UTM UNIVERSITI TEKNOLOGI MALAYSIA
INTERNATIONAL JOURNAL OF INNOVATIVE COMPUTING
ISSN 2180-4370
Journal Homepage : https://ijic.utm.my/

Digital Letter Archive and Signing System for Higher Education Administration

Muhammad Rofiqurrahman Ibnu Disya Ghazali
Faculty of Computing
Universiti Teknologi Malaysia
Skudai, Malaysia
rofiqurrahman@graduate.utm.my

Masitah Ghazali
Faculty of Computing
Universiti Teknologi Malaysia
Skudai, Malaysia
masitah@utm.my

The purpose of this system was to solve some issues experienced by the academic staff of Pamulang University and in turn affecting its students. These issues being the tediousness of collecting, reviewing, and signing student submitted letters and the transmission of digital letters between academic staff in a centralized and organized manner. This system presents a web-based solution that is designed to provide a platform for which students may submit letters intended for academic staff to approve via the automatic embedding of their digital signature onto the letter, and in addition, functionality for transmitting and receiving other letters of importance between academic staff. This solution aims to alleviate the pain of finding and organizing letters transmitted between academic staff, and eliminate the effort of personally contacting certain academic staff on part of the student to acquire their signature of approval for a specific letter.

Keywords — web application, digital letter, digital signature, higher education

I. INTRODUCTION

Letters are widely used for a variety of purposes in Indonesian higher education institutions. Some of the more common letters seen in institutions are letters of recommendation for example, requested by a student and requiring the acknowledgement of an educator such as a professor of their university. In cases such as these, the student would usually contact the appropriate lecturer or dean, sometimes with a letter prepared which needs their signature or simply ask for a letter to be made for a specified purpose. In other cases, letters are used to convey official information between academic staff in which case emails are usually the preferred method of transfer for these letters.

It is apparent that the process of creating, sending, and receiving letters of importance can be optimized using software. This could allow for a level of standardization of letters circulated within the educational institution and make categorization and archiving simpler from the perspective of academic staff and the institution, while also reducing direct communication between academic staff and students, as well as academic staff to other academic staff, improving the volume and concentration of useful information within the institution.

The aim of this project was to develop a responsive web system for students and academic staff as an interface for sending, receiving, signing, and archiving digital letters of importance using the MERN (MongoDB, Express.js, React.js, Node.js) stack to be employed in higher education institutions.

II. BACKGROUND AND RELATED WORKS

Currently there is no proper system that meets the requirements of the stakeholder due to the need to satisfy some unique use cases. The stakeholder however has highlighted two existing systems, based on their personal experience of use, that possess some parts of the capabilities the DLASS aims to have.

One such system is a letter management system used by Indonesia's national nuclear agency now part of Indonesia's agency for research and innovation called SITP or "Sistem Informasi Tata Persuratan. [1]" It is a website used internally by staff of the agency to send and receive letters of importance from other members of the agency. Letters are passed through the system from one user to another through a set chain of positions within the agency in a top-down manner where positions of management are sent copies of the letter for administration purposes during the process before reaching the destination of the intended receiver of the letter.

The other system is the digital signature module of the Intra BRIN [2] system used by Indonesia's national research and innovation agency. The Intra BRIN system is an internal information system to serve the operational needs and business of the agency. It includes a variety of features for members of the agency to use but the digital signature module specifically is of interest to the stakeholder and this project.

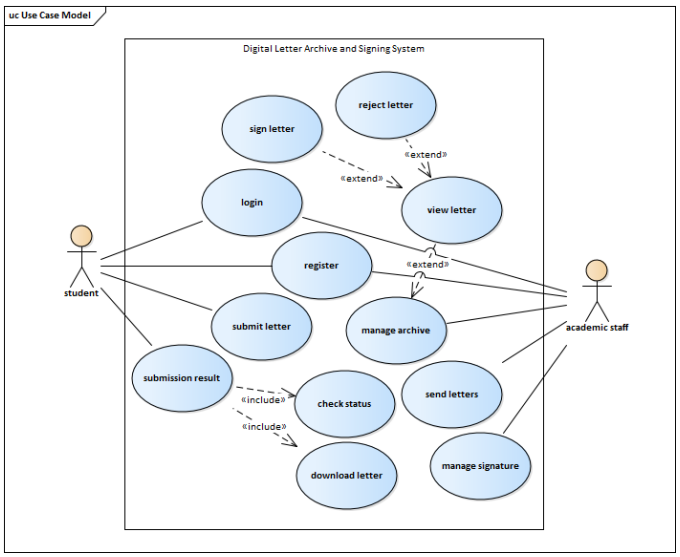
In short, a careful combination of features belonging to both highlighted systems are essential in the final product. Features from the highlighted systems will be considered based on the two core functionalities the stakeholder has conveyed which are the transmission of documents as demonstrated by the SITP, and electronic signing as seen in Intra BRIN.

III. METHODOLOGY

This section describes the intended method of development of the proposed system. The method chosen to develop the system was chosen based on the given time constraints for development and the architecture of the proposed system. Thus, the waterfall model was used which consists of a sequence of tasks that must be completed sequentially as phases with the entry criteria of each phase being a set of deliverables from the previous phase [3, 4].

A. Requirements Analysis

The first phase is requirements analysis and definition. In this phase, requirements are elicited from the stakeholders of the project. The stakeholders provide context for the basis to construct requirements which in the case of this project is elicited through multiple oral discussions with the stakeholder. The set of requirements are then understood and checked for testability, resulting in the production of the requirements specification document.

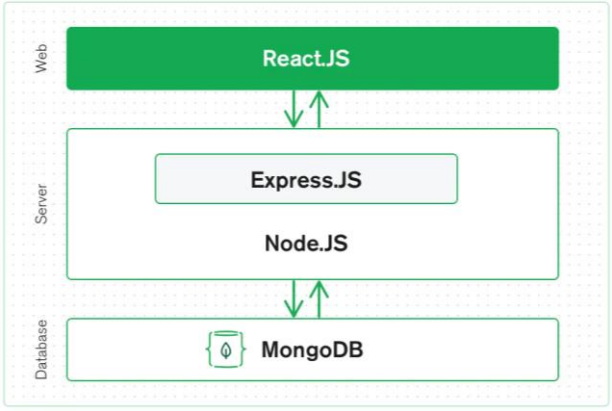


B. Fig. 1. Use case diagram of the system

C. System and Software Design

The second phase is system and software design. In this phase, a design of the final product is created based on the requirements elicited from the requirements specification document. The design of the final product is created using software modeling tools, namely Enterprise Architect for drawing diagrams such as class, package, component, sequence, activity, and use case diagrams which are documented in the software design document.

In order to develop the system, ReactJS, NodeJS, ExpressJS, and MongoDB, otherwise commonly known as the MERN stack were selected as the core technologies to build the web application, thus the system would have a three-layered architecture consisting of the web layer, server layer, and database layer [5]. The database consists of three entities being the user, letter, and signature.



D. Fig. 2. MERN architecture diagram [5]

E. Implementation and Unit Testing

Following the designs created from the system and software design phase of the project's development, the web application was built using software development tools. Visual Studio Code was the code editor that was used to write the entirety of the application [6]. The system was built as two repositories, namely the front-end code repository and the back-end code repository, both stored in GitHub to allow version control [7].

As explained in the system's architecture, the front end of the application was built using ReactJS. The core of the system's front end is made up of all the components of the system that represent the client views which are organized into a features folder that consists of the auth, users, submission, letters, and instructions features containing their associated views. These components are what the users of the system can interact with in order to operate the system.

The back end of the application was built as an API or Application Programming Interface. The back-end functions as an intermediary between what the users can see in the views and the resources stored in the database. It is primarily responsible for controlling what data is taken and served to the front end and the MongoDB database, which can be seen in files such as the lettersController, or usersController, among others. In addition to this, the back end is where database models are defined in the form of Mongoose schemas, which are seen in the User, Letter, and Signature models.

Throughout the programming of the web application, unit testing and integration testing was done in the localhost environment. Testing of the system's back end was done using Postman because it is an API. After the whole application is completely written, deployment was done using Render.com since it offers free hosting [8]. There are two deployments for this web application, namely a static site deployment for the front-end repository, and a web service deployment for the

back-end repository. The results of which is a web application accessible on https://unpamhive.onrender.com.

In this phase, the system itself is built by implementing the specified designs of all modules chosen for implementation from the previous phase through software code. Unit testing is also conducted where each unit of the system is tested for its functionality. This phase produces the system's source codes and databases.

F. Integration and System Testing

The fourth phase is integration and system testing. In this phase, the units produced during the previous phase are integrated into their subsystems and then the main system. A software testing document is produced which outlines the testing activities to be performed during this phase onto the integrated code to check if the system fulfills the specified requirements.

G. Operation and Maintenance

The last phase is the operation and maintenance phase. In this phase, the tested system is deployed onto the customer environment and released into the market. Once deployment is done and the product is used by the intended users, further issues may be encountered across time which must be fixed and therefore maintenance must be conducted. Possible improvements may also be discovered in which case, if implemented, the code must be updated.

IV. DIGITAL LETTER ARCHIVE AND SIGNING SYSTEM

The interfaces of the system are grouped into five feature categories which are organized in the features folder in the source code of the front end repository. The features associated with the interfaces being authentication, as found in the auth directory, regular letters accessible by academic staff only as found in the letters directory, student submissions as found in the submissions directory, instruction letters as found in the instructions directory, and digital signature control as found in the signatures directory.

The system has user interfaces for logging in users, registering users, and the dashboard page which are organized into the auth directory. User interfaces for creating, editing, and viewing regular letters individually, as well as viewing them in a list are organized into the letters directory. Similar to the letters directory, interfaces for creating, editing, and viewing student submissions, and instruction letters are organized into their respective submissions, and instructions folder. Lastly, the interface for creating and deleting an academic staff's signature is organized in the signatures directory. These user interfaces are core to the system's operation (see Fig. 3 to 5).

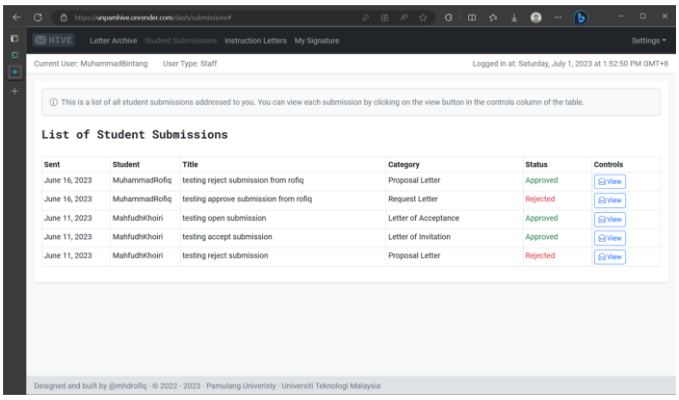


Fig. 3. List of student submissions page

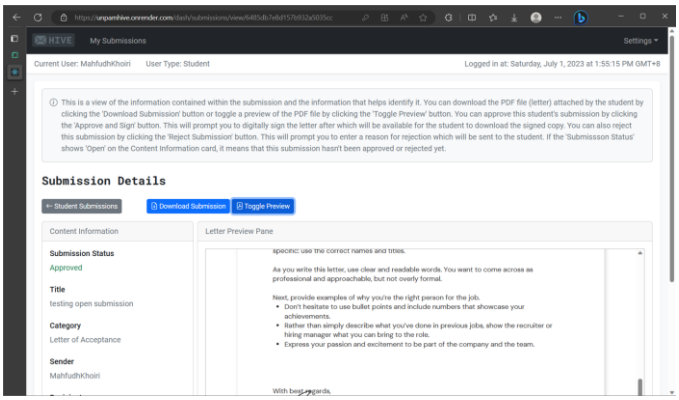


Fig. 4. Submission details page

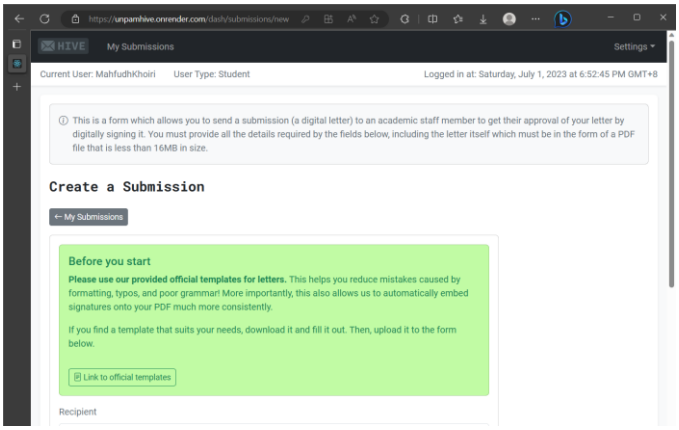


Fig. 5. Create a submission page

V. USABILITY TESTING AND RESULTS

Testing of the application with real users is conducted through a usability test where the application is measured on its level of usability to its users [9, 10]. Participants of the test were Dr Syaiful as the stakeholder of the project and an academic staff of Pamulang University, performing the test as an academic staff actor, and Diandra Nayyara, an undergraduate student English language major, performing as a student actor. Participants were given tasks to accomplish through the system and no directions or procedures. The time taken to complete

each task, level of satisfaction from 1 to 5 with 5 as the highest level of satisfaction, and notable issues were recorded during the test, presented in table 1 and table 2.

TABLE I. RESULTS OF USABILITY TESTING WITH STUDENT USER TYPE

Student User				
No.	***Given Task***	***Time (s)***	***Satisfaction***	***Issues***
1	Account registration	82	5	None
2	Login	20	5	None
3	Create submission	44	5	None
4	View submission	23	5	None
5	Check submission status	6	5	None
6	Edit submission	44	5	None
7	Download submission file	12	5	None
8	Delete submission	57	5	None
9	logout	5	5	None

TABLE II. RESULTS OF USABILITY TESTING WITH ACADEMIC STAFF USER TYPE

Academic Staff User				
No.	***Given Task***	***Time (s)***	***Satisfaction***	***Issues***
1	Account registration	105	5	None
2	Login	60	5	None
3	Create letter	93	5	None
4	View letter	44	5	None
5	Edit letter	60	5	None
6	Delete letter	18	5	None
7	Download letter	5	5	None
8	View student submission	10	5	None
9	Download student submission file	6	5	None
10	Reject student submission	26	5	None
11	Approve student submission	27	5	None
12	Add signature	20	5	None
13	Delete signature	8	5	None
14	Create instruction	70	5	None
15	View instruction	20	5	None
16	Download instruction file	10	5	None
17	Edit instruction	35	5	None
18	Delete instruction	5	5	None

Based on the results of the usability test, the web application is highly usable from the perspective of a student using the system as seen in high levels of satisfaction and an average task completion time of 32.5 seconds. For the academic staff, the average task completion time was 34.5 seconds, with high levels of satisfaction for all tasks. There were no issues with how any of the given tasks were executed and information was clear enough that the participants could understand what they were doing without additional information provided by the observer during the test.

VI. CONCLUSION

In conclusion, all the objectives of the project have been achieved. Beginning with the elicitation of requirements for the digital letter archive from the stakeholder of the project, Dr Syaiful which resulted in the creation of a set of use cases and the requirements of the project which is presented in a software requirements specification. The design of the system has been produced resulting in the system's domain model, package diagrams, class diagrams, and sequence diagrams for each module in addition to the entity relationship diagram and the initial user interface designs which is presented in a software design document. Then using the designs produced, the web application was built by implementing the MERN stack and deployed on Render.com. Finally, a usability test was conducted with real users from Pamulang University which measured how useful the user experience and interface is to its users.

VII. ACKNOWLEDGMENT

I would like to express my gratitude to my supervisor, Assoc Prof. Dr. Masitah Binti Ghazali, for her understanding, support, and feedback throughout the process of writing my thesis and supporting documents. It was a pleasure and great privilege to have had the chance to study under her guidance.

VIII. REFERENCES

[1] BATAN (2022). Sistem Informasi Tata Persuratan. https://sitp.batan.go.id/login.php. Available online 19 May, 2022.
[2] BRIN (2022). Intra BRIN. https://intra.brin.go.id/home. Available online 19 May, 2022.
[3] Sommerville I. Software Engineering. 9th Edition. Pearson. 2011.
[4] Tutorialspoint (2022). SDLC - Waterfall Model. https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.html. Available online 19 May 2022.
[5] MongoDB (2022), Mern Stack Explained. https://www.mongodb.com/mern-stack. Available online 19 May, 2022.
[6] Visual Studio Code (2022). https://code.visualstudio.com/. Available online 19 May, 2022.
[7] GitHub (2022), pdf.js. https://github.com/mozilla/pdf.js/. Available online 19 May 2022.
[8] Render (2023). https://render.com/about. Available online 30 June 2023.
[9] Nielsen Norman Group (2019), Usability Testing 101. https://www.nngroup.com/articles/usability-testing-101/. Available online 30 June 2023.
[10] Nielsen, J. (1993). Usability Engineering. Morgan Kaufmann.