

BAB III

PELAKSANAAN KERJA PROFESI

3.1 Bidang Kerja

Kegiatan Kerja Profesi dilaksanakan selama 4 bulan di PT Tirta Surya Raya. Dalam masa Kerja Profesi, Praktikan memiliki peran sebagai *IT* untuk pengembangan design aplikasi *Inventory*. Pengembangan aplikasi menggunakan kerangka kerja dari bahasa pemrograman PHP, yaitu Laravel. Sqliite, CSS, Javascrib, Electron, HTML. Praktikan memiliki tanggung jawab atas beberapa fitur dalam aplikasi *Inventory* seperti *Home*, *OS*, *Delivery Schedule* , *Barang Masuk*, *Barang Keluar*, *Inventory*. Selain itu, terdapat uji coba hasil pengembangan dari berbagai fitur tersebut dan melakukan perbaikan jika terdapat *error* pada saat pengembangan. Berikut beberapa poin-poin secara luas yang menjadi peran dan tanggung jawab sebagai *IT*, yaitu:

- 1) Menerapkan desain UI/UX buatan pengguna dalam kode menggunakan HTML (HyperText Markup Language), CSS (Cascading Stylesheets), dan bahasa pemrograman JavaScript untuk membuat layar interaktif.
- 2) Pastikan antarmuka dan fungsi aplikasi dapat bekerja dengan baik pada perangkat.
- 3) Pastikan kode ditulis secara efisien dan dapat dieksekusi dengan cepat. Termasuk penggunaan komponen yang dapat digunakan kembali untuk meningkatkan waktu respons dan kecepatan akses.
- 4) Memelihara dan menyelesaikan masalah yang muncul pada aplikasi website. Ini melibatkan kode pemecahan masalah, menangani kesalahan, dan meningkatkan fungsionalitas.
- 5) Bekerja dengan tim Manajemen TI untuk mengintegrasikan kode ke *database*.

3.2 Pelaksanaan Kerja

Pelaksanaan Kerja Profesi dimulai pada tanggal 27 Februari 2023 sampai 1 Juni 2023 di PT. Tirta Surya Raya yang dilaksanakan selama 5 hari kerja, yaitu Senin – Jumat pukul 08.00 – 17.00 WIB.

Selama periode Kerja Profesi, Praktikan telah melakukan beberapa tugas saat berkontribusi dalam pengembangan antarmuka aplikasi Inventory sebagai IT.

3.2.1 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem adalah tahap awal dan krusial dalam pembuatan dasar untuk pengembangan sistem yang akan datang. pada dasarnya, analisis kebutuhan sistem melibatkan analisis kebutuhan fungsional dan analisis kebutuhan non-fungsional. Analisis kebutuhan fungsional berfokus pada fungsi-fungsi atau fitur-fitur yang harus ada dalam sistem. Analisa kebutuhan non-fungsional berfokus pada aspek-aspek yang tidak berkaitan dengan fungsi langsung sistem, tetapi mempengaruhi kinerja, keamanan, dan pengalaman pengguna.

Berikut merupakan tabel rincian analisis kebutuhan fungsional sistem aplikasi *Inventory* secara keseluruhan:

Tabel 3.1 Analisis Kebutuhan Fungsional

No.	Kebutuhan Fungsional	Keterangan
1	Home Aplikasi	Penggunan dapat melihat dan memilih bagian apa yang aka di kerjakan
2	Inventory	Pengguna bisa melihat Stok terupdate
3	Barang Masuk	Pengguna dapat menambahkan stok barang kedalam aplikasi sehingga dapat terupdate

4	Barang Keluar	Pengguna dapat mengurangi stok barang sesuai dengan request pengiriman
5	OS	Pengguna dapat OS (Order Sheet) diperuntukan untuk dokument tambahan
6	Delivery Schedule	Pengguna bisa request pengiriman untuk semua barang

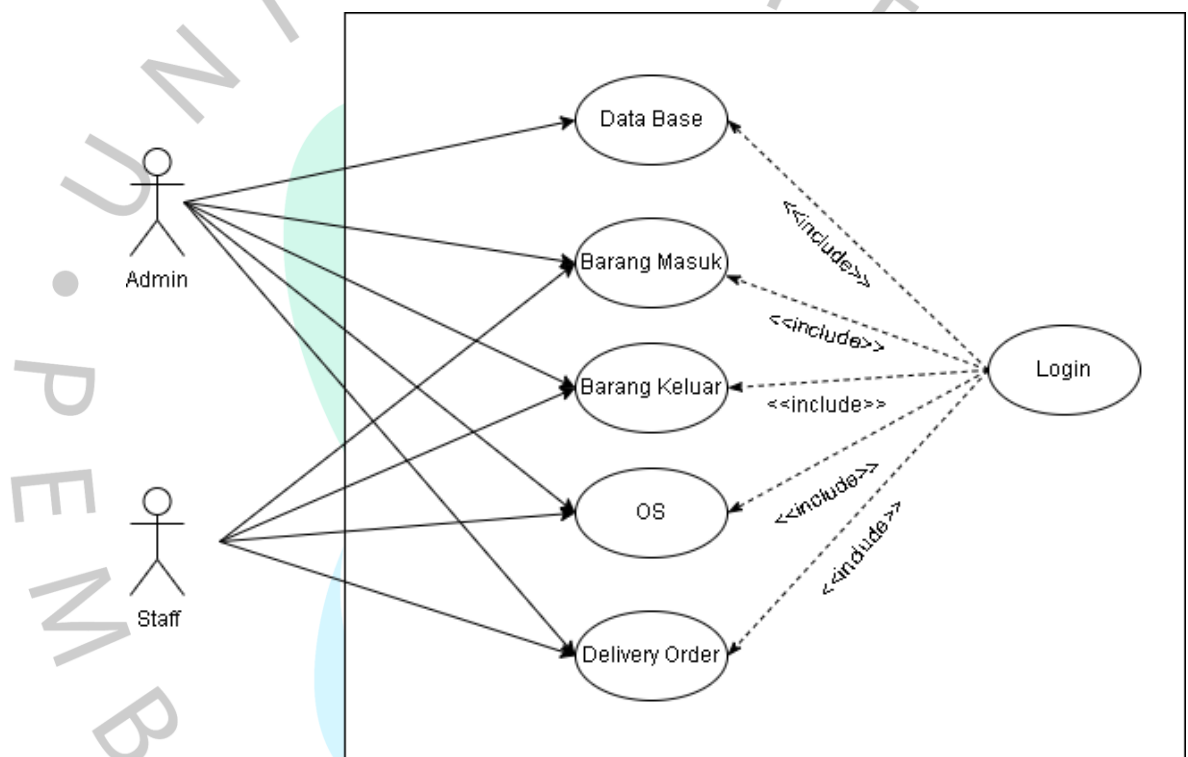
Selain itu, terdapat kebutuhan non-fungsional pada sistem aplikasi *Inventory*:

Tabel 3.2 Analisis Kebutuhan Non-Fungsional

No.	Kebutuhan Non-Fungsional	Keterangan
1	Responsif	Sistem harus memberikan pengalaman responsif dengan waktu respons yang cepat saat pengguna berinteraksi.
2	Keamanan	Sistem harus menyediakan mekanisme keamanan yang kuat, termasuk enkripsi data pengguna dan autentikasi yang aman.
3	Aksesibilitas tinggi	Sistem dapat diakses dan digunakan dengan mudah oleh pengguna.

4	Dapat di integrasi	Sistem harus dapat diintegrasikan dengan sistem yang ada di perusahaan.
---	--------------------	---

3.2.3 Use Case Diagram



Gambar 3.3 Use Case Diagram

Pada gambar 3.3 di atas, menggambarkan interaksi antara aktor dengan sistem. Terdapat dua aktor yang terlibat di dalam sistem aplikasi inventory. Admin bertindak sebagai Administrator yang dapat melakukan login, register, home, kelola inventory, kelola barang masuk, kelola barang keluar, kelola OS, kelola delivery schedule.

Staff yang bertindak sebagai guru dapat melakukan login, kelola inventory, kelola barang masuk, kelola barang keluar, kelola OS, kelola delivery schedule..

3.2.4 Use Case Scenario

Use Case Scenario adalah penjelasan sebuah interaksi antara pengguna (aktor) dan sistem yang sedang dalam pengembangan. Use case scenario menggambarkan langkah-langkah spesifik yang dilakukan oleh pengguna dan sistem guna mencapai suatu tujuan yang ditentukan. Berikut merupakan rincian tabel use case scenario aplikasi inventory.

Tabel 3.4 Use Case Scenario Register

<i>Use Case</i>	<i>Register</i>														
Deskripsi	Admin ingin membuat akun baru di sistem														
Aktor Utama	Admin														
Aktor Pendukung	-														
Pra-syarat	-														
Alur Utama	<table border="1"><thead><tr><th><i>Step</i></th><th><i>Action</i></th></tr></thead><tbody><tr><td>1</td><td>Admin membuka halaman register</td></tr><tr><td>2</td><td>menampilkan formulir registrasi yang meminta admin memasukkan informasi seperti nama, username, email, dan password</td></tr><tr><td>3</td><td>Admin mengisi formulir registrasi dengan informasi yang valid</td></tr><tr><td>4</td><td>Admin mengklik tombol "Register"</td></tr><tr><td>5</td><td>Sistem memvalidasi informasi yang dimasukkan Admin</td></tr><tr><td>6</td><td>Jika informasi valid, sistem membuat akun dan mengirimkan notifikasi</td></tr></tbody></table>	<i>Step</i>	<i>Action</i>	1	Admin membuka halaman register	2	menampilkan formulir registrasi yang meminta admin memasukkan informasi seperti nama, username, email, dan password	3	Admin mengisi formulir registrasi dengan informasi yang valid	4	Admin mengklik tombol "Register"	5	Sistem memvalidasi informasi yang dimasukkan Admin	6	Jika informasi valid, sistem membuat akun dan mengirimkan notifikasi
<i>Step</i>	<i>Action</i>														
1	Admin membuka halaman register														
2	menampilkan formulir registrasi yang meminta admin memasukkan informasi seperti nama, username, email, dan password														
3	Admin mengisi formulir registrasi dengan informasi yang valid														
4	Admin mengklik tombol "Register"														
5	Sistem memvalidasi informasi yang dimasukkan Admin														
6	Jika informasi valid, sistem membuat akun dan mengirimkan notifikasi														

	7	Sistem mengarahkan pengguna ke halaman login
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.5 Use Case Scenario Login

<i>Use Case</i>	<i>Login</i>	
Deskripsi	Admin/Staff ingin masuk ke aplikasi dengan akun yang valid	
Aktor Utama	Admin, Staff	
Aktor Pendukung	-	
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	
Alur Utama	<i>Step</i>	<i>Action</i>
	1	Admin / Staff membuka halaman login
	2	Sitem menampilkan formulir login yang meminta pengguna memasukkan username dan password
	3	Admin / Staff mengisi username dan password yang valid
	4	Admin / Staff mengklik tombol "Login"
	5	Sistem memvalidasi data login
	6	Jika data login valid, sistem mengarahkan pengguna ke halaman dashboard. Jika tidak, sistem menampilkan pesan kesalahan
Alur Alternatif	-	

Alur Pengecualian	Step	Action
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.6 Use Case Scenario dashboard

Use Case	Halaman Home	
Deskripsi	Admin/Staff ingin mengakses halaman home	
Aktor Utama	Admin, Staff	
Aktor Pendukung	-	-
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	
Alur Utama	<i>Step</i>	<i>Action</i>
	1	Admin / Staff berhasil login dan diarahkan ke halaman dashboard
	2	Sitem memvalidasi status login
	3	Sistem menampilkan halaman dashboard
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.7 Use Case Scenario Halaman Inventory

Use Case	Halaman Inventory
Deskripsi	Admin/Staff ingin mengakses halaman Inventory
Aktor Utama	Admin, Staff

Aktor Pendukung	-	-
Pra-syarat	-	Admin / Staff telah memiliki akun terdaftar di sistem
Alur Utama	<i>Step</i>	<i>Action</i>
	1	Admin / Staff berhasil login dan diarahkan ke halaman dashboard
	2	Admin / Staff mengarahkan ke halaman inventory
	3	Sistem menampilkan halaman inventory
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.8 Use Case Scenario Halaman Barang Masuk

<i>Use Case</i>	Halaman Barang Masuk	
Deskripsi	Admin/Staff ingin mengakses halaman barang masuk	
Aktor Utama	Admin, Staff	
Aktor Pendukung	-	-
Pra-syarat	-	Admin / Staff telah memiliki akun terdaftar di sistem
Alur Utama	<i>Step</i>	<i>Action</i>
	1	Admin / Staff berhasil login dan diarahkan ke halaman dashboard
	2	Admin / Staff mengarahkan ke halaman barang masuk

	3	Sistem menampilkan halaman barang masuk
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.9 Use Case Scenario Tambah Barang

<i>Use Case</i>	Tambah Barang	
<i>Deskripsi</i>	Admin/Staff ingin menambah barang	
<i>Aktor Utama</i>	Admin, Staff	
<i>Aktor Pendukung</i>	-	
<i>Pra-syarat</i>	- Admin / Staff telah memiliki akun terdaftar di sistem	
<i>Alur Utama</i>	<i>Step</i>	<i>Action</i>
	1	Admin / Staff berhasil login dan sistem memvalidasi status login
	2	Admin / Staff mengarahkan ke halaman barang masuk
	3	Admin / staff mengeklik ikon "Tambah" pada halaman barang masuk
	4	Sistem menampilkan halaman tambah barang
	5	Admin mengisi formulir nomor kode barang, nama barang, berat barang, jumlah barang tanggal masuk.
	6	Admin / staff mengeklik tombol "Submit"
7	Sistem memvalidasi data barang	

	8	Jika data barang valid, sistem menyimpan data barang ke dalam <i>database</i> .
	9	Sistem mengarahkan ke halaman barang masuk dan menampilkan pesan konfirmasi bahwa data barang berhasil ditambahkan
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.10 Use Case Scenario Ubah Data Barang Masuk

<i>Use Case</i>	Ubah Data Barang Masuk	
Deskripsi	Admin/Staff ingin mengubah barang masuk	
Aktor Utama	Admin, Staff	
Aktor Pendukung	-	
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	
Alur Utama	<i>Step</i>	<i>Action</i>
	1	Admin / staff berhasil login dan sistem memvalidasi status login
	2	Admin / staff mengarahkan ke halaman barang masuk
	3	Admin / staff mengeklik ikon "Edit" pada halaman barang masuk
	4	Sistem menampilkan halaman ubah barang masuk

	5	Admin / staff mengubah data formulir kode barang, nama barang, berat barang, jumlah barang tanggal masuk.
	6	Admin / staff mengeklik tombol "Update"
	7	Sistem memvalidasi data barang masuk
	8	Jika data barang masuk valid, sistem menyimpan data barang masuk ke dalam <i>database</i> .
	9	Sistem mengarahkan ke halaman barang masuk dan menampilkan pesan konfirmasi bahwa data barang masuk berhasil diubah
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.11 Use Case Scenario Hapus Data Barang Masuk

<i>Use Case</i>	Hapus Data Barang Masuk	
<i>Deskripsi</i>	Admin/Staff ingin mengubah barang masuk	
<i>Aktor Utama</i>	Admin, Staff	
<i>Aktor Pendukung</i>	- -	
<i>Pra-syarat</i>	- Admin / Staff telah memiliki akun terdaftar di sistem	
<i>Alur Utama</i>	<i>Step</i>	<i>Action</i>
	1	Admin / Staff berhasil login dan sistem memvalidasi status login
	2	Admin / Staff mengarahkan ke halaman barang masuk

	3	Sistem menampilkan halaman barang masuk
	4	Admin mengeklik ikon “Hapus” pada halaman barang masuk
	5	Sistem menghapus data barang masuk di <i>database</i>
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.12 Use Case Scenario Barang Keluar

<i>Use Case</i>	Halaman Barang Keluar	
Deskripsi	Admin/Staff ingin mengakses halaman barang keluar	
Aktor Utama	Admin, Staff	
Aktor Pendukung	-	-
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	
Alur Utama	<i>Step</i>	<i>Action</i>
	1	Admin / Staff berhasil login dan diarahkan ke halaman dashboard
	2	Admin / Staff mengarahkan ke halaman barang keluar
	3	Sistem menampilkan halaman barang keluar
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>

	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan
--	---	--

Tabel 3.13 Use Case Scenario Tambah Barang keluar

<i>Use Case</i>	Tambah Barang keluar	
<i>Deskripsi</i>	Admin/Staff ingin menambah barang keluar	
<i>Aktor Utama</i>	Admin, Staff	
<i>Aktor Pendukung</i>	- -	
<i>Pra-syarat</i>	- Admin / Staff telah memiliki akun terdaftar di sistem	
<i>Alur Utama</i>	<i>Step</i>	<i>Action</i>
	1	Admin / Staff berhasil login dan sistem memvalidasi status login
	2	Admin / Staff mengarahkan ke halaman barang keluar
	3	Admin / staff mengklik ikon "Tambah" pada halaman barang keluar
	4	Sistem menampilkan halaman tambah barang keluar
	5	Admin / staff mengisi formulir nomor kode barang, nama barang, berat barang, jumlah barang tanggal keluar, alamat customer, nama customer, pic.
	6	Admin / staff mengklik tombol "Submit"
	7	Sistem memvalidasi data barang keluar
	8	Jika data barang valid, sistem mengurangi data barang di dalam <i>database</i> .

	9	Sistem mengarahkan ke halaman barang keluar dan menampilkan pesan konfirmasi bahwa data barang berhasil dikurangi
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.14 Use Case Scenario Ubah Data Barang keluar

<i>Use Case</i>	Ubah Data Barang keluar	
Deskripsi	Admin/Staff ingin mengubah barang keluar	
Aktor Utama	Admin, Staff	
Aktor Pendukung	-	
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	
Alur Utama	<i>Step</i>	<i>Action</i>
	1	Admin / staff berhasil login dan sistem memvalidasi status login
	2	Admin / staff mengarahkan ke halaman barang keluar
	3	Admin / staff mengklik ikon "Edit" pada halaman barang keluar
	4	Sistem menampilkan halaman ubah barang keluar
	5	Admin / staff mengubah data formulir kode barang, nama barang, berat barang, jumlah barang tanggal keluar, alamat customer, nama customer, pic.

	6	Admin / staff mengeklik tombol "Update"
	7	Sistem memvalidasi data barang keluar
	8	Jika data barang keluar valid, sistem menyimpan data barang keluar ke dalam <i>database</i> .
	9	Sistem mengarahkan ke halaman barang keluar dan menampilkan pesan konfirmasi bahwa data barang keluar berhasil diubah
Alur Alternatif	-	
Alur Pengecualian	Step	Action
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.15 Use Case Scenario Hapus Data Barang Keluar

Use Case	Hapus Data Barang keluar	
Deskripsi	Admin/Staff ingin mengubah barang keluar	
Aktor Utama	Admin, Staff	
Aktor Pendukung	-	-
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	
Alur Utama	Step	Action
	1	Admin / Staff berhasil login dan sistem memvalidasi status login
	2	Admin / Staff mengarahkan ke halaman barang keluar
	3	Sistem menampilkan halaman barang keluar
	4	Admin mengeklik ikon "Hapus" pada halaman barang keluar

	5	Sistem menghapus data barang keluar di <i>database</i>
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.16 Use Case Scenario OS (Order Sheet)

<i>Use Case</i>	OS (Order Sheet)	
Deskripsi	Admin/Staff ingin mengakses halaman OS	
Aktor Utama	Admin, Staff	
Aktor Pendukung	-	
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	
Alur Utama	<i>Step</i>	<i>Action</i>
	1	Admin / Staff berhasil login dan diarahkan ke halaman dashboard
	2	Admin / Staff mengarahkan ke halaman OS
	3	Sistem menampilkan halaman OS
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.17 Use Case Scenario Tambah OS (Order Sheet)

Use Case	Tambah Barang keluar	
Deskripsi	Admin/Staff ingin menambah barang keluar	
Aktor Utama	Admin, Staff	
Aktor Pendukung	-	
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	
Alur Utama	Step	Action
	1	Admin / Staff berhasil login dan sistem memvalidasi status login
	2	Admin / Staff mengarahkan ke halaman OS
	3	Admin / staff mengklik ikon "Tambah" pada halaman OS
	4	Sistem menampilkan halaman tambah OS
	5	Admin / staff mengisi formulir nomor kode barang, nama barang, berat barang, jumlah barang tanggal keluar, alamat customer, nama customer, pic.
	6	Admin / staff mengklik tombol "Submit"
	7	Sistem memvalidasi data OS
	8	Jika data OS valid, sistem menambah data OS di dalam <i>database</i> .
	9	Sistem mengarahkan ke halaman OS dan menampilkan pesan konfirmasi bahwa data OS berhasil ditambah
Alur Alternatif	-	
Alur Pengecualian	Step	Action

	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan
--	---	--

Tabel 3.18 Use Case Scenario Ubah OS (Order Sheet)

Use Case	Ubah Data OS	
Deskripsi	Admin/Staff ingin mengubah OS	
Aktor Utama	Admin, Staff	
Aktor Pendukung	- -	
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	
Alur Utama	Step	Action
	1	Admin / staff berhasil login dan sistem memvalidasi status login
	2	Admin / staff mengarahkan ke halaman OS
	3	Admin / staff mengeklik ikon "Edit" pada OS
	4	Sistem menampilkan halaman ubah OS
	5	Admin / staff mengubah data formulir kode barang, nama barang, berat barang, jumlah barang tanggal keluar, alamat customer, nama customer, pic.
	6	Admin / staff mengeklik tombol "Update"
	7	Sistem memvalidasi data OS
	8	Jika data OS valid, sistem menyimpan data OS ke dalam <i>database</i> .
	9	Sistem mengarahkan ke halaman OS dan menampilkan pesan konfirmasi bahwa data OS berhasil diubah

Alur Alternatif	-	
Alur Pengecualian	Step	Action
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.19 Use Case Scenario Hapus OS (Order Sheet)

Use Case	Hapus Data OS	
Deskripsi	Admin/Staff ingin mengubah barang keluar	
Aktor Utama	Admin, Staff	
Aktor Pendukung	- -	
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	
Alur Utama	Step	Action
	1	Admin / Staff berhasil login dan sistem memvalidasi status login
	2	Admin / Staff mengarahkan ke halaman OS
	3	Sistem menampilkan halaman OS
	4	Admin mengklik ikon "Hapus" pada halaman OS
	5	Sistem menghapus data OS di <i>database</i>
Alur Alternatif	-	
Alur Pengecualian	Step	Action
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.20 Use Case Scenario Delivery Schedule

<i>Use Case</i>	Delivery Schedule	
Deskripsi	Admin/Staff ingin mengakses halaman Delivery Schedule	
Aktor Utama	Admin, Staff	
Aktor Pendukung	-	
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	
Alur Utama	<i>Step</i>	<i>Action</i>
	1	Admin / Staff berhasil login dan diarahkan ke halaman dashboard
	2	Admin / Staff mengarahkan ke halaman delivery schedule
	3	Sistem menampilkan halaman delivery schedule
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.21 Use Case Scenario Tambah Data Delivery Schedule

<i>Use Case</i>	Tambah delivery schedule	
Deskripsi	Admin/Staff ingin menambah delivery schedule	
Aktor Utama	Admin, Staff	
Aktor Pendukung	-	
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	

Alur Utama	Step	Action
	1	Admin / Staff berhasil login dan sistem memvalidasi status login
	2	Admin / Staff mengarahkan ke halaman delivery schedule
	3	Admin / staff mengklik ikon "Tambah" pada halaman delivery schedule
	4	Sistem menampilkan halaman tambah delivery schedule
	5	Admin / staff mengisi formulir nomor kode barang, nama barang, berat barang, jumlah barang tanggal keluar, alamat customer, nama customer, pic.
	6	Admin / staff mengklik tombol "Submit"
	7	Sistem memvalidasi data delivery schedule
	8	Jika data barang valid, sistem mengurangi data barang di dalam <i>database</i> .
	9	Sistem mengarahkan ke halaman delivery schedule dan menampilkan pesan konfirmasi bahwa data barang berhasil dikurangi
Alur Alternatif	-	
Alur Pengecualian	Step	Action
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.22 Use Case Scenario Ubah Data Delivery Schedule

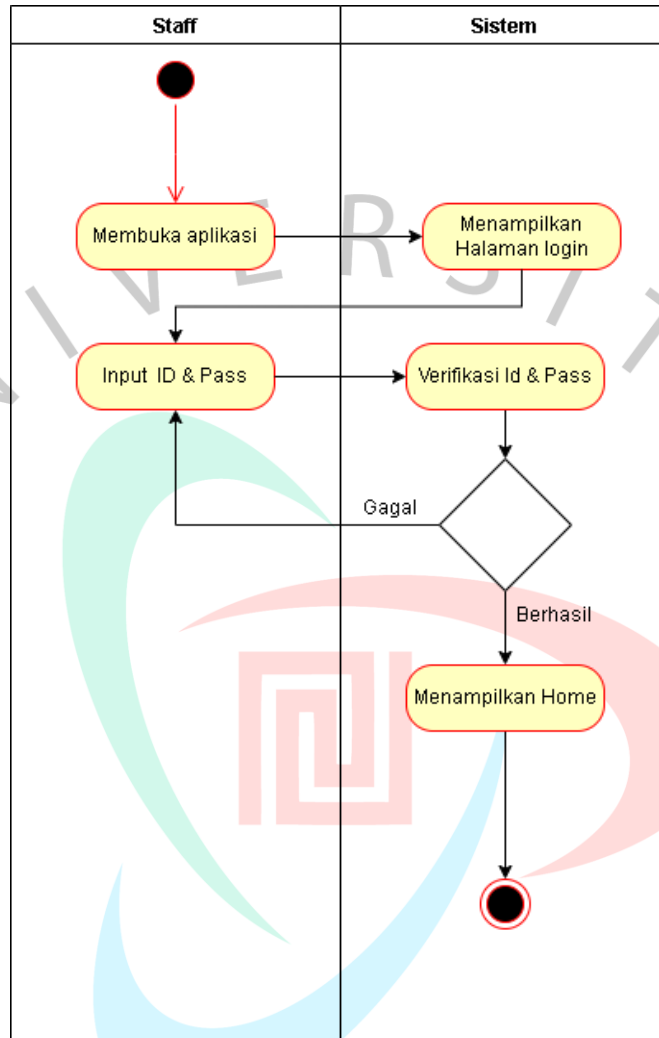
Use Case	Ubah Data Delivery Schedule
Deskripsi	Admin/Staff ingin mengubah delivery schedule

Aktor Utama	Admin, Staff	
Aktor Pendukung	- -	
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	
Alur Utama	<i>Step</i>	<i>Action</i>
	1	Admin / staff berhasil login dan sistem memvalidasi status login
	2	Admin / staff mengarahkan ke halaman delivery schedule
	3	Admin / staff mengeklik ikon "Edit" pada halaman delivery schedule
	4	Sistem menampilkan halaman ubah delivery schedule
	5	Admin / staff mengubah data formulir kode barang, nama barang, berat barang, jumlah barang tanggal keluar, alamat customer, nama customer, pic.
	6	Admin / staff mengeklik tombol "Update"
	7	Sistem memvalidasi data delivery schedule
	8	Jika data barang valid, sistem menyimpan data barang ke dalam <i>database</i> .
	9	Sistem mengarahkan ke halaman delivery schedule dan menampilkan pesan konfirmasi bahwa data delivery schedule berhasil diubah
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

Tabel 3.23 Use Case Scenario Hapus Data Delivery Schedule

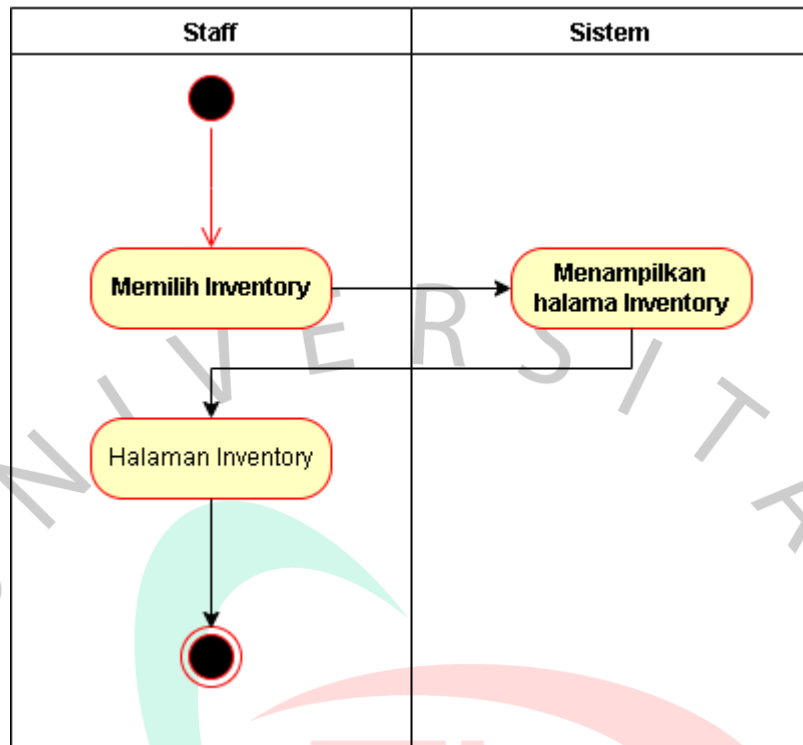
Use Case	Hapus Data Delivery Schedule	
Deskripsi	Admin/Staff ingin mengubah delivery schedule	
Aktor Utama	Admin, Staff	
Aktor Pendukung	-	
Pra-syarat	- Admin / Staff telah memiliki akun terdaftar di sistem	
Alur Utama	<i>Step</i>	<i>Action</i>
	1	Admin / Staff berhasil login dan sistem memvalidasi status login
	2	Admin / Staff mengarahkan ke halaman delivery schedule
	3	Sistem menampilkan halaman delivery schedule
	4	Admin mengklik ikon "Hapus" pada halaman delivery schedule
	5	Sistem menghapus data barang di <i>database</i>
Alur Alternatif	-	
Alur Pengecualian	<i>Step</i>	<i>Action</i>
	1	Jika informasi yang dimasukkan tidak valid, sistem menampilkan pesan kesalahan

3.2.5 Activity diagram



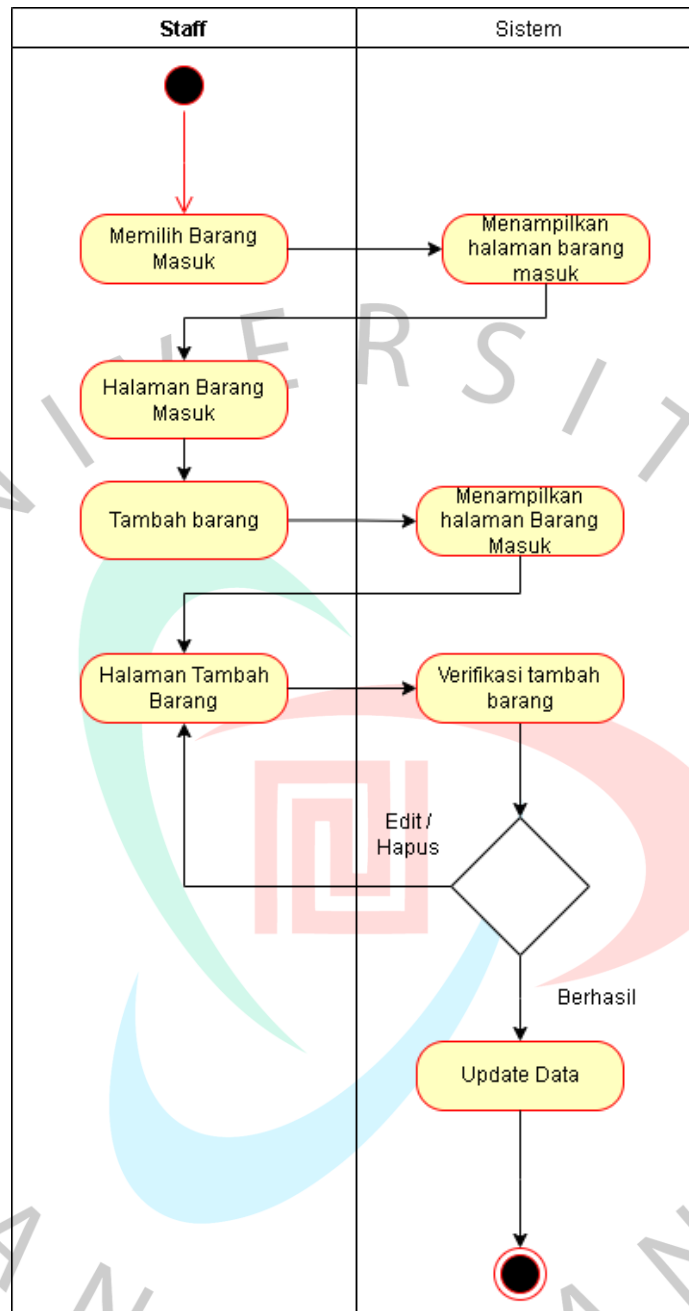
Gambar 3.1 Login Activity Diagram

Gambar 3.1 adalah menggambarkan login activity pada halaman ini dimana pengguna harus memasukan ID dan Password untuk memasuki aplikasi ini.



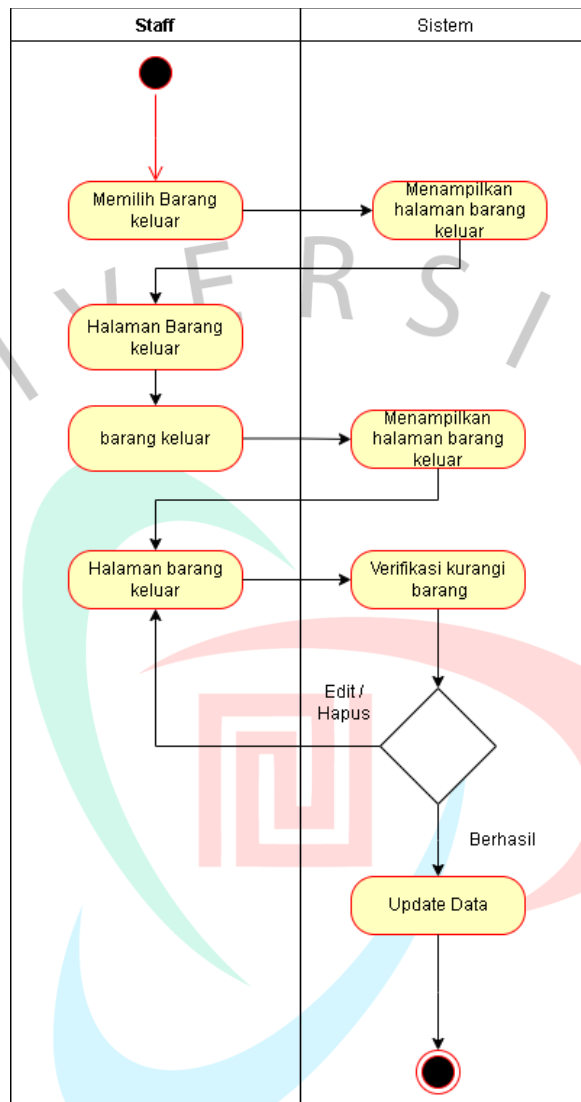
Gambar 3.2 Inventory Activity Diagram

Gambar 3.2 adalah menggambarkan inventory activity pada halaman ini semua staff dapat melihat update terbaru tentang stok barang yang ada.



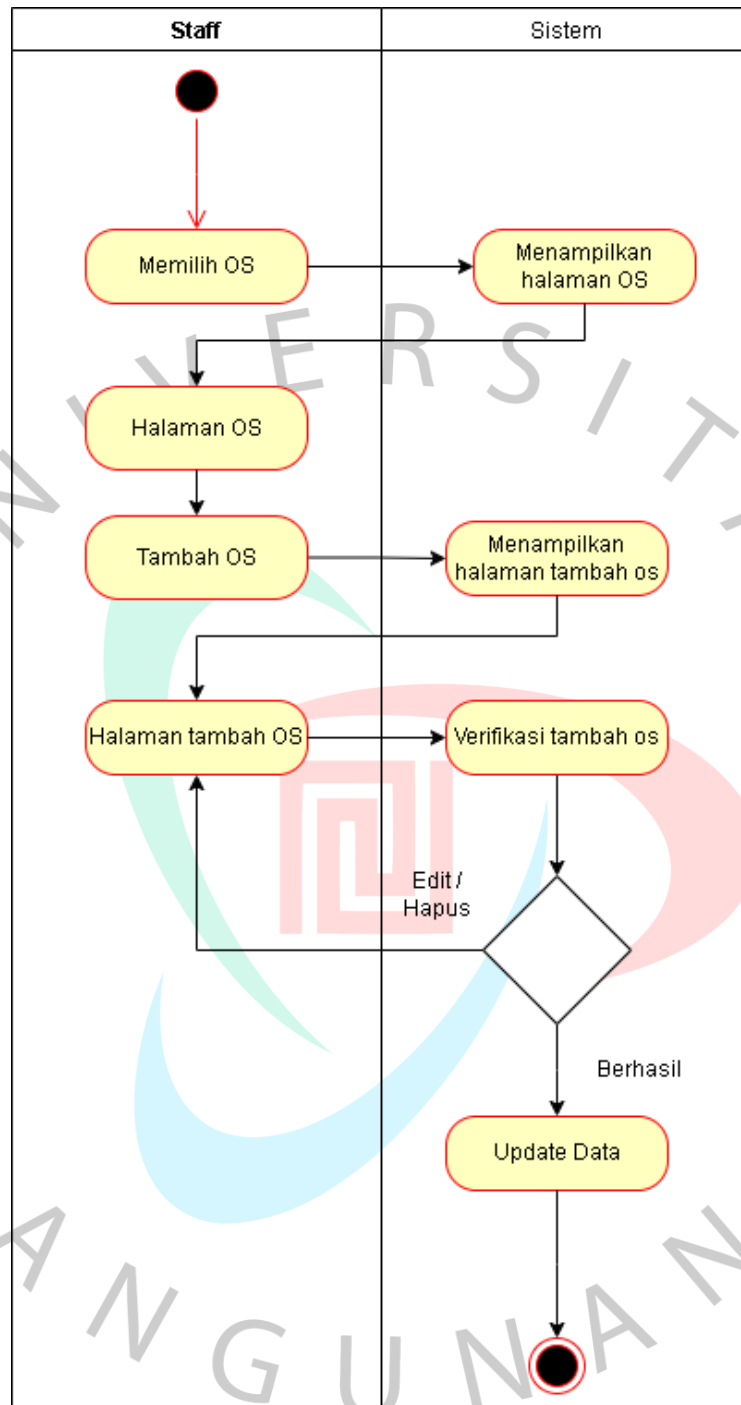
Gambar 3.3 Barang Masuk Activity Diagram

Gambar 3.3 adalah menggambarkan barang masuk activity pada halaman ini pengguna dapat menambahkan barang sesuai dengan barang yang datang mulai dari jumlah barang, nama barang, tanggal kedatangan.



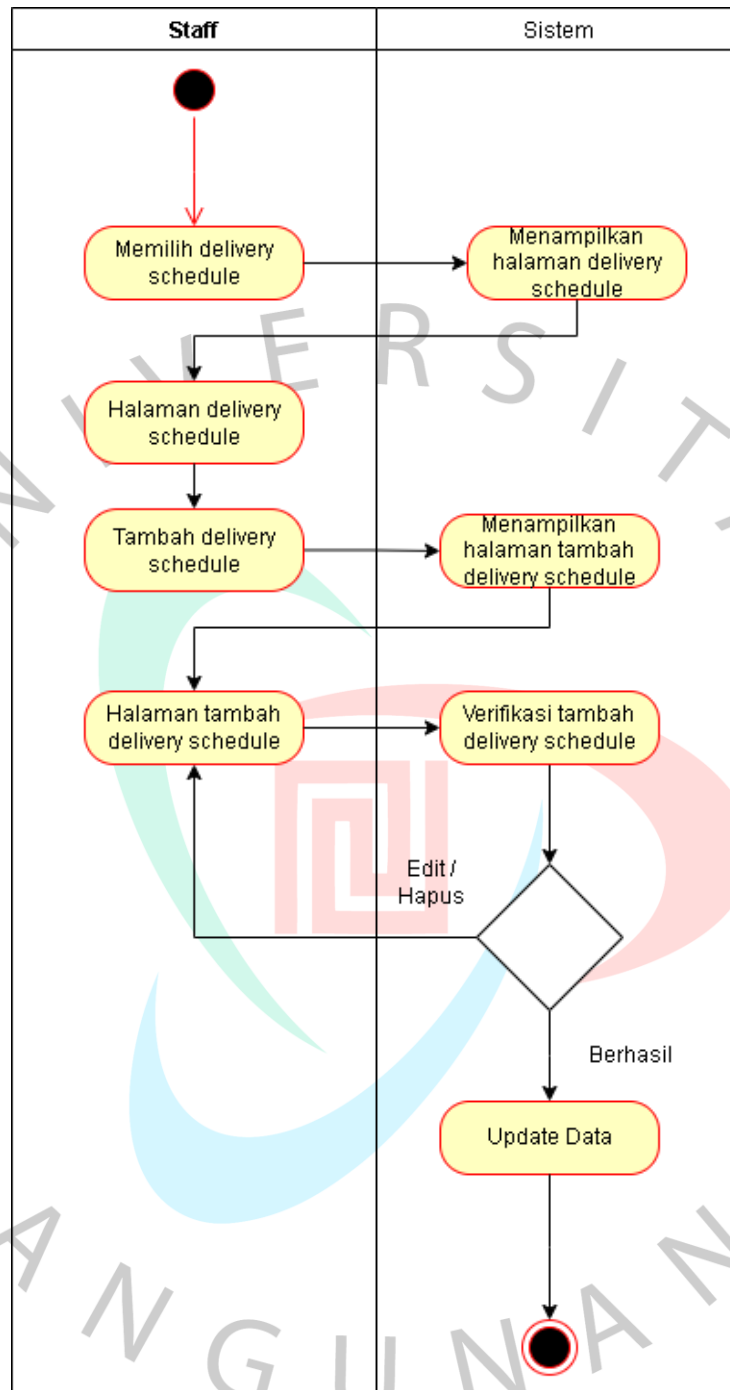
Gambar 3.4 Barang Keluar Activity Diagram

Gambar 3.4 adalah menggambarkan barang keluar activity pada halaman ini pengguna dapat mengurangi barang yang akan dikirim sesuai dengan jumlah customer order.



Gambar 3.5 OS Activity Diagram

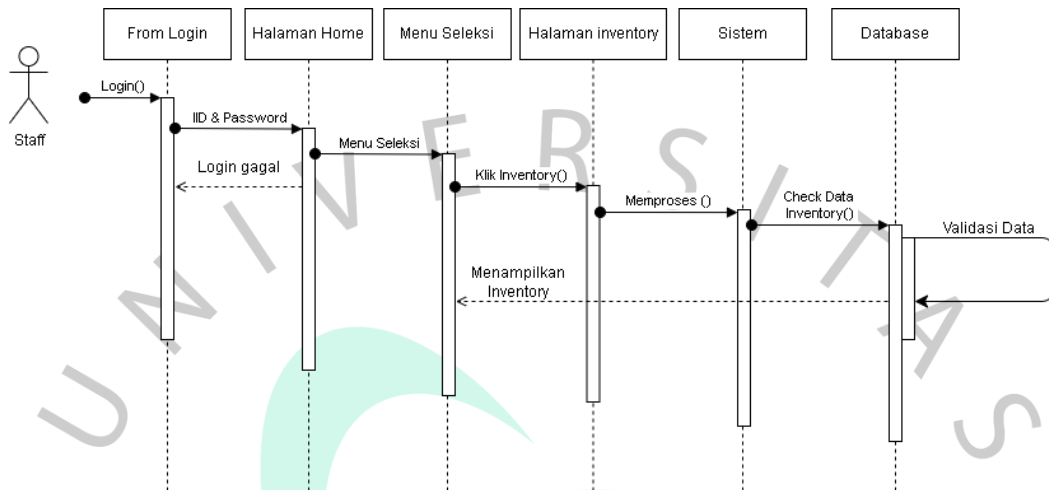
Gambar 3.5 adalah menggambarkan OS activity pada aplikasi yang di mana pada halaman ini pengguna dapat menambahkan dokumen yang menjelaskan orderan tersebut.



Gambar 3.6 Delivery Schedule Activity Diagram

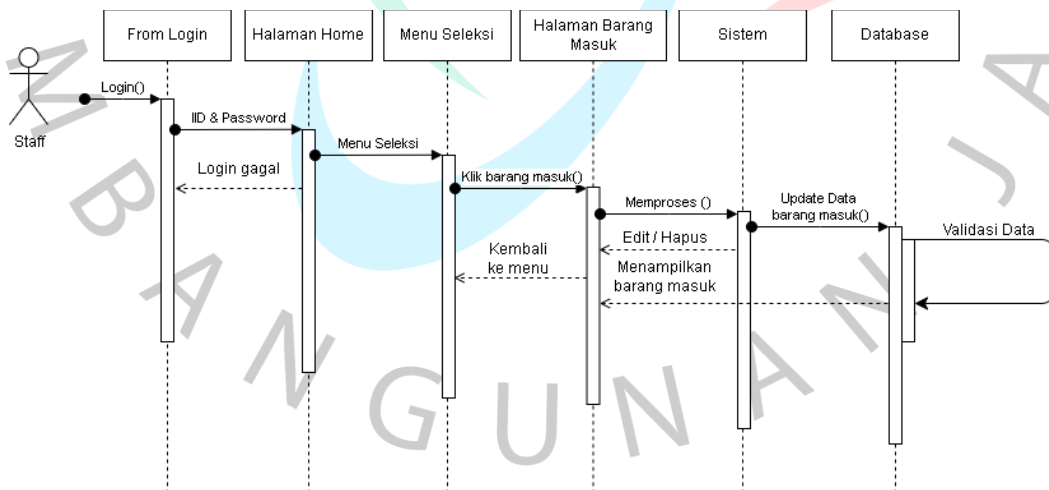
Gambar 3.6 adalah menggambarkan delivery schedule activity pada aplikasi yang di mana pada halaman ini pengguna dapat mengatur pengiriman sesuai tanggal yang di sepakati.

3.2.6 Sequence Diagram



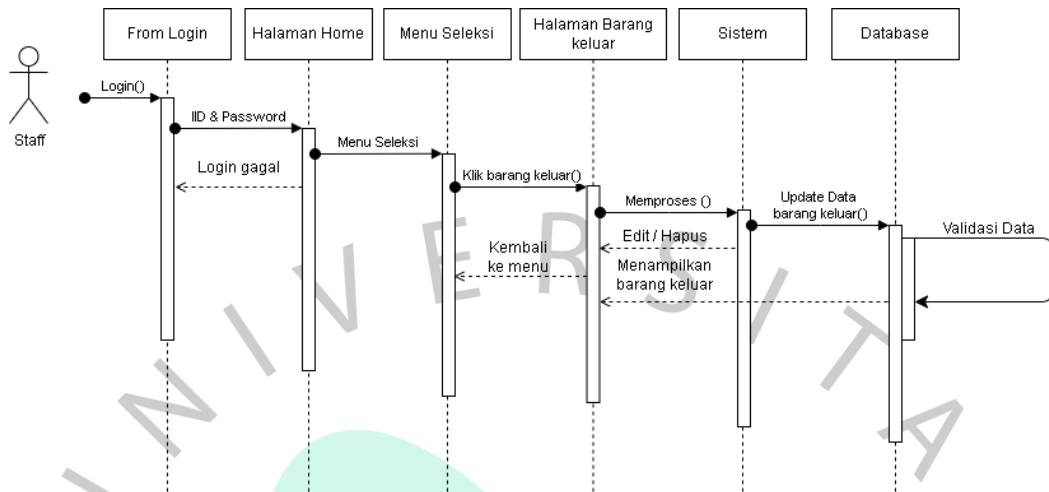
Gambar 3.7 Inventory Sequence Diagram

Gambar 3.7 adalah menggambarkan inventory sequence diagram pada halaman ini semua staff dapat melihat update terbaru tentang stok barang yang ada.



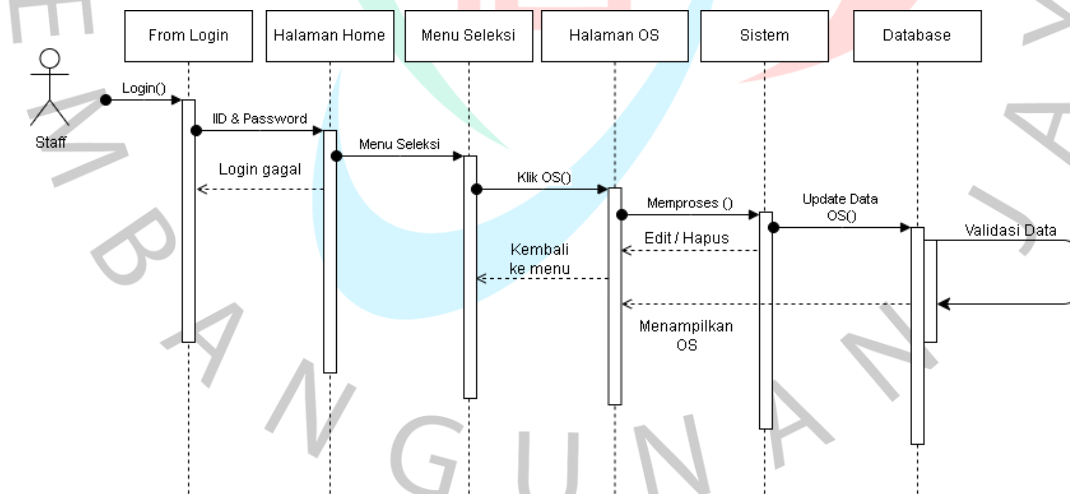
Gambar 3.8 Barang Masuk Sequence Diagram

Gambar 3.8 adalah menggambarkan barang masuk sequence diagram pada halaman ini pengguna dapat menambahkan barang sesuai dengan barang yang datang mulai dari jumlah barang, nama barang, tanggal kedatangan.



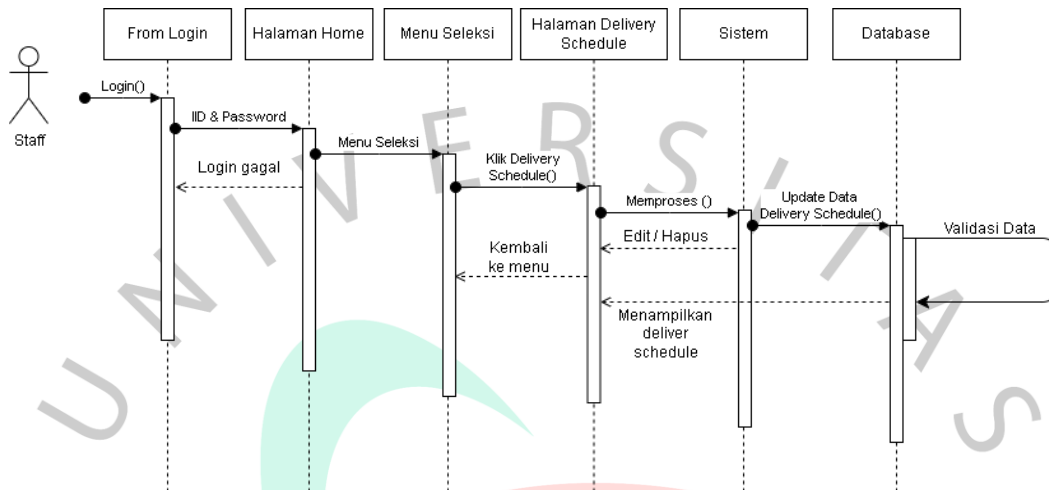
Gambar 3.9 Barang keluar Sequence Diagram

Gambar 3.9 adalah menggambarkan barang keluar sequence diagram pada halaman ini pengguna dapat mengurangi barang yang akan dikirim sesuai dengan jumlah customer order.



Gambar 3.10 OS Sequence Diagram

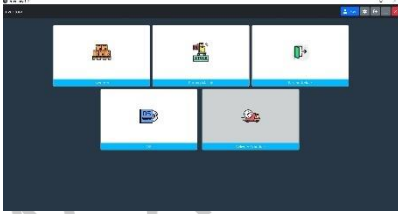
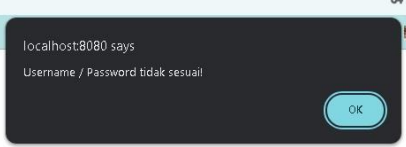
Gambar 3.10 adalah menggambarkan OS sequence diagram pada aplikasi yang di mana pada halaman ini pengguna dapat menambahkan dokumen yang menjelaskan orderan tersebut.


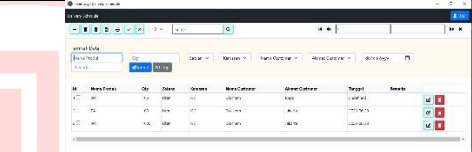
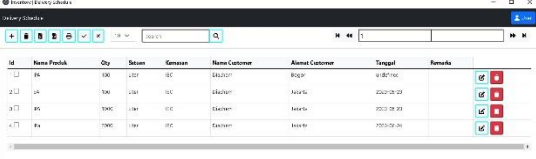


Gambar 3.11 Delivery Schedule Sequence Diagram

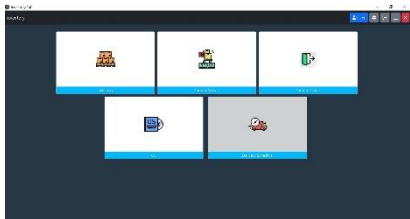
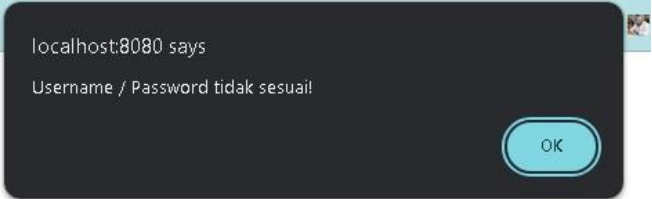
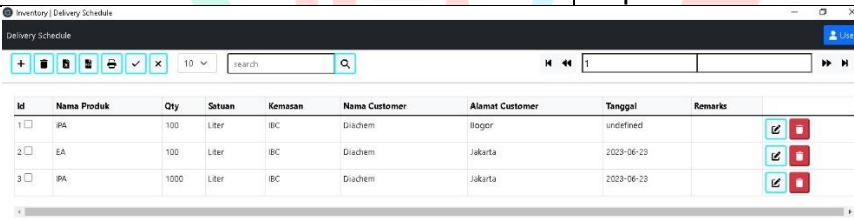
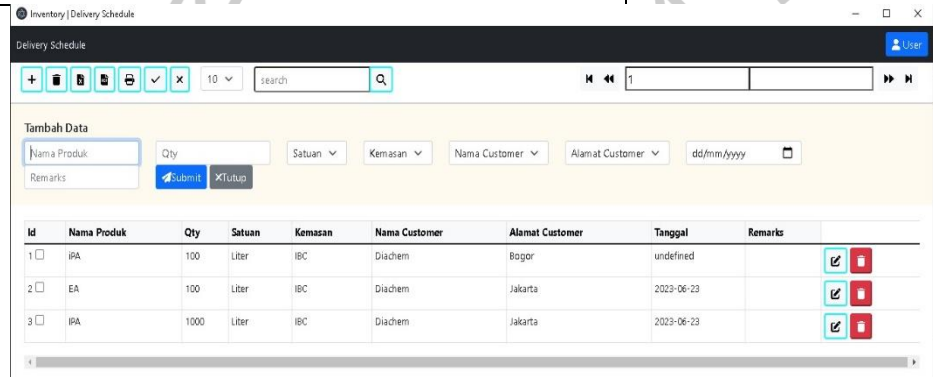
Gambar 3.11 adalah menggambarkan delivery schedule sequence diagram pada aplikasi yang di mana pada halaman ini pengguna dapat mengatur pengiriman sesuai tanggal yang di sepakati.

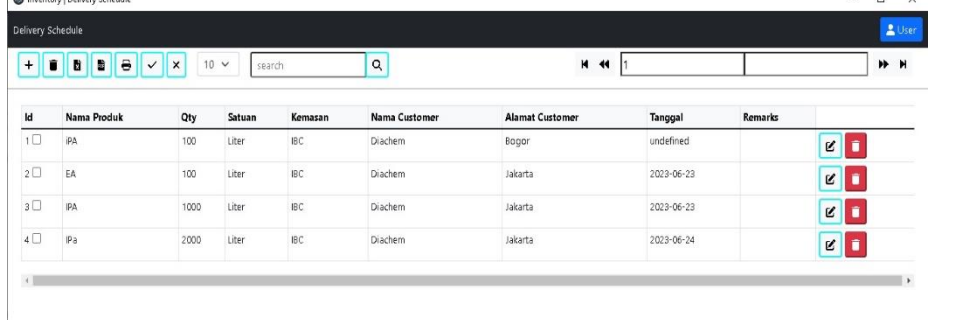
3.2.7 White Box

No.	Script	Hasil yang di tampilkan
1	<p>Login success</p> <pre> 1 login.php 2 3 if (\$SERVER['REQUEST_METHOD'] == "POST") { 4 \$username = \$_POST['username']; 5 \$password = \$_POST['password']; 6 if (\$query = mysqli_query(\$koneksi, "SELECT * FROM pengguna WHERE username='\$username' AND password='\$password'")) { 7 session_start(); 8 while (\$data = mysqli_fetch_assoc(\$query)) { 9 \$_SESSION['id_pengguna'] = \$data['id_pengguna']; 10 \$_SESSION['nama'] = \$data['nama']; 11 \$_SESSION['username'] = \$data['username']; 12 } 13 header("location: index.php"); 14 } else { 15 echo alert("Username / Password tidak sesuai!", "login.php"); 16 } 17 } else { 18 echo "Query error!"; 19 } 20 } </pre>	<p>Dapat masuk ke home</p> 
2	<p>Login Failed</p> <pre> header("location: index.php"); } else { echo alert("Username / Password tidak sesuai!", "login.php"); } } else { echo "Query error!"; } </pre>	<p>Terdapat Notifikasi error</p> 

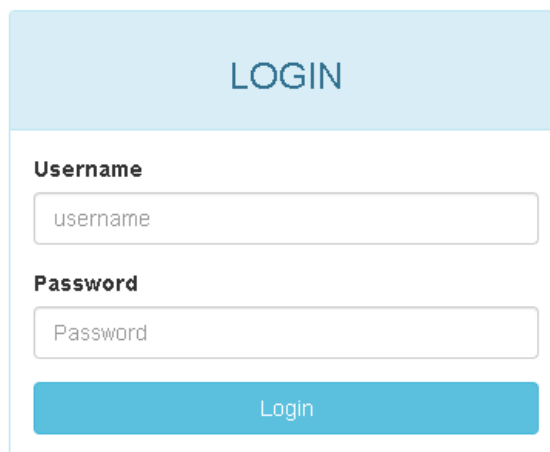
<p>3 Klik menu home</p> <pre> let mainWindow let productWindow mainwin = () => { mainWindow = new BrowserWindow({ webPreferences: { nodeIntegration: true, contextIsolation: false }, height: 550, resizable: false, title: 'TSR 1.0', autoHideMenuBar: true }) mainWindow.loadFile('index.html') db.serialize() => { console.log('we did it') } } </pre>	<p>masuk ke tab pilihan</p> 
<p>4 Menambahkan Item</p> <pre> // ... (previous code) ... // Add new item const addItem = () => { const formData = { nama: document.getElementById('nama').value, qty: document.getElementById('qty').value, satuan: document.getElementById('satuan').value, gamas: document.getElementById('gamas').value, namaCustomer: document.getElementById('namaCustomer').value, alamatCustomer: document.getElementById('alamatCustomer').value, tanggal: document.getElementById('tanggal').value, remarks: document.getElementById('remarks').value } db.serialize() => { db.run('INSERT INTO produk VALUES (?, ?, ?, ?, ?, ?, ?, ?)', [formData.nama, formData.qty, formData.satuan, formData.gamas, formData.namaCustomer, formData.alamatCustomer, formData.tanggal, formData.remarks], (err) => { if (err) console.log(err); else console.log('Item added successfully'); }) } } </pre>	<p>akan menampilkan tool bar tambah data</p> 
<p>5 Submit tambah data</p>	<p>Data yang sudah di submit akan masuk ke data base</p>
<pre> // ... (previous code) ... // Submit button const submit = () => { if (nama === '' qty === '' satuan === '' gamas === '' namaCustomer === '' alamatCustomer === '' tanggal === '' remarks === '') { alert('Please fill all fields'); } else { addItem(); } } </pre>	

3.2.8 Black Box

1	<p>Skenario Pengujian</p> <p>Melakukan login</p>	<p>Hasil</p> <p>Jika benar akan masuk ke tampilan home</p>
		
2	<p>Login gagal</p>	<p>Jika id dan password salah akan ada notifikasi “ id & pass tidak sesuai”</p>
		
3	<p>Masuk salah satu tab</p>	<p>Menampilkan tab yang di pilih</p>
		
4	<p>Menambah item</p>	<p>Klik tanda “ + “ pada toolbar dan akan muncul kolom tambah data</p>
		

5	Submit item	Ketika sudah sesuai itemnya klik menu “ submit “ maka data akan masuk ke data base
		

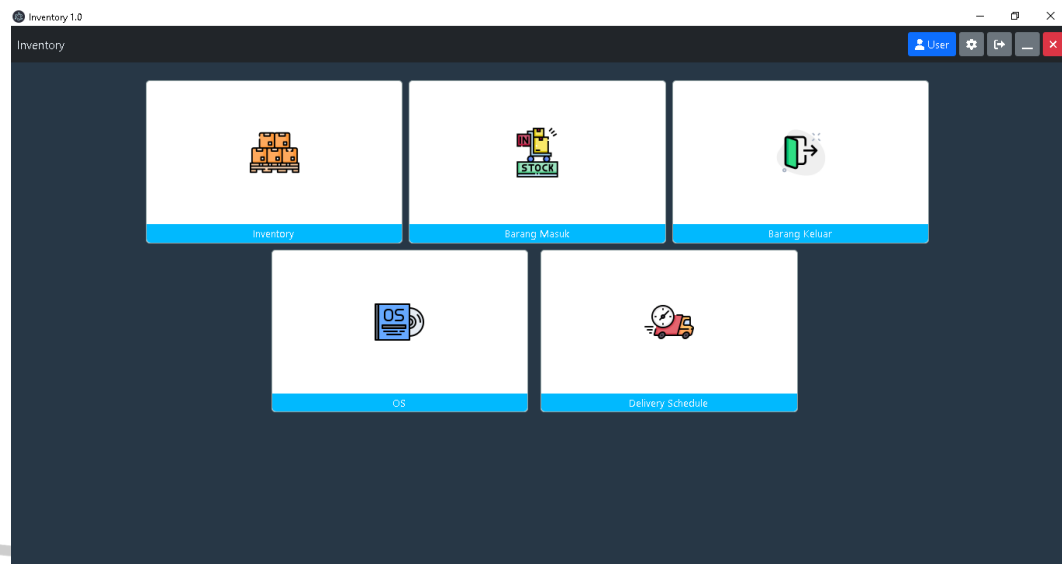
3.2.9 Hasil



Gambar 3.12 Halaman Login

Gambar 3.12 terlihat sistem yang menampilkan menu login untuk aplikasi inventory. Menu login ini merupakan bagian penting dalam sistem tersebut karena memungkinkan pengguna untuk mengakses dan menggunakan fitur-fitur aplikasi dengan aman dan terjamin. Dalam menu login, pengguna akan diminta untuk memasukkan informasi login yang benar, seperti email

dan password, untuk mengidentifikasi dan memverifikasi identitas mereka. Setelah data login diverifikasi, pengguna akan diberikan akses ke berbagai fitur aplikasi yang tersedia, seperti Inventory, Barang Masuk, Barang Keluar, OS, dan Delivery Schedule.



Gambar 3.14 Halaman Home

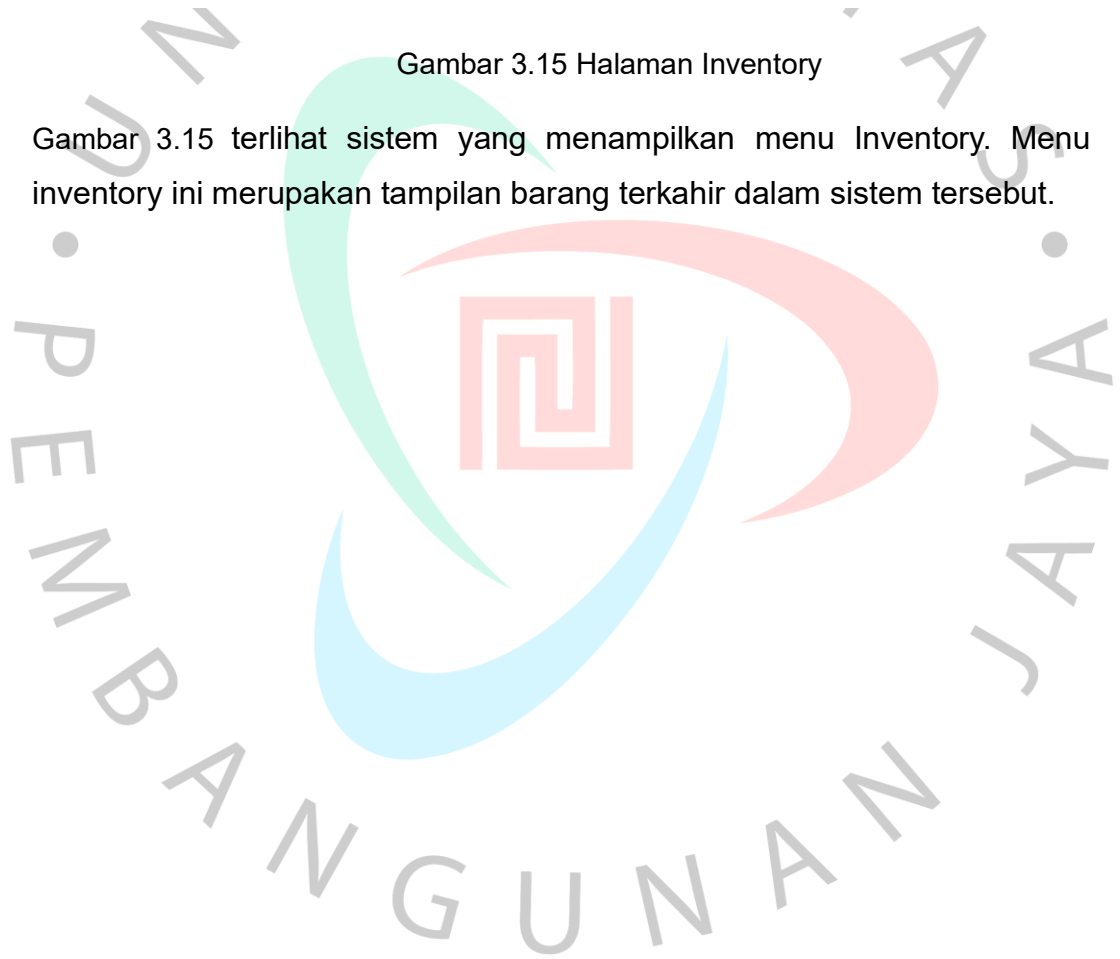
Gambar 3.14 terlihat sistem yang menampilkan menu Home untuk aplikasi inventory. Menu Home ini merupakan tampilan awal dalam sistem tersebut karena memungkinkan pengguna untuk mengakses dan menggunakan fitur-fitur aplikasi seperti Inventory, Barang Masuk, Barang Keluar, OS, dan Delivery Schedule.

The screenshot shows a web application window titled 'Inventory'. At the top, there is a navigation bar with a search input field containing the text 'search' and a magnifying glass icon. Below the navigation bar is a table with the following data:

Id	Nama Produk	Qty	Satuan	Kemasan
001	IPA	10000	Kg	Drum
002	TOLUEN	15000	Kg	Drum
003	HEXANE	20000	Kg	Bulk
004	A100	50000	Kg	Drum
005	A150	35000	Kg	Bulk

Gambar 3.15 Halaman Inventory

Gambar 3.15 terlihat sistem yang menampilkan menu Inventory. Menu inventory ini merupakan tampilan barang terkahir dalam sistem tersebut.



3.3 Tools dan Teknologi

Dalam praktik kerja IT dalam proyek pengembangan aplikasi inventory, ada beberapa alat yang dapat digunakan untuk membantu melaksanakan tugas dan tanggung jawab IT. Beberapa alat yang berguna untuk IT dalam pengujian perangkat lunak adalah:

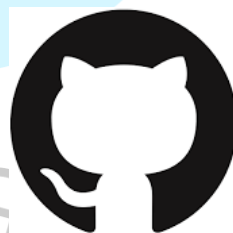
1) Visual Studio Code



Gambar 3.16 Visual Studio Code

Visual Studio Code adalah editor kode sumber lintas platform yang sangat populer yang dikembangkan oleh Microsoft. Visual Studio Code adalah alat yang sangat populer di kalangan pengembang perangkat lunak karena kekuatan, fleksibilitas, dan ketersediaannya yang lintas platform. Ini menyediakan lingkungan pengembangan yang kuat dan efisien untuk membuat dan mengedit kode sumber dengan nyaman.

2) GitHub



Gambar 3.17 GitHub

GitHub adalah platform hosting repositori git yang populer dan sangat digunakan dalam pengembangan perangkat lunak kolaboratif.

Ini menyediakan berbagai fitur dan alat yang memudahkan pengembang untuk mengelola proyek, berkolaborasi dengan tim, dan melacak perubahan dalam kode sumber.

3) MySQL



Gambar 3.18 MySQL

MySQL adalah salah satu RDBMS yang paling banyak digunakan di dunia, baik untuk aplikasi skala kecil maupun besar. Keandalannya, kinerja yang tinggi, dan fleksibilitas dalam mengelola data menjadikannya pilihan yang populer bagi pengembang dan organisasi dalam menyimpan, mengelola, dan mengakses data mereka.

4) Electron



Gambar 3.19 Electron

Electron adalah sebuah kerangka kerja pengembangan perangkat lunak yang open-source yang memungkinkan para pengembang untuk membuat aplikasi lintas platform menggunakan teknologi web, seperti HTML, CSS, dan JavaScript. Dikembangkan oleh GitHub, Electron awalnya diciptakan untuk membangun aplikasi desktop untuk platform GitHub.

5) Node.js



Gambar 3.20 Node JS

Node.js adalah *runtime* lingkungan pemrograman yang memungkinkan pengembang untuk menjalankan kode JavaScript di sisi server. Dengan model *event-driven* dan *non-blocking I/O*, Node.js dapat mengelola banyak permintaan dengan respons yang cepat.

3.4 Kendala Yang Dihadapi

Dalam proyek pengembangan aplikasi Inventory PT. Tirta Surya Raya, praktikan menghadapi beberapa kendala. Beberapa kendala dihadapi adalah sebagai berikut:

- 1) Jika terjadi perubahan persyaratan atau kebutuhan pengguna di tengah proyek, mungkin dihadapkan pada tantangan untuk menyesuaikan rencana pengujian yang telah dibuat sebelumnya. Hal ini dapat mempengaruhi jadwal dan efisiensi pengujian.
- 2) Memerlukan akses yang memadai terhadap data, termasuk data barang, dan data lainnya yang relevan. Namun, terkadang akses terhadap data ini terbatas atau terbatas oleh aturan privasi dan keamanan data.
- 3) Menghadapi situasi di mana hasil pengujian tidak sesuai dengan harapan atau ada kejanggalan yang sulit diidentifikasi. Dalam hal ini, diperlukan investigasi lebih lanjut dan upaya tambahan untuk menemukan dan memperbaiki masalah.

3.5 Cara Mengatasi Kendala

Berikut adalah beberapa cara untuk mengatasi kendala yang dihadapi oleh praktikan dalam proyek pengembangan *aplikasi inventory* PT. Tirta surya Raya:

- 1) Buat rencana pengujian yang komprehensif dengan mempertimbangkan persyaratan, skenario pengujian, dan kriteria. Rencana yang matang akan membantu menghindari perubahan mendadak dan mengatur pengujian dengan baik.
- 2) Jika terjadi perubahan persyaratan, komunikasikan secara efektif dengan tim pengembang dan cari solusi bersama. Lakukan penyesuaian yang diperlukan pada rencana pengujian dan prioritaskan pengujian berdasarkan perubahan tersebut.
- 3) Jalin komunikasi yang baik dengan tim pengembang dan pihak terkait lainnya. Diskusikan secara terbuka dan jelas mengenai persyaratan, masalah yang ditemukan, dan jadwal pengujian. Kolaborasi yang baik akan mempercepat resolusi masalah dan meminimalkan hambatan.

3.6 Pembelajaran Yang Diperoleh dari Kerja Profesi

Selama melaksanakan Kerja Profesi dalam proyek pengembangan *aplikasi inventory* PT. Tirta Surya Raya, berikut adalah beberapa pembelajaran yang dapat diperoleh:

- 1) Proses perencanaan yang matang sangat penting dalam pengujian perangkat lunak. Pembelajaran ini mengajarkan untuk secara teliti merencanakan pengujian, menyusun skenario pengujian, dan menetapkan prioritas berdasarkan risiko dan kepentingan. Dengan perencanaan yang matang dapat mengoptimalkan penggunaan sumber daya dan memastikan cakupan pengujian yang memadai.
- 2) Dalam pengembangan perangkat lunak, perubahan persyaratan atau kebutuhan pengguna sering terjadi. Pembelajaran ini mengajarkan untuk

menjadi fleksibel dan mampu menyesuaikan rencana pengujian dengan cepat ketika terjadi perubahan.

- 3) Hal ini membutuhkan komunikasi yang baik dengan tim pengembang dan kemampuan untuk mengidentifikasi dan mengelola perubahan dengan efektif.
- 4) Kolaborasi yang erat dengan tim pengembang, manajemen proyek, dan pemangku kepentingan lainnya adalah kunci kesuksesan dalam pengujian perangkat lunak. Pembelajaran ini mengajarkan untuk berkomunikasi secara efektif, berbagi informasi, dan bekerja sama dalam menyelesaikan masalah yang terkait dengan kualitas dan keandalan produk.
- 5) Pentingnya komunikasi dan kolaborasi yang efektif sangat penting dalam lingkungan bisnis dan pengembangan tim. Mereka meningkatkan efisiensi, mengoptimalkan pemanfaatan sumber daya, meningkatkan kualitas pekerjaan, mendorong inovasi, meningkatkan kepuasan dan keterlibatan tim, serta mengatasi hambatan dan konflik. Dengan memperkuat komunikasi dan kolaborasi, tim dapat mencapai kinerja yang lebih baik dan mencapai tujuan bersama dengan lebih efektif