Development of a Mail Management System (SMART) as a Solution to the Mail Archive Problem at the Malang Customs Office

Muhammad Iqbal Akbar[1]*✉, Annisa Putri Ayudhitama[1], Azhar Ahmad Smaragdina[1], Dila Umnia Soraya[1]

[1]Electrical and Informatics Engineering, Universitas Negeri Malang, Malang, Indonesia
Corresponding Author: iqbal.akbar.ft@um.ac.id

Article Information	ABSTRACT
	The study aims to create a digital mail management system, known as SMART, to address the inefficiencies of the current manual letter archiving system at the Malang Customs and Excise Supervision and Service Office. The existing system, which relies on Microsoft Excel, is time-consuming and prone to errors. This leads to difficulties in searching for specific letter scans and the potential loss of hard-file archives. The SMART system, designed using the Waterfall Method and the CodeIgniter framework, is a web-based information system that allows the management of incoming, outgoing, and internal letters in a digital format. The system includes features such as inserting, updating, and deleting letters, as well as uploading scanned letter files. According to the System Usability Scale (SUS) test results, the SMART application is highly usable, with an average score of 93.50, which exceeds the benchmark of 68. The implementation of SMART is expected to significantly improve the efficiency of the administration section's document archiving process, enhancing the accessibility and management of letter archives at the Malang Customs and Excise Supervision and Service Office.

How to Cite:

Akbar, M. I., et al. (2024). Development of a Mail Management System (SMART) as a Solution to the Mail Archive Problem at the Malang Customs Office. Jurnal Teknologi Informasi Dan Pendidikan, 17(2), 568-583. https://doi.org/10.24036/jtip.v17i2.868

1. INTRODUCTION

The Directorate General of Customs and Excise is authorized to conduct surveillance of export and import goods without disrupting the smooth process of tourists coming and

is authorized to take action if there are narcotics. In its application, the technical agency establishes prohibition and restriction regulations and notifies the Minister of Finance of the Republic of Indonesia to be implemented by the Directorate General of Customs and Excise in the field of special supervision, of course, must be given to the entry of prohibited goods that can disrupt the life of the nation and state [1-2].

Pengawasan dan Pelayanan Bea dan Cukai Tipe Madya Cukai (KPPBC TMC) Malang (Malang Customs and Excise Supervision and Service) is a type A3 customs and excise office established by the decision of the Director General of Customs and Excise Number KEP-46/BC/2008 on July 14, 2008. KPPBC TMC Malang is located in a very strategic location, namely in the middle of the city, precisely on Jalan Surabaya No. 2 Malang. KPPBC TMC Malang is under and responsible to the Head of Regional Office. KPPBC TMC Malang has the task of "Carrying out customs and excise services and supervision in the area and its authority based on applicable laws and regulations". This has been regulated in the Decree of the Minister of Finance of the Republic of Indonesia Number: 74 / PMK.01 / 2019 dated April 8, 2009 [3-4].

At the Malang Customs and Excise Supervision and Service Office, there is a general subdivision in the field of administration and employment that handles various kinds of letters and employment problems. in this subdivision, there are many problems that occur, especially in letters, namely recording every incoming, outgoing and delivery letter is still recorded in Microsoft Excel. Incoming and outgoing letters require a scanning process to save them as digital archives, where the digital archive is entered into Microsoft Excel in the form of a link to the scanned image of the letter. According to employees, this is not effective because it takes quite a long time to create links from these letters, while per day the general subdivision receives and issues many letters so that the letter archiving process no longer uses links and if employees want to find the letter archive they must search through the scan storage file of each letter. The problem occurs again if employees forget to scan and only store the letter archive in hard-file form. If the employee wants to find the desired letter, the employee must search in a pile of hard-files of the letter archive. This results in the need for quite a long time and is very inefficient in finding letter archives, moreover it is also possible if the hard-file archive is lost.

Research conducted by Syukhri investigated the challenges faced by the Administration department of SMA PGRI 1 Padang in managing a large volume of incoming and outgoing mail using a traditional paper-based system. The manual search process was found to be inefficient and prone to errors. In response, a web-based Archival Management Information System was developed using the CodeIgniter framework. This system is intended to assist the Administration section, Head of Administration, and Principal in the search and management of mail. The Waterfall development methodology was used for system construction, incorporating features such as user administration, data input and modification, deletion, search capabilities, and report generation. The

.

implementation of this application is expected to enhance the efficiency of the Administration section's document archiving process [5].

Research conducted by Siti Marlina and Made Leo Radhitya identified problems in the management of correspondence at the Batuyang Village Office, where letters stored in agenda books are at risk of being lost if not properly archived. The process of making letter dispositions took a long time due to dependence on the village head. To overcome this problem, they developed a web-based correspondence administration information system that enables efficient management of incoming and outgoing letters. The system provides officers with up-to-date data, facilitates printing of disposition sheets, and generates mail-related reports. Blackbox testing results show that all system features function properly, signalling the successful implementation of the system in improving the efficiency of correspondence administration at the village office [6].

Research carried out by Surya Guntur Guntur et al. revealed the importance of implementing information systems in modernising administration in the Beringin Helvetia sub-district urban village office, which currently still uses conventional methods for managing letter archives. This method often causes problems such as loss and damage to documents. This research aims to develop a web-based application that can record and store archives in a database that can be accessed at any time. The results show that this application provides significant benefits to the agency, including more structured and neat archive data management, as well as improving overall administrative efficiency [7].

Rikiti's research looked into the data management practices of the Sinergi Atap Negeri Foundation, a social organization that helps patients and their families. The current paper-based system is managed by a single administrator and lacks security protocols, which hinders efficiency and data integrity. As a solution, the study suggests implementing a web-based data management information system. The system was developed using the waterfall methodology and built with PHP, the CodeIgniter 3 framework, and a MySQL database. Its aim is to streamline data collection, patient submissions, and overall data management for the foundation. This application is expected to enhance staff performance by enabling faster and more efficient data entry, ensuring data security, and reducing the risk of errors [8].

Based on the findings from these studies, the recommended solution to the problem is to establish an official letters system called SMART (Sistem Manajemen Surat). SMART is designed to manage incoming, outgoing, and internal letters in a digital format. The aim is to replace the current manual letter archiving system that relies on Microsoft Excel. SMART will offer features such as inserting, updating, and deleting letters, as well as uploading scanned letter files, which is a crucial function. One key advantage of this system is that each newly archived letter will include an uploaded scan file. This will allow employees to search for specific letter scans by using the letter number. This approach is expected to effectively address letters issues within the organization, making it easier for employees to

.

access and view letter archives in digital format, rather than relying on physical files. SMART will be developed with Waterfall Method using the CodeIgniter framework, which is anticipated to simplify the creation of this web-based information system [5].

2. RESEARCH METHOD

This research uses the Waterfall development model. In traditional software development methodologies like the waterfall model, requirements engineering is essential in the early stages of the development life-cycle. It involves systematically gathering, analysing, validating, and managing software requirements. In the waterfall model, process activities are performed in a sequence of separate steps, with each step preferably finished before the next one begins [9]. This approach requires detailed software planning upfront and results in a significant amount of documentation, as each project stage must be completed before moving on to the next. The traditional approach starts with gathering and documenting a "complete" set of requirements, followed by architectural and high-level design, development, testing, and maintenance [10]. Figure 1 illustrates the development phases in the waterfall model.
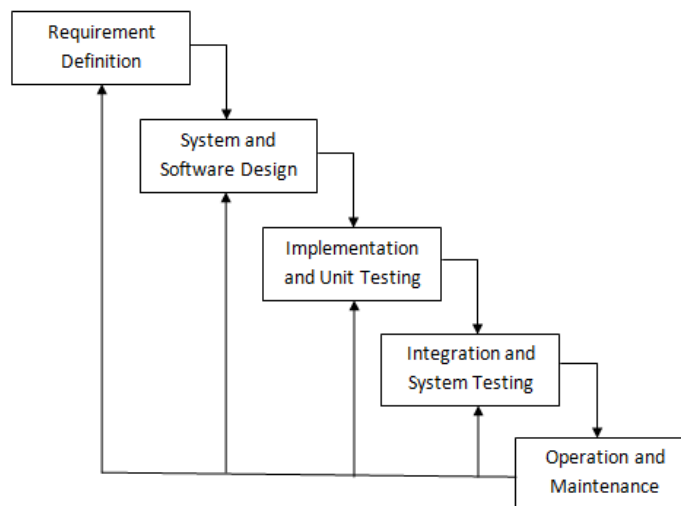

Figure 1. Waterfall Model

2.1. Requirement Definition

The first stage of the waterfall method involves analysing system requirements, which includes gathering data through research such as interviews or literature studies. Typically, a system analyst is responsible for this stage and conducts interviews and literature studies to gather information from users about their system requirements. The outcome of this stage is a user requirement document containing information about the

.

users' needs for the system, which will be translated into a programming language [11]. The requirements document for the development of SMART uses the Unified Modeling Language (UML), where the diagram used to describe the functions that will be implemented in SMART is the Use Case Diagram.

In making the SMART (Mail Management System) system, the first thing to do is to collect the required mail data. Starting from incoming mail data, outgoing mail and mail delivery. Furthermore, collecting data on letters in the general subdivision also requires interview activities. This activity is carried out to discover all types of letters used in the general subdivision. The interview was conducted with a resource person who already understood all kinds of letters in the general subdivision, namely the head of the General Subdivision of KPPBC TMC Malang.

2.2. System and Software Design

The second stage in this waterfall method is system design, where system design is a description that serves as a translation of system design before implementing the system with coding. The usual system design includes data structure design, system architecture, interface representation, and algorithm details. The results of the system design stage are in the form of software requirements documents, software requirements documents used by programmers in making the system [12].

2.3. Implementation and Unit Testing

The third stage in the waterfall method is coding and testing. This is the stage where the system's design is translated into a computer language to create the system the user needs. Coding is the process of implementing the design, while testing aims to find and fix errors in the system.

2.4. Integration and System Testing

The fourth stage in the waterfall method is implementation and testing, which is the final stage in working on the system. During this stage, the system is directly tested by its users. Users will use the system as intended and test it. If the system functions as intended and has no errors, then users can fully utilize the system.

To test this system, will use the System Usability Scale (SUS). SUS is a 10-item measurement tool used to assess users' perceptions of a product's usability. John Brooke developed this tool in the 1980s as a fast and freely applicable measurement method. The 10 question items used as a SUS measurement tool can be seen in Table 1. One of the advantages of SUS is the ability to be implemented quickly, has proven psychometric validity, and can be used in various domains and populations [13].

.

.

Usability is defined as a product's ability to fulfil a particular user's needs with effectiveness, efficiency, and satisfaction in the context of its use. A system with a high usability level should be easy to learn, efficient, and memorable. In addition, the system must also match the expectations and needs of users in terms of its physical and technical features. Companies' awareness of the importance of usability is increasing, and they are trying to design and develop more usable applications.

System Usability Scale (SUS) with adjective scale rating is one of the most famous and easy-to-use questionnaires to measure the level of usability of any product [14]. The adjective scale rating can be measured using the diagram in Figure 2 [15].

Table 1. Holden's Simplified System Usability Scale [16-17]

No	Scale
1	I think I would like to use this system frequently.
2	I found the system unnecessarily complex.
3	I thought the system was easy to use.
5	I found the various functions in this system were well integrated.
6	I thought there was too much inconsistency in this system.
7	I imagine that most people would be able to use this system without difficulty.
8	I found the system quite frustrating to use.
9	I felt confident using the system.
10	I needed to learn a lot of things before I could get going with this system.
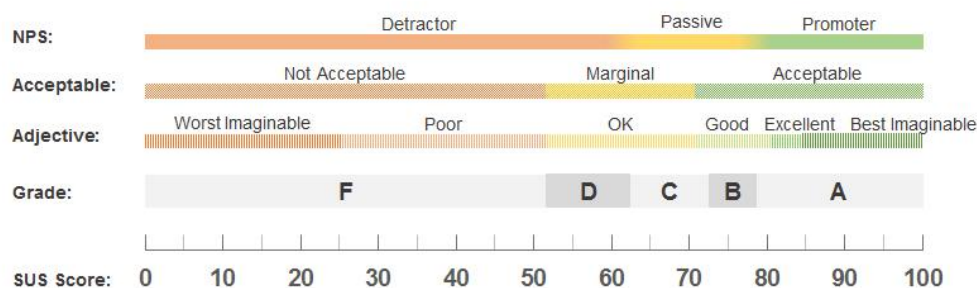
Figure 2. Grade, Adjective, Acceptable, and NPS associated with SUS scores

Table 1 shows the 10 standard questions used in SUS testing. SUS on SMART will be tested on 5 users. These users are the head of the general subdivision and administrative and personnel affairs employees, especially those who take care of the letters and archives section of the Malang Customs and Excise Supervision and Service Office. The value of the respondent's answer will be measured using a Likert scale. The Likert scale can be seen in table 2.

.

.

Table 2. Likert Scale

Skor	Description
1	Strongly disagree
2	disagree
3	Neutral
4	Agree
5	strongly agree

The Likert scale is a tool used to measure a person's attitude, opinion, or perception of a specific subject. It consists of five answer options, ranging from "strongly disagree" to "strongly agree".

On the Likert Scale, a "strongly disagree" response indicates that the respondent strongly rejects the given statement. A "disagree" response indicates the respondent's disagreement with the statement. A "neutral" response indicates that the respondent has no clear preference towards the statement. An "agree" response indicates that the respondent agrees with the statement. A "strongly agree" response indicates that the respondent strongly agrees with the statement [17].

In quantitative research, Likert scales are commonly used to measure various aspects such as customer satisfaction, public opinion on a policy, or employee perceptions of a company. After collecting individual scores from the System Usability Scale (SUS), the next step is to calculate the average of all respondents' scores. The minimum average SUS score that indicates an acceptable level of system usability is 68, which corresponds to a grade of C according to the Suaro-Lewis rating scale. When the application meets these criteria, it can be considered good and suitable for use by users [18]. You can find the Suaro-Lewis rating scale in Table 3.

Table 3. Suaro-Lewis curve scoring scale [18]

SUS Score Range	Grade
84.1 – 100	*A+*
80.8 84.0	*A*
78.9 – 80.7	*A-*
77.2 – 78.8	*B+*
74.1–77.1	*B*
72.6–74.0	*B-*
71.1–72.5	*C+*
65.0–71.0	*C*
62.7–64.9	*C-*

.

| 51.7–62.6 | *D* |
| 0.0–51.6 | *F* |

After potential users complete the questionnaire, we will determine the overall System Usability Scale (SUS) score by averaging the SUS scores of each respondent. To calculate the final SUS score, first, we subtract 1 from the score for odd-numbered questions (Q1, Q3, Q5, Q7, and Q9) and then subtract 5 from the score for even-numbered questions (Q2, Q4, Q6, Q8, and Q10). Next, we organize the scores into two subscales: adjusted ratings for odd-numbered questions, called the "Positive" subscale, and adjusted ratings for even-numbered questions, called the "Negative" subscale. The final SUS score is obtained by subtracting the "Negative" subscale score from the "Positive" subscale score, the sum of which is multiplied by 2.5. The resulting total SUS score ranges from 0 to 100, with higher scores indicating better usability. Scores exceeding 68 are considered above average, while scores exceeding 80 indicate excellent usability [19].

If individual SUS scores have been obtained, the next step is to calculate the average of all respondents' scores. An SUS score with a minimum average of 68 with a grade of C is considered acceptable for determining the usability level of the application. The assessment follows the Suaro-Lewis rating scale. If the application meets these criteria, then the application can be said to be good and feasible for use by users [18].

2.5. Operation and Maintenance

The fifth stage in this waterfall method is maintenance, where the system that has been used by users will always experience changes. Changes always occur if the operating system used by users is changed a new operating system is installed, or users need system development due to increased needs.

3. RESULTS AND DISCUSSION

3.1. Requirements Analysis

3.1.1. Functional Requirements

In making the SMART, a requirements analysis is needed. Functional requirements contain system requirements about all processes carried out by the system [20], as for the functional needs of the SMART system can be seen in table 4.

.

Table 4. Functional Requirement

No	Feature Requirements	Description
1	Login	
2	Manage the disposition, official notes and incoming letters (DS, ND, S)	DS = Disposisi ND = Nota Dinas S = Surat
3	Manage Minutes, notices, confidential official notes, official notes, permits, letters, orders, information, letter of information, and covering letters (BA, PEM, NDR, ND, SI, S, PRIN, KET, SP)	BA = Berita Acara PEM = Pemberitahuan NDR = Nota Dinas Rahasia ND = Nota Dinas SI = Surat Izin S = Surat PRIN = Perintah KET = Keterangan SP = Surat Pengantar
4	Sending letters	

3.1.2. Use Case Diagram

In many cases, the specification document is presented as a use-case model, which includes use-case diagrams and supplementary use-case definitions [21]. A use case diagram models the behaviour of an information system to be created. It describes the interaction between actors and the information system. Use cases are essential in creating information systems because they help identify the functions in the system and determine who has the authority to use these functions. The Use Case Diagram can be seen in figure 3 [22-24].
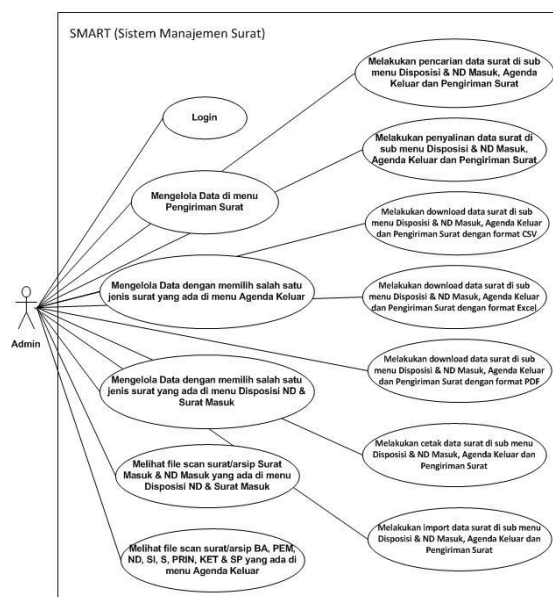
Figure 3. The SMART Use Case Diagram

.

.

The use case diagram of the SMART system consists of only 1 actor, namely the admin, the admin actor describes the employees who manage the letters in the general subdivision of the Malang customs office. The main tasks that can be performed by admin actors are.

1) Manage data by selecting one of the letter types in the ND Disposition and Incoming Mail.
2) Manage data by selecting one of the letter types in the Outgoing Agenda.
3) Manage data in Mailing
4) Manage scanned letter data files in ND Disposition and Incoming Letters selected as Incoming Letters or Incoming ND.
5) Manage the scan of letter data files in one of the selected Outgoing Agenda Disposition sub menus, namely BA, PEM, ND, SI, S, PRIN, KET, SP.

3.3. Design and Implementation SMART

The first process in designing the SMART is to collect letters data in the general subdivision, letter data in Excel form and each sub-type of letter has different columns. From the letter data collected, the letter columns used in each type of letter in the general subdivision are: (1) Letter Number; (2) Letter Date; (3) Letter Subject; (4) Receipt Date; (5) Disposition/From; (6) Remarks; (7) Letter Type; (8) To; (9) Leave Date; (10) ID; (11) Name; (12) Substitute Duty; and (13) Purpose.

Letter types are divided into 3 namely ND, Disposition, and Incoming Letters, Outgoing Agenda and Mail Delivery. ND Disposition and Incoming Letters based on the content of the letter data collected is a collection of incoming letter data received by the general subdivision, for example, such as internship permit applications, research permit letters, letters from other agencies and various letters from outside addressed to the Malang customs. The Outgoing Agenda is a collection of outgoing letter data made by the general subdivision for other sections in the Malang customs office or letters submitted to agencies outside customs such as internship permit reply letters, research permit reply letters and many others. Letter Delivery is a collection of letter data that must be sent out of the customs agency in Malang which has been put into a brown envelope.

On every page, always comes with features to import letter data, search letter data, copy letter data, export data to CSV, export data to Excel, export data to PDF, and print data. The development of this system uses the PHP programming language supported by the CodeIgniter framework in accordance with the requests of stakeholders.

3.3.1. Login

The login page is the initial page that appears when SMART is run. This page is used by users, namely employees who want to input letters and view, change, and delete letter

.

.

data. The login page consists of a username and password; users who can enter the SMART system are users whose data is already in the database. The login page is shown in Figure 4.

Figure 4. Login Page

3.3.2. Dashboard Page

The dashboard page is the main page of the SMART when the user has successfully logged in. In the dashboard page there are 3 main sub menus that are often used in the general subdivision, namely Incoming ND, Outgoing Agenda, and Mail Delivery. The dashboard page is shown in Figure 5.
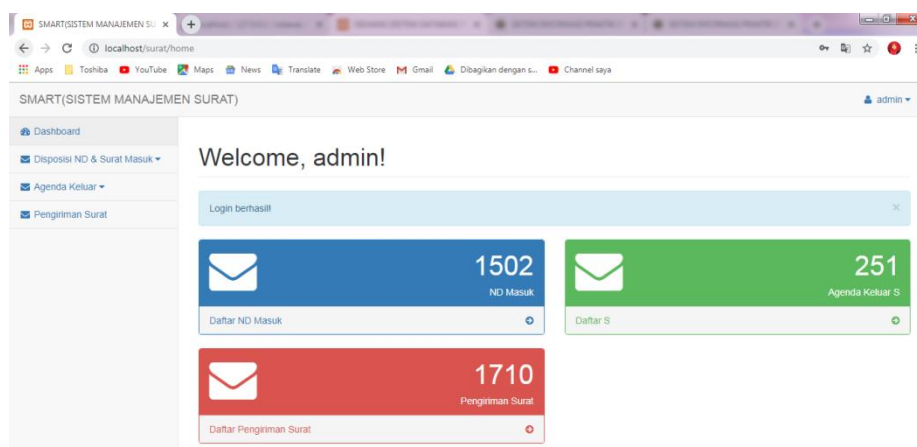

Figure 5. Dashboard Page

In this page, the admin can see the amount of incoming and outgoing mail data in general.

3.3.3. Disposition ND & Incoming Letters Page

The disposition letter (DS), official note (ND.S) sub menu is one of the sub menus of the ND & Incoming Letter Disposition menu which contains data on the disposition of warrants and official notes. The page of the DS, ND.S sub menus are shown in Figure 6.
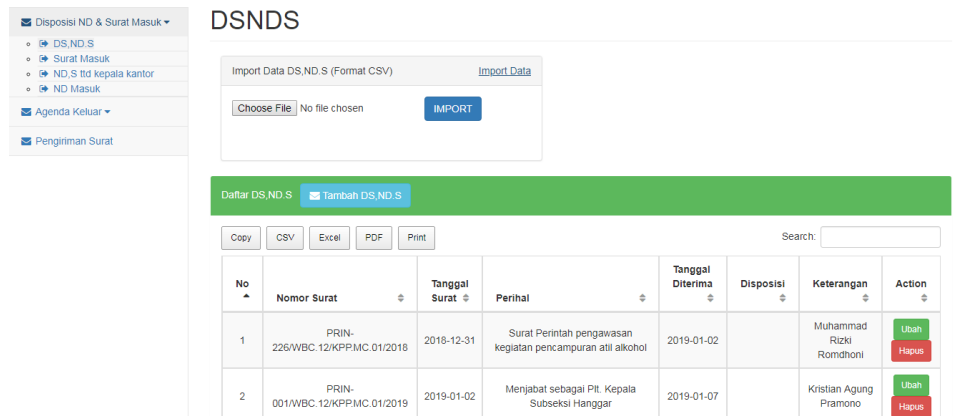
.

Figure 6. Disposition ND & Incoming Letters Page

On this page, the admin can add new DS, and ND.S, change and delete them. Besides that, there is an "ND, S. ttd Kepala Kantor" is one of the sub menus of the ND Disposition & Incoming Letters menu which contains data on official notes and letters signed by the head of the office. There is also an Incoming ND Sub menu which is one of the sub menus of the ND Disposition & Incoming Letters menu which contains incoming official memorandum data.

3.3.4. Outgoing Agenda Menu Page

In the BA menu, there are several sub menus, namely the BA sub menu which is one of the sub menus of the Outgoing Agenda menu which contains data on the numbering of minutes, the PEM sub menu which is one of the sub menus of the Outgoing Agenda menu which contains data on the numbering of notification letters, the NDR sub menu which is one of the sub menus of the Outgoing Agenda menu which contains data on the numbering of confidential official notes, the ND sub menu which is one of the sub menus of the Outgoing Agenda menu which contains data on the numbering of outgoing official notes, SI sub menu which is one of the sub menus of the Outgoing Agenda menu which contains employee leave permit numbering data, S sub menu which is one of the sub menus of the Outgoing Agenda menu which contains outgoing letter numbering data, PRIN sub menu which is one of the sub menus of the Outgoing Agenda menu which contains order letter numbering data, KET sub menu which is one of the sub menus of the Outgoing Agenda menu which contains certificate letter numbering data, and SP sub menu which is one of the sub menus of the Outgoing Agenda menu which contains cover letter numbering data. All of these sub menus have functions to add, change, and delete the letter data in them. The outgoing agenda menu page can be seen in Figure 7.
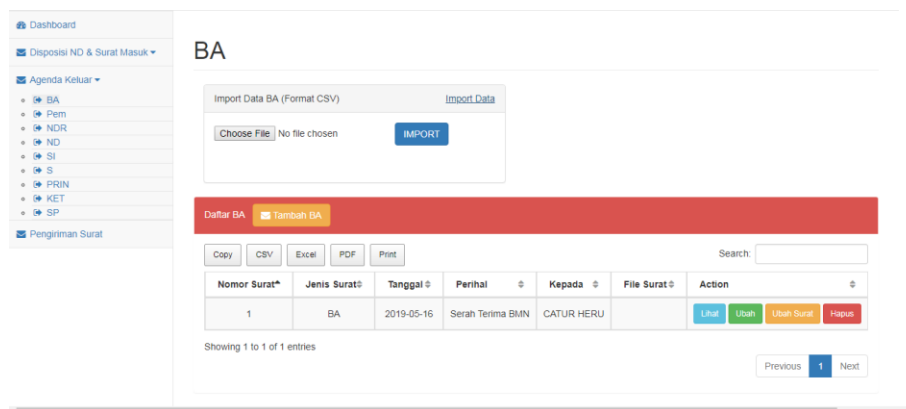
Figure 7. Outgoing Agenda Menu Page

3.4. Integration and Unit Testing

System integration was carried out in the general subdivision, especially in the field of administration and staffing which handles various kinds of correspondence. Before the integration was carried out, the head of the general subdivision and 4 staffs were gathered to conduct training on the use of SMART. The training was conducted to introduce SMART before SUS testing. After the training is conducted and deemed sufficient, SUS testing is carried out by distributing SUS questionnaires to prospective users who have gone through the training. SUS test results on SMART conducted on 5 users can be seen in Table 5.

Table 5. SUS Test Result on SMART

No	Respondent	Value
1	Respondent 1	100.00
2	Respondent 2	87.50
3	Respondent 3	92.50
4	Respondent 4	92.50
5	Respondent 5	95.00
	Average	93.50

As indicated in the table, the System Usability Scale (SUS) test results for the SMART application demonstrate a high level of usability perceived by the respondents. The scores are as follows: Respondent 1 was awarded a perfect score of 100.00, reflecting complete satisfaction with the application's usability. Respondent 2 provided a slightly lower score of 87.50, suggesting minor usability issues but overall, a positive assessment. Respondent 3 and Respondent 4 each scored the application at 92.50, indicating a strong usability experience with minor reservations. Respondent 5 rated the application at 95.00, again highlighting high satisfaction with the system's usability.

The average SUS score across all respondents is 93.50, which exceeds the benchmark 68, typically considered the threshold for acceptable usability. This average score places the SMART application well within the range of excellent usability, indicating that users find the application to be highly effective, efficient, and satisfactory. These results suggest that the SMART application is well-designed and meets the user's needs and expectations.

4. CONCLUSION

SMART is a web-based information system created to digitally manage incoming, outgoing, and internal letters. It offers features such as inserting, updating, and deleting letters, as well as uploading scanned letter files. This system aims to replace the manual letter archiving system, providing employees with the ability to search for specific letter scans using the letter number. SMART was developed using the Waterfall Method and the CodeIgniter framework, which are expected to simplify the creation of this information system.

Research results indicate that the System Usability Scale (SUS) test for the SMART application demonstrated a high level of usability, with an average score of 93.50, exceeding the benchmark of 68. This suggests that the SMART application is well-designed, effective, efficient, and satisfactory, meeting the users' needs and expectations.

In conclusion, the implementation of SMART as a digital letter management system is expected to significantly enhance the efficiency of the administration section's document archiving process by addressing the limitations of the current manual system. The positive SUS test results further support the potential success of SMART in improving the accessibility and management of letter archives at the Malang Customs and Excise Supervision and Service Office.

REFERENCES

[1] T. S. Ustun, C. Ozansoy, and A. Zayegh, "Recent developments in microgrids and example cases around the world—A review," *Renew. Sustain. Energy Rev.*, vol. 15, no. 8, pp. 4030–4041, Oct. 2011, doi: 10.1016/j.rser.2011.07.033.

[2] E. Syahputra *et al.*, "Penindakan Bea Dan Cukai Dalam Melakukan Penyidikan Terhadap Para Wisatawan Asing Yang Membawa Narkotika," *Nusantara Hasana Journal*, vol. 2, no. 3, pp. 46–53, Aug. 2022, [Online]. Available: https://nusantarahasanajournal.com/index.php/nhj/article/view/430

[3] M. Arianza Bima, R. Rahayu, S. Winata, and E. Syahputra, "Penindakan Bea Dan Cukai Dalam Melakukan Penyidikan Terhadap Para Wisatawan Asing Yang Membawa Narkotika," *Nusantara Hasana Journal*, vol. 2, no. 3, pp. 63–71, Aug. 2022, [Online]. Available: https://nusantarahasanajournal.com/index.php/nhj/article/view/431

[4] "Situs Resmi Kanwil Bea Cukai Jatim 2," Feb. 2024. https://kanwiljatim2.beacukai.go.id/ (accessed Nov. 12, 2024).

.

[5] S. Syukhri and P. Gusmayeni, "Design of Web-Based Archive Management Information System," *Jurnal Teknologi Informasi Dan Pendidikan*, vol. 14, no. 2, pp. 92–98, Sep. 2021, doi: 10.24036/tip.v14i2.429.

[6] S. Marlina and M. Leo Radhitya, "Mail Management System in the Digitalization of Village Administration: A Case Study of Batuyang Village Office, East Lombok," TECHNOVATE: Journal of Information Technology and Strategic Innovation Management, vol. 3, no. 1, pp. 148–155, Jul. 2024, doi: 10.52432/technovate.1.3.2024.148-155.

[7] S. Guntur Guntur, A. Ichsan, and I. Purnama Sari, "Designing a Web-Based Mail Management System at the Beringin Helvetia Sub-district Office," Altafani: Jurnal Pengabdian Masyarakat, vol. 1, no. 1, pp. 11–18, Mar. 2024, doi: 10.30596/altafani.v1i1.48.

[8] M. Z. A. Rikiti, E. Kusmayanto, and R. Kurniawati, "Design and development of Help Patient Data Management Information System at Sinergi Atap Negeri Foundation Web-Based using Codeigniter," *Journal of Applied Engineering and Technological Science*, vol. 3, no. 2, pp. 235–245, Jun. 2022, doi: 10.37385/jaets.v3i2.792.

[9] K. Curcio, T. Navarro, A. Malucelli, and S. Reinehr, "Requirements engineering: A systematic mapping study in agile software development," *Journal of Systems and Software/˜ the œJournal of Systems and Software*, vol. 139, pp. 32–50, May 2018, doi: 10.1016/j.jss.2018.01.036.

[10] I. Sommerville, Software Engineering. Addison-Wesley, 2015.

[11] A. De Lucia and A. Qusef, "Requirements engineering in agile software development," *Journal of Emerging Technologies in Web Intelligence*, vol. 2, no. 3, Aug. 2010, doi: 10.4304/jetwi.2.3.212-220.

[12] R. J. Holden, "A simplified System Usability Scale (SUS) for cognitively impaired and older adults," *Proceedings of the International Symposium of Human Factors and Ergonomics in Healthcare*, vol. 9, no. 1, pp. 180–182, Sep. 2020, doi: 10.1177/2327857920091021.

[13] A. Kaya, R. Ozturk, and C. A. Gumussoy, "Usability Measurement of Mobile Applications with System Usability Scale (SUS)," in *Lecture notes in management and industrial engineering*, 2019, pp. 389–400. doi: 10.1007/978-3-030-03317-0_32.

[14] T. Ananda, "Analisis Penggunaan Repository Perpustakaan Universitas Medan Area (UMA) dengan Menggunakan Metode iSystem Usability Scale (SUS)," Jurnal Teknologi Informasi Dan Pendidikan, vol. 16, no. 1, pp. 206–217, Oct. 2023, doi: 10.24036/jtip.v16i1.782.

[15] Y. Tian and C. Stewart, "History of E-Commerce," in IGI Global eBooks, 2006, pp. 559–564. doi: 10.4018/978-1-59140-799-7.ch090.

[16] E. Radcliffe, B. Lippincott, R. Anderson, and M. Jones, "A Pilot Evaluation of mHealth App Accessibility for Three Top-Rated Weight Management Apps by People with Disabilities," International Journal of Environmental Research and Public Health/International Journal of Environmental Research and Public Health, vol. 18, no. 7, p. 3669, Apr. 2021, doi: 10.3390/ijerph18073669.

[17] M. Mawardi, "Rambu-rambu Penyusunan Skala Sikap Model Likert untuk Mengukur Sikap Siswa," *Scholaria*, vol. 9, no. 3, pp. 292–304, Sep. 2019, doi: 10.24246/j.js.2019.v9.i3.p292-304.

[18] J. R. Lewis, "The system usability scale: past, present, and future," International Journal of Human-computer Interaction, vol. 34, no. 7, pp. 577–590, Mar. 2018, doi: 10.1080/10447318.2018.1455307.

[19] M. R. Pradana and N. Nuryuliani, "Redesign Of Ipusnas Application Using User Centered Design Method," *International Journal Science and Technology*, vol. 2, no. 1, pp. 73–79, Mar. 2023,

.

doi: 10.56127/ijst.v2i1.866.

[20] C. A. Crespo-Santiago and S. De La Cruz Dávila-Cosme, "Waterfall method: a necessary tool for implementing library projects," *HETS Online Journal*, vol. 1, no. 2, pp. 81–92, Nov. 2022, doi: 10.55420/2693.9193.v1.n2.91.

[21] E. Saleh and F. ElShahrani, "Extracting Functional and Non-Functional requirements for e-learning systems," ˜*Al-œMajallah Al-dawlῑYah Lil-ʿuluM Al-tarbawῑYah Wa-al-aDaB*, vol. 2, no. 5, Jan. 2023, doi: 10.59992/ijesa.2023.v2n5p3.

[22] B. Hnatkowska and M. Cebinka, "Activity Diagram Generation based on Use-Case textual Specification," *Computing and Informatics*, vol. 40, no. 4, pp. 772–795, Jan. 2021, doi: 10.31577/cai_2021_4_772.

[23] M. N. Arifin and D. Siahaan, "Structural and semantic similarity measurement of UML use case diagram," *Lontar Komputer/Lontar Komputer*, vol. 11, no. 2, p. 88, Jul. 2020, doi: 10.24843/lkjiti.2020.v11.i02.p03.

[24] R. Fauzan, D. Siahaan, S. Rochimah, and E. Triandini, "A different approach on automated use case diagram semantic assessment," *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 1, pp. 496–505, Feb. 2021, doi: 10.22266/ijies2021.0228.46.