CRIMSON PUBLISHERS
Wings to the Research

Peer Review Journal of Solar & Photoenergy Systems

Review Article

MVC Architecture: A Detailed Insight to the Modern Web Applications Development

Abdul Majeed[1]* and Ibtisam Rauf[2]

[1] *Korea Aerospace University, South Korea*

[2]*Department of Computer Science, Virtual university, Pakistan*

***Corresponding author:** Abdul Majeed, School of Information and Electronics Engineering, Korea Aerospace University, Deogyang-gu, Gyeonggi-do, Goyang-si, South Korea, Email: abdulmajid09398@kau.kr

Submission: September 08, 2018; **Published:** September 26, 2018

Abstract

Model, view and controller (MVC) is a well-known three-layer development architecture used for web applications developments. This paper presents the key insight related to the MVC layers, its uses, advantages and practices concerning MVC during web applications developments. We explain the all three layers in detail and their functionalities. The main objective of the study is to provide holistic view about each layer of the MVC and main functionalities and advantages of the MVC. Specifically, we explain the best practices concerning MVC based web applications developments. Furthermore, we explain the related technologies which are used in conjunction with MVC based web applications.

Keywords: Model view controller architecture; layers; Persistence; Functionalities

Introduction

Model view controller (MVC) [1] is an architectural pattern usually used in web-based applications. It provides three main layers; model, view, and controller. Many developers use MVC as a standard design pattern. It is a complete framework. MVC provide three types of classes:

A. Model- Model classes are used to implement the logic of data domains. These classes are used to retrieve, insert or update the data into the database associated with our application.

B. View- Views are used to prepare the interface of our application. By using that interface users interact with our application.

C. Controller- Controller classes are used to respond to the user's requests. Controller classes perform the users requested actions. These classes work with model classes and select the appropriate view that should be displayed to the user according to user requests.

MVC pattern architecture is basically a three-layered architecture. It separates the characteristics of application. Its first layer is related to the user input logic, second layer is related to the business logic and third layer is used to implement user interface logic. MVC provide very loose coupling among these three layers. MVC pattern are used to specify the location of each logic in application [2]. MVC patterns provide the facility of parallel development. It means that each layer of the application independent of each other i.e. three developer can work on the single application simultaneously [3]. One developer will be working on user input logic (controller logic), other developer will be working on the user interface logic (view) and third developer will be working on the business logic (model) at the same time. Second section of our research paper will be explained about when we can use the MVC pattern architecture. In this section we will also describe about some advantages of using MVC pattern architecture. In third section of the of our research paper we will describe the features of MVC framework.

In this section we will elaborate two different types of MVC frameworks which can be used to develop different web-based applications. In fourth section we will describe about the tools and technologies about which a developer should have deep knowledge so that he can create web-based application using MVC pattern architecture. In fifth section of our research paper we will talk about our research related work, in which we will give an example of e-commerce web application and explain that how we can develop that e-commerce web application using MVC pattern architecture. In six section of our research paper we will give result about our research related section. In seven section discussion we will talk about that how MVC pattern architecture influence the development of web-based applications. Also, in that section we will talk about the limitations of our study, as well as suggestions for developer. Also, we will provide the suggestions to the researchers. In next section we will draw a conclusion about our research paper. And in the last section we will provide references from where we gather the data from different resources.

When to use MVC pattern architecture

MVC pattern architecture gives us the idea of separation of concern, it helps us to implement the separation of concern among the model, view and controller classes within applications.

Separation of concern makes it easy for us to test our application as relation among different components of application is clearer and coherent. MVC help us to implement a test-driven development approach, in which we implement automated test cases before we write the code. These unit test cases help us predefine and verify requirements of new code before writing it.

If we are making an application with enough serious stimulating on the client side to refuse to go along with JavaScript alone. If we are developing an application which have a very high lifting on the server side and a little communication on the client side then we should not use the MVC pattern architecture, instead we should use simple setup such as web-based form model. The following are some characteristics that will help us whether to use MVC architecture in our application or not:

i. Our application needs asynchronous communication on the backend.

ii. Our application has a functionality which results in not to reload a full page for example commenting on a post while using Facebook or scrolling infinitely etc.

iii. Manipulation of data is mostly on the client side (browser) rather than server side.

iv. Same type of data is being delivered in different ways on a single page (navigation).

v. When our application has many insignificant connections that are used to modify data (button, switches).

Advantages of MVC architecture

a. MVC architecture helps us to control the complexity of application by dividing it into three components i.e. model, view and controller.

b. MVC does not use server-based forms, that's why it is ideal for those developers who want full control over their application behavior.

c. Test driven development approach is supported by MVC architecture.

d. MVC use front controller pattern. Front controller pattern handles the multiple incoming requests using single interface (controller). Front controller provides centralized control. We need to configure only one controller in web server instead of many.

e. Front controller provides support rich routing communications to design our web application.

Features of MVC framework

As we divide the logic of our application into three tasks (input logic, business logic, interface logic), testing of these components would become very easy. Testability is very fast and flexible, as we can use any unit testing framework compatible with MVC framework. It is an extensible and pluggable framework. We can design the components of our application in such a way that these are easily replaceable or can be modified easily. We can plug our own view engine, URL routing strategy, action method constraint serialization. Instead of depending on class to create objects we use a technique dependency injection (DI) which enable us to inject object into classes. Another technique inversion of control (IOC) is used to show dependency among objects, it specifies that which object need other which object. MVC provide URL mapping component that helps us to build using understandable and searchable URLs. Instead of using file name extensions MVC support URL naming patterns that are very useful for search engine optimization (SEO) and representational state transfer (REST) addressing. Some frameworks of MVC such as ASP.NET MVC framework provide us some built in features such as form authentication, session management, transactional business logic, web application security, object relational mapping, localization, membership and roles and URL authorization etc. The most popular frameworks available today are backbone.js, ember.js; angular.js and knockout.js.

I. Backbone. Js- Backbone.js framework is useful when our application need flexibility, we have uncertain requirements. Also, we want to accommodate change during application development.

II. Ember.js- When we want that our application should interact with JSON API than we should use ember.js framework in our application.

III. Angular.js- If we want more reliability and stability in our application, we want extensive testing for our application then we should use angular.js framework.

IV. Knockout.js- if we want to make a complex dynamic interface of application then knockout.js framework will be very useful for us.

Each framework has its own advantages and disadvantages. Developers can use any of the frameworks according to his requirements, which suit their web application.

Tools and technologies used with MVC

There are many tools and technologies which can be used to developed web application with the help of MVC architecture. Depending upon the interest of developers, they can use any of the tools and technologies to develop web application. Here are some tools and technologies which can be used to develop web application using MVC architecture:

Tools

Visual Studio: Visual studio is not just only a tool but a complete development environment which provide us facility to create different kinds of application. When we want to develop application using ASP.NET MVC framework then visual studio is very helpful for us.

1) MYSQL Server- relational database management server to maintain the database [4].

2) SQL Server- a database engine to maintain database just like MYSQL server [5].

3) MYSQL Workbench- a database design tool [6].

4) Net Beans- IDE (integrated development Environment) provide complete environment to develop different applications [7].

5) Glassfish Server: Java EE application server [8].

Technologies

A. HTML, CSS, JQUERY, AJAX for designing

B. Servlet and Java server pages (JSP) used with Net beans

C. EJB (Enterprise Java beans) technologies

D. JSTL (Java server pages standard tag libraries)

E. JPA (Java persistence API)

F. JDBC (Java database connectivity)

G. ASP.NET MVC used with Visual studio

There are many other tools and technologies which can be used with MVC architecture, but we have listed some of those tools and technologies which we are going to use in building our web application using MVC architecture [9].

Example of MVC architecture application

In our research study we have tried to make an Ecommerce web application using MVC architecture which will help the developers to understand the features of model view and controller pattern. E-commerce refers to the online selling and purchasing of goods over the internet. For example, amazon.com is today's most useful e-commerce website which provides different products of different categories. Customer can purchase those products online using internet. There are different types of e-commerce applications such consumer to consumer, Business to business, business to consumer.

In our application we have developed a business to consumer application which normally accommodates a B2C scenario consists of two components: first component store front which enables the customers to view different products, the data about the products is maintained in the database and is displayed to the customers on the pages dynamically. Second component is admin console which is protected area. It cannot be accessed by the customers instead it is available to the administration who manages the customer transactions such as customer orders, shipping and payment methods, CRUD (create read update delete) access to the database store.

The technology used in our application is Java. Java is a programming language which contains the fundamental concepts that can be grasped quickly. It sticks to an object-oriented technique which contained encapsulated objects that communicates with each other by passing information. Java include compile and run time checking to identify the errors. It also contains network security features. It is portable, application can easily be transferred from one platform to another platform with the need of a very little or no modification. Java interpreter is independent of run time environment, so we can run applications quickly. It also provides an automatic garbage collector to free unused memory.

Java is not only a programming language it is a complete platform which suggests software-based platform consist of two components. JVM (java virtual machine), engine used to execute the instruction translated by the java complier. It is considered as an instance of JRE (java runtime environment) and inserted in different products such as operating system, browsers and web servers etc. Second component of java platform is java application program interface (API) which provides prewritten codes which are prearranged into packages. These packages include classes for creating user interfaces such as font, button, menus etc. There are several editions of java platform such as Java standard edition, Java micro edition and java enterprise edition. Here we are working with java enterprise edition.

Java enterprise edition provides technologies used for developing and managing vigorous, consistent, scalable and secure server-side applications. These enterprise edition technologies are divided into two categories. These categories include web application technologies and enterprise application technologies. In our application as we must develop a web application so mostly we will be getting help from web application technologies. In our web application we are using Servlet, JSP, JSTL, EJB, and JPA enterprise technologies. Java EE technologies are mostly used technologies in the market as we have shown in a diagram taken from independent survey of European market perform in 2007 (Figure 1).

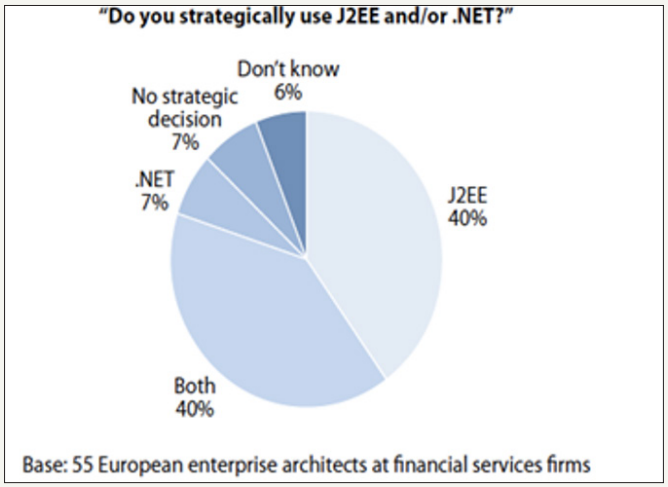

Figure 1: Survey findings about well-known IDEs.

The tools which we have used for the development of our application are net beans IDE, MySQL server and MySQL workbench. NetBeans is a development environment provides the facility of creating professional, enterprise desktop, mobile and web applications. MySQL server is a database engine used to maintain our database. MySQL workbench is used to visually design database such as ERD (entity relationship diagram).

Designing the application

Designing the application demonstrates that how our application can be applied to real world business requirements. Before designing the application, we try to gather the customer requirements. The customer requirements should be clear. As per requirements of our application, our application should meet the following customer requirements.

a. Online representation of data which should be stored in our database. The information about our products which will be displayed to the customer will be (name, image and description).

b. Add products temporary selected by our customer to a shopping cart.

c. Customer can update the quantity of the selected products.

d. Customer can remove the selected product from shopping cart.

e. Customer can view summary of shopping cart i.e. which product he selected and in how much quantity.

f. Customer can place an order and can make payment through a secure checkout process.

g. Our application should provide server-side administration so that administrators can view customer orders, add new products for the customers.

After gathering the requirements, we should design application according to customer requirements. We will try to make a use case which will describe that how our application will work and encapsulate its behavior. In our application we are making an e-commerce web site, so our use case will describe the following actions to be performed by different actors.

1) Customer can browse different products of the selected category.

2) Customer will add his desired products of different categories into shopping cart so that later he can purchase it.

3) Customer can view his shopping cart to see which items he has selected.

4) Customer can verify the shopping cart information so that he can proceed to checkout process.

5) Customer will provide his personal information as well he will see his order details on checkout page.

6) After viewing the checkout process page customer will confirm his order.

7) Administrator will view the orders of different customers after successfully login to our application.

8) Administrator can add new products into different categories.

According to above given use-case we must identify and design web pages which will be used in our application. We can create the mockups using different techniques. The technique which we use to design our application is paper prototyping in which we try to cooperate with customers by sketching ideas on the paper. In our application we have design five web pages i.e. welcome page (home page entry point to our application where our main categories of products will be shown), Category page (which will show us the list of different products also it will show us shopping cart option and proceed to checkout option), cart page (which will show us the list of all the product items which we have selected for purchasing), checkout page (which will be used to collect the user personal information as well as it will provide the order summary) and confirmation page (which will show the information that his order has been successfully placed) (Figure 2).

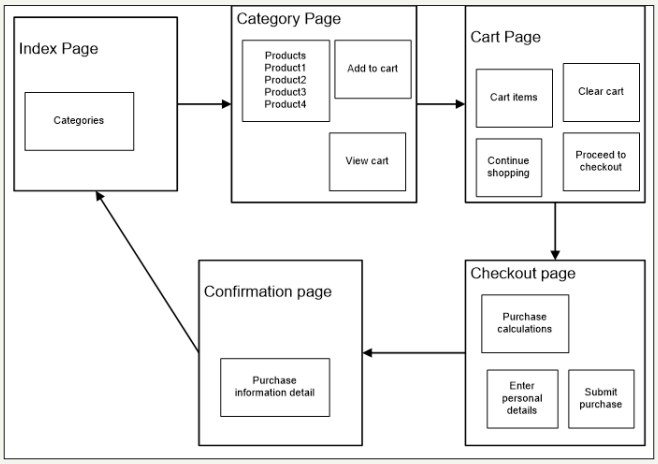

Figure 2: Purchasing an item using a web-application.

Determining the architecture

Before we start the development of application we should decide that which type of architecture we should use to develop our application. We should examine the functional components of our application and determine the way through which they will communicate with each other. There are many design patterns are available which can be used to develop an application. Here we are going to use MVC architecture pattern.

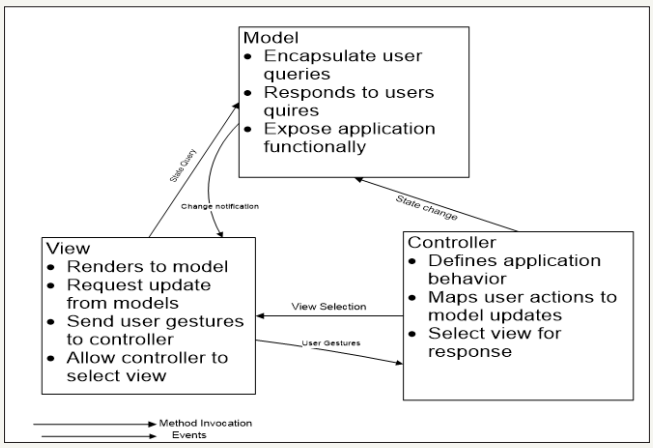

Figure 3: Functionality of each layer in MVC architecture.

MVC pattern divides our application into three major components i.e. Model, View and Controller. Model is used to

communicate with our database. View is used to get data from model and present it to the user. And finally, the controller is used to define the application behavior. It interacts with the inputs of the users. Normally in web-based application user's inputs are HTTP GET and HTTP POST. Detail about the MVC architecture pattern is given earlier in this paper in section I, section II, section III and Section IV (Figure 3).

Here we will be working with JSP technologies, our business logic into JSP pages will be coded using script lets. The script lets are snippets in <% %> tags. JSP pages will be compiled into Servlet. JSP was firstly introduced in 1999 and it included a description of two model architectures. Model one involves implementing business logic directly with JSP pages and second model applies the MVC pattern architecture which we are using in our application (Figure 4).

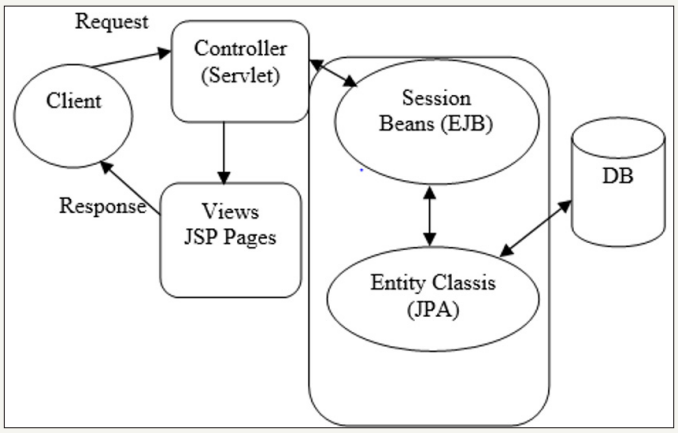

Figure 4: Interaction and overview of the MVC layers.

Designing the data model

In this step we should create a conceptual model of our storage system by defining the entities if our system and relationship among them. Our data model will contain both logical and physical design parameters required to generate a script using data definition language (DDL), which will be helpful in creating a database. As we have stated earlier we will use MySQL workbench to create data model, reverse-engineer SQL into visual presentations, forward-engineer data models into database schema and synchronize models with a running MySQL database server.

In our application we have identified four entities customer, category, product and order to construct an ERD (entity relationship diagram). We created a database schema from the ERD; we used IDE's Eclipse Link support to generate JPA classes from the database. We have used the technique of bottom up approach which begins with an existing database schema. It provides the easiest way to use forward engineering tool to take out the metadata from schema and generate the JPA entity classes.

Development of application

In development of application first we begin with the development of our application front end of application by creating JSP pages and placing them in proper locations, creating the header and footers which will be displayed in every view and setting up Servlet which will handle the entire incoming requests from the users. We also created a deployment descriptor (web.xml) which contained all the configuration information of the application.

After creating front-end of our application, we enabled our application that it can communicate with the database, for which we add sample data into the database and set up a data source and a connection pool on Glassfish server to proceed by JSP pages which will test the data source. We also use the JSTL library and SQL tag library to retrieve and show the images of different categories and products on the index and category pages.

After that we created java persistent API class for each table in the database, complete with named query annotations, fields (columns) and relationships (foreign keys). We also created the EJB session façade for each entity class to define their access methods. In EE developments following two technologies are used for development of Model component:

a) Enterprise Java Beans: it is a server-side component architecture which is used for fast and easy development of distributed, transactional, convenient application. It creates pool of component instance which is shared by users. It also provides thread safety, as well as transaction management. EJB provide the service of java authentication and authorization (JAAS) which provide security to our application.

b) Java Persistent: Java persistent automatically stores the data into the database contained by the java objects. It is an object relational technology which helps the application to manage the data between java objects and database in such a way that it is apparent to the developer. Transparency of data to the developer provides facilities of data caching and performance optimization.

Enterprise session beans are invoked by the users to perform a business operation. Session bean is available for some duration of time. Session bean is crashed when the EJB container is crashed and after crashing of session bean, user must re-establish a new session object to continue calculation. There are three types of session beans provided by the EJB i.e. stateful which is maintained across multiple method calls, stateless which is useful for those operations that can take place in single method call, and singleton which is instantiated only once per application and exists until the application remains active.

In our application we are using stateless session bean. We have used HTTP protocol which is stateless protocol; over which communication take place on the internet. We must create a session in which user related information is maintained for some duration before the information is properly stored into the database. We used HTTP Session for maintaining user's sessions (cookies). We have also secured our application by properly setting up the login process for administration console through which only authorized administrators will access console services and unauthorized users will be blocked. We have configured a secure data transport for customer checkout process as well as administration console.

For this purpose, we have used Glassfish server administration console to make users groups corresponds to our application. We also tested our application using JMeter kit plug-in. It enables us to check the performance of our application whether it is working properly or not.

Results

In our research paper we have discussed about MVC pattern architecture. We have discussed that how it can be useful for developing web-based application. We discussed about different benefits of the MVC architecture as well as we also discuss about some frameworks of MVC architecture. We provide a complete example of MVC architecture in which we develop an e-commerce application using different tools and technologies.

Discussion

MVC architecture has greater influence in the world of web-based application. It is very helpful for the developers. Developers can easily make their web applications using MVC architecture. It reduces the complexity of the application and divides the application into three major components i.e. model, view and controller. It provides full control over the web application to the developers. Using MVC architecture testing of different components of application becomes very easy. MVC architecture provides higher level of abstraction to their web applications.

Limitation of our study

In our research paper we have discussed about MVC architecture explicitly. We talked about its three major components model, view and controller and explained about communication of these components with each other. We do not provide enough detail about these components separately. We just give an example of web application and discussed about some of tools and technologies which are used to make these components more effective for web applications. There are also many other frameworks of MVC which can be used for the development of web application. We have only discussed front-end frameworks of MVC architecture in our research paper.

Suggestions for developers

Developers who are interested in working on web-based applications, MVC architecture is good choice for them because it provides many powerful features for web application as compared to the simple web form-based web applications. Development of web application becomes very easy and effective while using MVC architecture because it reduces the complexity of application by dividing the logic of web applications into three layers. Developers can work on these three layers independently. For those developers who want to make large scale and complex applications, MVC provide guidance and complete structure to those developers to make these web applications. Developers can write well structure, maintainable, java script code.

Developers can use MVC architecture with multiple high-level languages as this pattern is language independent. Developer can use java, asp.net etc. to develop web application while using MVC architecture. For those developers who are interested in developing their skills in java EE technologies and wanting to learn different web-based technologies MVC is a great option for them. We are providing a link to those developers who are interested in working on J2EE web-based application using MVC architecture [10] through which they can develop an e-commerce application by using different technologies such as JSP technologies by following different steps.

Suggestions for researchers

MVC architecture is independent of programming languages developer can use any programming language of their own interest to develop a web application. In our research paper we have provide a guideline to developers about the development of web applications using JSP technologies such as JSP, Servlet, EJB, JPA etc. Other researcher can work upon other MVC frameworks and provide the guidelines to the developers. Such as ASP.net MVC framework is most popular framework now a day. It provides many powerful features just like Ajax, cshtml, jQuery etc. to develop web applications. Researchers can work on ASP.net MVC frame work to explore its different features and provide the helps to developers.

Conclusion

MVC pattern architecture and integrated technologies just like JSP, Servlet and EJB on the platform of J2EE have simplified the development process of web applications. We can make scalable, transparent and portable web application with the help of MVC architecture. MVC architecture provide the concept of parallel development as it divides the logic of application into three layers, so each different developer can work simultaneously on these three layers of the same web application.

First layer of MVC architecture is model which is used to interact with the database. It is basically logic layer of our application which is used to insert, retrieve and update data in the database. Second layer is view which is related to the user interface, through which users interact with our application, how our application will be presented to the users, basically it is used for front end development. Third later of MVC architecture is controller which is used to get inputs from the users. It controls the updates of model object state and data displaying to the users according to user's inputs. User can request to our application and can get response through controller layer.

References

1. Gupta P, Govil MC (2010) MVC Design pattern for the multi framework distributed applications using XML, spring and struts framework. International Journal on Computer Science and Engineering 2(4): 1047-1051.
2. Shu-qiang H, Huan-ming Z (2008) Research on improved MVC design pattern based on struts and XSL. 2008 International Symposium on Information Science and Engineering, pp. 451-455.
3. Selfa DM, Carrillo M, Boone MDR (2006) A database and web application based on MVC architecture. 16th International Conference on Electronics, Communications and Computers (CONIELECOMP'06), p. 48.
4. Ahmed M, Uddin MM, Azad MS, Haseeb S (2010) MySQL performance analysis on a limited resource server. SpringSim 10 Proceedings of the 2010 Spring Simulation Multiconference, p. 1.

5. Anley C (2002) Advanced SQL injection in SQL server applications. Next Generation Security Software Ltd, UK.

6. Kroenke DM, Auer DJ. Database Concepts 7th Edition Online Appendix E SQL Views, SQL/PSM and Importing Data.

7. Benson C, Prove MM, Mzourek J (2004) Professional usability in open source projects. CHI '04 Extended Abstracts on Human Factors in Computing Systems, pp. 1083-1084.

8. Moulton B, Chaczko Z, Karatovic M (2009) International journal of digital content technology & amp; its Applications GYEONGJU. The International Association for Information, Culture, Human and Industry Technology.

9. Doyle B, Lopes CV (2008) Survey of technologies for web application development. University of California, USA.

10. Stratmann E, Ousterhout J, Madan S (2011) Integrating long polling with an MVC framework. Stanford University, USA, p. 10.