

Designing System Integration Testing to Test Reduction and Migration Payment Transaction Data at PT Telekomunikasi Indonesia, Tbk

Labbaika Putri Tiovani¹, Warih Puspitasari¹ and Umar Yunan K. S. Hedyanto¹

¹Information Systems, Telkom University, Jl. Telekomunikasi Jl. Terusan Buah Batu No.01, Bandung, Indonesia

Keywords: ERP, SAP, Data Reduction, Data Migration, SIT.

Abstract: To run transactions and operational activities, PT Telekomunikasi Indonesia, Tbk as one of the biggest telecommunication service providers in Indonesia requires an Enterprise Resource Planning (ERP) application that can integrate all business processes in real-time, namely SAP. Telkom Indonesia has been providing home telephone service since they established. By the time, POTS customers are increasing every year, hence data that is managed and stored in a database is increasing as well. Due to that problem, Telkom Indonesia experienced several issues such as failure during data backup, data restoration that took a long time that it interferes system performance. Because of that, Telkom Indonesia will migrate and reduce data. To ensure the consistency, completeness, and accuracy of data before and after the migration and reduction process, a test is needed with System Integration Testing (SIT) methodology. This study aims to design SIT on POTS customer payment transactions at PT. Telekomunikasi Indonesia, Tbk.

1 INTRODUCTION

ERP (Enterprise Resource Planning) is a system that is used to integrate all business processes in a company, especially an enterprise scale. ERP systems will collect transactional data on a company from a variety of different sources, then save, manage and analyse to help eliminate data redundancy and duplication, and can help companies make data-driven decisions (Rao and Kudtarkar, 2018).

Telkom Indonesia is a State-Owned Enterprise (BUMN) that is engaged in information and communication technology (ICT) services and telecommunications networks in Indonesia participating in the collaboration with other state-owned companies (Saputra, 2018) by implementing an ERP system. From several available ERP software, Telkom Indonesia chose SAP R / 3 which can integrate business units such as logistics, Human Resources, and Finance.

To transform into a digital telecommunication company, Telkom Indonesia offers six customer-oriented portfolio products, namely fixed, mobile, network infrastructure, wholesale, enterprise digital, and consumer digital. One of the main services is fixed-line telephone which is included in the fixed portfolio on fixed telephone segmentation (POTS). Recorded in the annual report of Telkom Indonesia in the year of 2017, POTS customers are increasing

every year shown in the following Figure 1:

Type of Customer	Years Ended December 31		
	2015	2016	2017
Fixed Wireline (POTS)	10.277.000	10.633.000	10.957.000

Figure 1: Number Of Fixed Phone Customers.

Based on Figure 1 number of fixed phone customers, it can be seen that with the increase in the number of customers in the POTS segment each year, the amount of data managed and stored in one database server will increase as well. This can cause system performance to decrease, restoration and data backup takes longer and the possibility of failing data backup is the main issue now. Currently, the data stored on Telkom Indonesia's database server is 34TB, to be able to maximize storage capacity, they need to reduce the capacity of database to 4TB, therefore, it can improve system performance. The reduction cannot be performed directly on production server due to the payment data keep increasing, backup all data and restored to simulation server. To ensure data consistency, accuracy and completeness, a test is needed using the system integration testing (SIT) method by conducting validation using quality assessment data. SIT will help to know if defects occur and give a solution to them before users try out the system (Ganesh et al., 2014). Therefore this study

aims to design SIT to ensure data migration and reduction for POTS customer payment transactions at PT Telekomunikasi Indonesia, Tbk so that it will help them to make decisions whether the reduced data is ready to be implemented or not.

2 LITERATURE STUDY

Literature studies are conducted to support research by seeking and reading information and material in relevant books and journals.

2.1 ERP

Enterprise Resource Planning (ERP) is an information system used by companies to help integrate and optimize all existing business processes such as factories, logistics, accounting, distribution, finance, and human resources (Al Aziz et al., 2018). Whereas according to (Tsai et al., 2010) Enterprise Resource Planning (ERP) is a software package that can provide the required data in full and accurate information where data and information can be generated at any time the user needs it.

ERP systems can help companies integrate all their operations that have function as computing environments throughout the company they have such as using a centralized database where stored and transmitted data is always consistent in real-time on all business functions (Monk and Wagner, 2012).

2.2 SAP

System Applications and Products in data processing or commonly called SAP is an application created by five former IBM employees namely Hans-Werner Hector, Klaus Tschira, Claus Wellenreuther, Dietmar Hopp, and Hasso Platter in 1972. This application was made to integrate all departments that is on the company on a centralized computer database (Ovidiu and Dascalu, 2010).

2.3 Types of Testing

There are three types of testing, namely Unit Testing, System Integration Testing, and User Acceptance Testing. The following is the explanation of all types of testing:

2.3.1 Unit Testing

Unit testing is the lowest level of system testing on SAP transactional level applications that are carried

out by developers. Negative and positive tests will be conducted on this testing phase. The main focus of this test is the master data, negative-positive testing, functional transaction conditions, and security roles and profiles (Ganesh et al., 2014).

2.3.2 System Integration Testing (SIT)

System Integration Testing (SIT) is a series of SAP transaction testing to check whether all integrated components are running according to the required specifications (Helfen and Trauthwein, 2010). The testing process begins with defining the needs and testing strategies (Ganesh, et al., 2014). This test is conducted by end-to-end for a number of related modules such as order-to-cash, purchase-to-pay, and hire-to-retire by using some input data that is done with several iterations (Helfen and Trauthwein, 2011).

2.3.3 User Acceptance Testing (UAT)

UAT is done by end-user of the system. The end-user will do the test based on test case so that will know how the end-user uses the system on production environment. Configuration and testing team are also involved to help when defects occur. This test will bring out a document as proof that the system that has been developed is accepted and fit to user's requirements (Ganesh et al., 2014).

2.4 Life Cycle Testing Phase

Life cycle testing is divided into four main phases namely preparation test, test design, execution test, and test evaluation. The following is an explanation of the testing phase (Dalal and Chhillar, 2012) (Hooda and Chhillar, 2015):

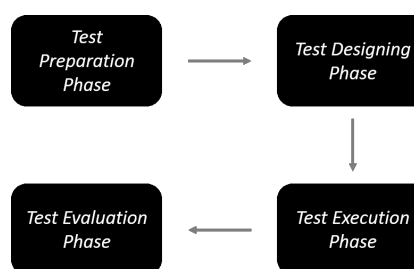


Figure 2: Life Cycle Testing Phase.

2.4.1 Test Preparation Phase

In the first phase, the test preparation explains how testing will be achieved. In this phase things will be done, such as defines the scope and objective of testing, identifies strategies or testing approaches, defines

risk, allocates resources, schedules testing, makes a data test (Dalal and Chhillar, 2012).

2.4.2 Test Designing Phase

In the second phase of the test design, which included the making of a test case and preparation of data tests Standard templates must be used to prepare test cases (Dalal and Chhillar, 2012). The test case specifies test inputs or data, execution conditions, and expected results. The test case is made by the QA team manually or automatically generated (Jamil et al., 2016).

2.4.3 Test Execution Phase

Tests will be conducted by the tester with the main purpose of finding defects. In this phase, the test case will be executed and the result will be an information pass / fail in the test script. When a tester finds defects, it has to give a solution to the defects found. The results of this phase are test status reporting and defects handling (Dalal and Chhillar, 2012).

2.4.4 Test Evaluation Phase

This phase will ensure test all system testing, integration, and user acceptance have passed the test and the decision of the evaluation results is taken if all requirements have been tested. At this stage, it also analyses defects and ensures documentation of each testing process (Hooda and Chhillar, 2015).

2.5 Test Scenario

The test scenario describes specifically the detailed instructions for testing. The test scenario is owned by the configuration team but there is also participation from members of the testing team and the development team (Namjoshi et al., 2011).

2.6 Test Case

A test case is a set of test inputs, execution conditions, and the expected result can be developed for a particular purpose so that it can validate specific functions in an application that will be tested. The test case forms the foundation for designing and developing test scripts (Lewis, 2017).

2.7 Test Script

Test script is a series of test cases arranged in a single flow of execution testing to test the functionality of a particular business. The test script contains what conditions will be tested, the number of test cases tested

on the script, prerequisites, test data needed for testing, and instructions to verify the test results. Test scripts are needed for testers so that they can still run tests not only limited to certain people who make test scripts (Lewis, 2017).

2.8 Data Quality

Data quality is a data level that states that the data is accurate, complete, timely updated, and consistent with business needs (Mosley and Mosley, 2008). In the real world, determining the quality of data is very possible to check whether each data in the database is correct, from where the data originates, and the cause of the error (Maydanchik, 2007). Quality data is very important due to several reasons, such as increased self-confidence when making decisions, reducing the risk of wrong decisions, avoiding the effects of complications from contaminated data (Mosley and Mosley, 2008).

3 METHODOLOGY

The research methodology in this study is shown on Figure 3. There are four phases to design SIT on this study that are review, collect, analysis, and interpretation. There will be four deliverables, which are test scenario, test case, test data, and test script. All deliverables will help to conduct the test.

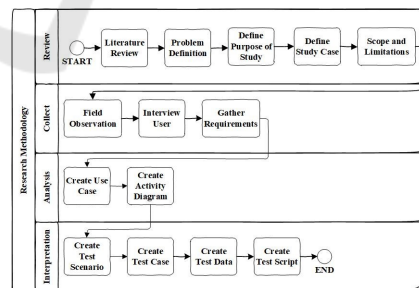


Figure 3: Research Methodology.

3.1 Limitations

The main focus of this study is designing a test therefore there are two phase namely test designing phase to design system integration testing for data reduction and migration test.

4 ANALYSIS AND RESULT

4.1 Test Designing Phase

4.1.1 Use Case

To be able to create a test script template, we need to analyse use case and activity diagram of the system. The results of the analysis will be a reference in making test scenarios, test cases, data tests so that we can make a template for test script to test the system.

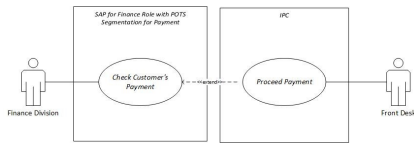


Figure 4: Use Case Diagram of SAP and IPC on POTS Segmentation for Payments.

Figure 4 is a use case diagram of the SAP application and IPC (Integrated Payment Center) of the Finance user role division with POTS segmentation for payments at Telkom Indonesia. Use case diagram illustrates how the system works. According to Figure 4, the finance division can check customer payments using SAP applications and the front desk processes POTS customer payments with the IPC system.

4.1.2 Activity Diagram of Proceeding Payment

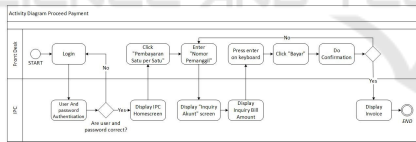


Figure 5: Activity Diagram of Proceeding Payment.

Figure 5 is an activity diagram for proceeding payment. The Activity diagram shows how workflows when processing customer payments using the IPC system. The output of this activity is payment invoice as proof that customers have completed their transaction.

4.1.3 Activity Diagram of Checking Customer's Payment

Figure 6 is an activity diagram to check customer payment. It shows how the system works to display all payment transaction of a particular customer.

4.2 Create Test Scenario

After analyzing the use case and activity diagram, the test scenario will be made as shown in Figure 7. Test

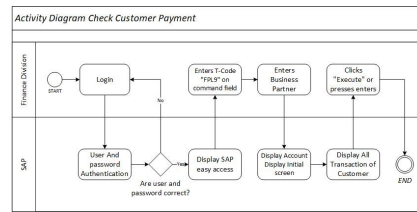


Figure 6: Activity Diagram of Checking Customer's Payment.

scenario is a flow of how the test will be executed. There are 2 scenarios, proceed payment and check customer's payment.

Use Case ID	Scenario ID	Test Scenario	Expected Outcome	Test Case ID
UC01	TS01	Proceed the payment	Inquiry bill amount of customer will be displayed based on telephone number	TC01
UC02	TS02	Check customer's payment	Customer's transactions will be displayed based on business partner	TC02

Figure 7: Test Scenario.

4.3 Create Test Case

Figure 8 and Figure 9 are test cases for proceeding payments and checking customer payments. The test case is used as a condition that must be tested in this study. There are 2 systems on this study, they are Integrated Payment Centre (IPC) and SAP. Each case has different expected results.

4.3.1 Test Case for Proceeding Payment

Test Case ID : TC01
 Test Title : Proceed payment
 Preconditions : User wants to check customer's payment and proceed it.

Steps No.	Test Steps	System	Transaction	Expected Result
1.	Displaying inquiry bill amount	IPC	Inquiry	Inquiry bill amount will be displayed according to the phone number of each customer
2.	Proceeding customer's payment	IPC	Inquiry	The output of this process is the payment invoice as proof to the customer that their transaction has proceeded.

Figure 8: Test Case for Proceeding Payment.

4.3.2 Test Case for Checking Customer's Payment

Test Case ID : TC02
 Test Title : Check customer's payment
 Preconditions : User wants to check customer's payment

Steps No.	Test Steps	System	Transaction	Expected Result
1.	Displaying all payment transaction of each customer	SAP	FFL9	All payment transactions data will be displayed for each customer according to business partner number

Figure 9: Test Case for Checking Customer's Payment.

4.4 Create Test Data

After the test case is made, a data test will be made to perform the test. Figure 10 and Figure 11 are input data that has been adjusted to the test conditions.

4.4.1 Test Data for Proceeding Payment

Steps No.	Test Steps	System	Transaction	Data Field	Data Input
1.	Displaying inquiry bill amount	IPC	Inquiry	Phone Number	XXXXXXXXXX
2.	Proceeding customer's payment	IPC	Inquiry	Confirmation	yes

Figure 10: Test Case for Proceeding Payment.

4.4.2 Test Case for Checking Customer's Payment

Steps No.	Test Steps	System	Transaction	Data Field	Data Input
1.	Displaying all payment transaction of each customer	SAP	FPL9	Business partner	XXXXXXXXXX

Figure 11: Test Data for Checking Customer's Payment.

4.5 Create Test Script

After designing test scenario, test case, and test data, then test script is created as a standard template to perform the test. Test script template contains system, transaction, description of transaction, detailed step on how to run the test, testing data. The result of the test will be recorded on the script as for the documentation to help future development.

There are two test script templates in this study to perform a test, the test script for proceeding payment and a test script for checking customer payment. Following is the test script:

4.5.1 Test Script for Proceeding Payment

Step	System	Transaction	T-Code Description	Detailed Step	Testing Data	Result (ok/fail/td)
1.	IPC	-	-	System will display "IPC Home Screen" User clicks "Inquiry" System will display "Inquiry"	1. XXXXXXXXX	
2.	IPC	-	-	1. User enters "No Penyanggill" User presses "Enter" on keyboard System will display "Inquiry Pembayaran" based on "No Penyanggill" User clicks "Bayar" System will display "Konfirmasi" User does confirmation		
3.	IPC	-	-			

Figure 12: Test Data for Checking Customer's Payment.

4.5.2 Test Script for Checking Customer's Payment

Step	System	Transaction	T-Code Description	Detailed Step	Testing Data	Result (ok/fail/td)
1.	SAP	FPL9	Display Account Balance	System will display "Account Display: balances" 1. User enters "Business Partner" User presses "Enter" on keyboard	1. XXXXXXXXX	

Figure 13: Test Data for Checking Customer's Payment.

5 CONCLUSIONS

Integration testing to test the data consistency between transactions on two integrated systems. Test designing occurs on the second phase of lifecycle testing phase which is test designing phase where test scenarios, test cases, and data tests are created. This test is designed to validate the quality of the data by looking at those four data quality dimensions. Hence, the design of the test is focused on master data and row data on transaction table payments.

Test script payment transaction design is prepared and created according to use case, activity diagrams for payment transaction, test scenario for payment, and test data for payment. Therefore, this test script can be used in the next phase of testing. The output of this study are test scenario, test case, test data and test script which are the design of integration testing of POTS customer payment transactions at PT Telekomunikasi Indonesia which focused on data validation.

The result of the test will ensure whether data that has been migrated and reduced in the new server is consistent, complete, and accurate by comparing the expected result of the test and the output the test, analyzing the defects and create solution for them. So that the final result will be a successful data reduction that is ready to be implemented to the production server and will be used on daily operation.

For further development of this research, this study can be as a guide by adjusting the purpose of research and case studies for each research and also developing this research to the last phase which is the evaluation phase of testing. Data testing can be done with several iterations to assess consistency, accuracy, and completeness of data.

REFERENCES

- Al Aziz, R., Amalia, F., and Brata, A. H. (2018). Pembangunan sistem enterprise resource planning pada griya laundry. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer e-ISSN*, 2548:964X.
- Dalal, S. and Chhillar, R. S. (2012). Software testing-three p's paradigm and limitations. *International Journal of Computer Applications*, 54(12).
- Ganesh, K., Mohapatra, S., Anbuudayasankar, S., and Sivakumar, P. (2014). *Enterprise resource planning: fundamentals of design and implementation*. Springer.
- Helfen, M. and Trauthwein, H. M. (2010). *Testing SAP solutions*. SAP PRESS.
- Hooda, I. and Chhillar, R. S. (2015). Software test process, testing types and techniques. *International Journal of Computer Applications*, 111(13).
- Jamil, M. A., Arif, M., Abubakar, N. S. A., and Ahmad, A. (2016). Software testing techniques: A literature review. In *2016 6th International Conference on Information and Communication Technology for The Muslim World (ICT4M)*, pages 177–182. IEEE.
- Lewis, W. E. (2017). *Software testing and continuous quality improvement*. Auerbach publications.
- Maydanchik, A. (2007). *Data quality assessment*. Technics publications.
- Monk, E. and Wagner, B. (2012). *Concepts in enterprise resource planning*. Cengage Learning.
- Mosley, M. and Mosley, M. (2008). *The DAMA dictionary of data management*. Technics Publications, LLC.
- Namjoshi, K., Zeller, A., and Ziv, A. (2011). Hardware and software: Verification and testing. *Lecture Notes in Computer Science*, 6405.
- Ovidiu, S. and Dascalu, C. (2010). The advantages and risks of using an erp system in the context globalization. *International Journal of Modern Manufacturing Technologies*, 2(2):83–88.
- Rao, S. and Kudtarkar, K. (2018). Implementation of odoo erp for business applications. *IOSR Journal of Computer Engineering*, pages 32–39.
- Saputra, E. (2018). Jaringan pemberitaan pemerintah (jpp).
- Tsai, W.-H., Chen, S.-P., Hwang, E. T., and Hsu, J.-L. (2010). A study of the impact of business process on the erp system effectiveness. *International Journal of Business and Management*, 5(9):26.