ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/291098214

# MVC Architecture Driven Design and Implementation of Java Framework for Developing Desktop Application

**Article** *in* International Journal of Information Technology · September 2014

DOI: 10.14257/ijhit.2014.7.5.29

CITATIONS
25

READS
7,151

2 authors:

Iqbal H. Sarker
Edith Cowan University

**234** PUBLICATIONS **9,765** CITATIONS

SEE PROFILE

Khalid Ibn Zinnah Apu
Chittagong University of Engineering & Technology

**8** PUBLICATIONS **74** CITATIONS

SEE PROFILE

All content following this page was uploaded by Iqbal H. Sarker on 21 September 2016.

The user has requested enhancement of the downloaded file.

International Journal of Hybrid Information Technology
Vol.7, No.5 (2014), pp.317-322
http://dx.doi.org/10.14257/ijhit.2014.7.5.29

# MVC Architecture Driven Design and Implementation of Java Framework for Developing Desktop Application

Iqbal H. Sarker and K. Apu

*Department of Computer Science & Engineering, Chittagong University of Engineering & Technology (CUET), Chittagong-4349, Bangladesh*
*iqbal@cuet.ac.bd, khalidex@yahoo.com*

## Abstract

*To avoid the problems of high complexity and low flexibility in Java based application Model–view–controller (MVC) framework is introduced to divide the whole application into model, view and controller, which can improve the software's reusability and flexibility. Java is one of the leading software development languages. However, the development model of existing Java organizes without a structure, which mixes the code of data access, the processing of business logic and graphical user interface (GUI) layer together. As a result, it creates many problems for software developers, meanwhile; it could not meet the rapid development of application software any more. In this paper, we design and implement a Java MVC framework for developing desktop based application which can separate the data, view and control of the software.*

**Keywords:** *Java, Framework Design, MVC Architecture, Software Engineering*

## 1. Introduction

Nowadays when information technologies represent one of the fastest developing business areas, many software companies can respond to client requirements with the same quality. In general, application software mainly contains three large modules, such as the interface, business logic and data. In traditional applications, the connections among these were very closely, so in the early days of software development, developers had to write all the code for its production of any application. Now, the emergence of the MVC framework, solve the above problems. MVC (Model-View-Controller) is the first letter of model, view, controller, separates the application input, business logic and output according to Model, View, and Controller [7]. MVC has demonstrated its benefits for interactive applications allowing multiple representations of the similar information, promoting the code reutilization, and helping developers to focus on a particular application feature.

The MVC framework has been widely become the standard in modern software development. Appropriate code reuse applied in the model layer, view layer and control layer apply may not only bring around the separation of underlying business logic, process control, and display, but also can make the software scalability and maintainability improved significantly. Code reuse, is one of the most common form of object-oriented reusability. When developing a software module, the modules internal should reuse as much as possible. Code reuse not only can greatly accelerate the speed of development, reduce development investment, but also can improve the quality of the system, to facilitate concentrate solution problems in the test stage. Code reuse is the vast majority of developers expect, also one of the goals of object-oriented. This paper describes fast development of a simple application generator for a specific application. Its main task is to justify development of a custom

ISSN: 1738-9968 IJHIT
Copyright ⓒ 2014 SERSC

generator from the very beginning in case of short time limits and specific project requirements.

The rest of the paper is organized as follows: Section 2 mentions the motivation of this framework. Section 3 provides a brief description of previous works related to MVC and framework. Section 4 introduces the overview of MVC components. Section 5 describes the design and implementation of framework. Section 6 provides experimental results and finally, Section 7 concludes this paper.

## 2. Motivation

The followings are the important benefits to develop a framework based on MVC.

- To group objects into separate Model, View, and Controller subsystems and decouple those subsystems within an application and enable multiple Views of the same Model and support easy addition, removal, and change of Views.
- To promote reuse of both Models and Views and simplify simultaneous development of Models and Views by different programmers.
- To organize application source code and develop application rapidly.

## 3. Related Work

A significant number of MVC techniques and software development frameworks have been implemented in recent years in software engineering area in order to reuse code and efficient application. Wei Cui *et al.*, [2] proposed a development framework of PHP based MVC design model, which might be an effective separation of data access, logic processing and user interface and thus it could promote the efficiency and quality of PHP development. But our concern is desktop software development based on Java. Abhinandan Jain *et al.*, [5] proposed a framework for rapid software development name YAM. This provides solutions for thorny development challenges associated with software reuse, managing multiple software configurations, developing of software product-lines, and multiple platform development and build management. The YAM framework integrates concepts spanning software organization, build management, release management, and software reuse to provide a nimble development processes for managing complex software development. But they don’t consider the MVC technique which can separate the presentation layer and business logic of the software. In [6], the authors present a practical management framework for commercial software development with OSS. In the framework, authors define a development process including activities, instructions, and artifacts to use OSS. But they don’t consider MVC technique. In this regard, we design and implement a MVC based framework based on Java for developing desktop application which can separate the data, view and control of the software and enhanced the efficiency, reliability, maintainability and scalability of application software.

## 4. MVC Components

Model–view–controller (MVC) is a software architecture concept considered as an architectural pattern in software engineering. It has three components such as Model, View and Controller. Figure 1 shows the relationship among the components of MVC architecture.

Copyright ⓒ 2014 SERSC

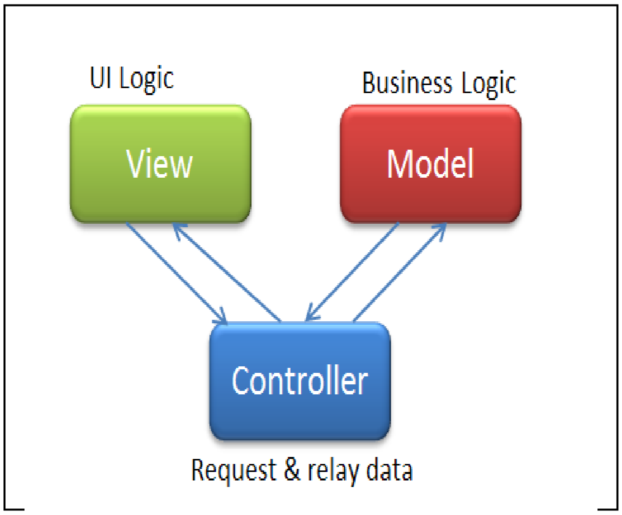


**Figure 1. MVC Components**

In an MVC framework, the view and controller belongs to the user interface. In the beginning, a user sends a request to a controller through a graphical user interface (GUI). Then the controller accesses model gives the data according to the user's request. After that, the model returns the data to the controller, and controller presents the data via the specified view. We also use database to store data and provide original data source in our system. We also summarize the following components.

- **Model-** Model represents an object or JAVA carrying data. It can also have logic to update controller if its data changes.
- **View-**View represents the visualization of the data that model contains.
- **Controller-**Controller acts on both Model and view. It controls the data flow into model object and updates the view whenever data changes. It keeps View and Model separate.

## 5. Design and Implementation

In this section, we briefly explain the design architecture and operation of the framework. This is a three-tier application framework that is organized into three major parts. The three parts are-

- Users
- MVC components and
- Database.

The first tier consists of only users. User can send request and get response by the "View" components of middle tier. Secondly, middle tier consists of three components named Model, View and Controller. These components can communicate and process data each other. Finally, third tier contains only database where data is stored permanently. Figure 2 illustrates how data flows throughout the system using MVC.

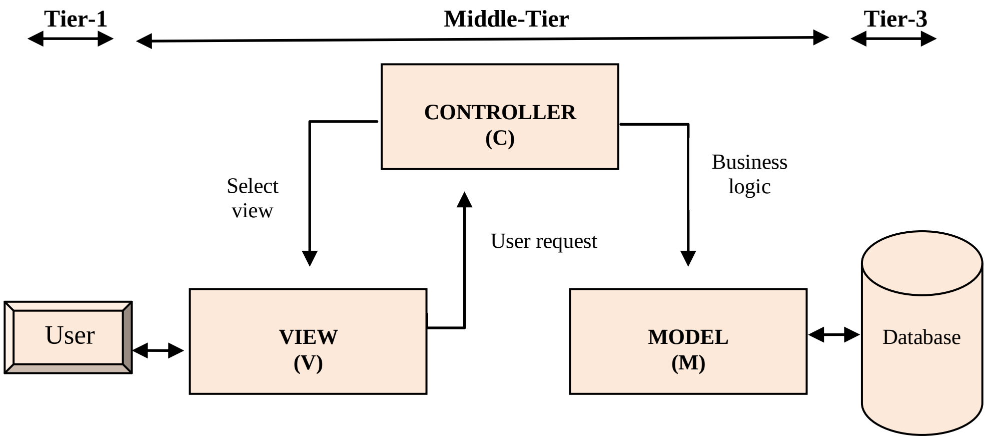


**Figure 2. Framework Architecture**

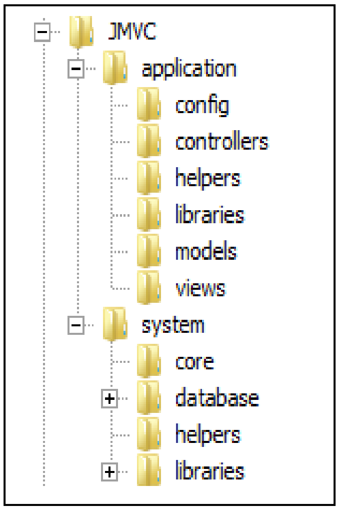


**Figure 3. General Folder Organization**

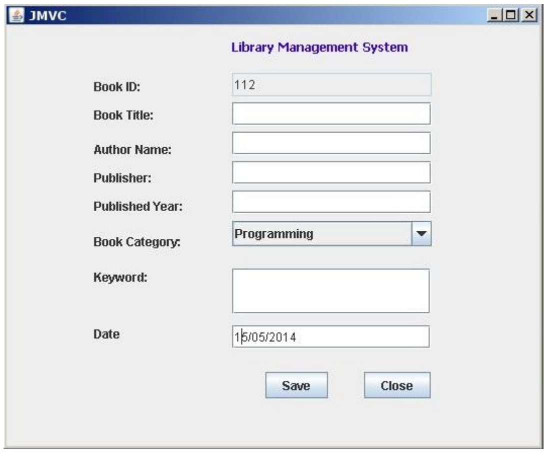
The operations are described in the following steps.

- User input is accepted by the component "View" through graphical user interface (GUI). In a stand-alone GUI client, user interactions could be button clicks or menu selections.
- The "Controller" processes the user requests. Based on the user request, the Controller calls methods in the View and Model to accomplish the requested action. A pure GUI controller accepts input from the user and instructs the model and viewport to perform action based on that input. The controller adapts the request to the model. The model represents, or encapsulates, an application's business logic or state. It captures not only the state of a process or system, but also how the system works.
- The view is responsible for the output of the model. A pure GUI view attaches to a model and renders its contents to the display surface.

Copyright ⓒ 2014 SERSC

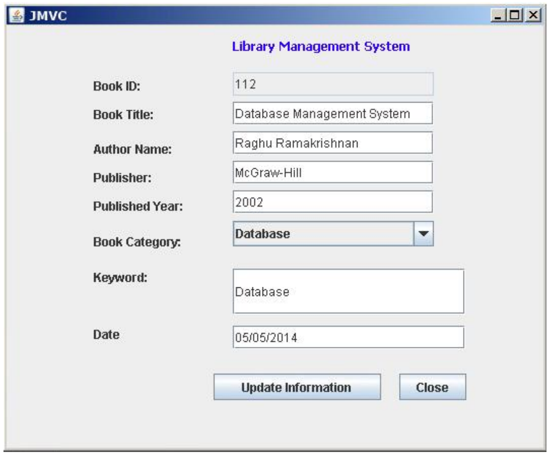
## 6. Experimental Results

In this section, we will discuss about the results of the framework. We developed a simple application named "Library Management System" to test this framework. Here, we displayed some snapshots of our application. We performed different types of operation (*such as data insert into database, data retrieve according to different conditions, data edit or update, different types of searching etc*). We got the appropriate results at every step.



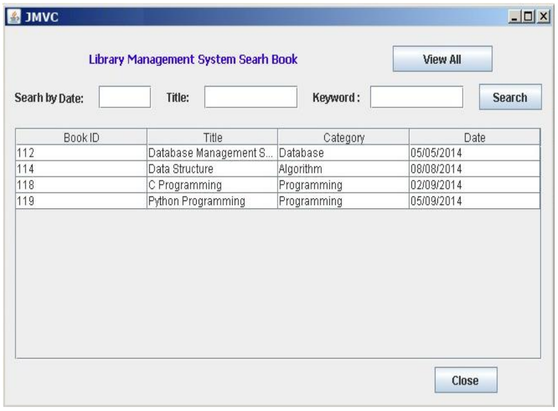
Book ID: 112
Book Title:
Author Name:
Publisher:
Published Year:
Book Category: Programming
Keyword:
Date 15/05/2014

Figure 4. Data Insertion GUI



Book ID: 112
Book Title: Database Management System
Author Name: Raghu Ramakrishnan
Publisher: McGraw-Hill
Published Year: 2002
Book Category: Database
Keyword: Database
Date 05/05/2014

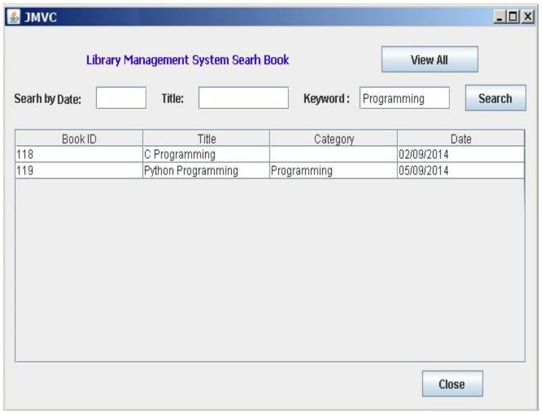
Figure 5. Data Update



Searh by Date: Title: Keyword :

| Book ID | Title | Category | Date |
|---|---|---|---|
| 112 | Database Management S... | Database | 05/05/2014 |
| 114 | Data Structure | Algorithm | 08/08/2014 |
| 118 | C Programming | Programming | 02/09/2014 |
| 119 | Python Programming | Programming | 05/09/2014 |

Figure 6. Data Retrieve Results



Searh by Date: Title: Keyword : Programming

| Book ID | Title | Category | Date |
|---|---|---|---|
| 118 | C Programming |  | 02/09/2014 |
| 119 | Python Programming | Programming | 05/09/2014 |

Figure 7. Searching Results

## 7. Discussion and Conclusion

In this paper, we have presented a java framework to develop desktop application software rapidly based on MVC. Software developer can use this framework to build software rapidly. Using this framework, not only achieve the completely separation of view, controller and the model of the MVC Mode, but also achieve a separation of business logic layer and presentation layer. From our testing application, we believe software can be effectively developed by using this MVC framework in a right way and this MVC framework can be one of active participants to software communities. The actual operation has proved that this framework is stable, efficient and able to develop high quality applications.

Copyright ⓒ 2014 SERSC

## References

[1] I. H. Sarker, "JMVC: A Java Framework for Rapidly Developing Desktop Application Software Based on MVC", Smart Computing Review, vol. 4, no. 4, **(2014)**.

[2] W. Cui, Lin Huang, LiJing Liang and Jing Li, "The Research of PHP Development Framework Based on MVC Pattern", 4th International Conference on Computer Sciences and Convergence Information Technology, IEEE, **(2009)**.

[3] "MVC website", http://www.en.wikipedia.org/wiki/Model-view-controller/.

[4] "Java website", http://www.en.wikipedia.org/wiki/Java_(programming_language)/.

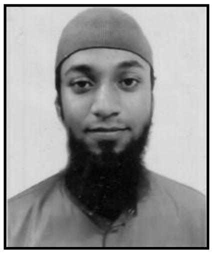
[5] A. Jain and J. Biesiadecki, "YAM- A Framework for Rapid Software Development", 2nd IEEE International Conference on Space Mission Challenges for Information Technology (SMC-IT'06), IEEE, **(2006)**.

[6] S. Chang, J. Lee and W. Yi, "A Practical Management Framework for Commercial Software Development with Open Sources", International Conference on E-Business Engineering, IEEE, **(2010)**.

[7] R. Yongchang, X. Tao, X. Zhaofeng and Z. Jianchun, "Application Research for Integrated SSH Combination Framework to Achieve MVC Mode," International Conference on Computational and Information Sciences, IEEE, **(2011)**.

## Authors

**Md. Iqbal Hasan Sarker** received his Bachelors of Science in Computer Science & Engineering from Chittagong University of Engineering & Technology, Bangladesh, in 2009. He also completed his Masters of Science in Computer Science & Engineering from the same university in 2014. Currently he is working as an assistant professor at the same department and university. His research interests include Software Engineering, Mobile Applications, Web Applications and Data Mining.



**Khalid IbnZinnah Apu** pursuing his B.Sc. degree in Computer Science & Engineering from Chittagong University of Engineering & Technology (CUET), Bangladesh. His research interest includes Cyber Security, Mobile Computing, Software Engineering, Cryptography and data mining.

Copyright ⓒ 2014 SERSC

View publication stats