DEVELOPMENT OF SHIP ARCHIVE INFORMATION SYSTEM IN XYZ COMPANY WITH SCRUM METHODOLOGY

Vika Fitratunnany Insanittaqwa¹, M. Ihsan Alfani Putera², M. Fathurrahman Shidiq³
Institut Teknologi Kalimantan, Balikpapan, Indonesia¹
E-mail address: vika.fitra@lecturer.itk.ac.id¹

Received: 26, January, 2024 Revised: 25, May, 2024 Accepted: 1, June, 2024

ABSTRACT
XYZ Company, a shipping agency in Balikpapan, handles the management of documents related to ships docking or sailing from the port. Currently, the document management process relies on manual, paper-based procedures, lacking a digital system to integrate all administrative processes. This manual approach is time-consuming, as staff need to search for and store physical documents in storage cabinets, which also occupy significant office space and pose a risk of document damage over time. The purpose of this research is to develop a web-based information system for document processing at XYZ Company, aiming to streamline the documentation process for ship archives. The development follows the Agile methodology using Scrum to adapt to evolving requirements throughout the development process. The stages of this research include Requirement Gathering, Sprint Planning, Daily Meetings, Sprint Reviews, Sprint Retrospectives, and Deployment. A total of 55 Product Backlogs were completed over three Sprints. Black Box testing was conducted at each iteration with the supervision of XYZ Company staff, yielding satisfactory results. The outcome of this research is an integrated Ship Archive Information System hosted on XYZ Company’s server, providing a robust solution to the ship documentation problem.

Keywords: Information System, Scrum, Ship Archive, Website, XYZ Company.

1. INTRODUCTION
Indonesia is an archipelago with more than 17,000 islands and more than 7,000 inhabited islands (Ghifari & Ahyudanari, 2021). The use of sea transportation as logistics capital for the distribution of primary commodities for economic growth is a suitable choice for this condition. XYZ Company is a company dealing in ship agency services that offers tanker ship transportation services, which are a form of sea transportation designed to transport raw fuels such as crude oil, liquid natural gas, finished fuels, and petrochemicals by sea. To carry out these functions, XYZ Company requires the use of information technology to support existing business processes.

Based on the Indonesian Government Regulation No. 20 Year 2010 on Water Transportation, companies must provide reports in writing relating to the documents about the ship that will enter and exit the port. The documents required by the ship include a passport for all crew members, last port clearance, maritime declaration of health, cargo manifest or nil cargo, last port of call or voyage memo, general list or nil list, vaccination list, ship’s store list including bonded list, crew effect declaration, crew arrival list, narcotics list, medicine list, ship’s
particulars, ship's health book, medical chest certificate, stowage plan or loading plan, inventory list, and water quality control certificate. However, the information management system at XYZ Company is still done manually using paper and physical storage in a cabinet located in the office.

XYZ Company does not yet have a system that integrates all of its administrative processes. This causes the processing of ship documents to be time-consuming and labor-intensive because staff must do manual input and archive manually by storing in the archive cabinet. The archiving process through manual methods also has a high risk of being lost or damaged due to various reasons (Ambarwati & Adianti, 2022; Yuniar et al., 2023).

The use of Office programs such as Microsoft Word and Microsoft Excel as archiving tools may be presented as the solution to the problem. However, the use of this software raises new problems, namely the lack of integration of one data with another and does not allow real-time access to the information by various stakeholders. In addition, there is also a risk of a complicated process of searching for data files if the staff want to make a report, where staff must find and open files again one by one which is certainly not an efficient solution to the initial problem (Gultom et al., 2024). Based on these problems, this research aims to create a system that integrates the document archiving process and provides real-time information that can be accessed by various stakeholders quickly, easily, and reliably.

Some literature provides solutions by utilizing website technology, including the ship data entry information system at PT Barakomindo Shipping Sorong Branch (Huwae et al., 2021), the archive e-filling application at PT Atosim Lampung Shipping (Juliyanto & Parjito, 2021), and the archive processing application at PT Daya Eka Samudera (Khoiriyah et al., 2022). The use of this web-based system allows access from anywhere and anytime with a device connected to the internet, so that archive data can be accessed by stakeholders whenever or wherever they are.

To develop the web-based system, this research uses the Agile method with Scrum (Ratnawati et al., 2023). The Agile method is one of the software development methods that is considered to have many advantages over traditional and linear development models, where the Agile method allows developers to adapt to changing needs that may occur by customers (Prabowo, 2020; Dawis et al., 2023; Sudipa et al., 2023). This method supports direct team involvement, early software releases, embracing change, incorporating technological needs and changes (Rasheed et al., 2021). Scrum is a variation of the Agile development method which is a combination of iterative and incremental development designed to accelerate the development process, align individual and corporate development goals, create a culture focused on development performance, support stakeholder value, require good communication, and improve individual development and quality of life (Srivastava et al., 2017; Sutherland, 2018; Aprilia, 2021; Tantraluhur, 2023). With the Agile method with Scrum, it is expected that the development of XYZ Company's information system can be carried out quickly and adaptively to changes by conducting intense communication between developers and the company.
The purpose of this research is the development of website-based software using the Agile method with Scrum to integrate data archives which are expected to solve archiving problems at XYZ Company so as to create efficiency both in time and effort.

2. THEORY

XYZ Company was founded in 1969 in Jakarta as a subsidiary of one of Indonesia’s top oil companies. The original purpose of XYZ Company was to engage in the maritime services industry with several functions, such as: to procure fuel distribution to all ports throughout Indonesia that cannot be reached by tankers, to procure maritime transportation for the development of projects spread throughout Indonesia, and to act as General Agent and Handling Agent for the parent company’s tankers on lease.

Scrum is a software development method that is derived from the Agile method. Scrum is done with the collaboration of the Scrum Team, which consists of one Scrum Master, one Product Owner, and Developers (Noll et al., 2017; Oomen et al., 2017). The responsibility of the Scrum Master is to establish the Scrum system, which involves several processes: the Product Owner orders the work into a list of Product Backlog, the Scrum Team turns a selection of the work into an Increment of value during a Sprint, the Scrum Team and the stakeholders inspect the results and adjust the development process for the next Sprint, and this cycle is repeated (Andre, 2023; Perdana, 2024; Team, 2024). Product Backlog is a prioritized list of project requirements in the form of: feature implementations, enhancements, or fixes that will be worked on and become a source of requirements for all changes that need to be applied to the software (Dewi, 2022; Susanto, 2024; Taliya, 2022). Sprint itself is the heartbeat of the Scrum, where the Scrum Team works to produce an increment in iterations (Magdalena, 2023). The duration of each Sprint can vary in different projects, with different numbers of iterations depending on the remaining Product Backlog to finish (Schwaber & Sutherland, 2020).

Each Sprint starts with Sprint Planning, where the Product Owner provides insight about the value of the current Sprint and discusses with the Developers to select items from the Product Backlog. The selected items are called the Sprint Backlog which is the goal of the Sprint. The Developer team can decompose the Backlog into smaller work items or tasks and work to deliver them by the end of the Sprint. To inspect the progress, 15-minute Daily Scrum is conducted by the Scrum Team to communicate the work by the Developers, to identify shortcomings, and to make a decision regarding solutions to the current work. At the end of the Sprint, Sprint Review is conducted to inspect the outcome where the Scrum Team presents the results of their work to key stakeholders and they discuss the progress toward the product goal. The Sprint is then finally closed with Sprint Retrospective, where the team plans ways to increase the quality and effectiveness of the work. The Scrum Team identifies the most helpful changes to be added to the Sprint Backlog for the next Sprint if necessary.

Use Case diagrams are used to display a general description of the software features that will be implemented to find out the limitations of development (Setiyani, 2021; Lesmono, 2024). Use Case diagram depicts the interaction between Actor (system user) and the System itself (Dharwiyanti & Wahono, 2003; Maulana, 2022). The perspective used in the design of the Use Case diagram is from the Actor's perspective to capture the needs of the Actor and to ensure
that the Actor's goals can be achieved (Kurniawan, 2018). Some important notations of this diagram are Actors, Use Cases, System Boundary, and Association Lines.

Activity Diagrams are used to model the dynamic aspects of a system that describes system activities, the data related to these activities, and the sequence of execution of these activities (Ahmad et al., 2019; Eshuis, 2006; Eshuis & Wieringa, 2004). Activity diagrams have several elements, namely Start Point, End Point, Activities, Forks, Joins, Decision Points, and Swimlanes.

Entity-relationships diagram (ERD) is a visual model used to describe the Entities that play a role in a system, the Attributes of these entities, and the relationship between one entity and another (Hartono, 2022; Lamanit, 2023). This model is usually the first step in designing a database for a system, which is very important in the software development process. ERD will usually be translated first to its logical data model by applying certain rules that allow the database to be implemented and integrated into an information system (Rashkovits & Lavy, 2021).

PHP or Hypertext Processor is a server-side programming language to develop internet-based applications. It is known as a language that is easy to learn, community-friendly, freely available as open-source software, and easy to deploy (Ganney et al., 2014) (Karayiannis & Karayiannis, 2019). Currently, there are several frameworks of PHP that can be implemented to promote the reuse of code based on the practice of each framework which could save time and costs in designing, developing, and testing the application (Haris & Hasim, 2019; Pleva, 2013; Uzayr, 2022). One of the most popular frameworks is CodeIgniter, which uses Model-View-Controller (MVC) architecture (Anttonen, 2018; Khan & Khanam, 2023). MVC separates the user interface, data, and process interfaces, making it possible to develop or maintain applications more effectively and efficiently (Abutaleb et al., 2021). CodeIgniter is claimed to have the fastest execution compared to other frameworks and that it can help programmers create web-based applications faster and easier (Purbo, 2021).

Black box testing is a method of testing in software to determine the functionality of an application (Dashti & Basin, 2020; Peled et al., 1999). The internal workings of the system is not known to the tester because the test scenarios are only concerned about the expected output of the application after being given certain input which should decide if the system works or not based on the software requirement specification (Verma et al., 2017).

3. METHOD
The method used in this research is shown below (Fig. 1). The Requirements of the Ship Archive Information System is specified after a series of interviews with the staff of XZY Company. A list of Product Backlog is produced in this stage. Afterward, the development of the software is conducted in several Sprints. In Sprint Planning, the Developer and Product Owner decides on several items from the Product Backlog as Sprint Backlog to deliver by the end of the Sprint. Each day, the Scrum Team conducts a Daily Meeting to evaluate the progress of the day. At the end of the Sprint, the Developer presents the outcome of the Sprint to the Product Owner to see the overall progress of the development in the Sprint Review stage. The Sprint ends with the Sprint Retrospective where the Scrum Team reflects on the current status
of the software and decides if the work is finished or improvements should be done on the next Sprint.

3.1 Requirement

This stage determines the requirements of the system by compiling a Product Backlog based on interviews with XYZ Company. This Product Backlog must be implemented and completed until there are no items left on this list at the end of development.

3.2 Sprint Planning

Sprint Planning is conducted with all Scrum Team members. This stage is important to determine what will be completed in the Sprint as outlined in the Sprint Backlog. The Sprint Backlog is selected from the Product Backlog that can be done within the specified Sprint time. In this study, the Sprint was carried out in a duration of 1 month as decided by the agreement of the Scrum Team.

3.4 Daily Meeting

The developer then implements the Sprint Backlog every day for the duration of the Sprint. Before implementing the program, the Developer also creates the Use Case Diagram, ERD, and Activity Diagram needed for the Sprint. Each day, the Scrum Team holds a Daily Meeting to determine what work should be done for the day. The duration of the Daily Meetings is 15 minutes. The software implementation that is carried out every day after the Daily Meeting produces program code written in PHP with CodeIgniter framework.
3.5 Sprint Review

Sprint Review is conducted at the end of the Sprint, where the Developer presents the results of his work to the Product Owner, which is a representative of XYZ Company. At the Sprint Review, feature testing is also carried out using Black Box Testing to see if the software has produced the desired output. This stage produces test documentation that will be used as a basis for the decision making in the next iteration to see if there are features that failed to be implemented.

3.6 Sprint Retrospective

Sprint Retrospective finalizes the Sprint by evaluating performance after the month is over. If the Sprint has been successfully implemented and all items on the Product Backlog have been implemented, then the software development process can proceed to the Development stage. However, if there are still items in the Product Backlog that have not been implemented or the Sprint still leaves a Backlog that has not been successfully tested, then additional Sprints need to be added. Sprint Retrospective also evaluates improvements that can be made for the next Sprint iteration to minimize errors in development.

3.7 Deployment

Deployment is carried out after all Product Backlog items have been successfully implemented and tested. At this stage, the information system is handed over to XYZ Company to be installed at its facilities. In addition, final testing with Black Box method is also carried out on the XYZ Company server to ensure that the information system is functioning properly and can be operated by the staff.

4. RESULTS AND DISCUSSION

Three Sprints have been carried out with a duration of 1 month each to build the ship archive information system. The Product Backlog explored at the Requirement stage amounted to 25 items, but in Sprint 2 there were additional features so that the total Product Backlog that had been implemented was 55 items.

4.1 Requirement

After several interviews with XYZ Company staff, a total of 25 Product Backlogs were listed, namely: Admin homepage, Add a User, Edit a User, Delete a User, Login, Logout, Index homepage, Document Index page, Add ship document, Add tanker document, Edit ship document, Edit tanker document, View document, Delete document, Upload document, Download document, Search document, Ship Index Page, Add ship information, Edit ship information, View ship information, Delete ship information, Download ship information, Upload ship information, and Search ship information. All of these Product Backlogs must be implemented in several Sprints according to the Scrum methodology.

4.2 Sprint 1

In this Sprint, 19 Sprint Backlogs were selected from the Product Backlog to be worked on. This selection is based on the Scrum Team's agreement at the Sprint Planning stage.
1) Sprint Planning
Sprint 1 started with Sprint Planning by selecting 19 Sprint Backlog items from the Product Backlog that focused on the Documents and Ship Information feature. The list of Sprint Backlog items to be worked on includes the Index Homepage, Document Index Page, adding and editing ship and tanker documents, viewing and deleting documents, uploading and downloading documents, searching documents, Ship Index Page, adding, editing, viewing, and deleting ship information, and uploading, downloading, and searching ship information.

2) Daily Meeting
After choosing the Sprint Backlog, the Developer immediately works on implementing the program every day. Every day a Daily Meeting is held with the Scrum Team to discuss what will be done in the day. In addition to implementing the code, the Developer has also designed diagrams related to the development in Sprint 1. The diagrams that have been designed include Use Case Diagram, Activity Diagram, and ERD found in Figure 2, Figure 3, and Figure 4, respectively.

![Use Case Diagram](image)

**Figure. 2. Use Case Diagram for Sprint 1.**

The Use Case described only has 1 Actor, namely the User who is a staff of XYZ Company. The staff is responsible for processing Document archives and Ship data. This Use Case Diagram will be updated throughout the development process in the next Sprints if needed.

Figure 3 shows one of the Activity Diagrams created in Sprint 1, which is for the process of adding ship documents. The process starts with the User choosing the Ship Document menu, then the system will display the Ship Document page. After the user presses the "Add" button on the page, the system will display the Add Document page. Then the User can input data as well as upload the ship documents. On this page the User can enter 2 types of documents, namely the Tanker Ship document or the general Ship Document. After the document data is filled and the document is uploaded, the User presses "Save" and the system will save all data into the database.
Figure 3. Activity Diagram of Add Ship Document.

Figure 4. ERD of Sprint 1.

Figure 4 shows that there are 3 Entities designed in this Sprint, namely Document, Document Type, and Ship. This ERD is important for the implementation of the database required to store...
inputs from users of the information system. This diagram will also be updated in the next Sprints if needed. After all the diagrams have been designed, the developer implements the program code. One example of a successfully implemented interface can be seen in Figure 5.

Figure 5 shows the Document Index Page which provides information on archives stored in the system. On this page Users can also add documents, change documents, view documents and delete documents. In addition to this page, there are also several other pages that have been successfully implemented, including the Index Homepage, Add Ship document, Add Tanker document, Edit Ship document, Edit Tanker document, View Document, Upload document, Download document, and Ship Index Page.

3) Sprint Review
After a month, Sprint Review was carried out to review the progress and to test the functionality of the system with the supervision of one of the XYZ Company staff using the Black Box testing. The test results can be seen in Table 1.

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Expected Output</th>
<th>Actual Result</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add Tanker Document</td>
<td>When using the add document feature, the entered document data and the file are stored in the database.</td>
<td>When using the add document feature, the entered document data and the file are stored in the database.</td>
<td>Success</td>
</tr>
<tr>
<td>Add Ship Document</td>
<td>When using the add document feature, the entered document data and the file are stored in the database.</td>
<td>When using the add document feature, the entered document data and the file are stored in the database.</td>
<td>Success</td>
</tr>
<tr>
<td>Test Case</td>
<td>Expected Output</td>
<td>Actual Result</td>
<td>Conclusion</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>Edit Tanker Document</td>
<td>When using the edit document feature, the entered document data is modified in the database including the document file address.</td>
<td>When using the edit document feature, the entered document data is modified in the database including the document file address.</td>
<td>Success</td>
</tr>
<tr>
<td>Edit Ship Document</td>
<td>When using the edit document feature, the entered document data is modified in the database including the document file address.</td>
<td>When using the edit document feature, the entered document data is modified in the database including the document file address.</td>
<td>Success</td>
</tr>
<tr>
<td>Delete Document</td>
<td>When using the delete document feature, the document data is deleted in the database and the document file is deleted from the server.</td>
<td>When using the delete document feature, the document data is deleted in the database but the document file is not deleted from the server.</td>
<td>Fails</td>
</tr>
<tr>
<td>View Document</td>
<td>When using the view document feature, the file will be displayed in a new web window.</td>
<td>When using the view document feature, the file will be displayed in a new web window.</td>
<td>Success</td>
</tr>
<tr>
<td>Download Document</td>
<td>When using the download document feature, the file will be downloaded by the system.</td>
<td>When using the download document feature, the file will be downloaded by the system.</td>
<td>Success</td>
</tr>
<tr>
<td>Add Tanker Document</td>
<td>When using the add document feature, the entered document data and the file are stored in the database.</td>
<td>When using the add document feature, the entered document data and the file are stored in the database.</td>
<td>Success</td>
</tr>
</tbody>
</table>

Table 1 shows the failure of one feature, namely Delete Document, which has not been successfully implemented because the document file has not been deleted on the server even though the data has been deleted in the database. This has the potential to cause low hard disk space in the future if too many files cannot be deleted from the server. Therefore, the implementation of this feature will be improved and will be added to the list of Backlog of Next Sprints.

4) Sprint Retrospective
In the Sprint Retrospective, it has been communicated that only 10 Sprint Backlogs have been implemented, namely Index Homepage, Document Index Page, Add ship document, Add tanker document, Edit ship document, Edit tanker document, View Document, Upload document, Download document, and Ship Index Page. The rest of the items in the Sprint Backlog were not successfully implemented within 1 month because there was a hindrance on the Developer side where he had to learn the new CodeIgniter 4 framework. This learning process took up too much time in the Sprint so the Sprint Backlogs cannot all be completed.
4.3 Sprint 2
In Sprint 2 the Product Owner added more required functionalities that focused on the management of financial and staff documents. This shows the advantages of the use of Scrum which can flexibly accommodate changes in software requirements in the middle of the development process. This addition results in an increase in the total of Product Backlog items that must be completed.

1) Sprint Planning
In Sprint Planning, the team determines the Sprint Backlog items. Some of the Sprint Backlogs taken from the Product Backlogs before the addition of features include deleting documents, searching documents, adding ship information, editing ship information, viewing ship information, deleting ship information, downloading ship information, uploading ship information, searching ship information, creating an admin homepage, adding a user, editing a user, deleting a user, logging in, and logging out.


2) Daily Meeting
After choosing the Sprint Backlog, the Developer designs and updates several diagrams. Use Case Diagrams were updated based on the Sprint Backlog added in this Sprint. Activity Diagrams were also designed for each page to be implemented. The ERD was also updated based on the new Entities. After the developers designed all the diagrams, the code was implemented to complete the Sprint Backlog in Sprint 2.

3) Sprint Review
Document, View Staff Document, Delete Staff Document, Download Staff Document, and Upload Staff Document. Black Box testing has also been conducted and resulted in a Success status for the entire Sprint Backlog that was successfully implemented.

4) Sprint Retrospective
Some items in the Sprint Backlog were not completed due to time limitations. In addition, the Developer had difficulty implementing the search feature to enable the users to get specific information from the index page. Therefore, the search features in all index pages should be implemented in the next Sprint.

4.4 Sprint 3
All remaining items in the Product Backlog are completed in this Sprint.

1) Sprint Planning
There are 17 Sprint Backlogs to be done in this Sprint, namely: Staff Index Page, Add Staff information, Edit Staff information, View Staff Information, Delete Staff information, Search Staff information, Search Staff document, Search document, Search ship information, Search finance document, Search Verified Finance document, Admin homepage, Add a User, Edit a User, Delete a User, Login, and Logout.

2) Daily Meeting
The developer updated and finalized the Use Case and ERD diagrams. In addition, the developer also designed the Activity Diagrams of the pages to be implemented. After that, the developer implemented the program to complete the ship archive information system.

3) Sprint Review
The developer has successfully completed the entire program implementation and also completed functional testing. No errors were found in the test, so it can be concluded that the test has been carried out successfully.

4) Sprint Retrospective
In the Sprint Retrospective, it has been communicated that the Developer has completed his duties well by implementing all Product Backlogs so that there are no remaining items. So it can be concluded that Sprint 3 is the last Sprint in the development of this software and the research can be continued to the Deployment stage.

4.5 Deployment
The information system has been handed over to XYZ Company and has been installed on the company server. Final functionality testing has also been carried out on the server to ensure all features are running properly using the Black Box method. This test was declared to have been carried out successfully because there were no errors. Thus, the development of the Ship Archive Information System has been successfully completed and is expected to be a solution to the XYZ Company's problems regarding the archive documentation system.

5. CONCLUSIONS AND SUGGESTIONS
The development of XYZ Company's web-based ship archive information system using the Scrum method has been completed. In this research, a total of 55 Product Backlogs have been implemented. The Product Backlogs were initially explored in the Requirement stage and had
been updated in the second Sprint. Overall, 3 Sprints have been carried out, each with a duration of 1 month. The development of the ship archive information system using the Scrum method shows the flexibility of this method which is shown by the freedom of the Product Owner to add features in the middle of development and the freedom of the Developer to complete the Product Backlog according to his ability in each Sprint because the Backlog that has not been successfully implemented in that Sprint can be moved over to the next Sprint. The test results show that the system has worked well and is expected to be a solution to the problem of the ship archive documentation system at XYZ Company.

It is suggested that the maintenance process of the program is carried out by XYZ Company because maintenance is an important part of the software life cycle. This maintenance is part of the software evolution process. When the company has depended on the use of software, maintenance must be carried out due to the possibility of changes from error discovery, changes in the software operating environment, and additional functional requirements (Ogheneovo, 2013). This maintenance can also be done with the Scrum method because it has a high level of flexibility with the application of the iteration process (ur Rehman et al., 2018). In the future, additional features that support the needs of XYZ Company can be added to digitalize the document archive management process. Design of a better program interface can also be done if necessary based on the needs of the company.

REFERENCES
Transactions on Software Engineering, 30(7), 437–447. https://doi.org/https://doi.org/10.1109/TSE.2004.33


