

## Rancang Bangun Aplikasi Berbasis Website Inventori *Jig* (Studi Kasus: PT Schneider Electric Manufacturing Batam)

Daniel Silitonga<sup>1</sup>, Maria<sup>2</sup>

Multimedia Engineering Technology Program, Batam State Polytechnic  
Animation Program, Batam State Polytechnic

---

### Article Info

#### Article history:

Received Jun 12<sup>th</sup>, 201x

Revised Aug 20<sup>th</sup>, 201x

Accepted Aug 26<sup>th</sup>, 201x

---

#### Keyword:

Inventori

*Jig*

Maintenance

Website

Waterfall

---

### ABSTRACT

*Jig* adalah salah satu alat penting yang dirancang untuk mendukung proses produksi di PT Schneider Electric Manufacturing Batam. Penelitian ini bertujuan untuk mengembangkan sebuah aplikasi berbasis website untuk inventori *Jig*. Aplikasi ini dirancang untuk mencatat persentase *Jig* yang rusak dan diperbaiki oleh tim teknisi Test & Maintenance PT Schneider Electric Manufacturing Batam serta untuk melakukan inventori *Jig*. Pengembangan metode Waterfall mengumpulkan data melalui observasi dan wawancara dengan klien. Alat ukur System Usability Scale (SUS) dan metode pengujian Blackbox digunakan untuk melakukan pengujian. Hasil evaluasi SUS menunjukkan skor 72, menandakan bahwa sistem telah diterima dengan baik oleh pengguna dan mendapat penilaian "Baik (B)". Ini mengindikasikan bahwa Aplikasi Website Inventori *Jig* PT Schneider Electric Manufacturing Batam telah mencapai tingkat kualitas yang memuaskan dan memiliki potensi untuk meningkatkan kinerjanya di masa mendatang.

Copyright © 201x Institute of Advanced Engineering and Science.  
All rights reserved.

---

### Corresponding Author:

Maria,

Animation Program,

Politeknik Negeri Batam,

Jl. Ahmad Yani, Tlk. Kering, Kec. Batam Kota, Kota Batam, Kepulauan Riau, 29461, Indonesia.

Email: maria@polibatam.ac.id

---

## 1. PENDAHULUAN

*Jig* merupakan salah satu alat yang paling penting dalam proses produksi produk-produk dari PT Schneider Electric Manufacturing Batam. *Jig* adalah alat khusus yang dirancang untuk mencari lokasi dan memegang dengan kuat benda kerja pada posisi yang tepat selama operasi manufaktur atau perakitan [1]. PT Schneider Electric Manufacturing Batam adalah cabang di Pulau Batam dari perusahaan Perancis Schneider Electric, yang memimpin transformasi digital dalam pengelolaan energi dan otomasi di seluruh dunia. [2].

Sebagai perusahaan yang memiliki perubahan besar di bidang digital, PT Schneider Electric Manufacturing Batam membutuhkan suatu aplikasi berbasis website inventori *jig* untuk mengimplementasikan persentase dari *Jig* yang telah rusak dan telah diperbaiki oleh tim teknisi Test & Maintenance. Departemen Test & Maintenance merupakan departemen dari PT Schneider Electric Manufacturing Batam yang memiliki tugas untuk memperbaiki dan memelihara mesin produksi maupun komponen yang terkait dalam proses produksi barang termasuk *Jig* [3]. *Jig* yang telah rusak akan diperbaiki oleh teknisi Test & Maintenance dan *Jig* yang telah diperbaiki juga didata dan disimpan ke dalam basis data departemen tersebut untuk melihat persentase dari *Jig* yang rusak dan sudah diperbaiki. Untuk mengatasi masalah tersebut, maka dikembangkan sebuah aplikasi berbasis website intranet.

Website berbasis intranet merupakan bentuk media masa yang publikasinya melalui jaringan yang hanya dapat digunakan di dalam internal Perusahaan [4]. Rancang bangun aplikasi berbasis website inventori *jig* merupakan website intranet yang dibuat oleh pengembang untuk menampilkan persentase *jig*, mengambil input data, mengedit data yang telah terinput terhadap *Jig* yang rusak maupun yang telah diperbaiki.

Pengembangan website ini dibuat menggunakan metode *Waterfall*. Metode *Waterfall* memaparkan berbagai pendekatan alur hidup perangkat lunak [5]. Metode ini digunakan untuk membangun situs web ini. Metode ini dipilih karena memiliki tahapan yang terstruktur dan konsisten, mulai dari analisis, desain, eksekusi program, pengujian, dan tahapan pemeliharaan perangkat lunak apabila ada masalah. Selain itu, aplikasi yang dirancang harus diuji menggunakan metode *System Usability Scale (SUS)* untuk menentukan kenyamanan, kemudahan, dan kepuasan pengguna saat menggunakan aplikasi tersebut. [6].

## 2. LANDASAN TEORI

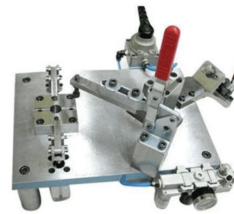
### 2.1. Jig

*Jig* adalah alat khusus yang dirancang untuk mencari lokasi dan memegang dengan kuat benda kerja pada posisi yang tepat selama operasi manufaktur atau perakitan [3]. Dalam proses produksinya, duplikasi part yang sangat akurat dan tepat dibutuhkan untuk memperoleh kualitas produk yang bagus dan tidak memiliki kecacatan maupun missing terhadap produk lainnya. Maka, *jig* merupakan komponen kuncinya [7].

Di Pt Schneider Electric Manufacturing Batam, digunakan beberapa jenis *jig*, yaitu *Jig Cable* dan *Jig Fixture*. Meskipun fungsi masing-masing berbeda, keduanya memiliki tujuan yang sama, yaitu untuk mendukung proses operasi manufaktur produk yang diproduksi oleh PT Schneider Electric Manufacturing Batam. Visualisasi dari *Jig Cable* dan *Jig Fixture* dapat dilihat pada gambar 2.1 dan gambar 2.2..



Gambar 2.1 *Jig Cable*



Gambar 2.2 *Jig Fixture*

### 2.2. Website

Sebuah website adalah platform daring yang menghubungkan berbagai dokumen, baik dalam jangkauan lokal maupun internasional. Dokumen-dokumen ini, yang disebut halaman web, dapat diakses menggunakan *browser tool* seperti Google Chrome, Mozilla Firefox, Microsoft Edge, dan lainnya. Fitur untuk beralih dari satu halaman ke halaman lain disebut *hypertext*, yang memungkinkan pengguna menelusuri berbagai halaman yang tersimpan di server yang sama di seluruh dunia. [8]

### 2.3. Intranet

Intranet adalah jaringan bersifat *private* yang memanfaatkan protokol internet TCP/IP untuk menjaga agar jaringan hanya dapat diakses dalam area terbatas. Intranet biasanya digunakan dalam lingkungan perusahaan untuk mencegah koneksi dengan pihak luar, sehingga data tetap aman dan hanya dapat digunakan oleh pihak internal. [9]

### 2.4. Blackbox Testing

Blackbox Testing merupakan pendekatan pengujian yang fokus pada fungsi-fungsi perangkat lunak. Dalam metode ini, perilaku perangkat lunak diuji berdasarkan input dari pengguna untuk memverifikasi bahwa hasil keluaran sesuai dengan yang diinginkan, tanpa memeriksa secara langsung proses internal atau kode program yang digunakan.. Oleh karena itu, metode ini dapat disimpulkan sebagai suatu teknik pengujian yang tidak memerlukan pengetahuan tentang kinerja internal perangkat lunak, sehingga pengujian diperlukan untuk memastikan fungsionalitasnya. [10]

### 2.5. System Usability Scale (SUS)

*System Usability Scale (SUS)* merupakan metode evaluasi yang digunakan untuk menilai kegunaan suatu aplikasi dengan cara yang efisien dan sederhana. Penggunaannya melibatkan pengisian kuesioner yang terdiri dari 10 pertanyaan atau pernyataan, dengan masing-masing memiliki 5 opsi jawaban. *SUS* sering digunakan untuk mengevaluasi berbagai jenis produk dan layanan, termasuk perangkat keras, perangkat lunak, aplikasi seluler, dan situs web.. [11]

### 2.6. Microsoft SQL Server

Microsoft SQL Server adalah sistem yang mengatur basis data client/server (DBMS) milik Microsoft yang dirancang untuk menyimpan data di server dan memungkinkan pengguna untuk terhubung ke server yang sama menggunakan alamat server tertentu. SQL Server dapat menampilkan dan memanipulasi data serta menyimpan, mengambil, dan mengamankan basis data. [12]

### 2.7. Visual Basic .NET

Visual Basic .NET adalah bahasa program berorientasi objek yang menggunakan .NET framework dan dijalankan melalui perangkat lunak Microsoft Visual Studio. VB.NET biasanya digunakan untuk

mengembangkan aplikasi Windows Forms dengan fitur-fitur seperti membuat, membaca, menyunting, dan menghapus data. [13]

### 2.8. ChartJS

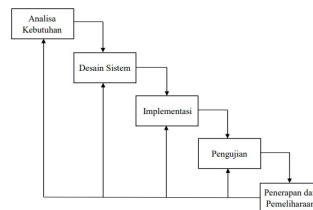
ChartJS adalah perangkat lunak untuk visualisasi data yang terkenal dan merupakan salah satu metode yang paling mudah untuk menggambarkan data secara visual. Dengan menggunakan bahasa pemrograman JavaScript, ChartJS dapat membuat berbagai jenis grafik seperti grafik batang, grafik garis, grafik radar, grafik pie, grafik campuran, dan lainnya.. [14]

### 2.9. Basis Data

Basis Data adalah koleksi informasi yang tersimpan secara terstruktur dalam komputer, yang memungkinkan program komputer untuk mengakses dan memanipulasi informasi yang tersedia.. [15]

## 3. METODE PENELITIAN

Dalam penelitian ini, digunakan metode Waterfall yang memiliki serangkaian tahapan seperti yang ditunjukkan dalam diagram berikut..



Gambar 3.1 Metode *waterfall*

### 3.1. Analisis Kebutuhan (*Requirement*)

Tahapan ini merupakan tahap mengumpulkan data melalui observasi dan wawancara kepada departemen maintenance yang nantinya akan dianalisa kemudian ditarik kesimpulan untuk menentukan perancangan alur kerja yang akan dibuat seperti data *jig* rusak yang telah dipindai ke dalam basis data, data *jig* yang telah diperbaiki, dan statistik *jig* yang paling banyak diperbaiki yang nantinya dikelompokkan berdasarkan tahunan, bulanan, harian dan perarea produksi.

### 3.2. Desain Sistem (*Design System*)

Tahap ini merupakan tahapan melakukan rancangan alur kerja aplikasi berdasarkan pengembangan dan bahasa pemrograman. Penerapan rancangan *input* dan *output* yang akan dihasilkan di aplikasi. Pengembang melakukan Rancang Bangun Aplikasi Berbasis Website Inventori *Jig* dengan bahasa pemrograman berorientasi objek menggunakan *Visual Basic .NET*.

### 3.3. Implementasi (*Implementation*)

Tahapan selanjutnya yaitu implementasi, pada tahapan ini bahasa pemrograman berorientasi objek yaitu *VB.NET* yang digunakan akan dikolaborasikan dengan *Microsoft SQL Server* sebagai basis data yang digunakan dengan tujuan membuat *website* dengan jaringan *intranet*.

### 3.4. Pengujian (*Integration and Testing*)

Pada fase ini, dilakukan pengujian terhadap implementasi program yang telah dieksekusi. Tujuannya adalah untuk memverifikasi bahwa aplikasi yang dibuat beroperasi dengan efektif sesuai dengan perancangan yang telah disusun. Pengujian awal dilakukan dengan menggunakan pendekatan *black box*, sementara pengujian lanjutan menggunakan analisis *System Usability Scale (SUS)*.

Tabel 3.1 Rancangan Kuesioner SUS

Nama:	
Usia:	
Jabatan:	
Pernyataan	
1.	Saya berpikir akan menggunakan aplikasi berbasis <i>website</i> inventori <i>jig</i> PT XYZ ini lagi.
2.	Saya merasa aplikasi berbasis <i>website</i> inventori <i>jig</i> PT XYZ ini sulit digunakan.
3.	Saya merasa aplikasi berbasis <i>website</i> inventori <i>jig</i> PT XYZ ini mudah digunakan.
4.	Saya membutuhkan bantuan orang saat menggunakan aplikasi berbasis <i>website</i> inventori <i>jig</i> PT XYZ.
5.	Saya merasa fitur-fitur (dari sisi navigasi, tombol, dan tampilan <i>website</i> ) yang ada di aplikasi berbasis <i>website</i> inventori <i>jig</i> PT XYZ ini berfungsi.
6.	Saya merasa banyak hal yang tidak konsisten dan perlu diperbaiki dalam aplikasi berbasis <i>website</i> inventori <i>jig</i> PT XYZ.
7.	Saya merasa orang lain akan memahami cara menggunakan aplikasi berbasis <i>website</i> inventori <i>jig</i> PT XYZ.
8.	Saya merasa aplikasi berbasis <i>website</i> inventori <i>jig</i> PT XYZ ini membingungkan (dari sisi navigasi, tombol, dan tampilan <i>website</i> ).
9.	Saya merasa tidak ada hambatan saat menggunakan aplikasi berbasis <i>website</i> inventori <i>jig</i> PT XYZ.
10.	Saya harus membiasakan diri terlebih dahulu dalam menggunakan aplikasi berbasis <i>website</i> inventori <i>jig</i> PT XYZ.

### 3.5. Pemeliharaan (*Maintenance*)

Tahapan ini merupakan tahap terakhir dalam penelitian menggunakan metode *waterfall*. Pemeliharaan aplikasi dilakukan agar aplikasi dapat terus menerus digunakan. Dari waktu ke waktu kebutuhan perangkat lunak pasti mengalami perubahan, maka dari itu perlu dilakukannya pemeliharaan terhadap aplikasi yang sudah dibuat agar terus mendapatkan pembaruan ataupun perbaikan pada fitur-fitur yang sebelumnya telah dibuat.

## 4. HASIL DAN PEMBAHASAN

### 4.1. Analisis Kebutuhan Sistem

Tahapan ini merupakan tahap mengumpulkan data melalui observasi dan wawancara kepada departemen *maintenance* yang nantinya akan dianalisa kemudian ditarik kesimpulan untuk menentukan perancangan alur kerja yang akan dibuat. Berikut adalah input dan output yang akan dihasilkan berdasarkan hasil observasi dan wawancara kepada departemen Test & Maintenance terhadap website yang akan dibuat:

#### 4.1.1. Input

- Data *jig* rusak yang telah dipindai ke dalam basis data
- Data *jig* yang telah diperbaiki
- Mapping data *jig* berdasarkan nama *jig*, nama area produksi yang menggunakannya, size atau ukuran *jig*, dan serial number *jig*.

#### 4.1.2. Output

- Persentase *jig* yang paling banyak diperbaiki yang nantinya dikelompokkan berdasarkan tahunan, bulanan, harian dan perarea produksi sesuai dengan identitas *jig* yang ditentukan oleh user.

#### 4.1.3. Kebutuhan Fungsional

Kebutuhan fungsional pada aplikasi dapat dilihat pada tabel 4.1

Tabel 4.1 Kebutuhan Fungsional

Fungsional	Keterangan
F001	Aplikasi dilengkapi dengan fitur login dan <i>authentication role</i> yaitu <i>role engineer</i> , <i>maintenance</i> , dan <i>operator</i>
F002	Aplikasi dapat membuat, menghapus, dan mengedit akun pengguna lainnya yang menggunakan aplikasi oleh pengguna dengan <i>role engineer</i> .
F003	Aplikasi dapat menampilkan persentase <i>jig reject</i> dan <i>jig repaired</i> dengan dikelompokkan berdasarkan tahunan, bulanan, dan harian.
F004	Aplikasi dapat menampilkan data <i>jig reject</i> dan <i>jig repaired</i>
F005	Aplikasi dapat melakukan <i>scan out</i> untuk melakukan <i>repair</i> pada <i>jig</i> oleh pengguna dengan <i>role maintenance</i> dan <i>engineer</i> .
F006	Aplikasi dapat melakukan <i>scan in</i> untuk melakukan <i>reject jig</i> oleh pengguna dengan <i>role operator</i> .
F007	Aplikasi dapat menghapus data pada <i>reject jig</i> oleh pengguna dengan <i>role engineer</i> .
F008	Aplikasi dapat melakukan <i>record</i> untuk melakukan <i>reject jig</i> dan <i>record</i> untuk melakukan <i>repaired jig</i> .

#### 4.1.4. Kebutuhan Non Fungsional

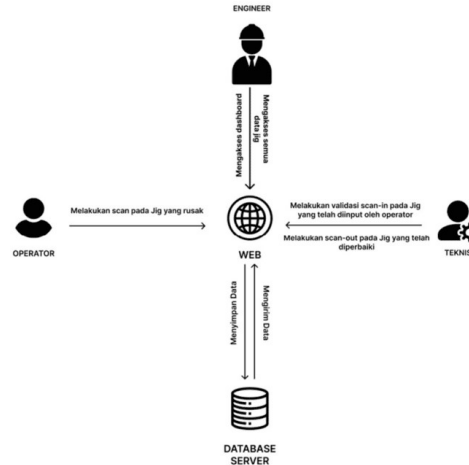
Kebutuhan Non Fungsional pada aplikasi dapat dilihat pada tabel 4.2

Tabel 4.2 Kebutuhan Non Fungsional

Non Fungsional	Keterangan
N001	Aplikasi harus memiliki tampilan yang mudah dipahami oleh pengguna

**4.1.5. Desain Sistem**

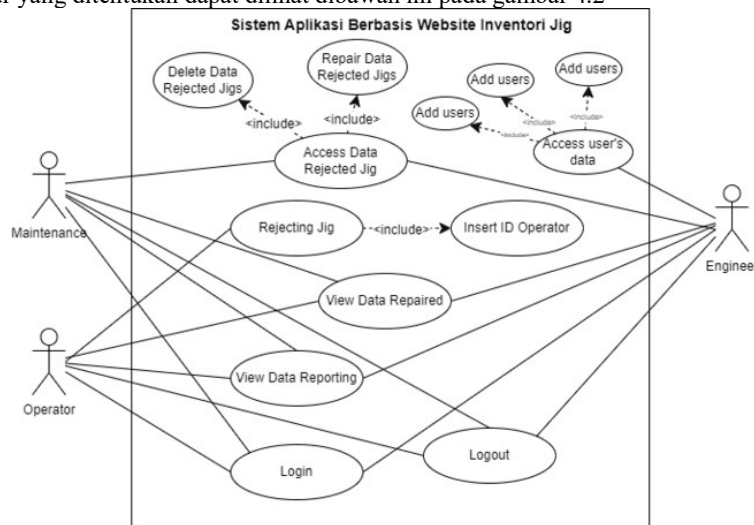
Sistem website ini mendukung tiga *role* pengguna: *engineer*, teknisi (*maintenance*), dan *operator*. *Engineer* memantau persentase *jig* melalui dashboard. Teknisi memperbaiki *jig* rusak dan mencatat detail perbaikan. *Operator* melakukan *scan in jig reject* untuk mencatat *jig* yang rusak. Dengan pembagian ini, setiap pengguna memiliki akses dan fungsi spesifik, memastikan pemantauan, perbaikan, dan pencatatan kondisi *jig* berjalan efektif.



Gambar 4.1 Gambaran Umum Sistem Website Inventori Jig

**4.1.6. Use Case Diagram**

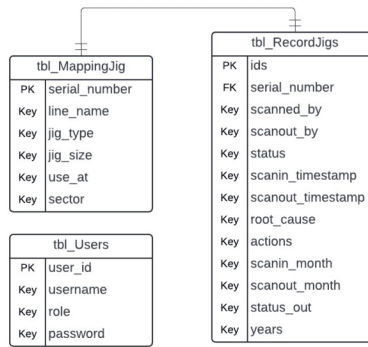
Dibawah ini merupakan use case diagram yang memiliki 3 *role user* yaitu *engineer*, *maintenance*, dan *operator*, semua user dapat melakukan login. Setelah login 3 *role user* tersebut terautentikasi untuk mengakses halaman dan fitur yang ditentukan dapat dilihat dibawah ini pada gambar 4.2



Gambar 4.2 Use Case Diagram

**4.1.7. Entity Relationship Diagram (Conceptual Model)**

Entity-Relationship Diagram (ERD) digunakan untuk melakukan pemodelan struktur basis data yang digunakan dalam sistem. ERD menggambarkan entitas-entitas utama dan hubungan antar entitas tersebut. Berikut adalah ERD sistem inventori *jig* pada gambar 4.3

Gambar 4.3 ERD *Conceptual Model*

#### 4.1.8. Wireframe

##### a. Wireframe *Login*

Halaman *login* adalah halaman untuk memulai masuk ke dalam *web* apabila *username* dan *password* salah maka tidak dapat meneruskan ke halaman selanjutnya. Wireframe *login* ditampilkan pada gambar 4.4.

Gambar 4.4 Wireframe *Login*

##### b. Wireframe *Select*

Halaman *select* merupakan tampilan setelah *login*.

Gambar 4.5 Wireframe *Select*

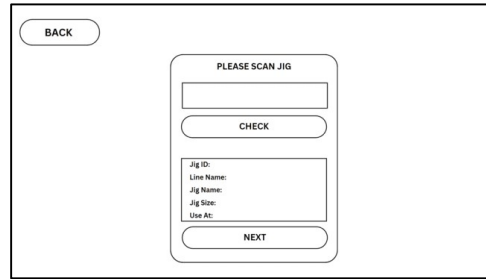
##### c. Wireframe *Repairing Jig*

Halaman *Repairing Jig* merupakan halaman untuk melakukan *repair* pada *jig*

Gambar 4.6 Wireframe *Repairing Jig*

##### d. Wireframe *Rejecting Jig*

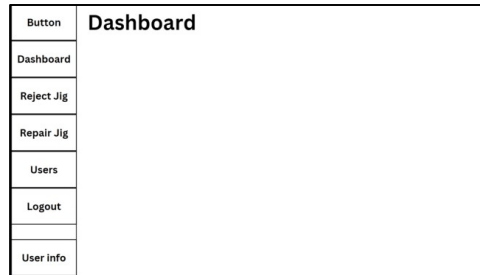
Halaman *Rejecting Jig* merupakan halaman untuk melakukan *rejecting* pada *jig*



Gambar 4.7 Wireframe *Rejecting Jig*

e. Wireframe *Dashboard*

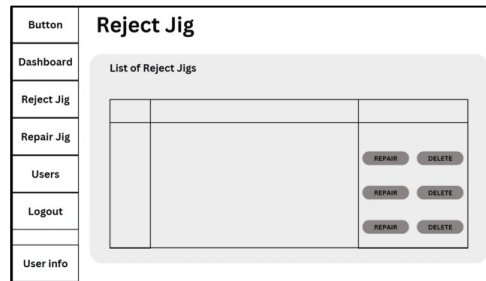
Halaman *Dashboard* merupakan halaman untuk menampilkan persentase *jig reject* dan *jig repaired*



Gambar 4.8 Wireframe *Dashboard*

f. Wireframe *Reject Jig*

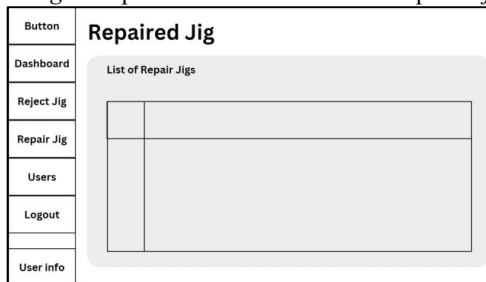
Halaman *Reject Jig* merupakan halaman untuk melihat data *jig reject* dan melakukan *repair* dan *delete* pada *jig*



Gambar 4.9 Wireframe *Reject Jig*

g. Wireframe *Repair Jig*

Halaman *Repair Jig* merupakan halaman untuk menampilkan *jig repaired*



Gambar 4.10 Wireframe *Repair Jig*

h. Wireframe *Users*

Halaman *Users* merupakan halaman untuk menampilkan data *user*, menambah *user*, dan menghapus *user*

Button	<b>Users</b>			
Dashboard	Add New User Account			
Reject Jlg	Username	Nickname	Password	Roles
Repair Jlg	<input type="text"/>	<input type="text"/>	<input type="password"/>	<input type="text"/>
Users	SUBMIT			
Logout	List Users			
User Info	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Gambar 4.11 Wireframe *Users*

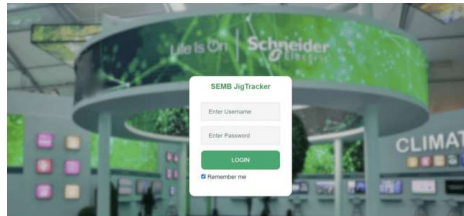
## 4.2. Implementasi

Pada tahap ini dilakukan implementasi berdasarkan dengan menggunakan materi yang di kumpulkan.

### 4.2.1. Implementasi Antarmuka

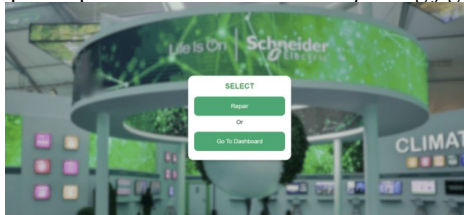
#### a. Halaman *Login*

Pada aplikasi ini terdiri dari 3 *role* pengguna yaitu *engineer*, *maintenance*, dan *operator*. Pengguna harus melewati tahap *login* terlebih dahulu untuk mendapatkan verifikasi pengguna aplikasi dan masuk ke dalam halaman *select*.

Gambar 4.12 Halaman *Login*

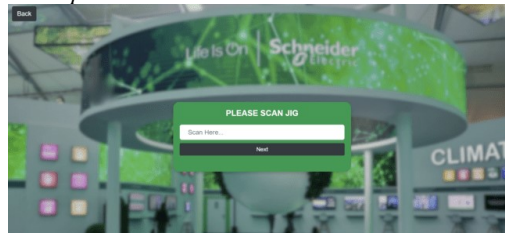
#### b. Halaman *Select*

Pada halaman ini menampilkan pilihan untuk melakukan *repairing jig*

Gambar 4.13 Halaman *Select*

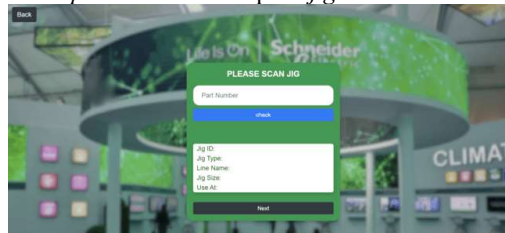
#### c. Halaman *Repairing Jig*

Pada halaman ini meminta *input serial number* untuk melakukan *scan repairing jig*

Gambar 4.14 Halaman *Repairing Jig*

#### d. Halaman *Rejecting Jig*

Pada halaman ini meminta *input serial number* pada *jig* untuk melakukan *reject jig*



Gambar 4.15 Halaman *Rejecting Jig*

e. Halaman *Dashboard*

Pada halaman *dashboard* menampilkan grafik berupa *bar chart* untuk melihat persentase *jig reject* dan *jig repaired* yang dikelompokkan berdasarkan tahunan, bulanan, dan harian



Gambar 4.16 Halaman *Dashboard*

f. Halaman *Reject Jig*

Pada halaman *reject jig* menampilkan data *jig reject* dan dapat melakukan *repair* dan *delete* pada *jig reject* yang diinginkan

Gambar 4.17 Halaman *Reject Jig*

g. Halaman *Repair Jig*

Pada halaman *repair jig* menampilkan semua data *jig repaired*

Gambar 4.18 Halaman *Repair Jig*

h. Halaman *Users*

Pada halaman *users* menampilkan data *user* dan dapat menambahkan, *edit*, dan *delete* data *user*

Gambar 4.19 Halaman *Users*


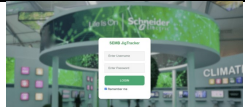

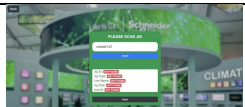







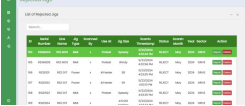
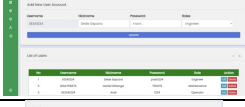
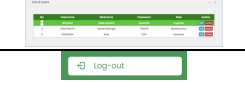
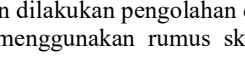
4.3. Pengujian

Pengujian dilakukan dalam dua tahap, yaitu pengujian alfa dan beta. Pengujian alfa berfokus pada fungsionalitas aplikasi menggunakan perangkat laptop melalui pendekatan *black box*. Pengujian beta dilakukan setelah semua fungsionalitas berjalan dengan baik dan sesuai dengan rancangan. Tahap ini melibatkan 20 responden [16] yaitu 1 *engineer*, 2 teknisi, 1 operator, 10 peserta magang, dan 6 staff dan di lakukan dengan cara menyebar kuesioner setelah uji aplikasi langsung.

Tabel 4.3 Pengujian Alfa Black Box

No	Aktivitas	Skenario	Hasil	Dokumentasi	Validasi
1	<i>Login</i>	<i>Username</i> benar <i>Password</i> benar	Berhasil <i>login</i> dan menampilkan halaman <i>select</i>		✓

*Title of manuscript is short and clear, implies research results (First Author)*

		<i>Username</i> benar <i>Password</i> salah	Tidak berhasil <i>login</i> , sistem melakukan <i>reload page</i>		
		<i>Username</i> salah <i>Password</i> benar	Tidak berhasil <i>login</i> , sistem melakukan <i>reload page</i>		
2	<i>Scan In Reject Jig</i> oleh <i>Operator</i>	Pengguna melakukan <i>input serial number</i> yang terdaftar dalam tabel <i>mapping jig</i>	Berhasil menyimpan data		✓
		Pengguna melakukan <i>input serial number</i> yang tidak terdaftar dalam tabel <i>mapping jig</i>	Tidak berhasil menyimpan data dan menampilkan notifikasi <i>scan gagal</i>		
		Pengguna melakukan <i>input serial number</i> yang sudah diinput ke dalam tabel <i>record jigs</i> dengan status <i>reject</i>	Tidak berhasil menyimpan data dan menampilkan notifikasi <i>jig</i> telah dilakukan <i>reject</i>		
3	<i>Scan ID</i> oleh seluruh user	Pengguna melakukan <i>input Id</i> karyawan tidak kosong	Berhasil menyimpan data		✓
		Pengguna melakukan <i>input Id</i> karyawan kosong	Tidak berhasil menyimpan data		
4	<i>Input Root Cause</i> dan <i>Action</i>	Pengguna melakukan <i>input root cause</i> dan <i>action</i> tidak kosong	Berhasil menyimpan data		✓
		Pengguna melakukan <i>input root cause</i> dan <i>action</i> kosong	Tidak berhasil menyimpan data		
5	Akses <i>Dashboard</i>	Pengguna mengakses halaman <i>dashboard</i>	Berhasil menampilkan <i>dashboard</i>		✓
6	<i>Action Buttons</i> di halaman <i>Reject Jig</i>	Pengguna melakukan <i>repair jig</i>	Berhasil menyimpan data		✓
		Pengguna melakukan <i>delete</i>	Berhasil menghapus data		
7	<i>Action Buttons</i> di halaman <i>Users</i>	Pengguna melakukan <i>edit user</i>	Berhasil memperbarui data		✓
		Pengguna melakukan <i>delete user</i>	Berhasil menghapus data		
8	<i>Buttons Logout</i> di navigasi	Pengguna mengklik <i>button logout</i>	Berhasil kembali ke <i>login</i>		✓

Pengujian beta dilakukan dengan mengumpulkan hasil kuesioner dan dilakukan pengolahan data dengan menghitung skor SUS (*System Usability Scale*) dari 10 pernyataan menggunakan rumus skor rata-rata menghitung skor SUS dan bobot nilai. [17]

$X = (\sum x)/n$   
 Keterangan:  
 X: Skor rata-rata  
 $\sum x$ : Jumlah skor SUS  
 n: Jumlah responden

Tabel 4.4 Pengujian Beta

Pengguna	Skor Akhir	Skor Akhir x 2,5
1	23	58
2	33	83
3	26	65
4	37	93
5	33	83
6	30	75
7	37	93
8	35	88
9	26	65
10	34	85
11	28	70
12	23	58
13	23	58
14	36	90
15	24	60
16	23	58
17	25	63
18	25	63
19	35	88
20	20	50

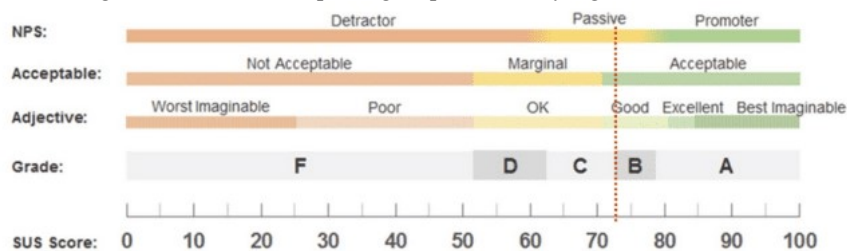
Berdasarkan data dari tabel 4.4 yang telah diolah, langkah kedepannya merupakan perhitungan menggunakan formula 2. Yaitu dengan mengalikan total seluruh skor dengan 2,5 yang menghasilkan jumlah **1440**, yang kemudian dibagi dengan jumlah responden sebanyak 20 orang.

Hasil akhirnya adalah **72**. Berdasarkan skor akhir dari perhitungan ini, aplikasi Berbasis Website Inventori Jig (Studi Kasus: PT Schneider Electric Manufacturing Batam) dapat diklasifikasikan sebagai "**Baik (B)**" berdasarkan tabel 4.5, dengan skor antara **68** hingga **80.3**.

Tabel 4.5 Penilaian System Usability Scale Website Inventori Jig

Skor SUS	Penilaian Adjektif	Penilaian Skala
>80.3	Luar Biasa	A
68 – 80.3	Baik	B
68	Cukup	C
51 – 68	Kurang	D
< 51	Buruk	F

Penelitian ini menunjukkan bahwa aplikasi berbasis website inventori jig, studi kasus PT Schneider Electric Manufacturing Batam, telah mencapai tingkat penerimaan yang memadai.



Gambar 4.20 Grafik Kesimpulan Skor SUS Aplikasi Berbasis Website Inventori Jig SEMB [18]

**4.4. Pemeliharaan**

Untuk tahap pemeliharaan yang diterapkan pada aplikasi berbasis website inventori jig meliputi pembaruan lingkungan operasional, seperti melakukan pembaruan pada kerangka basis data dengan melakukan integrasi ke basis data internal perusahaan dengan menyesuaikan rancangan yang telah dibuat dan memastikan website yang dibuat dapat dilakukan *publish*.

**5. KESIMPULAN**

Berdasarkan penelitian dan pengembangan yang telah dilakukan oleh penulis pada semua tahapan sebelumnya, dapat diambil kesimpulan yaitu dengan adanya Aplikasi Berbasis Website Inventori Jig Studi

*Title of manuscript is short and clear, implies research results (First Author)*

Kasus PT Schneider Electric Manufacturing Batam dalam mengelola pencatatan *jig* yang *reject* dan melakukan perbaikan pada *jig* yang juga dilakukan pencatatan dapat dijadikan bahan evaluasi dan perbaikan untuk kemajuan perusahaan itu sendiri. Pengujian fungsionalitas dengan metode *blackbox testing* pada website sudah tervalidasi bahwa semua fungsi telah dilakukan pengujian yang disertai dokumentasi. Berdasarkan evaluasi menggunakan *System Usability Scale* (SUS), aplikasi Website Inventori Jig PT Schneider Electric Manufacturing Batam mendapat skor 72. Skor ini menunjukkan bahwa aplikasi tersebut telah diterima dengan baik oleh pengguna, dengan penilaian "Baik (B)". Hal ini menunjukkan bahwa kinerja aplikasi telah memuaskan dan menunjukkan potensi untuk pengembangan lebih lanjut di masa mendatang.

## DAFTAR PUSTAKA

- [1] A. Saptari, W. Soon Lai, and M. Rizal Salleh, "Work Station Design and their Effect to Productivity," *Jordan Journal of Mechanical and Industrial Engineering*, vol. 5, no. 1, 2011.
- [2] S. Wardana and N. Fajrah, "Pengendalian Kualitas Produk Cacat PHX Toshiba Pada PT Schneider Electric Manufacturing Batam," *Jurnal Teknik Industri*, 2019.
- [3] J. Patalas-Maliszewska and S. Klos, "An approach to supporting the selection of maintenance experts in the context of industry 4.0," *Applied Sciences (Switzerland)*, vol. 9, no. 9, May 2019, doi: 10.3390/app9091848.
- [4] E. W. Fridayanthie And T. Mahdiati, "Rancang Bangun Sistem Informasi Permintaan Atk Berbasis Intranet (Studi Kasus: Kejaksaan Negeri Rangkasbitung)," *Jurnal Khatulistiwa Informatika*, Vol. 4, No. 2, 2016.
- [5] M. Badrul, "Penerapan Metode Waterfall Untuk Perancangan Sistem Informasi Inventory Pada Toko Keramik Bintang Terang," *Jurnal PROSISKO*, vol. 8, no. 2, 2021.
- [6] Prasetya, R. D., Khairy, F. M., Hibban, N., Rifa'i, D. B., & Pasya, R. I. (2023). Pengujian Usability Pada Website Kitabisa. Com Menggunakan Metode System Usability Scale (SUS). *METHODIKA: Jurnal Teknik Informatika Dan Sistem Informasi*, 9(2), 26-29.
- [7] Prasetiyo, H., Rispianda, R., & Adanda, H. (2016). Rancangan *Jig* Dan Fixture Pembuatan Produk Cover On-Off. *Teknoin*, 22(5).
- [8] I. N. Laily. (2022). Pengertian Website Menurut Para Ahli, Beserta Jenis dan Fungsinya, katadata. co. id.
- [9] Witi, F. L., & Mude, A. (2020). Analisis Jaringan Intranet Di Universitas Flores Menggunakan Quality Of Service (QoS). *Computer Based Information System Journal*, 8(1), 7-12
- [10] Febiharsa, D., Sudana, I. M., & Hudallah, N. (2018). Uji fungsionalitas (blackbox testing) sistem informasi lembaga sertifikasi profesi (silsp) batik dengan appperfect web test dan uji pengguna. *Joined Journal (Journal of Informatics Education)*, 1(2), 117-126.
- [11] Brooke, J. (2013). SUS: a retrospective. *Journal of usability studies*, 8(2), 29-40.
- [12] Subagia, A. (2017). Membangun aplikasi dengan codeigniter dan database SQL server. *Elex Media Komputindo*.
- [13] Wali, M. (2017). Membangun Aplikasi Windows dengan Visual Basic. *NET 2015 Teori dan Praktikum: Indonesia (Vol. 1)*. KITA Publisher.
- [14] Da Rocha, H. (2019). *Learn Chart. js: Create interactive visualizations for the web with chart. js 2*. Packt Publishing Ltd.
- [15] A. Andaru. (2018). Pengertian database secara umum, OSF Prepr
- [16] Nielsen, Jakob. 2012. How many test users in a usability study? Diakses pada tanggal 20 Januari 2017 dari <https://www.nngroup.com/articles/howmany-test-users/>
- [17] Z. Sharfina and H. B. Santoso, "An Indonesian adaptation of the System Usability Scale (SUS)," 2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS), Malang, Indonesia, 2016, pp. 145-148, doi: 10.1109/ICACSIS.2016.7872776.
- [18] Kesuma, D. P. (2021). Penggunaan metode System Usability Scale untuk mengukur aspek Usability pada media pembelajaran daring di Universitas XYZ. *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, 8(3), 1615-1626.