

Perancangan Sistem Informasi *Inventory Sparepart* Pesawat Di PT. Avia Teknik Solusindo Berbasis *Web* Menggunakan Metode Extreme Programming

Agung Maulana¹, Syahriani Syam², Febri Rismaningsih³

^{1,2}Teknik Informatika, Universitas Islam Syekh Yusuf, Tangerang, Indonesia

³Teknik Sipil, Universitas Islam Syekh Yusuf, Tangerang, Indonesia

¹ 1904030121@students.unis.ac.id, ² ssyam@unis.ac.id, ³ frismaningsih@unis.ac.id,

ABSTRAK / ABSTRACT	Kata Kunci / Keywords
<p>Inventory memungkinkan efisiensi dari pendistribusian serta penyimpanan barang. Sparepart pesawat adalah suatu bagian atau suku cadang baik itu yang terkecil maupun terbesar serta cair maupun padat. Sparepart tersebut harus dikelola dengan baik supaya tidak terjadinya kehilangan pada sparepart atau barang. Sistem informasi inventory sparepart pesawat berbasis web adalah suatu aplikasi berbasis web diperuntukan mengelola Data sparepart atau barang pesawat. Aplikasi ini bisa di akses melalui smartphone agar mempermudah dan flexible dalam mengelola Data sparepart atau barang pesawat. Adapun metode extreme programming digunakan dalam perancangan sistem informasi inventory sparepart pesawat berbasis web sebagai acuan dalam perancangan sistem.</p>	<p>Kata Kunci : sistem inventory, extreme programming, sparepart pesawat</p>
<p><i>Inventory enables the efficient distribution and storage of goods. An aircraft sparepart is a part or sparepart, whether it is the smallest or the largest, liquid or solid. These spareparts must be managed properly so that there is no loss of spareparts or goods. Web-based aircraft spareparts inventory information system is a web-based application intended to manage aircraft spareparts or goods data. This application can be accessed via smartphone to make it easier and more flexible in managing spareparts data or aircraft goods. The extreme programming method is used in designing a web-based aircraft spareparts inventory information system as a reference in system design.</i></p>	<p>Keywords : nventory system, extreme programming, aircraft spareparts</p>

I. PENDAHULUAN

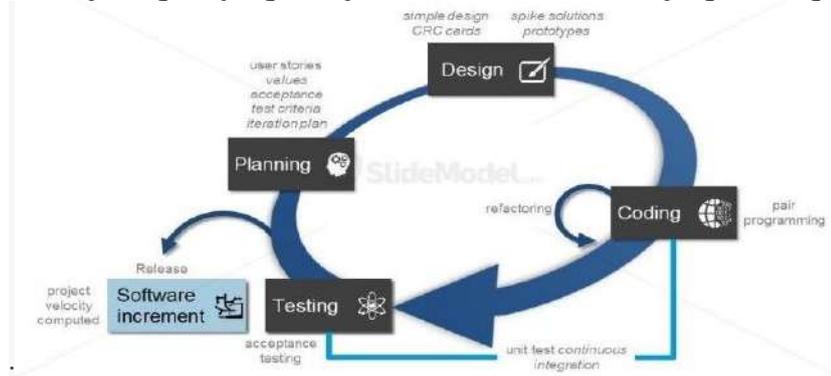
Banyak sekali faktor-faktor teknologi yang telah mempengaruhi manusia dan lingkungan sekitarnya seiring berjalannya waktu dan hal lainnya yang bisa membantu atau memperbaiki kinerja manusia. Teknologi sering dipakai oleh manusia agar mengurangi tingkat kesalahan manusia dan menjadi lebih cepat dan efisien. (Septiandito Saputra, 2021). Peran teknologi komputer di era *society* 5.0 sangat dibutuhkan dalam pengelolaan *Data* perusahaan, dengan menggunakan teknologi yang terkomputerisasi bisa menghasilkan suatu *Data* yang akurat dan berguna bagi perusahaan atau instansi. Seperti halnya sistem informasi *inventory*, sistem informasi *inventory* merupakan suatu sistem informasi yang digunakan untuk mengelola informasi *Data* penyimpanan barang. Sistem informasi *inventory* sangat diperlukan agar informasi *Data* penyimpanan barang menjadi lebih mudah untuk diproses dan memudahkan *user* pada saat pembuatan *report Data* barang masuk dan barang keluar. (Tarigan & Ardiansyah, 2020).

Pada hasil penelitian sebelumnya yang berjudul sistem informasi *inventory* bahan bangunan, dari penelitian tersebut didapatkan hasil bahwa sistem informasi *inventory* ini berbasis desktop dan *user* pada sistem ini hanya 1 (satu). (Rudiansyah & Purnia, 2022) melihat adanya kekurangan pada penelitian yang telah ada, maka dalam penelitian ini ingin mengembangkan suatu sistem *inventory* berbasis *website* agar lebih *flexible* dalam memasukan *Data* barang masuk maupun barang keluar, dan juga bisa diakses melalui *smartphone*, Selain itu dalam sistem ini terdapat 3 (tiga) *user* yang akan menggunakan sistem *inventory* ini.

Berdasarkan pemaparan permasalahan diatas peneliti memiliki ide yaitu merancang sistem informasi *inventory sparepart* pesawat berbasis *web* agar memudahkan pekerjaan agar lebih efisien, *flexible*, dan mengurangi terjadinya *human error*. Adapun metode yang digunakan yaitu dengan metode *Extreme Programming*, untuk meningkatkan efisiensi serta efektifitas dalam pengerjaan pengembangan perangkat lunak.

II. METODE

Dalam studi ini, digunakan pendekatan Extreme Programming, yang terdiri dari beberapa langkah yaitu perencanaan, perancangan, pembuatan kode, dan pengujian. Berikut adalah penjelasan mengenai setiap Langkah yang terdapat dalam metode extreme programming:



Gambar 1. Alur Metode *Extreme Programming*

Sumber: www.mehmetagile.com

1. Perencanaan Sistem (*Planning*)

Tahap *planning* dimulai dengan melakukan wawancara dan membuat listing daftar kebutuhan yang diinginkan dalam sebuah aplikasi.

Metode yang digunakan dalam proses wawancara bertujuan untuk mengidentifikasi sejumlah permasalahan yang muncul dalam pengaturan informasi terkait barang dan suku cadang. Wawancara ini melibatkan keterlibatan petugas yang bertugas di bagian penyimpanan dan pengadaan.

Pengkajian Pustaka dilaksanakan untuk memperoleh dasar referensi yang relevan dengan konsep dasar teori persediaan barang. Referensi berasal dari berbagai sumber seperti jurnal, dan situs-situs guna mendukung penyusunan laporan akhir.

2. Perancangan Sistem (*Design*)

Pada fase ini, terjadi proses perancangan terhadap aplikasi yang sedang direncanakan. Perancangan ini mencakup pembuatan scenario penggunaan, diagram aktivitas, struktur table ERD, dan juga desain antarmuka.

3. Implementasi Sistem (*Coding*)

Pada fase ini perancangan sistem, dilakukan penggunaan bahasa pemrograman PHP melalui *framework codeigniter*, serta penggunaan *HTML* dan *CSS*, ini dikarenakan sistem ini

melibatkan pembuatan basis data *MySQL* dan penerapan *framework codeigniter* sebagai bagian integral dari proses tersebut.

4. Pengujian (Testing)

Pada fase ini, dilakukan pengujian guna memverifikasi kesesuaian aplikasi yang telah diselesaikan dengan kebutuhan yang ada serta mendeteksi potensi bug pada aplikasi yang akan diuji. Saat dalam proses pengujian, peneliti menjalankan serangkaian tes terhadap aplikasi pengelolaan suku cadang yang telah dibangun. Dalam langkah ini, metode pengujian *black box* digunakan, dengan fokus utama pada pemeriksaan fitur dan fungsi aplikasi.

III. HASIL DAN PEMBAHASAN

Hasil penelitian ini berupa penerapan metode *extreme programming* pada aplikasi sistem *inventory sparepart* pesawat. Berikut pembahasan tahapan *extreme programming* :

1. Tahap Perancangan Sistem (Planning)

Tahap planning dimulai dengan melakukan wawancara, observasi, dan studi pustaka lalu membuat listing daftar kebutuhan yang diinginkan dalam sebuah aplikasi.

Metode yang diterapkan dalam proses wawancara bertujuan untuk mengidentifikasi sejumlah isu yang muncul dalam pengelolaan data barang atau suku cadang. Wawancara dilakukan dengan melibatkan petugas yang bertanggung jawab atas toko dan pengadaan.

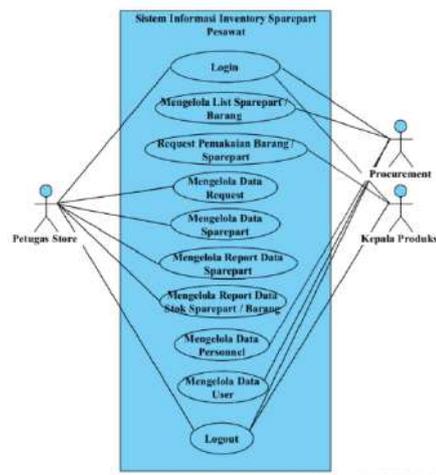
Studi kepustakaan dilaksanakan untuk memperoleh landasan referensial bagi penulis yang terkait dengan teori inventarisasi barang. Sumber-sumber literatur yang digunakan mencakup buku, jurnal, dan situs *web* guna memberikan kontribusi pada penyusunan laporan.

2. Perancangan Sistem (Design)

Pada fase ini, dilakukan penyusunan rencana untuk aplikasi yang akan dibuat. Rencana ini mencakup pembuatan scenario penggunaan, diagram aktivitas, struktur table *Entity-Relationship Diagram (ERD)*, serta desain antarmuka.

a. *Usecase Diagram*

Use case diagram rangkaian visualisasi interaksi komponen aplikasi dengan penggunaannya, berikut *use case* pada sistem informasi *inventory sparepart* pesawat.



Gambar 2. Usecase Diagram

Berikut deskripsi pada *use case diagram* tersebut :

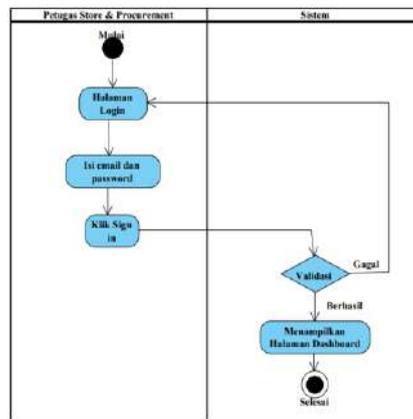
- 1) 3 actor yang melakukan interaksi : kepala produksi, *procurement*, dan petugas store.

- 2) Ada 10 *use case* yang dilakukan kepala produksi, *procurement*, dan petugas *store*.
 Kepala produksi: *login*, request pemakaian barang / *sparepart*, dan *logout*.
Procurement: *login*, mengelola list *sparepart*, mengelola *Data user*, mengelola *Data personnel* dan *logout*.
 Petugas *store*: *login*, mengelola *Data request*, mengelola *Data sparepart* / barang, mengelola *report Data sparepart* / barang, mengelola *report Data stok sparepart* / barang, dan *logout*.

b. *Activity Diagram*

Activity Diagram adalah suatu metode pemodelan yang mengilustrasikan berbagai jalur aktivitas dalam sebuah sistem. Diagram ini merinci bagaimana tiap jalur dimulai, keputusan yang mungkin timbul, dan titik akhir dari masing-masing alur. Selain itu, *Activity Diagram* juga memvisualisasikan proses yang bisa berjalan secara bersamaan dalam beberapa pelaksanaan yang berbeda.

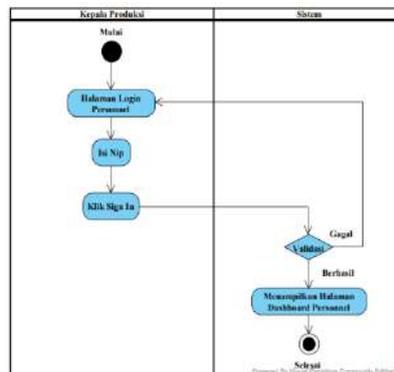
Activity diagram login petugas store dan procurement



Gambar 3. *Activity Diagram Login Petugas Store Dan Procurement*

Berdasarkan gambar 3 dijelaskan :

- a) 1(satu) *initial code*, untuk memulai aksi.
 - b) 4(empat) *action*, sistem yang mencerminkan eksekusi dari suatu aksi.
 - c) 1(satu) *decision*, sistem yang mencerminkan dimana jika ada aktivitas lebih dari satu.
 - d) 1(satu) *activity final node*, objek yang diakhiri.
- 1) *Activity diagram login kepala produksi*

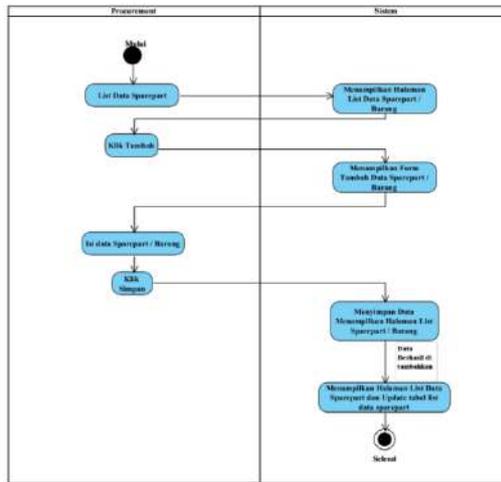


Gambar 4. *Activity Diagram Login Kepala Produksi*

Berdasarkan gambar 4 dapat dijelaskan:

- a. 1(satu) *initial code*, untuk memulai aksi.

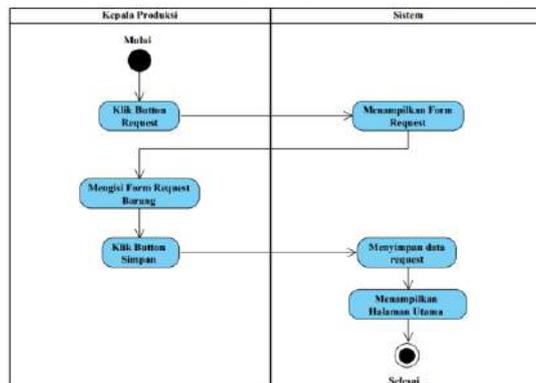
- b. 4(empat) *action*, sistem yang mencerminkan eksekusi dari suatu aksi.
 - c. 1(satu) *decision*, sistem yang mencerminkan dimana jika ada aktivitas lebih dari satu.
 - d. 1(satu) *activity final node*, sebagai objek yang diakhiri.
- 2) *Activity diagram* mengelola *list sparepart / barang*



Gambar 5. *Activity Diagram* mengelola *list sparepart / barang*

Berdasarkan gambar 5 dapat dijelaskan sebagai berikut:

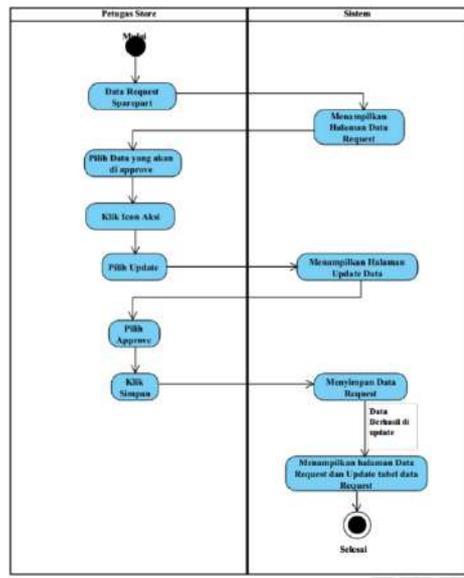
- a. 1(satu) *initial code*, untuk memulai aksi.
 - b. 8(delapan) *action*, sistem yang mencerminkan eksekusi dari suatu aksi.
 - c. 1(satu) *activity final node*, sebagai objek yang diakhiri.
- 3) *Activity diagram request* pemakaian barang / *sparepart*



Gambar 6. *Activity diagram request* pemakaian barang / *sparepart*

Berdasarkan gambar 6 dapat dijelaskan sebagai berikut:

- a. 1(satu) *initial code*, untuk memulai aksi.
 - b. 6(enam) *action*, sistem yang mencerminkan eksekusi dari suatu aksi.
 - c. 1(satu) *activity final node*, sebagai objek yang diakhiri.
- 4) *Activity diagram* mengelola data *request*

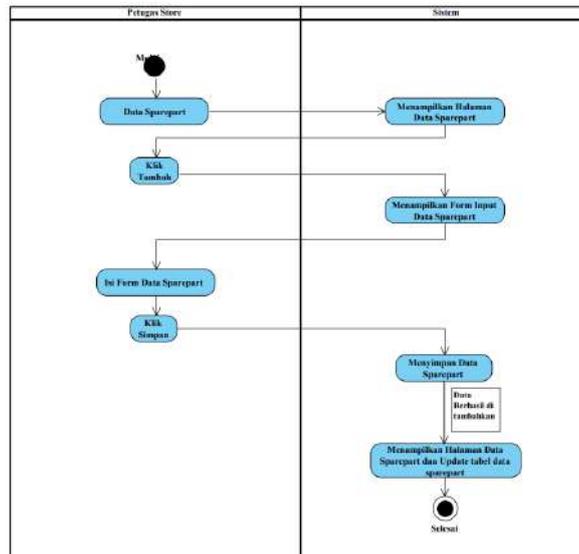


Gambar 7. Activity diagram mengelola data request

Berdasarkan gambar 7 dapat dijelaskan:

- a. 1(satu) *initial code*, untuk memulai aksi.
- b. 10(sepuluh) *action*, sistem yang mencerminkan eksekusi dari suatu aksi.
- c. 1(satu) *activity final node*, sebagai objek yang diakhiri.

5) *Activity diagram* mengelola data *sparepart*

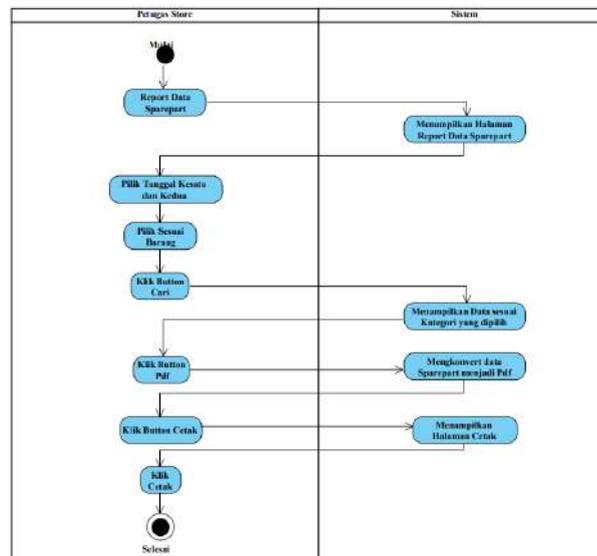


Gambar 8. Activity diagram mengelola data sparepart

Berdasarkan gambar 8 dapat dijelaskan:

- a. 1(satu) *initial code*, untuk memulai aksi.
- b. 10(sepuluh) *action*, sistem yang mencerminkan eksekusi dari suatu aksi.
- c. 1(satu) *activity final node*, sebagai objek yang diakhiri.

6) *Activity diagram* mengelola *report data sparepart*

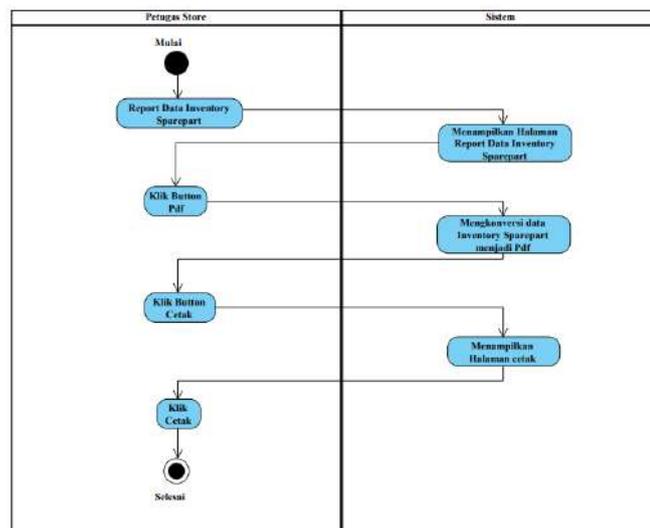


Gambar 9. Activity diagram mengelola report data sparepart

Berdasarkan gambar 9 dapat dijelaskan:

- a. 1(satu) *initial code*, untuk memulai aksi.
- b. 11(sebelas) *action*, sistem yang mencerminkan eksekusi dari suatu aksi.
- c. 1(satu) *activity final node*, sebagai objek yang diakhiri.

7) Activity diagram mengelola report data stok sparepart / barang

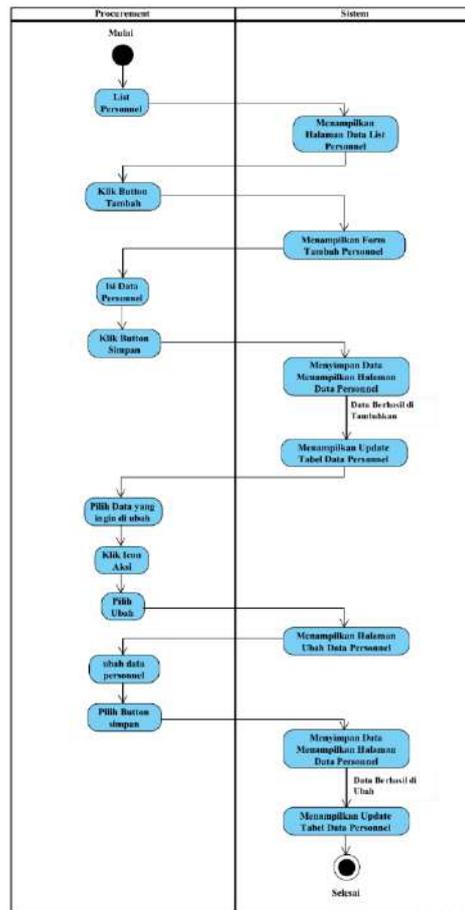


Gambar 10. Activity diagram report data stok sparepart / barang

Berdasarkan gambar 10 dapat dijelaskan:

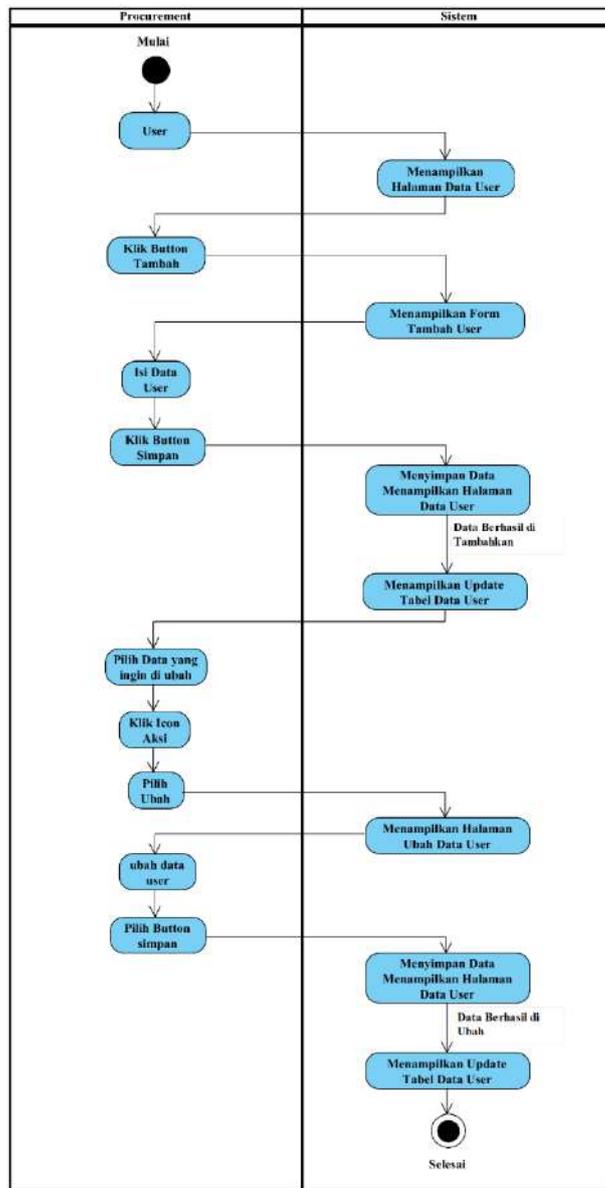
- a. 1(satu) *initial code*, untuk memulai aksi.
- b. 7(tujuh) *action*, sistem yang mencerminkan eksekusi dari suatu aksi.
- c. 1(satu) *activity final node*, sebagai objek yang diakhiri.

8) Activity diagram mengelola data personnel



Gambar 11. Activity diagram mengelola data *personnel*
 Berdasarkan gambar 11 dapat dijelaskan:

- a. 1(satu) *initial code*, untuk memulai aksi.
 - b. 16(enam belas) *action*, sistem yang mencerminkan eksekusi dari suatu aksi.
 - c. 1(satu) *activity final node*, sebagai objek yang diakhiri.
- 9) *Activity diagram* mengelola data *user*



Gambar 12. Activity diagram mengelola data user

Berdasarkan gambar 12 dapat dijelaskan:

- a. 1(satu) *initial code*, untuk memulai aksi.
 - b. 16(enam belas) *action*, sistem yang mencerminkan eksekusi dari suatu aksi.
 - c. 1(satu) *activity final node*, sebagai objek yang diakhiri.
- c. Struktur Tabel
- 1) Tabel *Users*

Tabel 1. Tabel *Users*

No	Name	Type Data	Value	Keterangan
1.	id_user	Int	11	Primary Key
2.	name	varchar	50	

3.	email	varchar	50
4.	password	varchar	50
5.	status	enum	'admin', 'staff'
6.	foto	varchar	255

2) Tabel Barang

Tabel 2. Tabel Barang

No	Name	Tipe Data	Value	Keterangan
1.	id_barang	Int	11	Primary Key
2.	kd_barang	varchar	10	
3.	nm_barang	varchar	50	
4.	stock	int	11	
5.	id_satuan	int	11	

3) Tabel Barang Masuk

Tabel 3. Tabel Barang Masuk

No	Name	Tipe Data	Value	Keterangan
1.	id_masuk	Int	11	Primary Key
2.	kd_barang	varchar	20	
3.	tgl_masuk	varchar	20	
4.	part_number	varchar	50	
5.	serial_number	varchar	50	
6.	qty	int	11	
7.	remarks	varchar	100	
8.	vendor	varchar	100	

4) Tabel Barang Keluar

Tabel 4. Tabel Barang Keluar

No	Name	Tipe Data	Value	Keterangan
1.	id_keluar	Int	11	Primary Key
2.	kd_barang	varchar	20	
3.	tgl_keluar	varchar	20	
4.	part_number	varchar	50	
5.	serial_number	varchar	50	
6.	qty	int	11	
7.	keterangan	varchar	100	

5) Tabel Request

Tabel 5. Tabel Request

No	Name	Tipe Data	Value	Keterangan
1.	id_request	Int	11	Primary

				<i>Key</i>
2.	kd_barang	varchar	20	
3.	tgl_request	varchar	20	
4.	nama	varchar	20	
5.	nip	varchar	20	
6.	qty	int	11	
7.	keperluan	varchar	50	
8.	approve	varchar	20	

6) Tabel *Personnel*

Tabel 6. Tabel *Personnel*

No	<i>Name</i>	<i>Type Data</i>	<i>Value</i>	<i>Keterangan</i>
1.	id_personnel	Int	11	<i>Primary Key</i>
2.	nama_personnel	varchar	100	
3.	id_depart	int	11	
4.	nip_personnel	varchar	50	
5.	status	varchar	10	

7) Tabel Satuan

Tabel 7. Tabel Satuan

No	<i>Name</i>	<i>Type Data</i>	<i>Value</i>	<i>Keterangan</i>
1.	id_satuan	Int	11	<i>Primary Key</i>
2.	satuan	varchar	100	

8) Tabel *Departement*

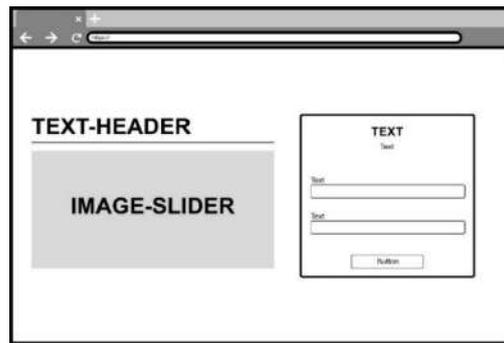
Tabel 8. Tabel *Departement*

No	<i>Name</i>	<i>Type Data</i>	<i>Value</i>	<i>Keterangan</i>
1.	id_depart	Int	11	<i>Primary Key</i>
2.	departement	varchar	20	

d. Desain Tampilan Antar Muka

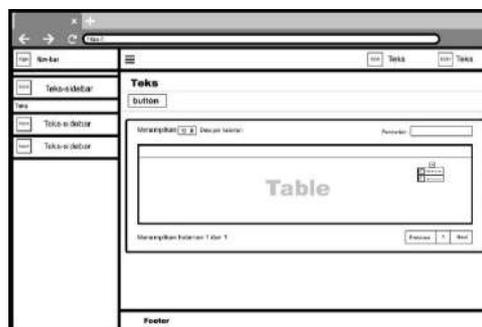
Pada implementasi antarmuka / *user interface* terdapat beberapa halaman yang menjelaskan proses untuk penggunaan aplikasi ini adalah sebagai berikut :

1) Antar Muka Halaman *Login*



Gambar 13. Tampilan Halaman *Login*

2) Antar Muka Halaman Mengelola Data *Sparepart*



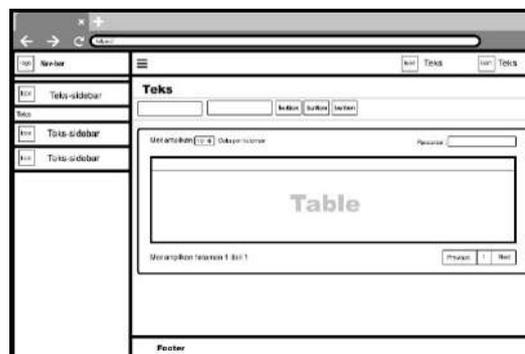
Gambar 14. Tampilan Halaman Mengelola Data *Sparepart*

3) Antar Muka *Form* Tambah Data



Gambar 15. Tampilan *Form* Tambah Data

4) Antar Muka Halaman *Report* Data

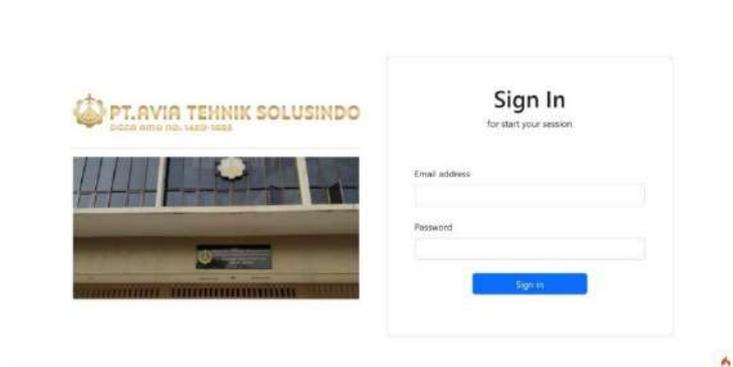


Gambar 16. Tampilan Halaman *Report* Data

3. Implementasi Sistem (*Coding*)

Pada proses perencanaan sistem, digunakan bahasa pemrograman *PHP* melalui kerangka kerja *CodeIgniter*, serta *HTML* dan *CSS*. Ini disebabkan oleh sifat sistem yang berbasis situs *web*, dan pengembang dilakukan dengan bantuan kode editor *Visual studio Code*. Langkah ini mencakup pembuatan basis data *MySQL* serta penerapan *framework CodeIgniter*.

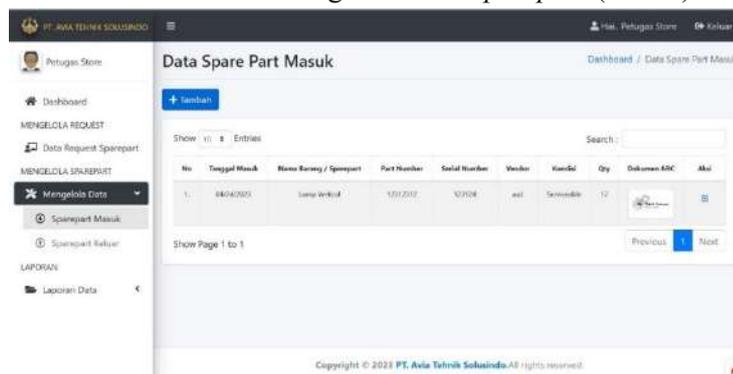
a. Antar Muka Halaman *Login*



Gambar 17. Antar Muka Halaman *Login*

Gambar 17 merupakan hasil tampilan dari *coding* pada halaman *login*.

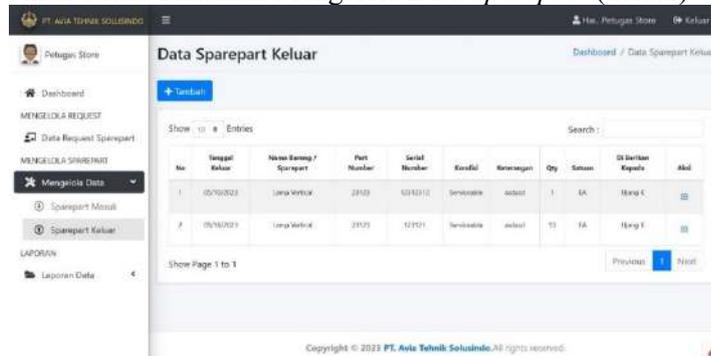
b. Antar Muka Halaman Mengelola Data *Sparepart* (masuk)



Gambar 18. Antar Muka Halaman Mengelola Data *Sparepart* (masuk)

Gambar 18 merupakan antar muka hasil dari *coding* pada halaman mengelola data *sparepart* masuk.

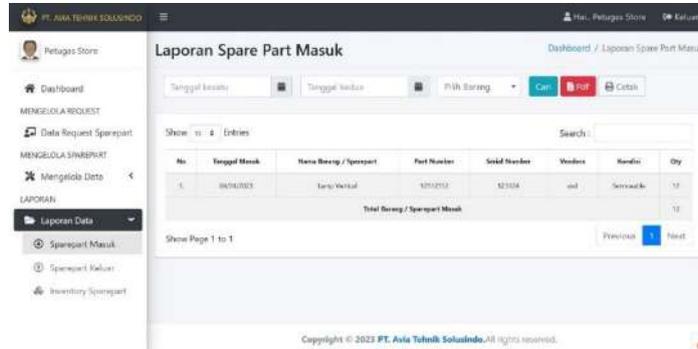
c. Antar Muka Halaman Mengelola Data *Sparepart* (keluar)



Gambar 19. Antar Muka Halaman Mengelola Data *Sparepart* (keluar)

Gambar 19. Antar muka dari hasil *coding* pada halaman mengelola data *sparepart* (keluar).

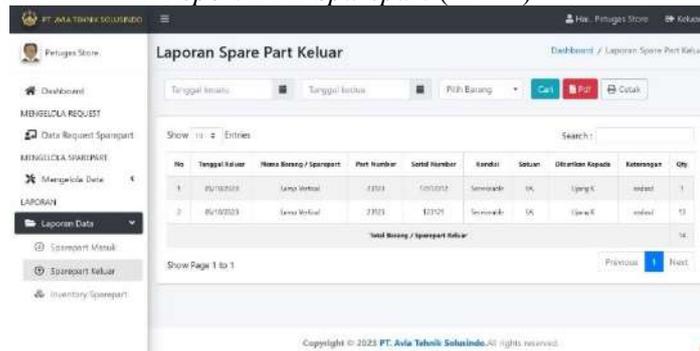
d. Antar Muka Halaman *Report Data Sparepart* (masuk)



Gambar 20. Antar Muka Halaman *Report Data Sparepart* (masuk)

Gambar 20. Antar muka hasil dari *coding* pada halaman *report data sparepart* (masuk).

e. Antar Muka Halaman *Report Data Sparepart* (keluar)



Gambar 21. Antar muka Halaman *Report Data Sparepart* (keluar)

Gambar 21. Antar muka hasil dari *coding* pada halaman *report data sparepart* (keluar).

4. Pengujian (*Testing*)

Pada proses pengujian, penulis menguji aplikasi inventaris suku cadang yang telah dibuat. Dalam tahap ini, pendekatan pengujian yang digunakan adalah *black-box testing*, di mana penekanan diberikan pada pengujian fitur dan fungsi aplikasi.

Tabel 9. Pengujian *Black-box*

No	Pengujian	Test Case	Hasil Yang diharapkan	Hasil Pengujian	Keterangan
1.	<i>Email</i> dan password kosong, kemudian klik tombol sign in.		Menolak akses login dan meminta lengkapi kolom " <i>Email</i>		Valid sesuai yang diharapkan

			<i>must be insert” dan “password must be insert”</i>		
3.	<i>email dan password terisi</i>		Sistem akan melanjutkan ke halaman dashboard		Valid sesuai yang diharapkan :
4.	Menambahkan <i>Data list sparepart</i>		Sistem akan menyimpan <i>Data</i> yang telah ditambahkan		Valid Sesuai yang diharapkan :
6.	Mengelola <i>Data Request</i>		Sistem akan menyimpan perubahan <i>Data</i>		Valid sesuai yang diharapkan
7.	Mengelola <i>Data sparepart</i>		Sistem akan menyimpan <i>Data</i> yang telah di tambahkan		Valid sesuai yang diharapkan
8.	Menampilkan <i>Data sparepart</i> berdasarkan kategori tanggal dan barang. Lalu klik cari		Sistem menampilkan <i>Data sparepart</i> berdasarkan kategori		Valid sesuai yang diharapkan
9.	Mengelola <i>report Data sparepart</i> , membuat file pdf <i>Data sparepart</i> , klik tombol pdf		Sistem akan mengeksport file pdf <i>Data sparepart</i>		Valid sesuai yang diharapkan

10.	Mengelola <i>report Data sparepart</i> , Cetak <i>Data sparepart</i>		Sistem akan menampilkan halaman cetak <i>Data</i>		Valid sesuai yang diharapkan
11.	Mengelola <i>report Data stok sparepart</i> , Cetak <i>Data sparepart</i>		Sistem akan menampilkan halaman cetak <i>Data stok sparepart</i>		Valid sesuai yang diharapkan
12.	Mengelola <i>report Data stok sparepart</i> , pdf		Sistem akan mengekspor file pdf <i>Data stok sparepart</i>		Valid sesuai yang diharapkan

Berdasarkan tabel di atas menunjukkan bahwa hasil pengujian valid dengan yang diharapkan. Semua perintah atau fungsi pada saat digunakan berjalan dengan baik.

IV. KESIMPULAN

Dapat disimpulkan bahwa hasil penelitian ini telah berhasil merancang sistem informasi *inventory sparepart* pesawat dengan tujuan membantu dan memudahkan dalam melakukan pengelolaan data *sparepart* dan penyimpanan berkas *data sparepart* atau barang sudah tersimpan baik di dalam *database*. Pada implementasi metode *extreme programming* memudahkan dalam perancangan sistem informasi *inventory sparepart* pesawat. Dan perancangan aplikasi menjadi lebih teratur sesuai dengan tahapan-tahapannya.

V. DAFTAR PUSTAKA

Fatmawati, F., & Munajat, J. (2018). Implementasi Model Waterfall Pada Sistem Informasi Persediaan Barang Berbasis Web (Studi Kasus: PT.Pamindo Tiga T). *Jurnal Media Informatika Budidarma*, 2(2), 1–9. <https://doi.org/10.30865/mib.v2i2.559>

Lesmana, R. A., Bakti, A. M., Studi, P., Informatika, T., & Ilmu, F. (2023). *Usability Testing Pada Website SMK Negeri 1 Suak Tapeh Menggunakan System Usability Scale (SUS)*. 365–371.

Pribachtiar, R. A., & Utomo, A. P. (2021). Perancangan Sistem Informasi Inventory Barang (E-Gudang) Pada Cv Jaya Water Solusindo Berbasis Website. *Jurnal IKRAITH-INFORMATIKA*, 5(3), 54–63.

Rudiansyah, I., & Purnia, D. S. (2022). *Sistem Informasi Inventory Bahan Bangunan (SIABANG) Studi Kasus CV. Sinar Jaya Tangerang Menggunakan Metode*. 8(2), 2018–2023.

- Rudiansyah, I., & Purnia, D. S. (2022). *Sistem Informasi Inventory Bahan Bangunan (SIABANG) Studi Kasus CV . Sinar Jaya Tangerang Menggunakan Metode. 8(2)*, 2018–2023.
- Septiandito Saputra, A. (2021). Pengaruh Teknologi Informasi Pada Koperasi Di Era Industri 4.0. *Transekonomika: Akuntansi, Bisnis Dan Keuangan, 1(5)*, 505–510. <https://doi.org/10.55047/transekonomika.v1i5.77>
- Supriyatna, A. (2018). Metode Extreme Programming Pada Pembangunan Web Aplikasi Seleksi Peserta Pelatihan Kerja. *Jurnal Teknik Informatika, 11(1)*, 1–18. <https://doi.org/10.15408/jti.v11i1.6628>
- Tarigan, R., & Ardiansyah, D. (2020). Perancangan Aplikasi Inventory Barang Pada Cv. Mr Lestari Berbasis Web. *Jurnal Sistem Informasi Dan Informatika (Simika), 3(2)*, 77–94. <https://doi.org/10.47080/simika.v3i2.985>