Development of a Web-Based School Payment Administration Information System Using the Laravel Framework

Samsul Arifin*, Asroni, and Aprilia Kurnianti

Universitas Muhammadiyah Yogyakarta, Jln.Brawijaya, Tamantirto, Kasihan, Bantul, Yogyakarta 55183, Indonesia
Corresponding author: samsul.arifin.2014@ft.umy.ac.id

Abstract

A school's payment administration management system is crucial for its well-being because all activities require payment administration to run successfully. The administration department is responsible for managing the payment, including recording payment of Education Development Contribution (SPP) and other contribution fees and generating reports. Unfortunately, these tasks are carried out manually and have not been digitized, resulting in the possibility of erroneous data and a lengthy recording procedure. Accordingly, it necessitates developing a web-based system for school payment administration management to decrease the risk of payment and recording errors. Laravel was utilized as the application's framework, with PHP as the language. Database management system employed MySQL. Observation, interviews, and system analysis were performed to create a web application. This application could resolve issues, making payment data recapitulation more effective and efficient while reducing the risk of data inaccuracy.

Keywords: School payment, PHP, Website, Laravel framework.

1. Introduction

Current technological development significantly impacts the advancement of information systems, which will evolve alongside technological development. Improving science and technology at Madrasah Aliyah (MA) Grogolpenatus is one of the efforts that should be performed to attain this development. Schools are institutions established to deliver educational services under the direction of teachers. In this regard, MA Grogolpenatus is a high school-level Islamic education institution combining the Islamic curriculum with the standard one. Its activities are supported by payment administration. The challenge encountered by the school is the payment administration procedure, which is still performed manually using the school-provided payment form, thereby increasing the risk of data errors and the time required to generate payment reports.

Given the current issues, it is required to design a payment administration application that will limit the possibility of data errors. This web application was created using Laravel Hypertext Preprocessor (PHP) programming. PHP is an embeddable programming language that can be placed into Hypertext Markup Language (HTML) documents. The Laravel framework is an open-source web application employing the Model View Controller (MVC) design pattern. This web application was developed with Sublime Text.

Research conducted [1] entitled "Analysis and Design of Student Financial Administration Information System at Superior Vocational High School Jambi Sakti. In

this study aims to analyze and design Student Financial Administration Information System at SMK Unggul Sakti Jambi. With the design of this system produces services in the form of information such as information on new student data, information on payment of tuition fees and tutoring fees computers, recapitulation of SPP receipts reports and computer lessons every day and every month, as well as recapitulation of arrears reports of students who have not paid their tuition fees and computer lessons every month. According to research conducted [2] with title "Information System for Payment of Educational Development Contributions (SPP) At SMAN 1 Gaung Web-Based". In this study aims to build a system that can facilitate officers in the process management of financial data or school financial recapitulation, which will be later with this system will be able to provide convenience in service school administration as well as in presenting data and recapitulation information payment.

Research conducted [3] with the title "School Payment Administration Information System at Darun Nurjati Middle School Bekasi city". In this study aims to build a system information regarding the administration of Darun Nurjati Middle School students in Bekasi City dealing with a problem regarding SPP administration payment data and student education. By designing this system can speed up the process search for payment transaction data because it has been computerized. According to research conducted by [4] which entitled "Web-Based Application for School Financial Management (Case Study: Ma Exactly Tarogong Garut)". In this study aims to produce financial reports with accurate results, so that it can be ascertained uniformity of recording format by each user and minimizing human error in the financial management of MA Persis Taragong Garut. Based on the research above, generate conclusions on the use technology is needed in making administrative information systems school finance, namely by designing and making this application, you can Facilitate the processing of payments and financial administration preparation to be more effective and efficient.

1.1. Theoretical Basis

1.1.1. Definition of Administration

Researchers performed administration in a narrow sense, namely administrative activities whose essence is the routine activity of recording, documenting activities, organizing correspondence with all aspects and prepare reports. Educational administration in the sense broadly is the entire process of mobilizing and integrating everything both personnel, spiritual and material related to achievement educational purposes [5].

1.1.2. Website

Researchers performed Understanding Website is "a collection of web pages that have been published on the internet network and has a domain/URL (Uniform Resource Locator). can be accessed by all internet users by entering the address. This is made possible by the existence of World Wide Web (WWW) page technology the website has a Hyper Text Markup Language (HTML) document format, which can be accessed using HTTPS, HTTPS itself is a protocol that used to convey various information from the website server to addressed to users or users using a web browser [6].

1.1.3. Information Systems

Researchers performed the system is a collection of several elements that interact with each other in order to achieve a goal certain. Information is data that is processed and shaped to be more useful and means more to those who receive it. The information system itself is a system that within an organization that meets processing needs daily transactions, support operations, are managerial and strategic activities of an organization and provide certain external parties with reports than needed [7].

1.1.4. Unified Modeling language (UML)

Researchers performed the Unified Modeling Language (UML) is a standard used to perform a visualization, design and documenting software systems in the industry. UML delivers a standard for designing models of systems. Using UML can create models for many types of software applications, where those applications can run on any hardware, operating system and network, and written to in any programming language. Because UML uses class and operation in its basic concept, it will be more suitable for use writing software in an object-oriented language such as C++, Java, or VB NET [8].

1.1.5. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) is a model for describes the relationship between data in the database based on basic objects data that is often encountered in system development activities by system analysts. Researchers performed ERD is a data model uses some notation to describe data in terms of entities and the relationships described by the data [9].

1.1.6. Data Flow Diagram

Researchers performed Data Flow Diagram (DFD) or in language Indonesia is also known as Data Flow Diagram (DAD). DFD is a model the logic or process created to describe where the data comes from, and where is the destination of the data coming out of the system or software, where is the data stored and what process generates the data. Objects- data objects in the depiction of DFD usually use arrows labeled, circle- circles which are also known as bubbles [10].

2. Method

The Software Development Life Cycle (SDLC) model was utilized for system design and manufacture, particularly for system development and maintenance. The model selected was prototyping.
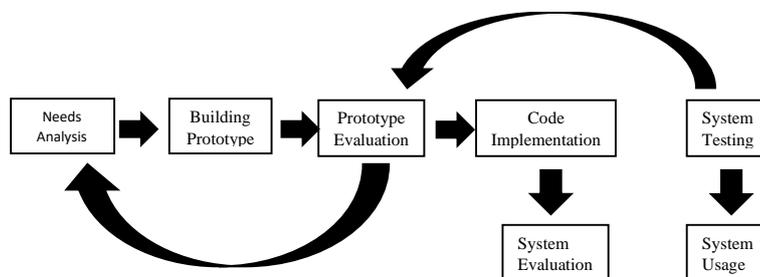
Figure 1 Prototype

Figure 1 outlines numerous phases of the prototype method.

1) Needs Analysis : Both users and developers defined the software format and identified the needs and the system to be created.
2) Building Prototype : It refers to a system design by making a temporary design focusing on presenting it to the users.
3) Prototype Evaluation : This stage was carried out by the users, whether the prototype followed their needs and wants. It would be revised by repeating the previous steps if it was inappropriate. However, if appropriate, it proceeded to the subsequent step.
4) Code Implementation : The agreed prototype was then translated into a programming language at this stage.

5) System Testing : Once the system was completed and became software, it underwent testing to minimize errors.
6) System Evaluation : At this stage, the users evaluated whether the system met their expectations. The developer would repeat the fourth and fifth steps if they detected several issues.
7) System Usage : Software tested and approved by the users was ready to use.

3. Results

3.1. Needs Analysis

The developed web application was accessible to both administrators and users, where the administrators were the administrative employee, and the users referred to students. The administrators acquired complete access to this application and could manage financial data for reporting. In contrast, the users were only able to view the data. The needs analysis was conducted through conversations with the project manager. The following are the needs analysis results on the payment administration web application.

a. There was a "Login" allowing the administrators and users to access the application.
b. There was "Add student data" to input student data.
c. There was "Add data for the academic year" to fill the school year.
d. There was "Add data type of cost" to fill in the cost type.
e. There was a print feature to print the data results in a pdf form.

3.2.Building Prototype

Every application requires an interface design. It offers a comprehensive overview of the entire application. This interface design describes each menu page and its functionality, created primarily for login into the application. Administrators and users must log in to access the application. In this study, user and administrator login pages were identical. Figure 2 depicts an overview of the interface design.
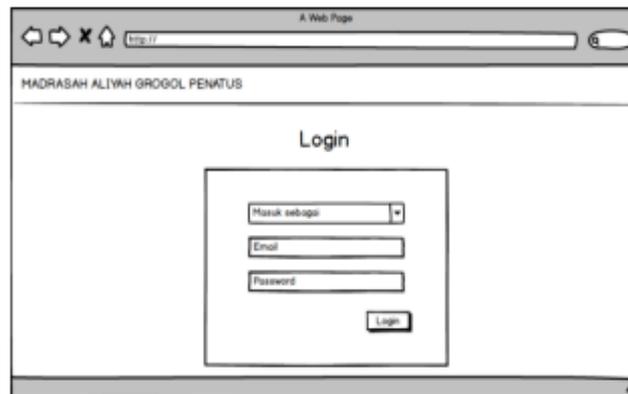
Figure 2 Login Interface Design

The administrators then proceeded to the administrator home page after logging in. The home page was the main view of the administrator interface design. Figure 2 illustrates that this page offers an administrator main menu interface.

Figure 3 Homepage Interface Design

Developing a data-based website requires a database. This web application was developed using XAMPP software and MySQL as the database. Apache, MySQL, PHP, and Perl were the components of XAMPP acting as the server (localhost). Figure 3 demonstrates how to activate Apache and MySQL for use with databases and local hosting.
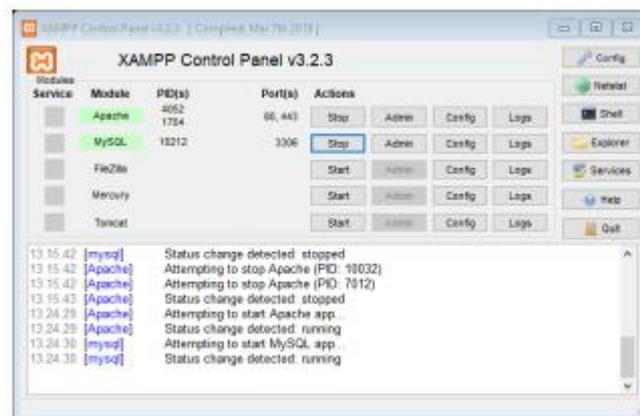
Figure 4 XAMPP Server

The image depicts the activation of Apache, a web server (HTTP Server), and MySQL, a database management system. Once the activation was complete, the database could be accessed at http://localhost/phpmyadmin/, as exhibited in Figure 4.

Figure 5 Database

Figure 5 illustrates a database with the name sekolah_sikeu, containing 13 tables.

Testing is an essential phase of the application development life cycle. It aims to ensure that all functions are fully operational and to detect any lingering problems or flaws in the developed application. Table 1 summarizes the testing results of the application.

Table 1 Administrator Functional Testing

No	Test Case	Expected Results	Results Obtained	Status
1	Login page	The administrator can successfully enter the login page.	After clicking the button, the administrator entered the login page.	Succeed
2	Login button on the login page with email and password as the input	A notification will appear that the email or password is incorrect.	A notification popped out when the administrator entered the wrong email or password.	Succeed
3	Home page	After successfully logging in, the administrator can enter the home page.	After clicking the administrator login button, it went directly to the home page.	Succeed
4	Logout button	The administrator goes back to the login page.	After clicking the logout administrator button, it immediately returned to the login page.	Succeed
5	Employee menu	The administrator successfully enters the employee page.	After clicking the employee menu, it went directly to the employee data page.	Succeed
6	"Add employee data" button on the employee page	The administrator successfully enters the "create employee" page.	After clicking the add data button, it went directly to the "create employee" page.	Succeed
7	Incorrect employee data input	An alert appears when data input is incomplete.	After clicking the submit button, an alert appeared.	Succeed
8	Correct employee data input	The data created are stored in the employee table.	After clicking the submit button, it went straight back to the employee page.	Succeed
9	Update button on the employee page	Admin successfully enters the employee update page.	After clicking the update button, it went directly to the employee update page.	Succeed
10	The submit button on the employee update page	An alert appears when the data have not been changed or are empty.	After clicking the submit button, an alert appeared.	Succeed

Table 2 User Functional Testing

No	Test Case	Expected Results	Results Obtained	Status
1	Login page	The user successfully enters the login page.	After clicking the button, the user entered the login page.	Succeed
2	Login button on the login page with email and password as the input	A notification will appear that the email or password is incorrect.	A notification appeared when the user entered the wrong email or password.	Succeed
3	Home page	After successfully logging in, the user can enter the home page.	After clicking the login button, the user immediately went to the home page.	Succeed
4	Logout button	The user returns to the login page.	After clicking the logout button, the user immediately returned to the login page.	Succeed
5	Bill payment menu	The user enters the bill payment page.	After clicking the bill payment menu, the user went directly to the bill payment page.	Succeed
6	Payment history menu	The user enters the payment history page.	After clicking the payment history menu, it went directly to the payment history page.	Succeed
7	SPP payment bill menu	The user enters the SPP payment bill page.	After clicking the SPP payment bill menu, it immediately went to the SPP payment bill page.	Succeed
8	Tuition payment history menu	The user enters the SPP payment history page.	After clicking the SPP payment history menu, it immediately went to the SPP payment history page.	Succeed
9	Menu for the annual suitcase fee	The user enters the annual component cost page.	After clicking the annual component cost menu, it immediately went to the annual component cost page.	Succeed

Because both administrators and students required access to the MA Grogolpenatus financial administration web application, it included a login form on its website. Those designated as school administrators had access to this web application. The administrators could perform everything on the data, including adding, editing, removing, searching, and printing them out. However, users could only view what the administrators made available to them.

The primary feature of this web application was processing school payment data. Data processing was under the administrators' authority, including additions, changes, and deletions. Data name search was another tool available to the administrators for locating certain records. The payment data could also be printed out and reviewed by the administrators using the print function.Tables 1 and 2 exhibit the results of functional testing. Thus, it can be safely assumed that this web application was up and functioning as expected.

4. Conclusion

Testing and feedback on the web application of the MA Grogolpentus financial administration system resulted in the following conclusions. The school financial administration system web application was suitable for administrative staff and students. The web application could be utilized to manage the financial records of a school. The administrators could save, edit, and delete data as necessary. Users could only access their bill payments using this web application. The administrators could print the payment data.

References

[1] Siahaan, K. (2017). Analisis Dan Perancangan Sistem Informasi Administrasi Keuangan Siswa Pada Smk Unggul Sakti Jambi. 2(3), 19.

[2] Z, A., Usman, U., & Ilyas, I. (2019). Sistem Informasi Pembayaran Sumbangan Pembinaan Pendidikan (Spp) Pada Sma Negeri 1 Gaung Berbasis Web. Sistemasi, 8(1), 153. Https://Doi.Org/10.32520/Stmsi.V8i1.444.

[3] Irawan, B., & Retnoningsih, E. (2017). Sistem Informasi Administrasi Pembayaran Sekolah Pada SMP Darun Nurjati Kota Bekasi. 16.

[4] Gumilar, S. K. Y., & Muhammad, W. (2017). Aplikasi Berbasis Web Pengelolaan Keuangan Sekolah(Studi Kasus: Ma Persis Tarogong Garut). 14.

[5] Abu, S. N. (2014). Pembinaan Guru Oleh Kepala Sekolah Dalam Pengelolaan Pembelajaran Di Sekolah Dasar. 2, 9.

[6] Nofyat, Ibrahim, A., & Ambarita, A. (2018). Indonesian Journal On Information System. 10.

[7] Mahdiana, D. (2011). Analisa Dan Rancangan Sistem Informasi Pengadaan Barang Dengan Metodologi Berorientasi Obyek : Studi Kasus Pt. Liga Indonesia. 8.

[8] Sulistyorini, P. (2009). Pemodelan Visual Dengan Menggunakan UML Dan Rational Rose. 7.

[9] Hasugian, H., & Shidiq, A. N. (2012). Rancang Bangun Sistem Informasi Industri Kreatif Bidang Penyewaan Sarana Olahraga. 7.

[10] Afyenni, R. (2014). Perancangan Data Flow Diagram Untuk Sistem Informasi Sekolah (Studi Kasus Pada Sma Pembangunan Laboratorium Unp). 2(1), 5.