The design of a letter archiving application using the Model View Controller (MVC) concept

B D D Arianti*, H Kuswanto, H A Januari and J Jamaluddin
Department of Informatics Education, Faculty of Mathematics and Science, Universitas Hamzanwadi, Lombok, Indonesia

*ariantibaiq@hamzanwadi.ac.id

Abstract. Letters are an essential means of communication in an institution or organization, and they are closely related to the times' progress that cannot be abandoned, including today's digital era. Based on observations, letter archiving at Hamzanwadi University is still conventional and not digitally recorded. So this research aims to design a letter archiving application by applying the Model View Controller (MVC) concept with a development model using ADDIE (Analysis, Design, Development, Implementation, and Evaluation). This Design consists of system architecture, flowchart, data flow diagram, and ERD.

1. Introduction

Letters is a written communication to convey information that certain parties need to know, either through individuals or agencies. Letters are not only for communication but also as reliable evidence because they have original signatures and stamps, so it is important to archive the letter [1,2]. Archives are essential sources of information supporting administrative process activities [3-6]. Archives are not only of the agency or organization's activities, but archives can also be created and received from other agencies or organizations and stored as reliable evidence [7-8].

Universitas Hamzanwadi, as a university, cannot be separated from correspondence activities. Based on the results of observations, Universitas Hamzanwadi has been archiving letters conventionally. The registration of letters still uses archival books and prone to being dirty, damaged, and lost. Not all letters are stored in filing cabinets, and it is still not following procedures in the filing process. It is less effective and efficient. To overcome this, Universitas Hamzanwadi requires a software system to archive a letter. Therefore, this study tries to develop a letter archiving application using the Model View Controller (MVC) concept [9-12].

The MVC concept is applied to make it easier to maintain and develop applications without worrying about other modules affected by the modified module [11]. The MVC concept separates the development of the main components that build an application, such as data manipulation (models), views, and processing algorithms (controllers), into different parts. Although the MVC concept extensively applies to the PHP framework, everyone who wants to use and switch from one framework to another must first learn about the coding style, the functions provided, and the different folder structure of each framework [12].
When using a framework, it must follow the coding style of the framework. Otherwise, the program logic will not work correctly [13,14]. Where native PHP uses a coding style that many people commonly use, it is built on the programmer's thoughts and can be object-based or structural programming according to the programmer's capabilities [14].

2. Methods
This study uses research and development methods, with a development model using the ADDIE development model, which consists of Analysis, Design, Development, Implementation, and Evaluation [15].

2.1. Analysis
The analysis stage is the first stage in the ADDIE development model. It will analyze user needs and system requirements [15].

2.2. Design
The design stage or planning stage is the stage for designing applications that will be developed following the analysis stage results.
2.3. Development
At this stage, application development begins following the results of planning at the design stage. Development carried out using the Visual Studio Code application as a code editor, XAMPP as a local server, Google Chrome, and Mozilla Firefox as a web browser to run application coding results.

2.4. Implementation
At this stage, a product trial that involves the user directly after product development at the development stage is complete. Product trials are carried out based on ISO 25010 standards. This stage determines users' feedback on the products produced in comments, suggestions, and others orally and in writing.

2.5. Evaluation
After getting feedback from users when testing the product in the previous stage, then at this stage, there will be improvements to the product based on the feedback provided by the user so that the product is ready for use again.

3. Results and discussion
This article only discusses two stages, namely the analysis and design stages. The results obtained are as follows:

3.1. Analysis
The system requirements analysis results are that the system developed can manage incoming and outgoing letter data. The system can provide reports on an incoming and outgoing letter. The system can forward the incoming letter to its destination appropriately and complete it quickly without space and time constraints. Meanwhile, the user needs analysis results: the system can running by two users, namely Admin and Pimpinan. Admin is in charge of managing all system data. Meanwhile, the leadership can position the incoming letter according to the purpose of the letter.

3.2. Design
3.2.1. System architecture

![System architecture diagram](image)

**Figure 3.** System architecture.

In figure 3 shows how users access data from a letter archiving application. Both users can manage letter data to and from the database via the internet network.
3.2.2. Flowchart

Figure 4. Administrator flowchart.

Figure 4 shows how the flowchart for administrators. This flowchart describes how the administrator will run the processes. From login, managing letter data, sending letter data to leaders, changing passwords, and logout.

Figure 5. Pimpinan flowchart.

Figure 5 shows an overview of the flowchart for Pemimpin. The processes carried out by the Pemimpin include checking the letter disposition link, checking the disposition status, sending the disposition, and inputting the letter disposition data.
3.2.3. Data Flow Diagram (DFD)

Figure 6 shows a context diagram of the application. The context diagram showed that this software is running by two users, namely Administrator and Pemimpin. Administrators manage login data, incoming letter, outgoing letter, letter disposition, institution data, management data, report data, and profile data. Meanwhile, the Pemimpin users receive and forward the letter disposition data.

Figure 7. DFD level 1.
Figure 7 is a Level 1 of the Data Flow Diagram. Level 1 DFD illustrates how data flows from the tables in the database to the system's processes. Eight processes use a database, including the login process, incoming letter, outgoing letter, disposition, agency data processing, management data, reports, and profiles.

3.2.4. Entity Relationship Diagram (ERD)

Figure 8. Entity relationship diagram.

Figure 8 is the ERD of this application system. ERD describes all entities' relationships in this application, including incoming letter, outgoing letter, users, agencies, links, dispositions, and reports.

4. Conclusion

The use of the Model-View-Controller (MVC) concept in making letter archiving applications has been easier to conceptualize so far, even the algorithm for this application is complex. It can be seen at the system design stage, starting from making flowcharts, DFD, to ERD, it is easier to apply the algorithm in this application system.

References

[1] Farell G, Saputra H, Novid I 2018 JTIP: Jurnal Teknologi Informasi dan Pendidikan 11(2) 55-56
[2] Rustamin Z, Dewi A P 2016 Simtek: Jurnal Sistem Informasi dan Teknik Komputer 1(2) 165-172
[3] Rosalin S 2017 Manajemen Arsip Dinamis (Malang: Universitas Brawijaya Press)
[4] Riasmiati A 2016 Khazanah: Jurnal Pengembangan Kearsipan 9(2) 65-70
[5] Agustina F 2016 Khazanah: Jurnal Pengembangan Kearsipan 9(2) 57-61
[6] Hendrawan M R and Ulum M C 2017 Pengantar Kearsipan (Malang: Universitas Brawijaya
Manulangga G C, Bataona D S, Laumal F E 2018 *Letters in Information Technology Education (LITE)* 1(1) 9-13

Simangunsong A 2018 *Jurnal Mantik Penusa* 2(1) 11-19

ElSayed N A, Smith R T, Marriott K, Thomas B H 2018 *Journal of Visual Languages & Computing* 44 1-12

Aniche M, Bavota G, Treude C, Gerosa M A, Deursen A V 2018 *Empirical Software Engineering* 23(4) 2121-2157

Simajuntak P 2016 *Journal Information System Development (ISD)* 1(2) 56-66

Pop D P, Altar A 2014 *Procedia Engineering* 69 172-179

Putra E K, Witanti W, Saputri I V, Pinasty S Y 2020 *IKRA-ITH Informatika : Jurnal Komputer dan Informatika* 4(2) 55-64

Padmanaba A, Kumalasari E, Andayati D 2020 *Jurnal SCRIPT* 8(1) 1-6

Arianti B D D, Wirasasmita R H, Mustaniuddin 2020 *Journal of Physics: Conference Series* 1539 012045