a working, and genuine software by using it. Also, we discussed the use of Model view controller makes software flexible, reliable, and scalable. And we had found that modification doesn't affects the entire model so if anything is required to change such as screen layout, fonts, extra supports for mobile phones then it can be done easily.

MVC has played a vital role in object-oriented software development. It has made greater influence in the web development and iOS development. IT has helped in reducing complexity of application software and separates the application in three major components i.e. Model, View and Controller. It gives a developer full control over the developed application. Three developers can work simultaneously. Also, testing of the software would be easier and more effective.

6   Conclusion

To conclude, this research is done to analyze how MVC plays an important role in Object Oriented Software development. An amazing architectural design pattern i.e. Model-View-Controller design pattern helps to maintain and define the good quality of software. By using MVC in object-oriented software development can make software more flexible, clear, reliable and scalable. It has a plethora of advantages in comparison to its disadvantages. For the development of both iOS and dynamic web application it has been widely used.

7   References

[1] M. RATHORE, "Analysis and Impact of using Model/View/Controller(MVC) Pattern for Web Applications," *sodhganga,* p. 54, 2018.

[2] J. G. D. David A. Botwe, "A Comparative Study of Web Development Technologies Using Open Source and Proprietary Software," *International Journal of Computer Science and Mobile Computing ,* vol. 4, no. 2, pp. 154-165, 2015.

[3] L. Hao, "Application of MVC Platform in Bank E-CRM," *International Journal of u-and e-service ,* vol. 6, no. 2019, p. 42, 2013.

[4] A. M. a. L. Rauf, "MVC Architecture: A Detailed Insight to the Modern Web Applications Development," *Crimson Publishers,* vol. 1, no. 1, pp. 1-7, 2018.