FLYWHEEL: JURNAL TEKNIK MESIN UNTIRTA

Homepagejurnal: http://jurnal.untirta.ac.id/index.php/jwl

Design of Bridging System for Reporting Loading and Unloading of Goods in Imports and Online TPS Application at the Directorate General of Customs and Excise (DJBC) (Case Study: PT. PolyChemLindo)

Anita Megayanti, M.Kom[1*], Roy Amrullah Ritonga, M.Kom[2],

[1]*STTIKOM Insan Unggul Cilegon, JL Sultan Ageng Tirtayasa Kav. 25-28 No. 146 424*
[2]*STIKOM Alkhairiyah Cilegon, JL. H. Enggus Arja No. 1 42441*

*Corresponding author : anita.megayanti@gmail.com

ARTICLE INFO

Received 26/03/2020
revision 26/04/2020
accepted 29/04/2020
Available online 30/04/2020

ABSTRACT

Temporary Piling Places are used to hoard imported goods while awaiting their release from customs areas following the provisions of customs law. PT. Polychem Lindo is a company that functions as a TPS (Temporary Hoarding Site) that reports data starting from goods coming off the ship to goods leaving the gate or vice versa.
At present, the data reported to DJBC (Directorate General of Customs and Excise) is still standard, even though a web service has been made to integrate the data into DJBC Online TPS. Meanwhile, that information about goods data in real-time and monitor its movements from time-to-time.
The method used in application development is a waterfall. The design of the bridging application system of the reporting documents of hoarding of loading and unloading of imported goods with the Online TPS DJBC application (Directorate General of Customs and Excise) is carried out so that the document entry process is only done in one transaction. This research produces a Bridging System program that functions to bridge the two running systems so that the data input process runs more efficiently. The bridging system is the use of web service information technology that allows two different systems at the same time to be able to carry out two processes without interference from one system on another system.

Keywords: *Bridging Application System, TPS Online Web Service Item Licensing Application*

1. PRELIMINARY

The development of technology, especially information technology in this decade, is so fast that almost everyone is very familiar with information technology. The development of information technology that is fast can not be separated from the demands of companies as users in the use of information technology to meet the needs of information needed in decision making [1]. The utilization of data based on information technology is a must now if the company wants to compete in the current era of globalization [2].

PT. Polychemlindo is a company providing temporary stockpiling of importing goods, the type of transactions of which are tanks. The storage tank will be temporarily stored by limiting the period for the stocking of products inside. The thirty-day period provided is deemed sufficient to provide an opportunity for the stakeholders to immediately remove their goods from the Temporary Piling Place so as not to disrupt the smooth flow of products at the port (congestion). Hoarding of imported goods which has passed the deadline determined by the Customs and Excise officials will be declared as goods

not controlled. Polychem Lindo, as the TPS (Temporary Hoarding Site) company, is responsible for the import duty on unpaid, imported goods which are piled up at the TPS before the customs notification is submitted by the importer [3].

Goods Import Declaration (PIB) is a notification document by the importer to the customs on imported goods, based on customs complementary documents according to the self-assessment principle. Self-assessment is a principle that requires taxpayers to calculate, pay, and report taxes according to the provisions of the law [4].

Also, the TPS company being required to pay and report taxes, the Direktorat Jenderal Bea Cukai (DJBC), has developed an application to use the internet as a medium for the delivery of import-export documents and documents related to other customs. The aim is not to input data manually by the DJBC. The problem that arises in TPS entrepreneurs is providing an application to input document data that will be reported to DJBC then sending data to DJBC, because it has been prepared by DJBC's web services [5].

A Bridging system aims to increase the effectiveness of data entry processing, efficient use of resources, and faster reporting documents [2]. Furthermore, the Bridging system could also integrate smart home devices with connected vehicles based on promising de-facto standards [6]. Meanwhile, Risk analysis using Fit / Gap Analysis and FMEA as a solution to minimize the failure of implementing system integration. This method can reduce of 92.67% risk failure [7]**.** Moreover, a bridging system can improve the effectiveness of SIMRS Application and Virtual Claim Application registration services at the Assyifa Sukabumi Islamic Hospital. At the same time, the transaction process has become simpler [8].

In our previous research, we had conducted the Bridging system between BPJS applications with the SIMRS in Krakatau Medika Hospital. In this paper, we introduce the bridging design process between the form of loading and unloading documents, loading and unloading of imported goods into the DJBC online, TPS application consisting of an explanation of the web service. Moreover, the bridging system could make recapitulation reports of incoming/outgoing goods easier and monitored in real-time.

2. RESEARCH METHODS

2.1. THEORETICAL BASIS

2.1.1. DESIGN AND BUILD

According to *Pressman*, the design means a series of procedures of programming language to describe in detail how the system components are implemented [9]. The notion of a building or building system is the activity of creating new systems or replacing or improving existing systems as a whole.

Building design is closely related to system design which is a unity to design and build an application, it can be concluded that the system design is an activity of translating the results of analysis into the form of software products then creating the system or improving the existing system.

2.1.2. BRIDGING SYSTEM

Bridging is to connect systems with formats and structures suitable for one type of computer with systems that are suitable for other networks [10].

The bridging system is to harmonize two different systems without the intervention of each system with each other so that data security is maintained [11].

Bridging comes from the word "Bridge," which means bridge, bringing together, bridging. As the name implies, this tool is used to bridge two networks. But unlike repeater, which only functions as a physical bridge, the bridge can also function as a logical bridge such as dismantling and arranging rescue packages, buffering, and others. Thus the bridge can be used to connect two kinds of networks with different packet formats or have different speeds.

2.1.3. WEB SERVICE

The making of this document application system for reporting incoming / outgoing imported goods is how to utilize all DJBC electronic services for stockpiling reporting so that TPS entrepreneurs can report in real-time. The solution is to use web services that have been provided by DJBC [12].

Web service is a software system designed to support machine-to-machine operations through computer network interactions with web server applications, world wide web (WWW) HyperText Transfer Protocol (HTTP) [13,14]. Web services allow communication between various applications using open standards such as HTML, XML, WSDL, and SOAP. Web services can provide services with different operating systems and programming languages. Web services use XML as a standard for exchanging messages and data [15].

2.2. RESEARCH METHODS

2.2.1. DATA COLLECTION METHODS

Data collection methods are done in 2 (two) ways, namely:

a. Observation

Application designed according to the management conditions of PT. Polychem Lindo and can be executed following expectations. Therefore it is necessary to conduct inspections or research at PT. Polychem Lindo Merak to obtain relevant information following the conditions of the appropriate TPS company.

b. Interview

Considering how important the matching of information between the topics raised in the study with the intended object, it is necessary to conduct interviews with relevant parties, among others: management of PT. Polychemlindo regarding data reported to DJBC while discussions with DJBC related to the bridging system using web services.

c. Literature review

To support the presented information, it is also necessary to collect data from various sources, be it books, scientific journals, or the internet about the bridging system and reporting of temporary stockpiling of goods in / out.

2.2.2. METHOD OF ANALYSIS AND DESIGN OF INFORMATION SYSTEMS

The bridging of the application system design method uses a waterfall model which includes:

a. System Requirements Analysis

At this stage, an analysis of the needs and problems experienced by PT. Polychemlindo. The software that is made must have functions following the requirements as desired from the Company, namely PT. Polychemlindo.

There are several report documents submitted to DJBC to be made into the application for loading and unloading documents on loading and unloading of imported goods. In the reporting application, a master menu will be made following the reference data from DJBC, so that the process of sending data can be validated correctly to the DJBC online TPS application.

b. Design

Based on the needs analysis, it can be seen what functions the software must-have, and then the software design is made with the appropriate functions.

The design phase of reporting applications for incoming/outgoing goods documents includes input design and output design. Input design that is made is adjusted to some data that must be input based on the data needed. To make it easier to display data and store data, a database is required to hold data that has already been saved. Both data are to clarify the inputted data in detail, either displayed in tabular form or the form of reports such as in pdf and excel form.

c. Code Generation

Each object runs well as expected, so specific codes are needed in moving the object in question. The code referred to in the application is designed in the form of basic code and SQL queries.

d. Testing

The testing phase is the next stage after the code generation stage. At this stage, Black Box testing is done to check whether the system in question can run according to what is expected or not. If the program not following what is expected, both damaged and deficient, the program will undergo correction or be developed again. Testing focuses on software in terms of logic and functional and ensure that all parts have been tested; this is done to minimize errors (errors) and provide the resulting output as desired.

e. Support

The new system that runs is used following company requirements. During its lifetime, the system will be periodically maintained. Changes are made if a problem arises or if there is a new need. Furthermore, the Company will use the improved system.

A good system must run as expected, so it needs continuous maintenance. The maintenance process continues regularly both from the system and to the improvement of the system if there are obstacles in its operation due to technical problems that are not indicated in the system development process.

3. RESULTS AND DISCUSSION

3.1. SOFTWARE REQUIREMENT ANALYSIS

3.1.1. ANALYSIS PHASE

With the bridging system, coari and codeco data reporting of bulk goods piled in incoming/outgoing storage tanks will be sent using a username and password and fstream following the web service owned by the DJBC online TPS application. Previously PT. Polychemlindo first registered service access to the DJBC peacock sector to get a username and password.

Each reporting data sent from the Polychemlindo to the DJBC Online Application must have a different ref_number. This because the ref number element is the primary key in the DJBC Online TPS system, which will automatically create as a key when sending the unloading list or the hoarding expense list.

The requirements specification of the bridging system for loading and unloading goods reporting application with DJBC TPS online application is as follows:

a. The reporting clerk gets a login
b. The clerk can manage the loading and unloading data of the landfill
c. Officers can send data to the DJBC Online TPS application
d. The clerk can print the Goods Expense Report
e. The clerk can log out

3.1.2. USE CASE DIAGRAMS

The use case diagram is used to describe what the system should do. The use case diagram provides a way to describe the system's external view and its interactions with the outside world. Figure 1 shows the reporting system of loading and unloading for hoarding goods.

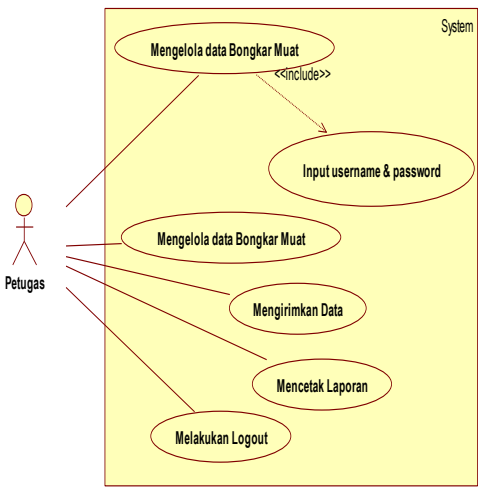

Figure 1. Use case diagrams

3.1.3. ACTIVITY DIAGRAM

The use case diagram is used to describe the activities that occur in the system. From the first to the end, this diagram shows the steps in the working process of the system that we make (Fig. 2).

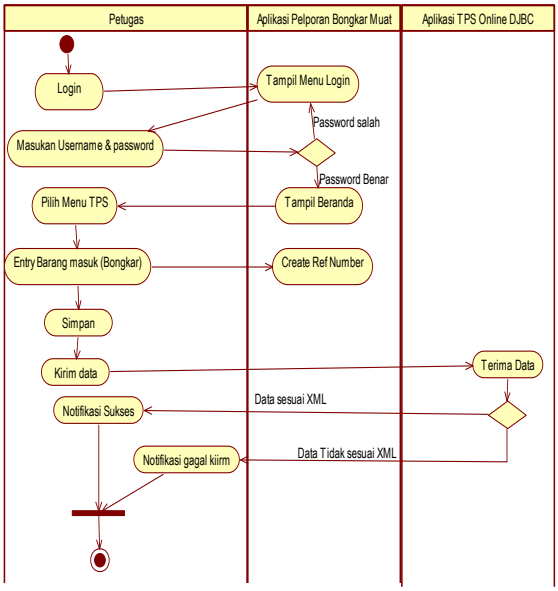

Figure 2. Activity diagrams

3.2. DESAIN

3.2.1. ER-DIAGRAM

At the design stage, the database will be designed, software architecture design, and interface design of the document reporting system for incoming/outgoing goods at PT. Polychem Lindo (Fig. 3).

3.2.2. LOGICAL RECORD STRUCTURE

Figure 4 presents the logical record structure for our design system.

3.2.3. COMPONENT DIAGRAM

Figure 5 shows the component diagram of the software. At the first stage, the software provides a front page which that process flow to the web service.

3.2.4. DEPLOYMENT DIAGRAM

Figure 6 presents the deployment diagram of the bridging system. This bridging system was integrating between PT. Polychemlindo server with Direktorat Jenderal Bea Cukai server.

3.3. USER INTERFACE

3.3.1 LOGIN PAGE

To access the TPS (Temporary Hoarding) application online via the internet to http://www.polychemlindo.com/ login will appear, as shown in figure 7.

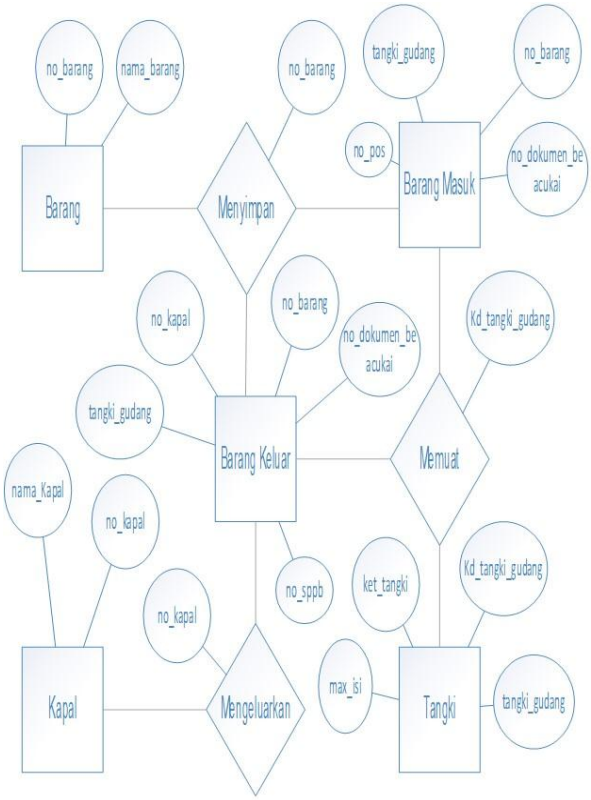

Figure 3. Er-diagrams

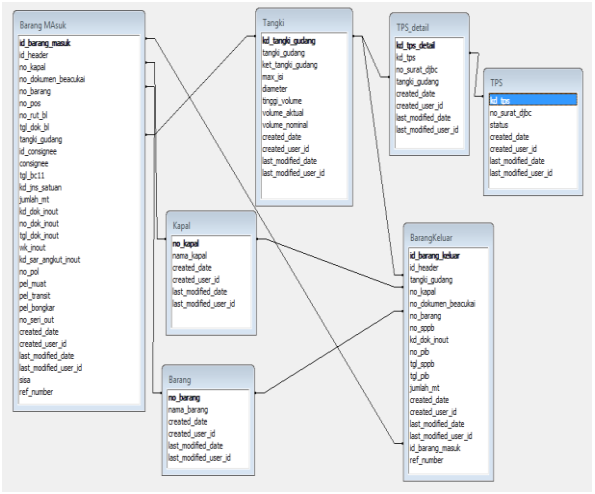

Figure 4. Logical Record Structure

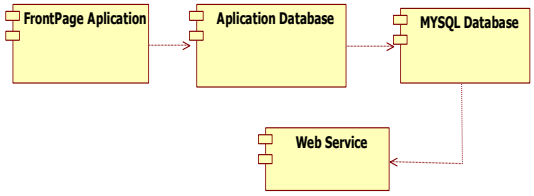

Figure 5. Component Diagram

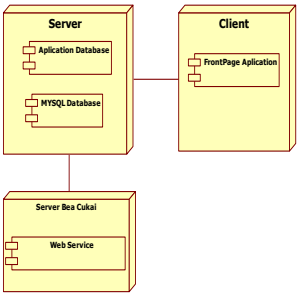

Figure 6. Deployment Diagram

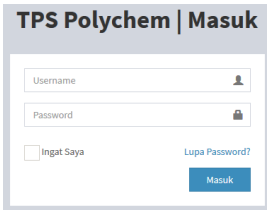

Figure 7. Login

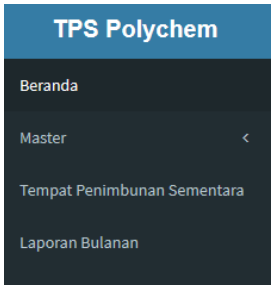

Figure 8. Main Menu

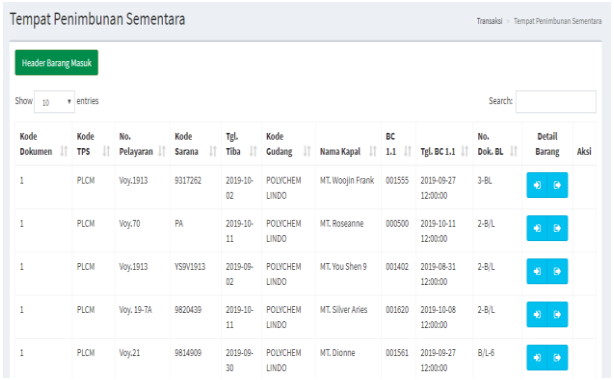

Figure 9. Transaction Menu

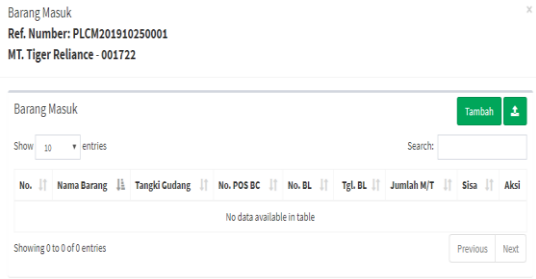

Figure 10. Item Entry Menu

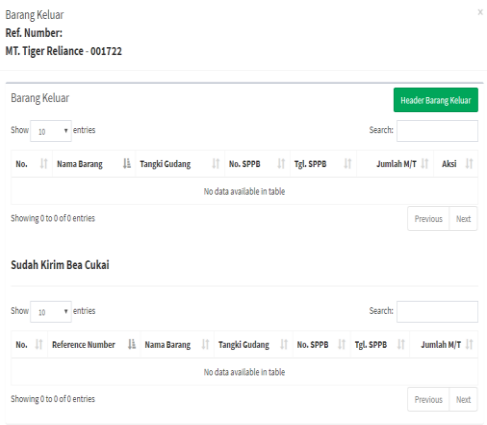

Figure 11. Item Exit Menu

3.3.2 HOME PAGE

The main page consists of the homepage, master, temporary storage area, and monthly reports presented in figure 8.

3.3.3 TRANSACTION MENU TEMPORARY STORAGE

This temporary hoarding menu page serves to enter incoming and outgoing goods, where any incoming goods that have received a permit from customs can already be reported by integrating customs web services (Fig.9).

3.3.4 ITEM ENTRY MENU PAGE

This entry menu functions as the entry of incoming goods, which will be stockpiled at Polychemlindo then reported to customs through the web service (Fig.10).

3.3.5 ITEM EXIT MENU PAGE

This outgoing goods menu page functions as the entry of outgoing goods from the landfill at Polychemlindo, the data is reported to customs through the web service (Fig. 11).

3.4. DATA DELIVERY

On data transmission, it will be connected to the web service that has been provided by customs. The web service is used to report data on imported goods unloaded / stockpiled and their expenditure, as well as reports on exported goods entering the TPS and loading/loading data. The XML Schema forms the data to be sent to the customs as follows:

'<? xml version = "1.0" encoding = "utf-8"?>
<DOCUMENT xmlns = "cocotangki.xsd">
<COCOTANGKI>
<HEADER>
<KD_DOK> '. $ GetHeader [' kd_dok '].' </KD_DOK>
<KD_TPS> '.substr ($ getHeader [' kd_tps '], 0.4).' </KD_TPS>
<NM_ANGKUT> '.substr ($ getHeader [' nm_ transport '], 0.25).' </NM_ANGKUT>
<NO_VOY_FLIGHT> '. Substr ($ getHeader [' no_voy_flight '], 0.20).' </NO_VOY_FLIGHT>

```
    <CALL_SIGN> '.substr ($ getHeader [' call_sign '], 0.8).' </CALL_SIGN>
    <TGL_TIBA> '.substr (date (' Ymd ', strtotime ($ getHeader [' tgl_tiba ')]), 0.8).' </TGL_TIBA>
    <KD_GUDANG> '. Substr ($ getHeader [' kd_gudang '], 0.4).' </KD_UDUD>
    <REF_NUMBER> '. Substr ($ ref_number, 0.16).' </REF_NUMBER>

</HEADER>
<Detail>
'$ details.'
</DETIL>
</COCOTANGKI>
</DOCUMENT> '
);
$ result = $ soap-> CoCo Tank ($ data);
date_default_timezone_set ('Asia / Jakarta');
$ this-> MY_Model-> updateData ('tps_header', array ('ws_result' => $ result-> CoCoTangkiResult, 'ws_date' => date ('Ymd H: i: s')), array ('ref_number' = = > $ ref_number));
$ _userdata = $ this-> ion_auth-> user () -> row ();
$ this-> MY_Model-> insertData ('log',
array (
'data' => json_encode ($ data),
'datetime_send' => date ('Y-m-d H: i: s'),
'userid' => $ this -> _ userdata-> id,
'return' => $ result-> CoCoTangkiResult
)
);
$ res = array ('success' => (strpos ($ result-> CoCoTangkiResult, 'Success')? true: false), 'message' => $ result-> CoCoTangkiResult);
echo json_encode ($ res);
}
$ data = '
        <TANGKI>
        <SERI_OUT> '.substr ($ getDetail [' no_seri_out '], 0.11).' </SERI_OUT>
        <NO_BL_AWB> '. Substr ($ getDetail [' no_rut_bl '], 0.30).' </NO_BL_AWB>
        <TGL_BL_AWB> '. Substructure (date (' Ymd ', strtotime ($ getDetail [' tgl_dok_bl ')]), 0.8).' </TGL_BL_AWB>
        <ID_CONSIGNEE> '.substr ($ getDetail [' id_consignee '], 0.15).' </ID_CONSIGNEE>
        <CONSIGNEE> '.substr ($ getDetail [' consignee '], 0.60).' </CONSIGNEE>
        <NO_BC11> '.substr ($ getHeader [' no_dokumen_beacukai '], 0.6).' </NO_BC11>
        <TGL_BC11> '.substr (date (' Ymd ', strtotime ($ getDetail [' tgl_bc11 ')]), 0.8).' </TGL_BC11>
        <NO_POS_BC11> '. Substr ($ getDetail [' no_pos '], 0.12).' </NO_POS_BC11>
        <NO_TANGKI> '.substr ($ getDetail [' tank_gudang '], 0.20).' </NO_TANGKI>
        <JML_UIT>> '. Substr ($ getDetail [' number_mt '], 0.8).' </JML_UIT>
        <JNS_SATUAN> '.substr ($ getDetail [' kd_jns_united '], 0,2).' </JNS_SATUAN>
        <KD_DOK_INOUT> '. $ GetDetail [' kd_dok_inout '].' </KD_DOK_INOUT>
        <NO_DOK_INOUT> '. $ No_dok_inout.' </NO_DOK_INOUT>
        <TGL_DOK_INOUT> '. Substructure (date (' Ymd ', strtotime ($ getDetail [' tgl_dok_inout ')]), 0.8).' </TGL_DOK_INOUT>
        <WK_INOUT> '.substr (date (' YmdHis ', strtotime ($ getDetail [' wk_inout ')]), 0.16).' </WK_INOUT>
        <KD_SAR_ANGKUT_INOUT> '.substr ($ getDetail [' kd_sar_ freight_inout '], 0.1).' </KD_SAR_ANGKUT_INOUT>
        <NO_POL> '.substr ($ getDetail [' no_pol '], 0.10).' </NO_POL>
        <PEL_MUAT> '.substr ($ getDetail [' pel_load '], 0.5).' </PEL_MUAT>
        <PEL_TRANSIT> '.substr ($ getDetail [' pel_transit '], 0.5).' </PEL_TRANSIT>
        <PEL_BONGKAR> '.substr ($ getDetail [' pel_block '], 0.5).' </PEL_BONGKAR>
        </TANGKI> ';
        return $ data;
        }
public function postDataAll ($ ref_number, $ details)
        {
        $ soap = new SoapClient ('https://tpsonline.beacukai.go.id/tps/service.asmx?wsdl',array (
        "stream_context" => stream_context_create (
        array (
        'ssl' => array (
        'verify_peer' => false,
        'verify_peer_name' => false,
        )
        )
        )
        ));
        $ getHeader = $ this-> getData ('tps_header',
        array (
        'where' => array (
        'ref_number' => $ ref_number
        )
        )
        );
        $ this-> load-> helper ('xml');
```

3.5. CONCLUSION

From the discussion above regarding the design of Bridging System Application Reporting on the unloading / stockpiling of imported goods at PT. This Polychemlindo can be concluded as follows:

1. The making of this web application is expected to be able to assist officers reporting data on imported goods, unloaded/stockpiled, and released, as well as reports on data on exported goods entering the TPS and loading/loading data. Especially in the process of sending data to customs, so reporting is more efficient and reaction time because it is directly reported at one time.

2. It is providing convenience to PT. Polychemlindo to report loading/unloading data, as well as monitoring the remaining goods contained in each tank.

REFERENCES

1. Belloum ASZ, Koulouzis S, Wiktorski T, Manieri A. Bridging the demand and the offer in data science. Concurr Comput Pract Exp Publ. 2019;31(e5200):1–14.
2. Vtyurina A, Fourney A, Morris MR, Findlater L. Bridging Screen Readers and Voice Assistants for Enhanced Eyes-Free Web Search. In: Proceedings of the 2019 World WideWeb Conference (WWW '19). San Francisco, CA, USA: nternational World Wide Web Conference Committee; 2019. p. 3590–4.
3. Katzin IE, Kumnick PL, Us AZ, Harris TD, Us CA, Faith P, et al. (12) United States Patent The MERCHANT - CONSUMER BRIDGING PLATFORM. Vol. 2. United states: 13 / 366 , 083; US 10,204,327 B2, 2019.
4. Mardi. Sistem Informasi Akuntansi. Sikumbang R, editor. Jakarta: Ghalia Indonesia; 2011. 174 p.
5. Admin Web Bea dan Cukai. Bea Cukai Belawan Adakan Sosialisasi Implementasi TPS Online. KEMENTERIAN KEUANGAN DIREKTORAT JENDERAL BEA DAN CUKAI. 2018.
6. Lee W, Lee S. Design and Implementation of Bridging System for Integrating Smart Home with Connected Vehicles. In: 2018 International Conference on Information and Communication Technology Convergence (ICTC). IEEE; 2018. p. 1043–8.
7. Wibisono MI, Awaludin M, Sistem I, Planning ER. Evaluasi integrasi sistem enterprise dengan sap ecc 6.0 pada perusahaan fmcg 1,2. In: Prosiding SINTAK 2017. Fakultas Teknologi Informatika Universitas Stikubank; 2017. p. 237–43.
8. Wahyudin Y, Suhada S, Hidayatulloh T, Firmansyah DA. Rancang Bangun Bridging System Aplikasi Simrs Dan Aplikasi Virtual Claim Di Rumah. J SWABUMI. 2019;7(1):84–9.
9. Pressman RS. Software Engineering A Practitioner's Approach. 7th ed. Schilling FM, editor. New York: Mc Graw Hill;
10. Kamuslengkap.com. Arti kata " Bridging " - Istilah Komputer [Internet]. Kamuslengkap.com. 2020 [cited 2020 Nov 5]. p. 1–6. Available from: https://kamuslengkap.com/kamus/komputer/arti-kata/bridging
11. Herman. 22 Rumah Sakit Provider BPJS Kesehatan Sudah Implementasikan "Bridging System" [Internet]. beritasatu.com. 2014 [cited 2020 May 11]. p. 1–2. Available from: https://www.beritasatu.com/kesehatan/194257-22-rumah-sakit-provider-bpjs-kesehatan-sudah-implementasikan-bridging-system
12. Hosono S, Shimomura Y. Bridging On-site Practices and Design Principles for Service Development. Procedia CIRP [Internet]. 2017;60:422–7. Available from: http://dx.doi.org/10.1016/j.procir.2017.02.018
13. Booth D, Haas H, McCabe F (Fujitsu L of A, Newcomer E (Iona), Champion M (Software A, Ferris C (IBM), et al. Web Services Architecture [Internet]. w3.org. 2004. Available from: https://www.w3.org/TR/ws-arch/wsa.pdf
14. Kamarudin K, Kusrini K, Sunyoto A. Uji Kinerja Sistem Web Service Pembayaran Mahasiswa Menggunakan Apache JMeter. J RESPATI J Inform Telekomun Elektron Vol [Internet]. 2018;13(1). Available from: http://jti.respati.ac.id/index.php/jurnaljti/article/view/215
15. Battle R, Benson E. Bridging the semantic Web and Web 2 . 0 with Representational State Transfer (REST). Web Semant Sci Serv Agents World Wide Web. 2008;6:61–9.