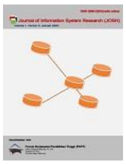


Journal of Information System Research (JOSH)
Volume 5, No. 2, Januari 2024, pp 542−553
ISSN 2686-228X (media online)
https://ejurnal.seminar-id.com/index.php/josh/
DOI 10.47065/josh.v5i2.4791

Application of RESTful Web Service in the Development of Letter Management Information System of Inspectorate

M. Fauzan Fikri, Ken Ditha Tania*

Fakultas Ilmu Komputer, Sistem Informasi, Universitas Sriwijaya, Palembang
Jl. Srijaya Negara, Bukit Lama, Kec. Ilir Bar. I, Kota Palembang, Sumatera Selatan, Indonesia
Email Penulis Korespondensi: kenya.tania@gmail.com

Submitted: **10/01/2024;** Accepted: **21/01/2024;** Published: **22/01/2024**

Abstrak−Perkembangan teknologi yang semakin pesat dapat meningkatkan efisiensi dan efektivitas di seluruh bidang dengan mengoptimalkan penggunaan teknologi. Inspektorat Provinsi Sumatera Selatan merupakan instansi pemerintahan yang berlokasi di ibu kota Sumatera Selatan, Palembang. Permasalahan yang ditemukan pada Inspektorat Provinsi Sumatera Selatan adalah disposisi surat yang belum terdigitalisasi sehingga disposisi surat tidak dapat dipantau statusnya dan surat rentan hilang. Dibutuhkan lebih banyak waktu dan penyimpanan dalam melakukan pencarian surat. Dengan sistem yang masih manual, inspektorat provinsi sumatera selatan membutuhkan anggaran lebih untuk media kertas serta kurangnya efisiensi dan efektivitas pada sistem manual. Sehingga diperlukan sistem manajemen surat yang terdigitalisasi yang dapat diakses dengan perangkat mobile maupun pc. Dengan adanya Web Service diyakini dapat menunjang integrasi data antara mobile dan pc tanpa aplikasi pihak ketiga. Penerapan RESTful API dapat meningkatkan keamanan dalam hak akses user. Hasil penelitian ini berupa aplikasi dengan metode Rapid Application Development mengadopsi perancangan sistem Unified Modelling Language dengan pengujian Blackbox diketahui bahwa aplikasi berjalan dengan baik sesuai dengan kebutuhan pengguna mengindikasikan bahwa aplikasi dapat mengatasi permasalahan pada manajemen surat yang terjadi di inspektorat provinsi sumatera selatan.

Kata Kunci: Manajemen Surat; Rapid Application Development; RESTful API; Sistem Informasi; Unified Modelling Language; Web Service

Abstract−The rapid development of technology can increase efficiency and effectiveness in all fields by optimizing the use of technology. The Inspectorate of South Sumatra Province is a government agency located in the capital city of South Sumatra, Palembang. The problem found in the Inspectorate of South Sumatra Province is the disposition of letters that have not been digitized so that the disposition of letters cannot be monitored and the range of letters is lost. It takes more time and storage in searching for letters. With a system that is still manual, the inspectorate of the province of South Sumatra requires more budget for paper media and a lack of efficiency and effectiveness in the manual system. So a digitized letter management system is needed that can be accessed by mobile devices or PCs. With the Web Service, it is believed that it can support data integration between mobile and PC without third party applications. The application of RESTful API can increase security in user access rights. The results of this study are in the form of applications with the Rapid Application Development method adopting the Unified Modeling Language system design with Blackbox testing, it is known that the application runs well according to user needs, indicating that the application can solve problems in letter management that occur in the inspectorate of the province of South Sumatra.

Keywords: Information System; Letter Management; Rapid Application Development; RESTful API; Unified Modelling Language; Web Service

1. INTRODUCTION

Increasingly advanced technological developments provide improvements in terms of efficiency and effectiveness in all areas, especially performance. The benefit gained from technological developments is that data management can be done with fast and accurate precision. The use of the system is known to increase effectiveness and efficiency [1]. A good agency should have applications that are useful and can be used to support the agency's performance [2]. Therefore, optimizing technology in an agency must be considered to increase efficiency and effectiveness as well as the performance of human resources in an agency or company.

Inspektorat Daerah Provinsi Sumatera Selatan is a government agency tasked with carrying out internal supervision of regional government administration led by an Inspector. In carrying out its duties, the provincial inspectorate cannot be separated from administrative functions such as the disposition of letters. Letter disposition is the process of distributing letters, where the steps for distributing letters are by delivering the letters to the address and person they are addressed to, classifying the letters by arranging them based on their level of importance and determining which official should handle the letters by including a disposition sheet [3].

In accordance with the Peraturan Menteri Negara Pendayagunaan Aparatur Negara Dan Reformasi Birokrasi Nomor 6 Tahun 2011 [4], it appealed to all state institutions, central and regional governments, state universities, and BUMNID to prepare instructions for implementing electronic document in their respective agencies by referring to the guidelines.

However, based on the results of interviews with the agency, it was stated that the letter disposition process had not used technology optimally. Thus, it is vulnerable to loss of files and the disposition status cannot be monitored directly, which has an impact on the performance of the agency. The absence of electronic documents has resulted in the agency having to provide more physical media such as paper and file shelves, which requires a larger budget by the day. This means that the agency’s performance in letter disposition is not optimal.

Based on an interview conducted with Kasubag Umum dan Kepegawaian, it is stated that a system that can be accessed with laptop/pc and mobile devices is needed. One of the solutions to that requirement is to implement a web service. Web service is one of the web technologies based on open standards that is used as communication means between systems interoperably across platforms [5], [6], [7]. Web services could bring benefits such as cross-platform integration, reducing complexity and increasing maintainability as well as reusability [7], [8].

Kasubag Umum and Kepegawaian in an interview stated their needs a system that have high security to minimize the occurrence of piracy. The use of RESTful Web Service is one of the technological advancements that can be a solution to the problems described previously. RESTful API have high security and can store the data securely [9].

This is reinforced by [10] that is stated RESTful API expose back-end, so it is crucial to highly secure the data, keep the security retrieve and strongly that also can protect crucial data. Representational State Transfer (REST) is an architectural style that can be defined as a set of constraints that should be met in development of network-based applications [6], [11].

REST became most popular in almost any web-related development because of its complexity. REST can support from basic web applications to highly complex web service [6]. The author's reason for implementing RESTful because it is one of the most frequently used web services is RESTful [7], [12], [13]. Therefore, the author uses the application of RESTful web services in developing a letter disposition system.

The use of RESTful Web Services is one of the technological advancements that can be a solution to the problems described previously. RESTful web services can be integrated with applications from various platforms and can be used by many applications as service consumers without having to change the code at all [7]. This is in accordance with the needs of Inspektorat Daerah Provinsi Sumatera Selatan which states that it requires a letter disposition system that can be accessed on PC devices and mobile devices. The author's reason for implementing RESTful because it is one of the most frequently used web services is RESTful [7]. Therefore, the author uses the application of RESTful web services in developing a letter disposition system.

Previous research entitled Penerapan RESTful Web Service pada Disain Arsitektur Sistem Informasi pada Perguruan Tinggi which used the Requirements, analysis, design, and implementation method resulting in a RESTFul STARS application that is implemented into STARS application. This research concluded that CRUD and HTTP methods have a correlation and the RESTful application should be developed in accordance to the REST constraints [14].

Then the next research entitled Penerapan RESTful Web Service dengan Framework Laravel untuk Pembangunan Sistem Informasi Manajemen Sumber Daya Manusia aims to create new innovations by optimizing the use of technology using RESTful by designing systems using UML. The results of this research produce applications with high security by applying authorization to the access rights of each client or actor so that not all clients can access all REST APIs [9].

Furthermore, in the research entitled Pengembangan Sistem Informasi Pelatihan Berbasis Website Menggunakan Teknologi Web Service dan Framework Laravel, adopting a web service with the aim of integrating data which can make it easier to access URLs in a browser using prototyping development method, the results of the research show that the application runs smoothly according to the requirements of the user [15]. Research conducted by Ahmad Syarifudin entitled Sistem Informasi Manajemen Surat Berbasis Website di STMIK Pringsewu aims to minimize errors in manual systems and digitize letter disposition using the 4-D development method. With this application, you can activate the use of office stationery and save budgets, so that the system can facilitate employee performance [1].

The next research researched by Ida Bagus Gede Sarasvanda with the title Pendekatan Metode Extreme Programming untuk Pengembangan Sistem Informasi Manajemen Surat Menyurat pada LPIK STIKI applies the extreme programming method which consists of four stages with system design using UML. The results of the application are in accordance with user needs and run well which is proven to be efficient [16].

The difference between this research and previous research is in terms of the methods used by previous researchers in the form of System Development Life Cycle and Extreme Programming methods, while this research uses the Rapid Application Development method. Previous research emphasized security and minimizing errors, while this research emphasizes the needs required by users, namely innovation from manuals to integrated systems, but still pays attention to security and efficiency. The significant difference between this research and previous research is that this research adopts the RESTful API with NextJS, ExpressJS, and Expo frameworks which make the system both website-based and mobile-based, so that the level of accessibility in this research is higher, while in previous research it was limited to a website-based system.

Based on the description of the problem above as well as the literature study that has been carried out by the author. The author realizes that there is a need for a Letter Management Information System at Inspektorat Daerah Provinsi Sumatera Selatan which can be accessed via computers or mobile devices. Given this need, the author adopted the implementation of a Web Service by implementing a RESTful API to support data integration while still paying attention to the security. It is hoped that this research will contribute to overcoming the problems that occur. With this Letter Management Information System application, hopefully it will be able to increase efficiency and effectiveness for human resources at Inspektorat Daerah Provinsi Sumatera Selatan in managing letters and official documents.

2. RESEARCH METHODOLOGY

2.1 Research Stages

This research has five stages in the form of data collection, problem analysis, design, implementation, integration, and testing, and maintenance which can be seen in **Figure 1** below.

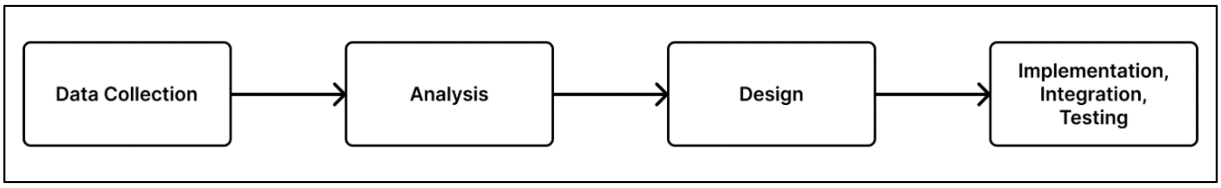


Figure 1. Research Flow

Figure 1 explains the research stages carried out by the author, starting from the data collection stage by observing the agency and conducting literature studies, after the data was collected the author conducted interviews to find out the problems that occurred in the agency and find solutions to the problems that occurred, then carried out designing a system that suits user requirements starting in from color to layout, after that implementation, integration, and testing related to the system that has been built in the agency will be carried out.

2.2 System Development Method

This research uses the Rapid Application Development (RAD) method which is divided into three stages as attached in Figure 2 Rapid Application Development (RAD) is a methodology for developing software in a short time. Rapid application development speeds up the application creation process cycle so that it can save resources [17]. The RAD method is a suitable method to be applied in this research due to time constraints in working on the system, but even though time is limited, research results and applications can still be developed according to user requirements and prioritizing quality in the application.

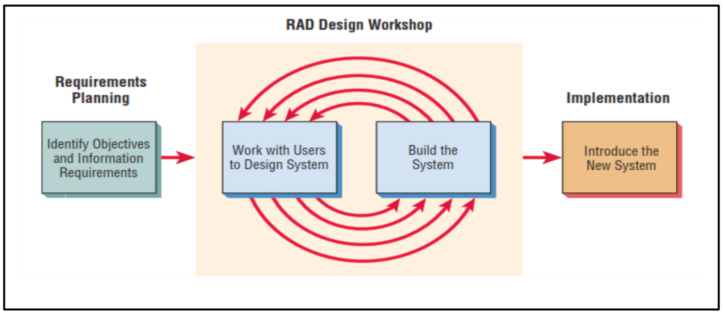


Figure 2. RAD Phases

In Figure 2 is RAD phases that is used in this research adopted from RAD phases in [18], the stages carried out by researchers in developing the system are attached as follows:

a. Requirement Planning, at this stage a needs analysis is carried out based on existing business processes by conducting observations, interviews, and documentation. Starting with defining ongoing business processes and analyzing existing problems. Then, from the problems obtained, the requirements for the new system are formulated. It can be concluded that in this stage the output obtained is analysis of ongoing business processes, problem analysis, and system requirements analysis.
b. Design Workshop, at this stage the system is designed according to the needs outlined in the requirements planning phase, the system design will be built into a prototype that can be run. In this system there are three main components in the form of a database, RESTful web service, and client application. Through this stage, business processes are depicted using UML in the form of diagrams that are use cases, sequences and component diagrams.
c. Implementation, at this stage the author will implement the accepted prototype that was created in the previous stage in the form of website and mobile application into production. Through this stage, black-box testing will also be carried out to find out whether the system is running well and in accordance with user requirements. If the application runs well and meets the user's needs, it will be installed at Inspektorat Daerah Provinsi Sumatera Selatan.

2.3 Data Collection Technique

Data collection is carried out to support this research, the author used the following data collection techniques:

a. Interview, the author conducted a semi-formal interview in the form of questions and answers with Mr. Andri Wijaya as Sub Bagian Umum dan Kepegawaian. Through interviews, it is known that letter management is still carried out manually using paper, the disposition of letter is also carried out manually depending on the recipient of the letter, and letters are vulnerable to loss. Apart from that, it is difficult to search through letters and to monitor letters, which results in the need for more time to search for letters or recreate letters.

b. Observations, the author made direct observations at Inspektorat Daerah Provinsi Sumatera Selatan regarding the flow of letter disposition that runs in the agency and collected data that can support this research.

c. Literature study, the author conducted a literature study by reading and analyzing research that had been previously carried out by previous researchers sourced from journals, books and internet. Through this literature study, the author obtained references in the form of theory and practice for building an optimal system.

2.4 System Analysis

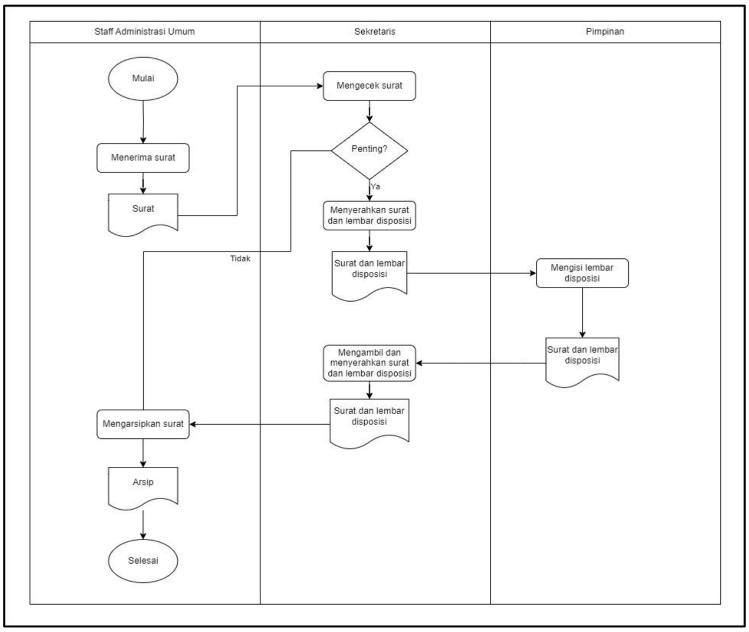

Staff Administrasi Umum
Sekretaris
Pimpinan
Mulai
Menerima surat
Surat
Mengecek surat
Penting?
Ya
Tidak
Menyerahkan surat dan lembar disposisi
Surat dan lembar disposisi
Mengisi lembar disposisi
Surat dan lembar disposisi
Mengambil dan menyerahkan surat dan lembar disposisi
Surat dan lembar disposisi
Mengarsipkan surat
Arsip
Selesai

Figure 3. Old System Flow

In Inspektorat Daerah Provinsi Sumatera Selatan, letter management still uses physical evidence such as paper media, so data is vulnerable to loss and damage, and requires more storage and requires more time to search for data and monitor the disposition of letters, this has a bad impact if the problem occurs over a long period of time. The problem that occurs in managing letters is that they still use paper media so that the disposition of letters cannot be traced. This results in employees having to look for letters in several places to monitor letters. Apart from that, the disposition of the letter also depends on the recipient of the letter, if the recipient is not present, it is quite difficult to carry out the disposition process. Based on the problems that occurred, a proposed system design is carried out to develop the system that is currently running. The following is a description of the proposed system.

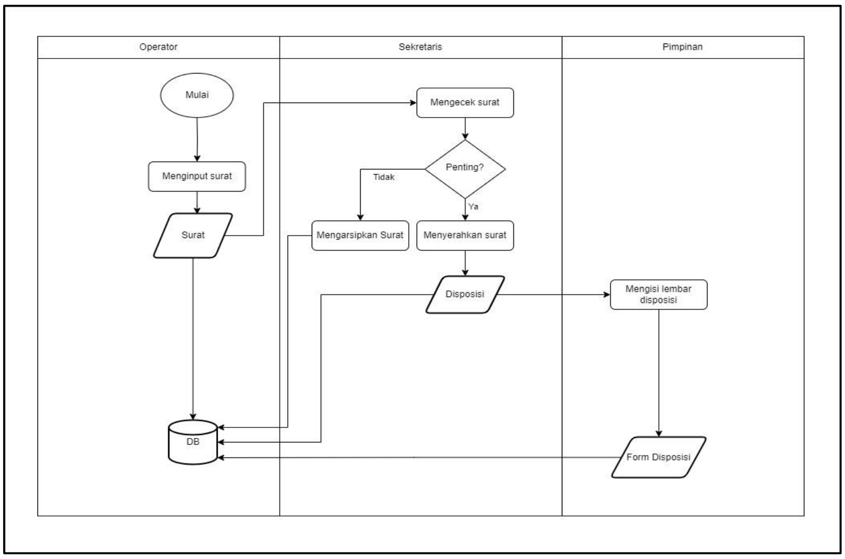


Figure 4. Proposed System Flow

3. RESULT AND DISCUSSION

3.1 System Design

At this stage, Unified Modeling Language modeling is used which involves requirements analysis to identify system requirements, system based on user requirements, identification of system users, database, and system flow. Unified modeling language is a technique for modeling systems consisting of use case diagrams, sequence diagrams, activity diagrams, and component diagrams.

a. Use Case

In this research, use case diagram is used with a total of four actors and a total of seven use cases. Use cases are used to describe the way a business system interacts with its environment [19]. Use cases are used to describe the basic functions of an information system [20]. In this research, the user as the primary actor generalizes between petinggi, operator and operator petinggi. Lastly, admin as secondary actor.

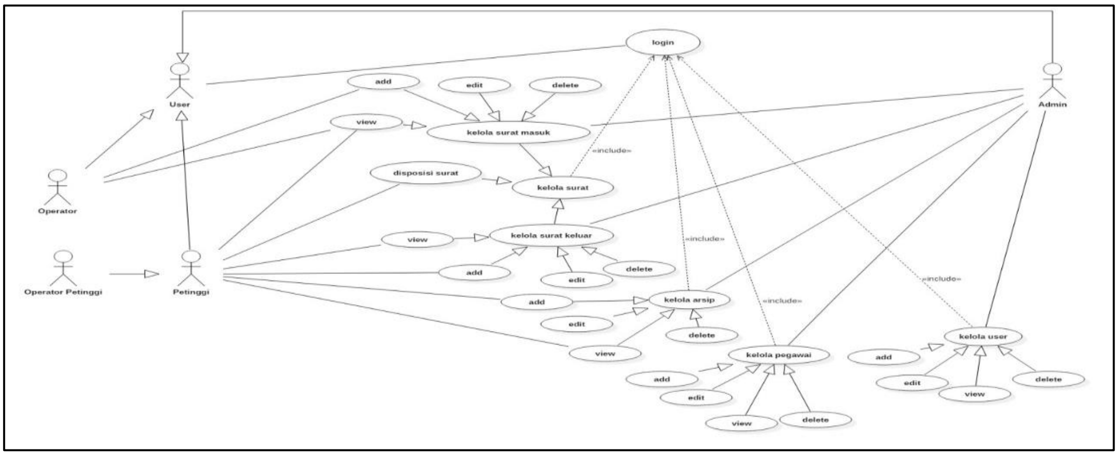


Figure 5. Use Case Diagram

As it can be seen in Figure 5 above there are a total of seven use cases, they are: login, manage incoming letter, letter disposition, manage outgoing letter, manage letter disposition, manage employee, and manage user.

1) Login use case, in this use case all users can log in to access the system.
2) Manage incoming letter, in this use case operator can view and add new letters. Petinggi or operator petinggi can only view the letters. While admin can do all the actions (view, add, edit, and delete letter).
3) Letter disposition, in this use case petinggi or operator petinggi can do letter disposition.
4) Manage outgoing letter, in this use case operator can view and add new letters. Petinggi or operator petinggi can only view the letters. While admin can do all the actions (view, add, edit, and delete letter).
5) Manage letter disposition, in this use case admin can do all the actions (view, add, edit, and delete letter disposition).
6) Manage employee, in this use case admin can do all the actions (view, add, edit, and delete employee).
7) Manage employee, in this use case admin can do all the actions (view, add, edit, and delete user).

b. Activity Diagram

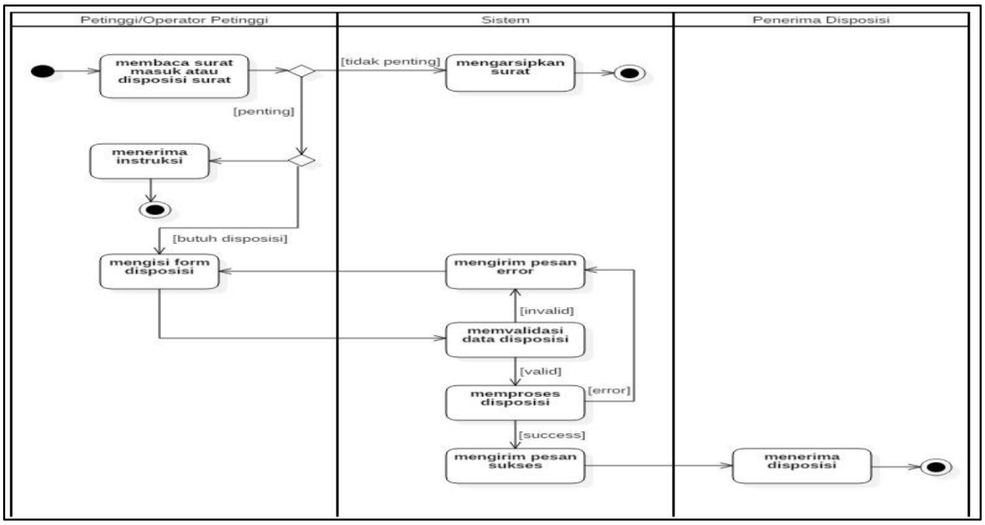


Figure 6. Activity Diagram

Figure 6 describes the letter disposition workflow which is initiated by petinggi or operator petinggi by looking at incoming letters. If the letter is important, it needs to be followed up to receive orders or fill out the disposition form. If you have filled out the disposition form, the system will validate the disposition data and process it until it sends a success message which will then be forwarded to the disposition recipient.

c. Sequence Diagram

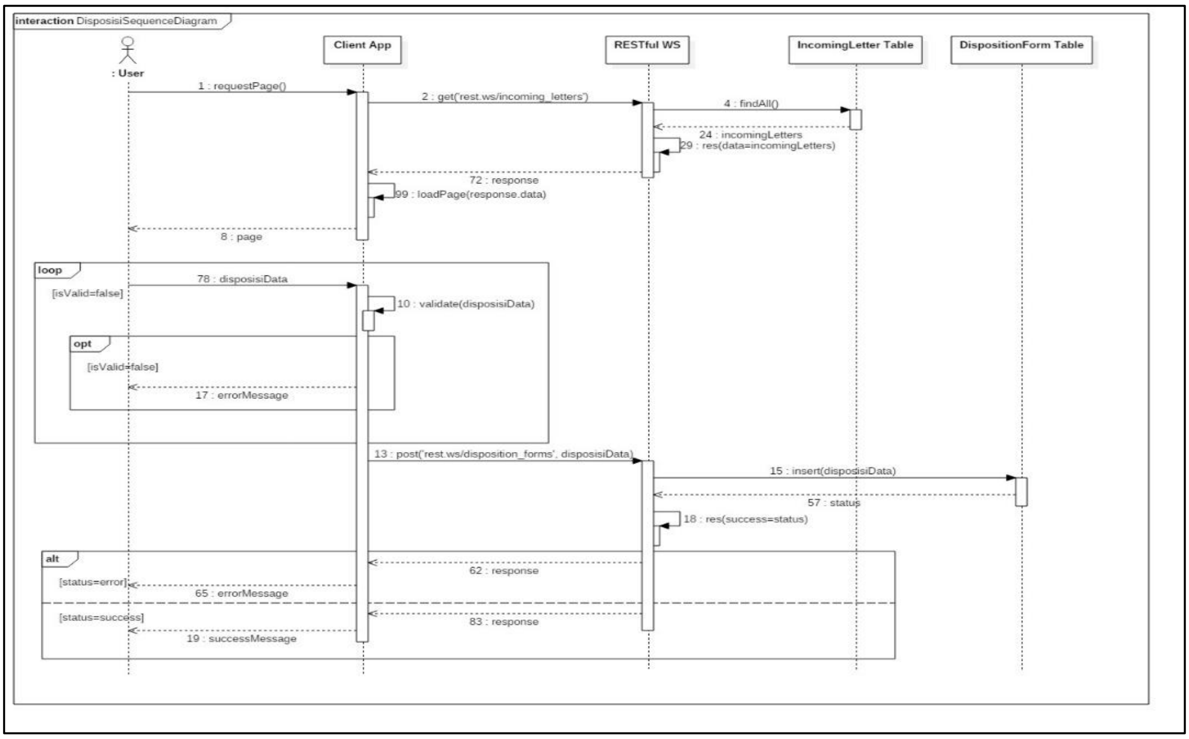


Figure 7. Sequence Diagram

Figure 7 describes the sequence diagram used to describe the relationship between actors and systems which presents how the user will request a letter disposition page with client app. Then, the user needs to fill out the disposition form. Lastly, if the form is valid, it will be stored in the database through RESTful web service.

d. Component Diagram

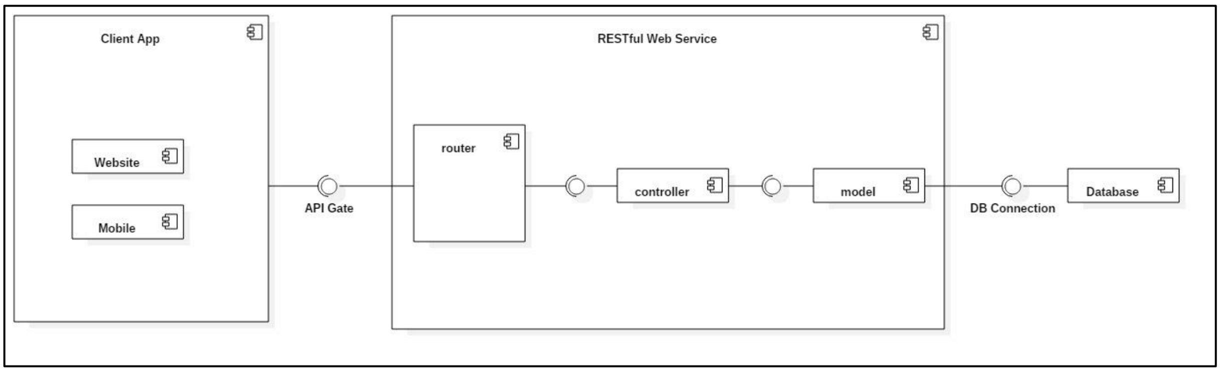


Figure 8. Component Diagram

Figure 8 describes the architecture of the system. As can be seen, there are three main components of the system, those are Client App, RESTful Web Service, and Database. Client App, can also be referred to as front-end, is an interface that users can interact with. RESTful Web Service acts as intermediary between Client App and Database. It provides interface to data layer that can be consumed by Client App. RESTful Web Service component also supports data integration across Client Apps. Lastly, the Database component is used as storage for system data.

3.2 RESTful Web Service Design

As stated in **Section I**, REST is a collection of constraints to be implemented. These constraints are as follows [6], [11]:

1) Resources are identified by a mechanism (in this article, URI schema is used).
2) Each resource has a representation.

3) All resources can be accessed by allowed methods and have the same semantics across resources.
4) Stateless interactions.
5) Idempotency and caching.
6) Layered system (while optional, it is recommended).

While these constraints are not mandatory, it is encouraged to be implemented as it brings benefits such as scalability and security [6]. In this case caching techniques are not used as this application needs real-time data. In this RESTful web service, there are thirteen resources in total. All resources are represented in JSON format. If allowed, user can perform CREATE, UPDATE, and DELETE operation on the resource by taking inputs. For security, endpoints are protected by performing authorization using API key and token using JWT token on every request. Without API key or token, requests will not be processed. Another thing to be noted is if a resource has a reference to another resource, it will give URI pointing to the referenced resource instead of the representation of the source itself (marked as **string***, for example: { user:’https://rest.api/users/1’ }). These resources are:

Table 1. Resource Design

Resource	Representation
Authentication	{ user: **string***}
User	{ id: number, username: string, userLevel: **string***, employee: **string***, createdAt: Date, updatedAt: Date | null }
Employee	{ id: number, name: string, nip: string | null, nik: string | null, section: string, position: string createdAt: Date, updatedAt: Date | null }
IncomingLetter	{ id: number, refNo: string, sender: string, about: string, date: Date, status: string, path: string | null, createdAt: Date, updatedAt: Date | null }
OutgoingLetter	{ id: number, refNo: string, to: string,about: string,date: Date, status: string, path: string | null, createdAt: Date, updatedAt: Date | null }
Disposition	{ id: number, incomingLetter: **string***, status: string, createdAt: Date, updatedAt: Date | null }
DispositionForm	{ id: number, from: **string***, to: **string***, notes: string | null, instruction: string, disposition: **string***, createdAt: Date, updatedAt: Date | null}
Status	{ id: number, name: string }
DispositionStatus	{ id: number, name: string }
Section	{ id: number, name: string }
UserLevel	{ id: number, level: string }
Instruction	{ id: number, name: string }

Each resource has an identifier using URI schema to access the resource. The URI schema for resources is as followed:

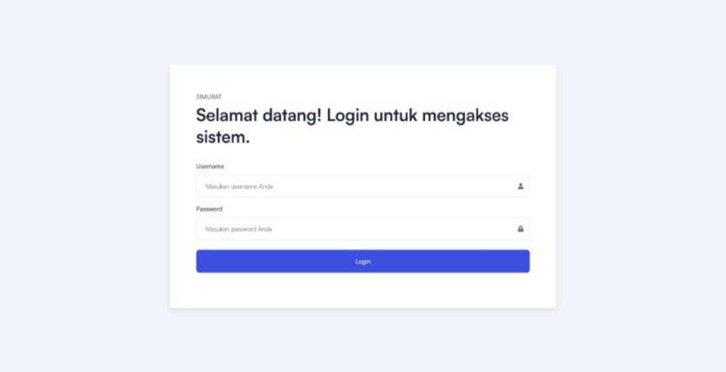
Table 2. URI Schema

URI	HTTP Method	Purpose
	Authentication	
/auth	POST	Setting accessToken and refreshToken cookies. Retrieve user that match given credentials.
/auth/token	POST	Authenticate token.
/auth/logout	POST	Removing accessToken and refreshToken cookies.
	User	
/users?api_key=value	GET	All users list.
/users/{id}?api_key=value	GET	Retrieve user by id.
/users	POST	Create a new user.
/users	PUT	Update a user by id.
/users	DELETE	Delete a user by id.
	Employee	
/employees?api_key=value	GET	All employees list.
/employees/{id}?api_key=value	GET	Retrieve employee by id.
/employees	POST	Create a new employee.

URI	HTTP Method	Purpose
/employees	PUT	Update an employee by id.
/employees	DELETE	Delete an employee by id.
	IncomingLetter	
/incoming_letters?api_key=value	GET	All incoming letters list.
/incoming_letters/{id}?api_key=value	GET	Retrieve incoming letter by id.
/incoming_letters	POST	Create a new incoming letter.
/incoming_letters	PUT	Update an incoming letter by id.
/incoming_letters	DELETE	Delete an incoming letter by id.
	OutgoingLetter	
/outgoing_letters?api_key=value	GET	All outgoing letters list.
/outgoing_letters/{id}?api_key=value	GET	Retrieve outgoing letter by id.
/outgoing_letters	POST	Create a new outgoing letter.
/outgoing_letters	PUT	Update an outgoing letter by id.
/outgoing_letters	DELETE	Delete an outgoing letter by id.
	Disposition	
/dispositions?api_key=value	GET	All dispositions list.
/dispositions/{id}?api_key=value	GET	Retrieve disposition by id.
/dispositions	POST	Create a new disposition.
/dispositions	PUT	Update a disposition by id.
/dispositions	DELETE	Delete a disposition by id.
	DispositionForm	
/disposition_forms?api_key=value	GET	All disposition forms list.
/disposition_forms/{id}?api_key=value	GET	Retrieve disposition form by id.
/disposition_forms	POST	Create a new disposition form.
/disposition_forms	PUT	Update a disposition form by id.
/disposition_forms	DELETE	Delete a disposition form by id.
	Status	
/status?api_key=value	GET	All status list.
	DispositionStatus	
/disposition_status?api_key=value	GET	All disposition status list.
	Section	
/sections?api_key=value	GET	All sections list.
	UserLevel	
/user_levels?api_key=value	GET	All user levels list.
	Instruction	
/instructions?api_key=value	GET	All instructions list.

3.3 System Implementation

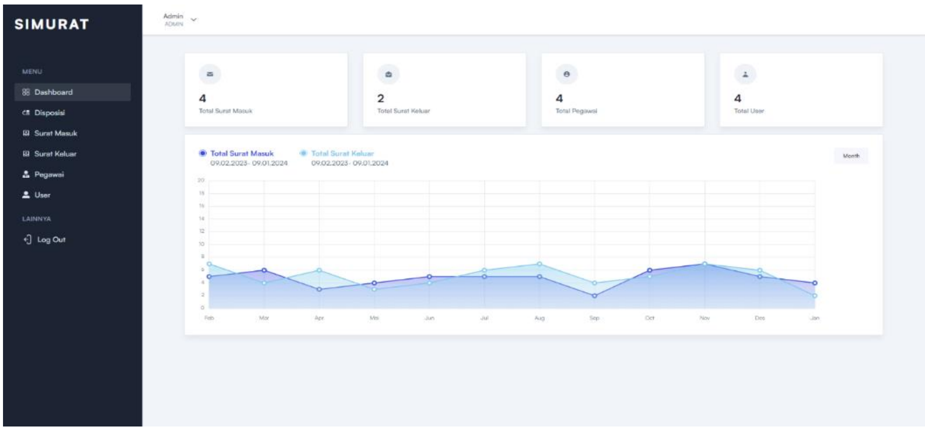
System implementation is the stage of implementing a system that has been designed into a form that can be used by users after going through testing.


SIMURAT

Selamat datang! Login untuk mengakses sistem.

Figure 9. Login Page

Figure 9 is the initial page that will be seen by the user, the user will be displayed with a login page when first accessing the system. On the login page, the user is required to enter a username and password, then if authenticated, the user login will be successful and receive authorization to access features according to the user's role.



4 Total Surat Masuk

2 Total Surat Keluar

4 Total Pegawai

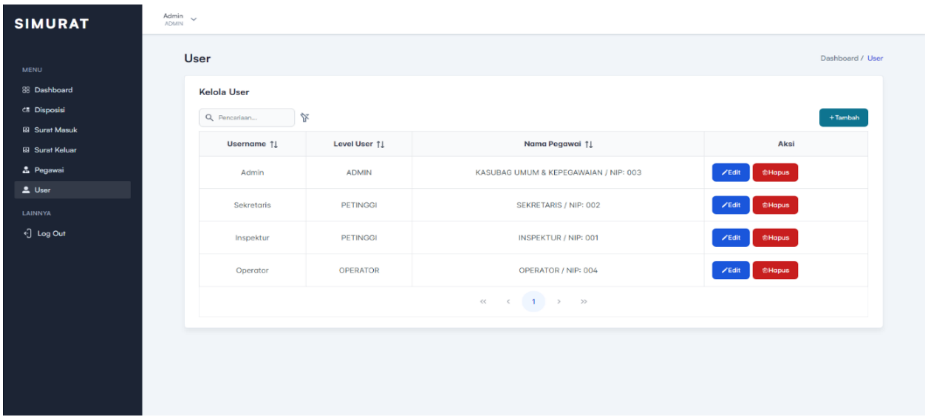
4 Total User

Total Surat Masuk 09.02.2023 - 09.01.2024

Total Surat Keluar 09.02.2023 - 09.01.2024

Figure 10. Dashboard Page

Figure 10 is the dashboard page when the user successfully logs in. The dashboard page contains statistics on incoming and outgoing letters in numbers and diagrams. Apart from that, the dashboard page contains the number of employees and users.



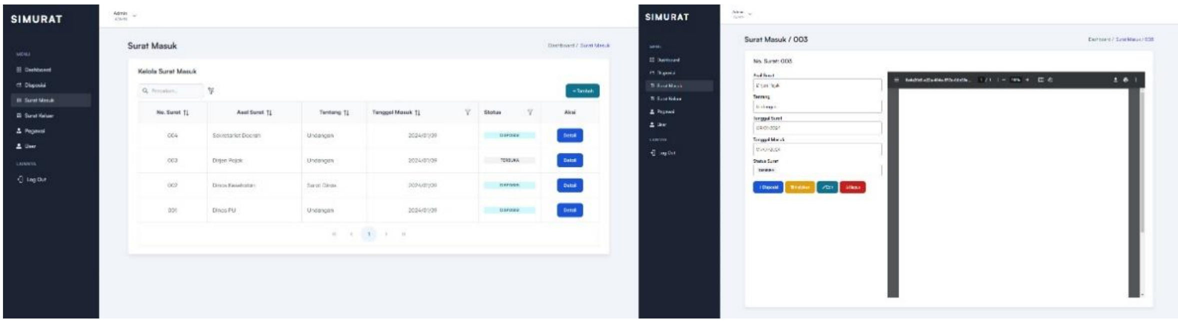
User

Kelola User

Username ⇅	Level User ⇅	Nama Pegawai ⇅	Aksi
Admin	ADMIN	KASUBAG UMUM & KEPEGAWAIAN / NIP: 003	Edit Hapus
Sekretaris	PETINGGI	SEKRETARIS / NIP: 002	Edit Hapus
Inspektur	PETINGGI	INSPEKTUR / NIP: 001	Edit Hapus
Operator	OPERATOR	OPERATOR / NIP: 004	Edit Hapus

Figure 11. User Page

Figure 11 shows the employee management page, on this page the admin can carry out the create, read, update, and delete (CRUD) process on user accounts. Apart from that, admins can also sort and search user data.



Surat Masuk

Kelola Surat Masuk

No. Surat ⇅	Asal Surat ⇅	Tentang ⇅	Tanggal Masuk ⇅	Status	Aksi
004	Sekretariat Daerah	Undangan	2024/01/09	DISPOSISI	Detail
003	Dirjen Pajak	Undangan	2024/01/09	TERBUKA	Detail
002	Dinas Kesehatan	Surat Dinas	2024/01/09	DISPOSISI	Detail
001	Dinas PU	Undangan	2024/01/09	DISPOSISI	Detail

Surat Masuk / 003

No. Surat: 003

Asal Surat

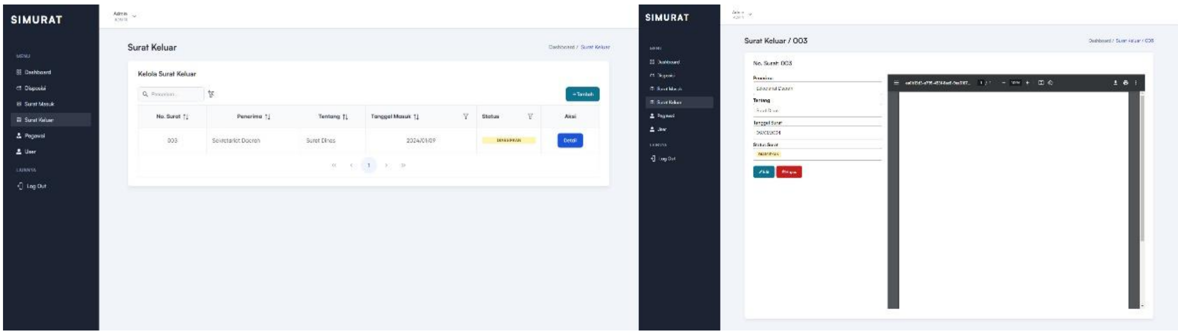
Tentang

Tanggal Surat

Tanggal Masuk

Status Surat

Figure 12. Incoming Letter Page



Surat Keluar

Kelola Surat Keluar

No. Surat ⇅	Penerima ⇅	Tentang ⇅	Tanggal Masuk ⇅	Status	Aksi
003	Sekretariat Daerah	Surat Dinas	2024/01/09	[illegible]	Detail

Surat Keluar / 003

No. Surat: 003

Penerima

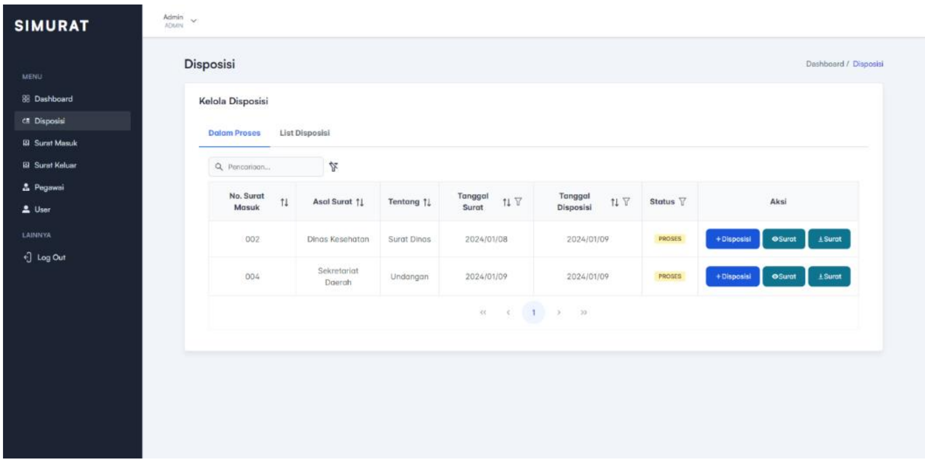
Tentang

Tanggal Surat

Status Surat

Figure 13. Outgoing Letter Page

Figure 12 and Figure 13 are the display of the incoming and outgoing mail pages. The admin can manage incoming and outgoing mail starting from create, read, update, and delete (CRUD).



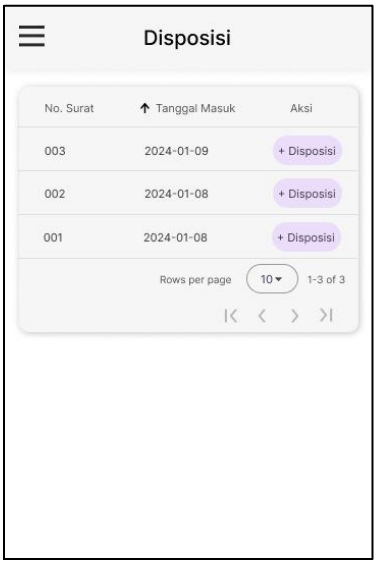
Disposisi

Kelola Disposisi

No. Surat Masuk	Asal Surat	Tentang	Tanggal Surat	Tanggal Disposisi	Status	Aksi
002	Dinas Kesehatan	Surat Dinas	2024/01/08	2024/01/09	PROSES	+Disposisi ⊙Surat ⤓Surat
004	Sekretariat Daerah	Undangan	2024/01/09	2024/01/09	PROSES	+Disposisi ⊙Surat ⤓Surat

Figure 14. Disposition Page

Figure 14 is a display of the disposition page. On this page users and admins can monitor the disposition status of letters, search, sort, and follow up on whether letters need disposition or not.



Disposisi

No. Surat	↑ Tanggal Masuk	Aksi
003	2024-01-09	+ Disposisi
002	2024-01-08	+ Disposisi
001	2024-01-08	+ Disposisi

Figure 15. Disposition Page on Android

Figure 15 is a display of the disposition page on android, on this page users and admins can monitor the disposition status of letter, search, sort by phone.

3.3 System Testing

To do system testing, the author uses the black box method with the aim of finding out whether the system has passed the test or not. Black-box testing is carried out with users who will use the system. The testing method using black-box testing is easy to implement because it only requires lower and upper limits of the expected data, and estimates can be calculated from the amount of test data for the number of input fields to be tested, the input rules that must be filled in, and the upper and lower limits that are filled in [21].

Table 3. User Black-box Testing

No.	Testing	Input	Test Result	Description
1	Login	username, password	Authenticated	Success
2	Kelola Surat Masuk	refNo, about, sender, date, file	Data Saved	Success
3	Kelola Surat Keluar	refNo, about, to, date, file	Data Saved	Success
4	Disposisi	letterId, instruction, to, notes	Data Saved	Success

The black-box test was carried out on the user page and can be seen in the table. The results of the test successfully indicated that there were no errors, and the system can ran smoothly according to the user's requirements.

Table 4. Admin Black-box Testing

No.	Testing	Input	Test Result	Description
1	Login	username, password	Authenticated	Success
2	Kelola Surat Masuk	refNo, about, sender, date, file	Data Saved	Success
3	Kelola Surat Keluar	refNo, about, to, date, file	Data Saved	Success
4	Disposisi	letterId, instruction, to, notes	Data Saved	Success
5	Kelola User	username, password, userLevelId	Data Saved	Success

As can be seen in the table above, it is known that the application functions and features on the admin page run well without any errors. This indicates that all functions and features are in accordance with the user's needs. The test results on the admin and user pages were successful without any errors or the need to do further improvements. The results in this study are in line with previous research which concluded that there is a correlation between CRUD and HTTP, besides that RESTful web service must be developed in accordance with REST restrictions to run well [14].The application that has been made also has high security while still paying attention to authorization, so that hijacking is minimal because the access rights of each user are different, this is in line and has the same results as previous research [9]. Based on the results of testing, it is known that the system has met user needs and is able to facilitate employee performance in line with previous research[1].

4. CONCLUSION

Implementing RESTful Web Service to Letter Management Application at Inspektorat Daerah Provinsi Sumatera Selatan helps the performance of agency employees and increases efficiency and effectiveness, especially in the field of letter disposition carried out by Sub Bagian Umum dan Kepegawaian. The letter disposition process can be well documented in a transparent system. Through the implementation of this Web Service, applications can be accessed on mobile devices and PCs without having to use third party applications. The implementation of this RESTful API increases security by using user authentication and authorization. The test results using black-box testing ran well without any errors or improvements to the application. The application is in accordance with user needs and requirements. Based on the results of the design and implementation of the system being built, the admin is expected to back up data periodically to minimize data loss. For further research, it is hoped that this system can be developed following future trends which continue to prioritize quality and increase efficiency.

ACKNOWLEDGMENT

Thank you to Inspektorat Daerah Provinsi Sumatera Selatan for giving the opportunity for researchers to carry out this research and also to the parties involved who have helped so that the research and article writing can be completed.

REFERENCES

[1] Ahmad syarifudin, A. P. Nanda, and S. Hartati, "Sistem Informasi Manajemen Surat Berbasis Website di STMIK Pringsewu," SAINS DAN INFORMATIKA : RESEARCH OF SCIENCE AND INFORMATIC, vol. 7, no. 1, pp. 17–22, Apr. 2021, doi: 10.22216/JSI.V7I1.340.

[2] I. Imasita and H. Hirman, "APLIKASI PENGELOLAAN DISPOSISI SURAT BERBASIS WEB PADA DINAS PEKERJAAN UMUM KOTA MAKASSAR," Jurnal Instek (Informatika Sains dan Teknologi), vol. 7, no. 1, p. 106, 2022.

[3] M. A. Londa, Y. A. Wee, and M. Radja, "Implementasi Sistem Informasi Monitoring Disposisi Surat Masuk dan Surat Keluar Berbasis Website," MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer, 2022, [Online]. Available: https://api.semanticscholar.org/CorpusID:248790953

[4] Kementerian Pendayagunaan Aparatur Negara dan Reformasi Birokrasi Indonesia, "Peraturan Menteri Pendayagunaan Aparatur Negara dan Reformasi Birokrasi Nomor 6 Tahun 2011." 2011

[5] R. J. Anthony, "Chapter 6 - Distributed Systems," in Systems Programming, R. J. Anthony, Ed., Boston: Morgan Kaufmann, 2016, pp. 383–474. doi: https://doi.org/10.1016/B978-0-12-800729-7.00006-6.

[6] A. Archip, C. M. Amarandei, P. C. Herghelegiu, C. Mironeanu, and E. Serban, "RESTful Web services-A question of standards," 2018 22nd International Conference on System Theory, Control and Computing, ICSTCC 2018 - Proceedings, pp. 677–682, Nov. 2018, doi: 10.1109/ICSTCC.2018.8540763.

[7] R. A. Bahlool and A. M. Zeki, "Comparative Study between Web Services Technologies: REST and WSDL," in 2019 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), 2019, pp. 1–4. doi: 10.1109/3ICT.2019.8910298.

[8] K. Sangsanit, W. Kurutach, and S. Phoomvuthisarn, "REST web service composition: A survey of automation and techniques," in 2018 International Conference on Information Networking (ICOIN), 2018, pp. 116–121. doi: 10.1109/ICOIN.2018.8343096.

[9] O. D. Arianto and Y. A. Susetyo, "PENERAPAN RESTFUL WEB SERVICE DENGAN FRAMEWORK LARAVEL UNTUK PEMBANGUNAN SISTEM INFORMASI MANAJEMEN SUMBER DAYA MANUSIA," JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika), vol. 7, no. 2, pp. 522–532, May 2022, doi: 10.29100/JIPI.V7I2.2870.

[10] D. Felício, J. Simão, and N. Datia, “RapiTest: Continuous Black-Box Testing of RESTful Web APIs,” Procedia Comput Sci, 2023, [Online]. Available: https://api.semanticscholar.org/CorpusID:257706162

[11] R. T. Fielding et al., “Reflections on the REST architectural style and ‘principled design of the modern web architecture’ (impact paper award),” Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering, 2017, [Online]. Available: https://api.semanticscholar.org/CorpusID:13213930

[12] A. Belkhir, M. Abdellatif, R. Tighilt, N. Moha, Y.-G. Guéhéneuc, and É. Beaudry, “An Observational Study on the State of REST API Uses in Android Mobile Applications,” 2019 IEEE/ACM 6th International Conference on Mobile Software Engineering and Systems (MOBILESoft), pp. 66–75, 2019, [Online]. Available: https://api.semanticscholar.org/CorpusID:198981360

[13] F. Halili and E. Ramadani, “Web Services: A Comparison of Soap and Rest Services,” Mathematical Models and Methods in Applied Sciences, vol. 12, p. 175, 2018, [Online]. Available: https://api.semanticscholar.org/CorpusID:56080712

[14] P. F. Tanaem, A. F. Wijaya, A. D. Manuputty, and G. N. Huwae, “Penerapan RESTFul Web Service Pada Disain Arsitektur Sistem Informasi Pada Perguruan Tinggi (Studi Kasus: STARS UKSW),” JASIEK (Jurnal Aplikasi Sains, Informasi, Elektronika dan Komputer), vol. 2, no. 1, pp. 11–20, 2020.

[15] T. M. E. Nathanael, “PENGEMBANGAN SISTEM INFORMASI PELATIHAN BERBASIS WEB MENGGUNAKAN TEKNOLOGI WEB SERVICE DAN FRAMEWORK LARAVEL,” Jurnal Techno Nusa Mandiri, 2019, [Online]. Available: https://api.semanticscholar.org/CorpusID:146046427

[16] I. B. G. Sarasvananda, I. G. K. A. G. Wiguna, and S. Styawati, “Pendekatan Metode Extreme Programming untuk Pengembangan Sistem Informasi Manajemen Surat Menyurat pada LPIK STIKI,” Jurnal Informatika Universitas Pamulang, vol. 6, no. 2, pp. 258–267, 2021, doi: 10.32493/INFORMATIKA.V6I2.9482.

[17] R. Naz and M. N. A. Khan, “Rapid applications development techniques: A critical review,” International Journal of Software Engineering and Its Applications, vol. 9, no. 11, pp. 163–176, 2015.

[18] K. E. Kendall et al., “Systems analysis and design,” p. 518, 2014, Accessed: Oct. 26, 2023. [Online]. Available: https://thuvienso.hoasen.edu.vn/handle/123456789/11193

[19] A. R. Dennis, B. Wixom, and D. P. Tegarden, “Systems Analysis and Design: An Object-Oriented Approach with UML,” 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:54115814

[20] F. Indriyani, Yunita, D. A. Muthia, A. Surniandari, and Sriyadi, Analisa Perancangan Sistem Informasi, 1st ed. Graha Ilu, 2019.

[21] A. C. Praniffa, A. Syahri, F. Sandes, U. Fariha, Q. A. Giansyah, and M. Hamzah, “Pengujian Sistem Informasi Parkir Berbasis Web Pada UIN SUSKA RIAU Menggunakan White Box dan Black Box Testing,” Jurnal Testing dan Implementasi Sistem Informasi, vol. 1, no. 1, pp. 1–16, Mar. 2023, [Online]. Available: https://www.journal.al-matani.com/index.php/jtisi/article/view/321