CoreID JOURNAL
Vol. 1 No. 1, March 2023 (21-32)
ISSN: 2987-6990
DOI: https://doi.org/10.60005/coreid.v1i1.6

Designing Website-Based Administration Management Information System

Nurfitria Khoirunnisa[1], Haryati[2], Eulis Jubaedah[3], Tia Rostiawati[4]
[1,2,3,4] Information System Department, Politeknik Negeri Subang, Indonesia

Article Info

Article history:

Received February 25, 2023
Revised March 17, 2023
Accepted March 27, 2023

Keywords:

Administrative
Information System
Letter
Organization
Website

ABSTRACT

Administration is one of the essential things for the running of an organization. If in an organization the administrative management is good, the organization will also be able to run well. The current administration can be used for decision-making, evaluating the activities carried out, and even knowing whether or not the organization is being run. The same happened at the Informatics Management Student Association (HIMMI) at the Subang State Polytechnic campus. Administrative management that is carried out is still conventional by storing documents in several storage folders so that there is a buildup of documents. Even documents will be scattered everywhere. Making letters is still done by checking the last number on the general ledger, so it has difficulty to make letter numbers. The submission of letters is hampered because it takes a long time because goes through a fairly long flow. Therefore, a solution was found, namely the Design of a Website-Based Administration Management Information System using the Waterfall method to assist secretaries in managing secretarial administration at the Informatics Management Student Association. The result obtained is an administrative management information system that can facilitate the storage of documents that have been integrated with the database. In addition, the system can simplify the process of issuing outgoing letters, overcome the problem of double mail numbering, streamline the attendance process, provide a timeline of work programs, and can provide information related to the reports needed. The test results conducted using the black-box testing method on 102 test items were declared 100% valid. It showed no errors or failures in the system at the time of running.

Corresponding Author:

Nurfitria Khoirunnisa
Information System Department, Politeknik Negeri Subang
Jl. Brigjen Katamso No.37, Dangdeur, Kec. Subang, Kabupaten Subang, Jawa Barat 41211, Indonesia
Email: nurfitria@polsub.ac.id

1. INTRODUCTION

An organization or agency must have administration. Administration, in a narrow sense, is data and information processing activities that include record-keeping, correspondence, and bookkeeping activities in writing needed by an organization [1]–[4]. Administration is one of the essential things for the running of an organization. If in an organization the administrative management is good, the organization will also be able to run well [5], [6]. The current administration can be used for decision-making, evaluating the activities carried out, and even knowing whether or not the organization is being run. [7]. The same happened at the Himpunan Mahasiswa Manajemen Informatika (HIMMI) at the Subang State Polytechnic campus. The administration that is most often carried out is the administration of correspondence, submission of proposals, attendance, recording timelines, and making other documents. Sometimes in each period of management, these documents need to be stored correctly, so it will cause several problems that can occur, one of which is the absence of reference documents or evaluation materials for the next activity. Based on the results of observations we made with related parties, namely the secretary of the association in charge of taking care of administrative matters, administrative management at the Informatics Management Student Association is still carried out conventionally by storing documents in several storage folders so that there is a buildup of documents and even

documents will be scattered everywhere, making letters is still by checking the last number on the ledger so that it has difficulty in making Letter number, the submission of the letter is hampered because it takes a long time because through a reasonably long flow from the chief executive who asks for a letter to be made by the secretary, then the secretary makes an outgoing letter and waits for revision or approval from the supervisor. The outgoing letter can be submitted to the relevant party. In addition, making reports by recording one by one often need to remember attendance even though it is essential to assess the performance and contribution of the board.

Based on research that has been conducted previously by Rendi Pradipta (2017) entitled "Design and Build a Mail Administration Management Information System using the Laravel Framework in the Department of Electrical Engineering" states that the Administrative Management Information System of the Department of Electrical Engineering helps administrate incoming mail, outgoing mail, and making letters carried out by students. The testing results with black box testing obtained the expected results [4]. So that the mail administration management information system of the Department of Electrical Engineering is feasible to be applied [4]. Gusti Purnama Sari, entitled "Design and Build of Electronic Mail and Disposition Information System of Jambi University," stated that the Electronic Mail and Disposition Information System of Jambi University developed was declared feasible and considered good according to aspects of functionality, efficiency, reliability, and usability [8].

With the problems that have been described, a solution was found, namely the Design of a Website-Based Administrative Management Information System to assist the secretary in managing secretarial administration at the Informatics Management Student Association.

2. RESEARCH METHOD

The method that supports the problems that have been raised is using the waterfall method. Broadly speaking, the waterfall method consists of several stages: requirements definition, system and software design, implementation and unit testing, integration and system testing, and operation and maintenance [2], [9]-[13]. However, at this time, the design of information systems is only up to the integration and system testing stage. The following is a system design framework that refers to the waterfall method can be seen in Figure 1 below.

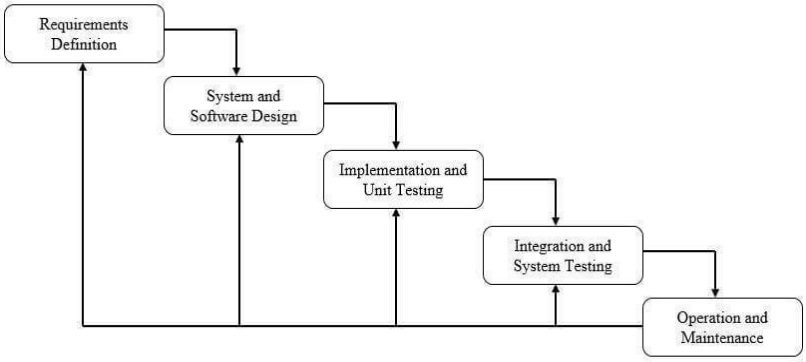

Figure 1. Waterfall Method [10], [11], [14]

2.1. Requirement Definition

Requirement Definition is a stage where the various requirements of the system are to be built [15]–[17]. This stage generates a user requirements document. At this stage of collecting, literature studies, observations, and interviews are needed to identify factors that cause these problems so that they can decide a way out of problems regarding administrative management.

a. Literature Study
   Literature studies are carried out by reading and collecting data based on books, journals, and previous research related to the problems to be discussed.
b. Observation and Interview
   Observations and interviews were conducted by interviewing one of the related users, the secretary of the Subang State Polytechnic Information Management Student Association.

2.2. System And Software Design

System And Software Design is the stage where designing the appearance of the system interface and the entire business process or workflow of a system. The design aims to describe a system to be used as a guide in building the system [15]–[17]. This design stage is based on the previous needs analysis through various methods. At the modeling stage using tools found in Unified Modeling Language (UML), namely use case diagrams, activity diagrams, sequence diagrams, and class diagrams [18]–[22]. In addition, at the user interface design stage, using Figma mockup facilitates the design of administrative management information systems.

2.3. Implementation And Unit Testing

Implementation And Unit Testing is the stage of creating and building a system based on needs analysis and design [15]–[17]. The system is created using programming language coding to implement the system. The programming language used uses PHP 8 programming language with Laravel Framework, and the database uses Mysql.

2.4. Integration And System Testing

Integration and system testing is the last stage in the design and development of a system. Testing is testing on the system carried out by testers. This stage is carried out to know whether the system built is running with its function or by design, in addition to finding out whether the user needs have been met or not [15]–[17]. Testing is carried out with black box testing techniques, where testing is carried out to observe the input and output results of the software or test the functionality of each menu, feature, or button contained in the system [8].

3. RESULT AND DISCUSSION

3.1. Modeling

Modeling aims to describe a system so that it can be used as a guide in developing systems that will be created and designed based on the needs analysis carried out previously through various methods. Modeling carried out at this stage includes the design of UML systems, databases, and user interfaces. Some of these explanations are outlined in the subchapters below.

3.1.1. Use Case Diagram

Use case diagrams are used to show the interaction of actors and systems [23]. The use case of the mail management information system can be seen in Figure 2 below.

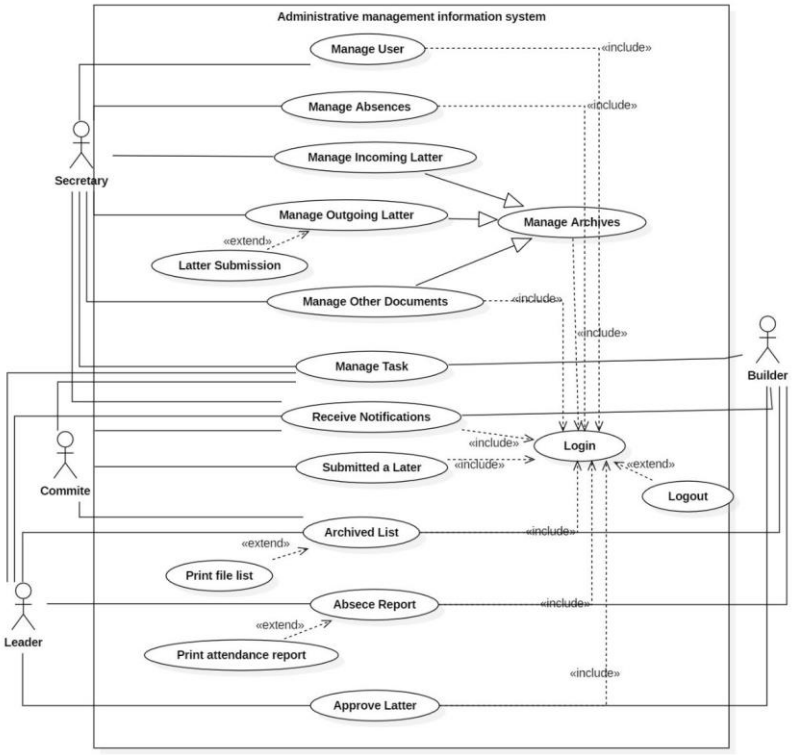

Figure 2. Use Case Diagram Administrative Management Information System

The use case diagram above has four users. The following definition of actors in administrative management information systems can be seen in Table 1 below.

Table 1. Definition Actor

No	Actor	Description
1	Secretary (Admin)	Secretary is an admin who has access rights in terms of managing user data, managing attendance, managing incoming mail, managing other documents,

No	Actor	Description
		managing outgoing mail, managing agendas, following up on mail submissions, and receiving notifications.
2	Chairman of the Association (User)	The association's chairman is a user with access rights in terms of submitting letters, managing agendas, receiving notifications, viewing attendance data, and viewing mail data.
3	Builder (User)	A Builder is a user with access rights in managing agendas, receiving notifications, viewing attendance data, viewing mail data, and approving letters.
4	Management (User)	An administrator is a user with access rights in terms of submitting letters, managing agendas, receiving notifications, and viewing archive data. The board here includes the vice chairman, treasurer, disciplinary commission, department chair, and department members.

The following is a definition for every "use case" in an administrative information system, which can be seen in Table 2 below:

Table 2. Description Use Case

No	Use Case	Description
1	Login	It is *the process of the* user and admin logging into the website system.
2	Logout	It is the process of exiting *users* and admins from the website system.
3	Manage user data	It is a process carried out by the admin in managing user accounts (users*)* who have access rights to the website system, namely viewing users, adding users, editing users, and deleting *users.*
4	Manage attendance	It is a process carried out by the admin in managing attendance reports related to administrator attendance information in the association, such as adding management and attendance data, namely viewing attendance, adding attendance, changing attendance, deleting attendance, and printing attendance.
5	Manage archives	It is a process carried out by admins in managing archiving.
6	Manage Incoming Mail	It is the process of archiving incoming mail carried out by the admin, namely viewing incoming mail, adding incoming mail, changing incoming mail, deleting incoming mail, and printing incoming mail.
7	Manage Outgoing Mail	It is the process of archiving outgoing mail carried out by the admin, namely viewing outgoing mail, adding outgoing mail, changing outgoing mail, deleting outgoing mail, and printing outgoing mail.
8	Manage other documents	It is another document archive process carried out by the admin, namely, viewing other documents, adding other documents, changing other documents, deleting other records, and printing other documents.
9	Mengajukan surat	It is a process by users, namely administrators and heads of associations, to submit outgoing letters.
10	Managing the Agenda	It is a process carried out by admins, namely secretaries and *users*, namely administrators, heads of associations, and coaches, in terms of managing the agenda of activities in the association, both viewing the agenda, adding to the agenda, changing the agenda, and deleting the agenda.
11	Get notifications	It is a process carried out by admins, namely secretaries and *users*, namely administrators, heads of associations, and coaches.
12	Managing the Agenda	It is a process carried out by admins, namely secretaries and *users*, namely administrators, heads of associations, and coaches, in terms of managing the agenda of activities in the association, viewing the agenda, adding to the agenda, changing it, and deleting it.
13	Get notifications	It is a process carried out by admins, namely secretaries and *users*, namely administrators, heads of associations, and coaches.
14	View archive data	It is a process carried out by *users*, namely the head of the association, management, and coach, where *users* can see archive data, incoming mail, outgoing letters, and other documents.
15	Print an archive report	It is a process carried out by users, namely the head of the association, management, and coach, where users can see archive data, incoming letters, outgoing letters, and other documents.
16	Approve the letter	It is the process of receiving information related to changes in data that can be received by the user.
17	View mail submissions	This is a process that can be done by the admin when they want to see the submission of letters that have been made by the board or chairman.

3.1.2. Class Diagram

Class diagrams are static diagrams of an application. Class diagrams are used to visualize, describe, and also document various aspects of a system [23]. The class diagram in the design of this administrative management information system illustrates the functions and attributes to be used in building this system using the MVC

concept. The controller is used to receive requests and data from the user and then determine what will be processed. Models deal directly with databases to manipulate data. The view is used to display data to the user. The following is a class diagram of the administrative management information system that can be seen in Figure 3.

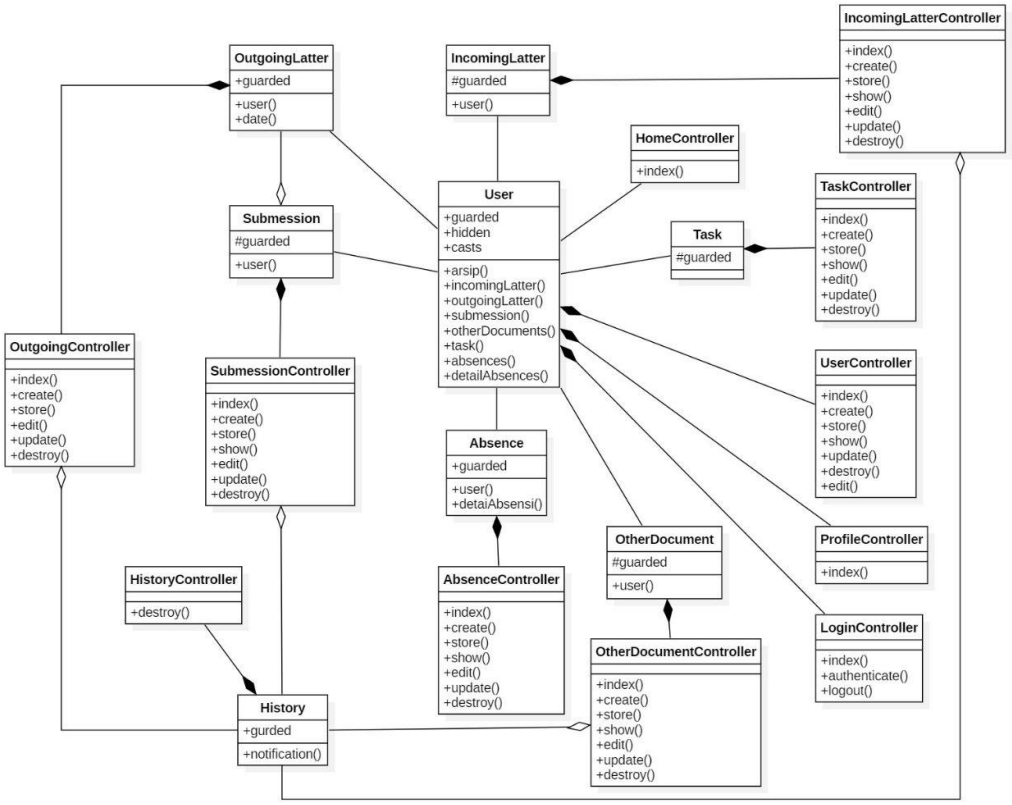

Figure 3. Class Diagram Administrative Management Information

3.2. Design *Database*

Entity Relationship Diagram (ERD) is a modeling tool for primary data. The purpose is to explain the connection between data in a database, based on every database object that is related [24]–[26]. The following are ERD from the mail Management Information system that can be viewed in Figure 4 below

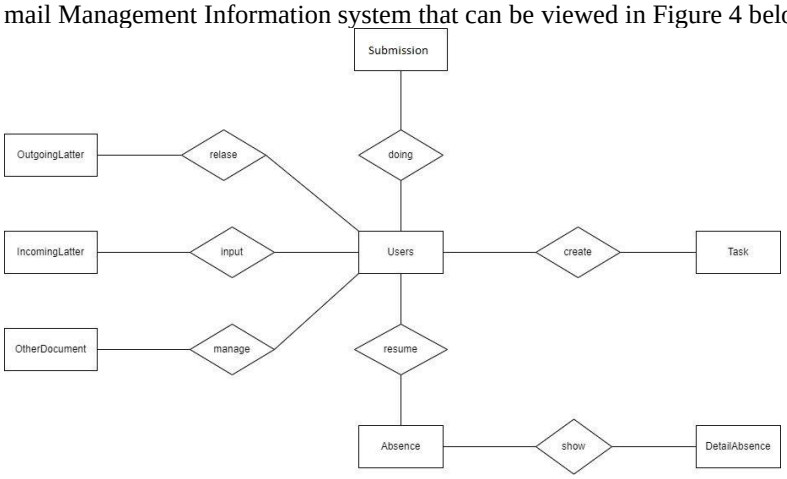

Figure 4. ERD Administrative Management Information System

The following are Structure table based on the entity of ERD Administrative Management Information System

Table 2. Structure for User Table

Attribute Name	Type_Data	Table Referensi	PK/FK
id	bigint(20) Unsigned		Primary Key
name_ users	varchar(255)		
Username	varchar(255)		Unique

Attribute Name	Type_Data	Table Referensi	PK/FK
Registration_number	varchar(255)		Unique
position	varchar(255)		
status	varchar(255)		
Image	varchar(255)		
generation	varchar(255)		
email	varchar(255)		Unique
password	varchar(255)		

Table 3. Structure for Submission Table

Attribute Name	Type_Data	Table Referensi	PK/FK
id	bigint(20) Unsigned		Primary key
user_id	bigint(20) Unsigned	Table users	Foreign Key
registration_number	varchar(255)		Unique
date	Date		
letter_destination	varchar(255)		
regarding	varchar(255)		
status	varchar(255)		
comments	text		
file	varchar(255)		

Table 4. Structure for Submission Table

Attribute Name	Type_Data	Table Referensi	PK/FK
id	bigint(20) Unsigned		Primary Key
user_id	bigint(20) Unsigned	Table users	Foreign Key
date	date		
name	varchar(255)		
present	varchar(255)		
absent	varchar(255)		
comments	varchar(255)		

Table 5. Structure for Details Attendance Table

Attribute Name	Type_Data	Table Referensi	PK/FK
id	bigint(20) Unsigned		Primary Key
present _id	bigint(20) Unsigned	Table absences	Foreign Key
user_id	bigint(20) Unsigned	Table users	Foreign Key
presence	varchar(255)		
comments	varchar(255)		

Table 6. Structure for Tasks Table

Attribute Name	Type_Data	Table Referensi	PK/FK
id	int(10) unsigned		Primary Key
user_id	bigint(20) Unsigned	Table users	
name	varchar(255)		
description	text		
task_date	date		
background	varchar(255)		
text	varchar(255)		

Table 7. Structure for Incoming Mail Table

Attribute Name	Type_Data	Table Referensi	PK/FK
Id	bigint(20) Unsigned		Primary Key
user_id	bigint(20) Unsigned	Table users	Foreign Key
registration_number	varchar(255)		
date	date		
letter_origin	varchar(255)		
subject	varchar(255)		
file	varchar(255)		

Table 8. Structure for Outgoing Mail Table

Attribute Name	Type_Data	Table Referensi	PK/FK
id	bigint(20) Unsigned		Primary Key
user_id	bigint(20) Unsigned	Table users	Foreign Key

Attribute Name	Type_Data	Table Referensi	PK/FK
registration_number	varchar(255)		
date	date		
letter_destination	varchar(255)		
subject	varchar(255)		
status	varchar(255)		
comments	text		
fill	text		
file	varchar(255)		
kahim	varchar(255)		
secretary	varchar(255)		
ketuplak	varchar(255)		

Table 9. Structure for Other Document Table

Attribute Name	Type_Data	Table Referensi	PK/FK
id	bigint(20) Unsigned		Primary Key
user_id	bigint(20) Unsigned	Table users	Foreign Key
name	varchar(255)		
File	varchar(255)		

3.3. User Interface

Design Interface Administrative Management Information System for Subang State Polytechnic Information Management Student Association consist of many pages, The following are several pages. The first thing that the system display is a login page, in the login page there is an input form such as email and password for the user that wants to access administrative management information. If the email and password are correct, the system would open a front page. If failed it will appear login failed notification. The following is the display login page that showed in Figure 5.

After the login is successful, the system will be directed to the dashboard page, in the dashboard secretary will appear numbers of incoming mail, outcoming mail, another document, total attendances, and displays agenda, as shown in Figure 6.

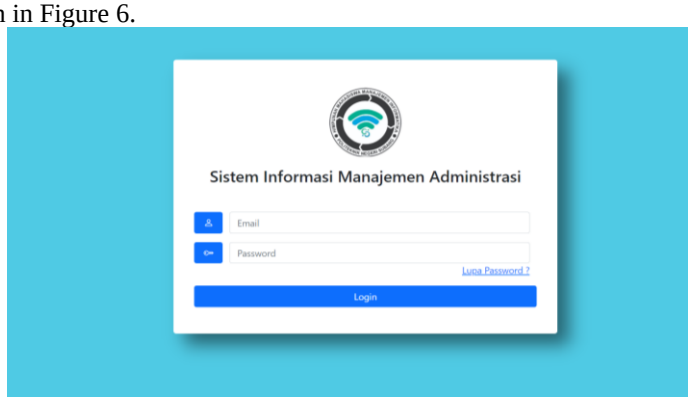

Figure 5. Login Display

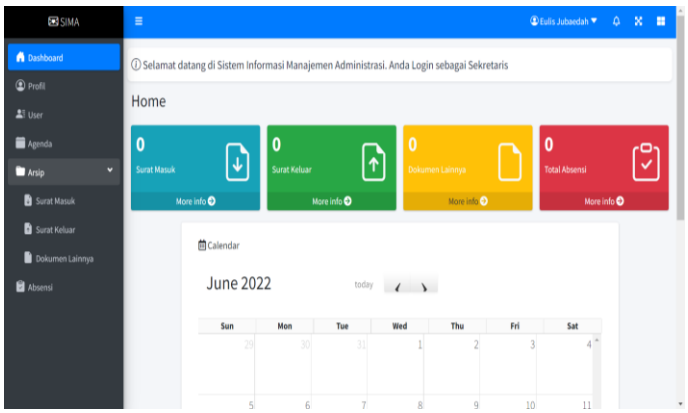

Figure 6. Dashboard Display

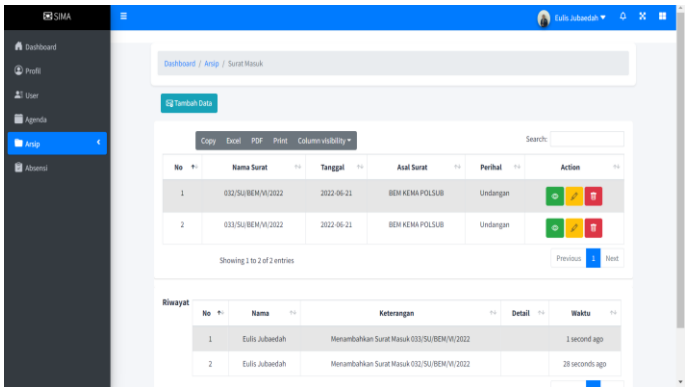

Figure 7. Incoming Mail Display

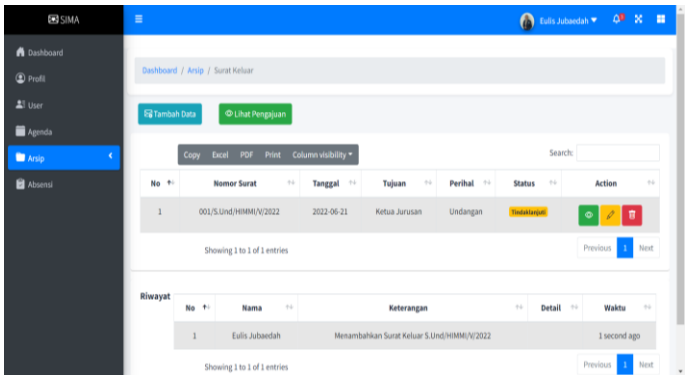

Figure 8. Outgoing mail Page Display

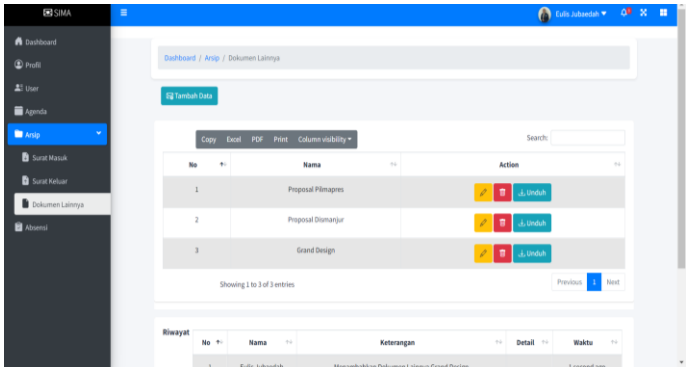

Figure 9. Another Document Page Display

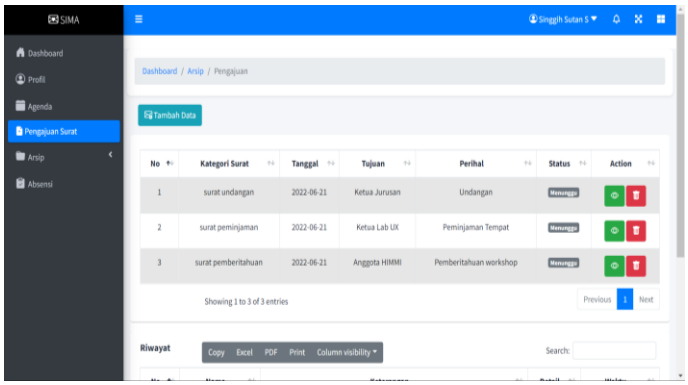

Figure 10. Submission Mail Page Display

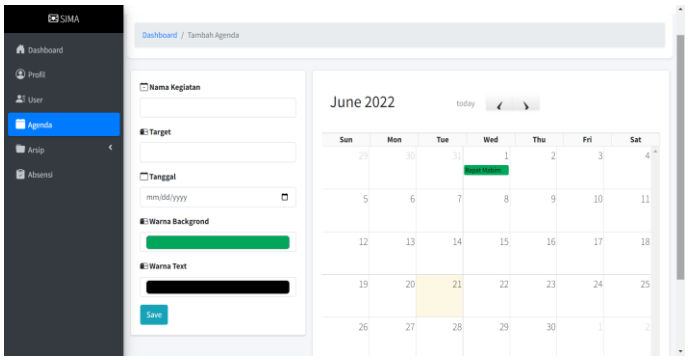

Figure 11. Agenda Page Display

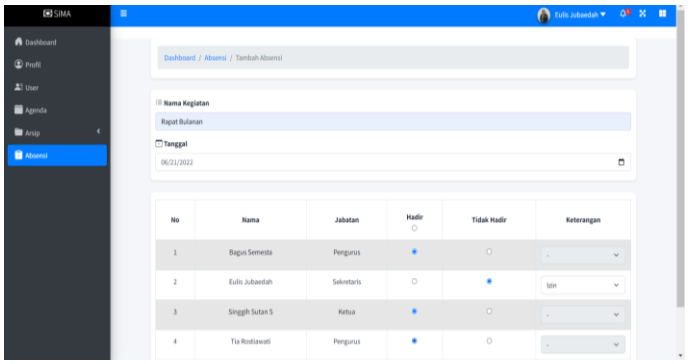

Figure 12. Attendance Page Display

3.4. Coding

The administrative Management Information System has used Laravel 9 framework with MVC architecture (*Model*, *View*, *Controller*). The framework is a basic conceptual structure that use to solve complex problems. This terminology was used in software scope to explain the design software system that is reusable[8]. Concept MVC was an architectural pattern that separate into 3 main component logic. Model related to data and interaction for database contains queries in the administrative mail system. The Model contains a class and function for fetching, updating, and deleting data in the Administrative mail system database. "View" is used to display the webpage in the administrative mail system that contains html, jquery, javascript, and CSS, While the controller in the administrative mail system acts as a bridge to connect data with "view". There are several source code model figures, views, and controllers that are shown in Figure 13, Figure 14, and Figure 15.

```
32    /**
33     * Show the form for creating a new resource.
34     *
35     * @return \Illuminate\Http\Response
36     */
37    public function create()
38    {
39        return view('surat-masuk.create', [
40            'title' => 'Tambah Surat Masuk',
41            'tasks' => Task::all(),
42            'notif' => Riwayat::notif()
43        ]);
44    }
45
46    /**
47     * Store a newly created resource in storage.
48     *
49     * @param  \Illuminate\Http\Request  $request
50     * @return \Illuminate\Http\Response
51     */
52    public function store(Request $request)
53    {
54        $suratMasuk = SuratMasuk::max('id');
55        $validatedData = $request->validate([
56            'user_id' => 'required',
57            'nomor_surat' => 'required',
58            'tanggal' => 'required',
59            'asal_surat' => 'required',
60            'perihal' => 'required',
61            'berkas' => 'nullable|file|max:16384'
62        ]);
```

Figure 13. Source Code Controller

```php
<?php

namespace App\Models;

use App\Models\User;
use App\Models\Pengajuan;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Database\Eloquent\Factories\HasFactory;

class SuratMasuk extends Model
{
    use HasFactory;

    protected $guarded = ['id'];

    public function user()
    {
        return $this->belongsTo(User::class);
    }

    public function pengajuan()
    {
        return $this->belongsTo(Pengajuan::class);
    }
}

```

Figure 14. Source Code Model

```
@extends('layouts.dashboard')

@section('container')
    <div class="container">
        <div class="row bg-white mt-4 pt-3 p-1 rounded-3 shadow-sm">
            <nav style="--bs-breadcrumb-divider: '>';" aria-label="breadcrumb">
                <ol class="breadcrumb">
                    <li class="breadcrumb-item"><a href="/home">Dashboard</a></li>
                    <li class="breadcrumb-item"><a href="/arsip">Arsip</a></li>
                    <li class="breadcrumb-item active" aria-current="page">{{ $title }}</li>
                </ol>
            </nav>
        </div>
        @if (session()->has('success'))
            <div class="alert alert-success alert-dismissible fade show mt-4" role="alert">
                {{ session('success') }}
                <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>
            </div>
        @endif
        @if (auth()->user()->jabatan === 'Sekretaris')
            <a href="/surat-masuk/create" class="btn btn-info text-white m-3"><i class="bi bi-envelope-plus"></i> Ta
                Data</a>
        @else
            <div class="my-5">
            </div>
        @endif
        <div class="table-responsive bg-white text-center">
            <div class="card">
                <!-- /.card-header -->
                <div class="card-body">
                    <table id="example1" class="table table-bordered table-striped">
                        <thead>
```

Figure 15. Source Code View

3.5. Result

System testing is the testing of a complete and integrated software program. Software or often known as software is just a unit of elements of a larger computer-based system. Usually, the software is connected with other software and hardware.

System testing is carried out to find out whether the system built is running as its function or not, but also to find out whether user needs have been met or not in the implementation and design results. System testing is carried out using the black box testing method or functional testing that tests the inputs and outputs produced, administrative management information system testing is carried out by 3 testers on 46 test items consisting of 102 action procedures, resulting in 102 valid action procedures and 0 invalid ones. This shows that the function of the system is 100% running well, there are no errors or errors during operation.

4. CONCLUSION

Based on the result of Designing Website-Based Administration Management, we can conclude several points:

A. Administrative Management Information System facilitates in-store documents, because it's integrated with a database then secure and structured.
B. Administrative Management Information System can accelerate the publishing of outgoing mail process, starting with the submission mail process by the administrator and head, then followed up by the secretary with the addition of outgoing mail, then it can be revised and approved by the supervisor.
C. Administrative Management Information System affords to solve the problem of numbering multiple emails, make the attendance process effective, provide a work programs timeline, and can provide information related to the required reports.
D. Administrative Management Information System stated eligible based on test results on 102 test items 100% valid, this means that there is no error or failure when the system is running.

ACKNOWLEDGEMENTS

The researcher also gives thanks to Politeknik Negeri Subang and many people that support and contribute to this research.

REFERENCES

[1] L. Marliani and L. Marliani, "Definisi Administrasi dalam Berbagai Sudut Pandang," *Din. J. Ilm. Ilmu Adm. Negara*, vol. 5, no. 4, pp. 17–21, Jan. 2019.

[2] M. A. Lestari, M. Tabrani, and S. Ayumida, "Sistem Informasi Pengolahan Data Administrasi Kependudukan Pada Kantor Desa Pucung Karawang," *J. Interkom J. Publ. Ilm. Bid. Teknol. Inf. dan Komun.*, vol. 13, no. 3, pp. 14–21, Oct. 2018.

[3] R. Wardhani and M. H. Yaqin, "Game Dasar- Dasar Hukum Islam Dalam Kitab Mabadi ' ul Fiqh Jilid I," *Teknika*, vol. 5, no. 2, pp. 473–478, 2013.

[4] R. Pradipta, H. Wibawanto, and A. Mulwinda, "Rancang Bangun Sistem Informasi Manajemen Administrasi Surat Menggunakan Framework Laravel pada Jurusan Teknik Elektro," *Pros. Sintak*, pp. 284–289, 2017.

[5] Ushansyah, "Pentingnya Administrasi Sekolah Untuk Kemajuan Pendidikan," *Ittihad J. Kopertais Wil. XI Kalimantan*, vol. 15, no. 27, pp. 13–22, 2017.

[6] N. F. Luaylik, "Dari Negara menjadi Publik Sentris: Kajian Historis Pelaksnaan Administrasi Indoensia Tahun 1900-an," *Al-Isnad J. Islam. Civiliz. Hist. Humanit.*, vol. 1, no. 1, pp. 63–73, 2020.

[7] A. Budaya *et al.*, "Analisis Budaya Organisasi Dalam Meningkatkan Prestasi Kerja Pegawai Administrasi UIN Datokarama Palu," *J. Integr. Manaj. Pendidik.*, vol. 1, no. 1, pp. 20–29, Mar. 2022.

[8] G. P. Sari, J. Marzal, and M. Mauladi, "Rancang Bangun Sistem Informasi Persuratan dan Disposisi Elektronik Universitas Jambi," *JUSS (Jurnal Sains dan Sist. Informasi)*, vol. 1, no. 1, pp. 20–29, Mar. 2018.

[9] A. C. Nugroho, "Rancang Bangun Sistem Informasi Manajemen Surat Tugas Berbasis Web Menggunakan Waterfall Model," *J. Inform. J. Pengemb. IT*, vol. 4, no. 2, pp. 146–151, May 2019.

[10] C. Tristianto and C. Tristianto, "Penggunaan Metode Waterfall untuk Pengembangan Sistem Monitoring dan Evaluasi Pembangunan Pedesaan," *J. ESIT (E-Bisnis, Sist. Informasi, Teknol. Informasi)*, vol. 12, no. 1, Jan. 2022.

[11] E. Darmawan, "C4.5 Algorithm Application for Prediction of Self Candidate New Students in Higher Education," *J. Online Inform.*, vol. 3, no. 1, p. 22, Jun. 2018.

[12] N. Lukman, O. T. Kurahman, S. Respiawati, and A. Sulaeman, "Integration of Repository System in Optimization Data for Graduates' Scientific Paper," *Khizanah al-Hikmah J. Ilmu Perpustakaan, Informasi, dan Kearsipan*, vol. 10, no. 2, pp. 209–218, Dec. 2022.

[13] N. Lukman, F. Umar, and Y. Aditia Gerhana, "Digitalisasi Layanan Minimun Desa Sebagai Upaya Peningkatan Layanan Desa Binaan," *AL Khidm. J. Ilm. Pengabdi. Kpd. Masy.*, vol. 5, no. 1, 2022.

[14] R. Layona and B. Yulianto, "Aplikasi Pencarian Informasi Dan Lokasi Tempat Makan Pada Perangkat Mobile Berbasis Android," *J. Nas. Teknol. dan Sist. Inf.*, vol. 2, no. 2, pp. 9–16, 2016.

[15] T. Kusuma, T. P. P. Kusuma, and D. G. Rahayu, "Sistem Informasi Pengelola Bank ASI Berbasis Website (Studi Kasus: Posyandu Kelurahan Bantarsoka)," *Edumatic J. Pendidik. Inform.*, vol. 5, no. 1, pp. 41–49, Jun. 2021.

[16] M. Nasrullah, N. Dwi Angresti, S. Harits Suryawan, and dan Faizal Mahananto, "Requirement Engineering terhadap Virtual Team pada Proyek Software Engineering," *J. Adv. Inf. Ind. Technol.*, vol. 3, no. 1, pp. 1–10, May 2021.

[17] M. M. Fajar and U. Chotijah, "Sistem Informasi Manajemen Layanan Kearsipan (SI MALAK) Berbasis Web," *J. Inform. dan Tek. Elektro Terap.*, vol. 10, no. 3, pp. 2830–7062, Aug. 2022.

[18] D. Alita, I. Tubagus, Y. Rahmanto, S. Styawati, and A. Nurkholis, "Sistem Informasi Geografis Pemetaan Wilayah Kelayakan Tanam Tanaman Jagung dan Singkong pada Kabupaten Lampung Selatan," *J. Soc. Sci. Technol. Community Serv.*, vol. 1, no. 2, pp. 1–09, Oct. 2020.

[19] F. Sonata and V. Winda Sari, "Pemanfaatan UML (Unified Modeling Language) Dalam Perancangan Sistem Informasi E-Commerce Jenis Customer-To-Customer," *J. Komunika J. Komunikasi, Media dan Inform.*, vol. 8, no. 1, pp. 22–31, Jun. 2019.

[20] L. Jin and X. Liang, "Modeling of instant messaging system based on RUP and UML," in *Proceedings - 2016 International Conference on Computational Intelligence and Applications, ICCIA 2016*, 2016, pp. 61–66.

[21] S. Dharwiyanti and R. S. Wahono, "Pengantar Unified Modeling LAnguage (UML)," *IlmuKomputer.com*, pp. 1–13, 2003.

[22] M. O'Docherty, *Object-Oriented Analysis, and Design : Understanding System Development with UML 2.0*, 1st ed. USA: John Wiley & Sons Inc, 2005.

[23] I. Sommerville, *Software Engineering*, 9th ed. Boston: Pearson Education, 2011.

[24] N. A. Istiqomah, K. Imayah, N. Saidah, and M. A. Yaqin, "Pengembangan Arsitektur Data Sistem Informasi Pondok Pesantren," *Jurasik (Jurnal Ris. Sist. Inf. dan Tek. Inform.*, vol. 5, no. 1, pp. 27–35, Mar. 2020.

[25] Q. Li and Y.-L. Chen, "Entity-Relationship Diagram BT - Modeling and Analysis of Enterprise and Information Systems: From Requirements to Realization," Q. Li and Y.-L. Chen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 125–139.

[26] B. Nugroho, D. Remawati, and B. Widada, "IMPLEMENTASI METODE WP (WEIGHTED PRODUCT) UNTUK MENDUKUNG KEPUTUSAN PENJURUSAN SISWA DI SMA NEGERI 1 GROBOGAN," *J. Teknol. Inf. dan Komun.*, vol. 4, no. 2, Oct. 2016.