



Service Information System Animal Product Testing at Bogor's Quality Testing & Certification Center for Animal Products (BPMSPH)

Sandra Jamu Kuryanti¹, Siti Nur Khasanah², Eko Yulianto³

^{1,2,3}Information Systems, University of Bina Sarana Informatika, Kramat Raya No 98, Central Jakarta, 10420, Indonesia

E-mail: sandra.sjk@bsi.ac.id, siti.skx@nusamandiri.ac.id, eko.eui@bsi.ac.id

ARTICLE INFO

ABSTRACT

Article history:

Received: 10/03/2021

Revised: 20/03/2021

Accepted: 21/03/2021

Keywords:

Information System Design,
Testing Service, Website

The Bogor Animal Product Quality & Certification Testing Center (BPMSPH) in carrying out the testing service process still uses the system of meeting directly with service users to carry out tests and use paper media for recording service user data. Services and data processing like this require considerable effort and time, the risk of damage to paper as a medium for recording service user data is also a problem. Therefore, a web-based animal product testing service information system design was made at the Bogor Animal Product Quality & Certification Testing Center (BPMSPH) which uses the waterfall model as a software development method and the data collection techniques used consist of observations, interviews and literature studies. This designed system provides online testing services where service users can fill out test forms anywhere without having to go directly to BPMSPH. Hopefully this web-based testing service system can make it easier for service users to carry out tests at the Bogor Animal Product Quality & Certification Testing Center (BPMSPH).

Copyright © 2021 Jurnal Mantik.

All rights reserved.

1. Introduction

The demand for inspection and quality testing services for animal products continues to increase in terms of both the number of samples and the types of testing required (2017 BPMSPH Performance Report). BPMSPH Bogor has a testing service mechanism, that is, service users must communicate with the BPMSPH Bogor management to make test submissions, the testing service process that is established is through available telephone contacts or by visiting the BPMSPH Bogor management directly to make test submissions. Furthermore, when making a test submission, service users are required to fill out a test submission form. The test submission form is intended as a place for service user confirmation as well as for the purposes of archiving service user data for the management of BPMSPH Bogor.

In the testing service mechanism described earlier, it can be seen that the testing service process that occurs between the management of BPMSPH Bogor and prospective service users has been well managed, but the problem is that the testing service process is still by telephone and the recording of service user data is still ongoing. manually using paper, of course, this is very risky because paper media is easier to cause accidental or intentional damage.

With the problems faced by BPMSPH Bogor, the solution taken is to improve the management of the testing service process at BPMSPH Bogor by making a web-based testing service system. The creation of a web-based testing service system at BPMSPH Bogor is very important because the system will later make it easy for service users to submit test submissions to BPMSPH Bogor online, can provide security for all data, and make it easier for managers to process data on the number of testing services and User data for testing services at any time for the management of the Bogor BPMSPH management.

2. Literature Review

2.1 Unified Modeling Language (UML)

UML is a visual language for modeling and communication about a system using diagrams and supporting texts [1]

2.2 Activity Diagram

Activity diagrams describe the workflow (work flow) or activities of a system or business process or menu in the software [1]



2.3 Sequence Diagram

Sequence diagrams describe the behavior of objects in use cases by describing the life time of objects and messages sent and received between objects [1]

2.4 Enterprise Relationship Diagram (ERD)

Entity Relationship Diagram is a diagram that describes the relationship between tables and fields in a database system [2]

There are three types of relations between tables in an ERD chart [2] :

- a. One To One
describes the relationship of one field in the first table to the second table. This relation is the simplest.
- b. One to Many
This relation describes the relationship of one field in the first table to two or more fields in the second table.
- c. Many to Many
The relation in this case is an N to N relation, which means that one or more fields in the first table can be connected to one or more.

2.5 Logical Record Structure (LRS)

The Logical Record Structure (LRS) is formed with a number of the record type ". Some record types are described by rectangular boxes and with unique names. The difference between LRS and E-R diagrams is that the record type name is outside the record type field box is placed. LRS consists of links between record types. This link shows the direction of one record type, the fields that appear on both record type links. The depiction of the LRS begins by using an understandable model. Two methods can be used, starting with the relationship between the two models which can be converted to LRS, the other method starting with ER-Diagram and directly converted to LRS [3]

3. Research Method

The research method used by the author is the waterfall model research method, which is often called the linear sequential model or classic life flow. The waterfall model provides a systematic as well as sequential approach to software development, starting with the specification of user requirements and then continuing through the stages of planning, modeling, construction, and handing over the system to customers / users [1]

a. Software Requirements Analysis

The process of gathering needs is carried out intensively and specifically against device needs in order to understand what the user needs. The specification of software requirements at this stage needs to be documented.

b. Design

Software design is a multi-step process that focuses on the design of a software program including data structures, software architecture, interface representations, and coding procedures. This stage translates the software requirements from the needs analysis stage into a design representation so that it can be implemented into a program at a later stage. The software design produced at this stage also needs to be documented.

c. Program Code Creation

The design must be translated into a software program. The result of this stage is a computer program in accordance with the designs that have been made at the design stage.

d. Testing

Testing focuses on the software logically and functionally and ensures that all parts are tested. This is done to minimize errors and ensure the resulting output is as desired.

e. Support or Maintenance

It does not rule out the possibility of a software change when it is sent to the user. Changes can occur due to errors that appeared that were not detected during testing or the software had to adapt to a new environment. The support or maintenance stage can repeat the development process from the specification analysis stage to new software changes.

4. Discussion

4.1 Use Case Diagram

Use cases or use case diagrams are modeling for the behavior of an information system to be created [4]

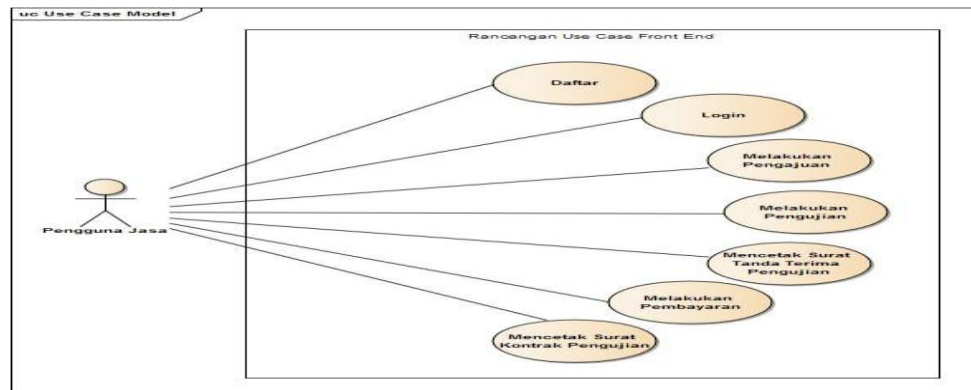


Fig 1. Service User Use Case Diagram

In the usecase image above illustrates the Service User Diagram which is the process of making a test submission

4.2 Diagram Activity

Activity diagrams describe the workflow (work flow) or activities of a system or business process or menu in the software

a. Submission Activity Diagram

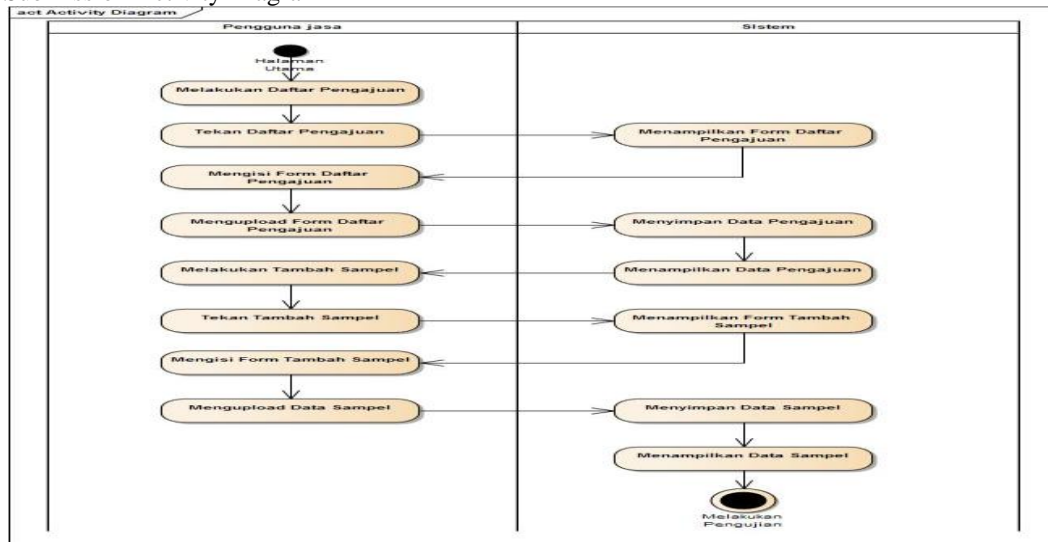


Fig 2. Submission Activity Diagram

In the Submission Activity Diagram image above describes the process of making a test submission

b. Testing ctivity Diagram

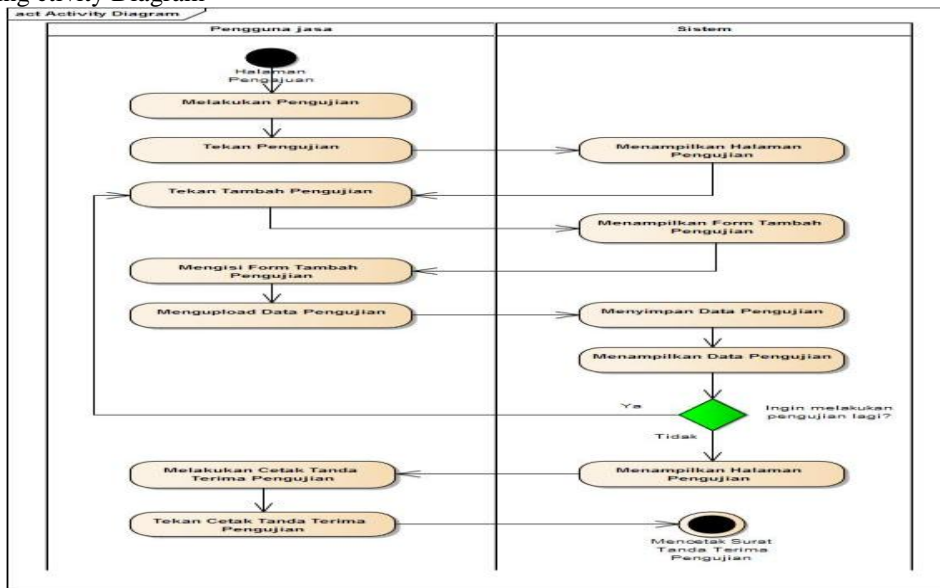


Fig 3. Testing Activity Diagram

In the Figure 3 the test activity diagram above illustrates the process of conducting testing

c. Payment Activity Diagram

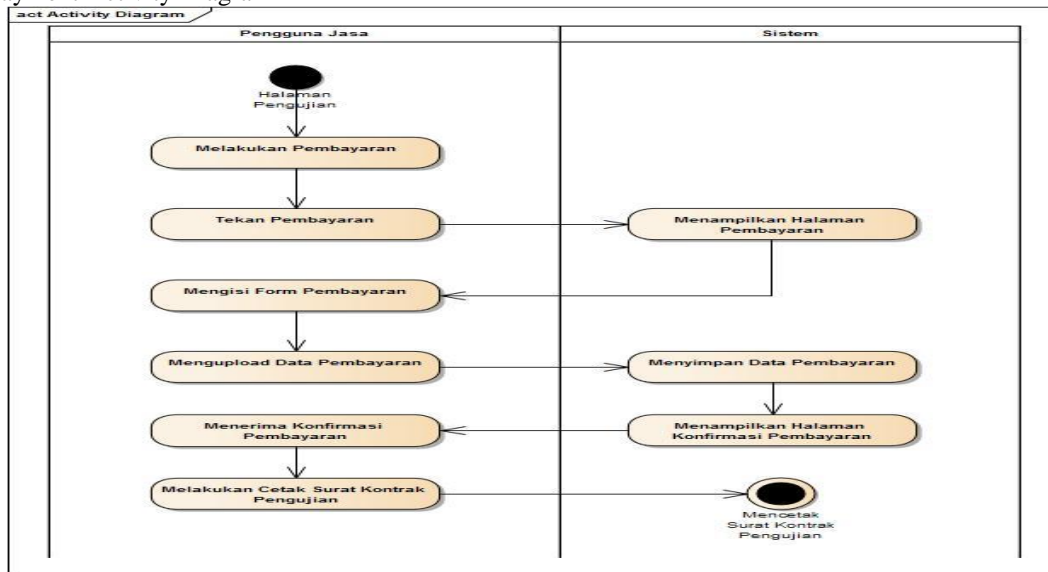


Fig 4. Payment Activity Diagram

In the payment Activity Diagram image illustrates the process of making payments for the tests to be carried out

4.3 Display



Fig 5. Front End Login Page

PENDAFTARAN
Silahkan Lengkapi Formulir

Email

Nama Lengkap

Alamat

No Handphone

Password

Fig 6. Service User Account List Page

BUAT PENGAJUAN PENGUJIAN

Nama Perusahaan

Alamat Perusahaan

Kode Pos

No Telpn

No Fax

Foto KTP / Diri

Lampiran

Fig 7. Submission List page

INFORMASI

PERIHAL : PERMINTAAN PENGUJIAN RANGAN

KODE PEMBAYARAN : \$127617281

Nama :
Muhammad Rizky Suhendar

Nama Perusahaan :
Ayam Geprek Haha

Alamat Perusahaan :
Jl. Bareng Jadian Engga

Kode Pos :
112233

No Telpn :
0892313445

No Fax :
-

No	Kode Pengujian	Kode Sampel	Nama Sampel	Jenis Sampel	Nama Pengujian	Harga	Keterangan
1	19.00.001	Aj01	Ayam	Daging Ayam	Uji Fisik (bau, rasa, warna)	Rp. 42.500,00	

Total Pengujian Rp 42.500,00

Nama Bank

Nomor Rekening

Nama Pemilik Rekening

Fig 8. Payment page

5. Conclusions

This web-based testing service system that has been created can be further developed such as adding a notification feature to the admin every time a test is entered, so that the admin does not have to monitor the test submission page continuously.

6. References

- [1]. Shalahuddin & Rosa A.S. (2015). *Rekayasa Perangkat Lunak*. Bandung: Informatika Bandung
- [2]. Nurhadi, A. (2018). Penerapan Metode Waterfall Dalam Sistem Informasi Penyedia Asisten Rumah Tangga Secara Online. *Jurnal Khatulistiwa Informatika*, 6(2), 97–106. <https://doi.org/10.31294/khatulistiwa.v6i2.150>
- [3]. Kuryanti, S. J. (2016). Rancang Bangun Sistem E-Learning sebagai Sarana Pembelajaran. *Jurnal Khatulistiwa Informatika*, 4(1), 84–92. <https://doi.org/10.1089/pho.2010.2784>
- [4]. Anna, A., Nurmalasari, N., & Yusnita, A. E. (2018). Rancang Bangun Sistem Informasi Akuntansi Penerimaan dan Pengeluaran Kas pada Kantor Camat Pontianak Timur. *Jurnal Khatulistiwa Informatika*, 6(2), 107–118. <https://doi.org/10.31294/khatulistiwa.v6i2.153>

