

BAB IV

PERANCANGAN

Dalam bab ini, peneliti akan menjelaskan secara detail bagaimana kebutuhan spesifikasi sistem, cara kerja sistem, rancangan antarmuka dan rancangan pengujian sistem yang akan dibuat.

4.1 Analisis Sistem Terdahulu

Analisis sistem terdahulu bertujuan untuk memahami sistem, mengidentifikasi kelebihan dan kekurangan dari sistem terdahulu. Pada PT Maggot Indonesia Lestari belum memiliki sistem prediksi harga melalui website, PT maggot Indonesia Lestari masih menggunakan sistem manual, yang dimana pada sistem manual ini memiliki kekurangan, seperti memerlukan waktu yang lebih lama dalam mendapatkan informasi tentang harga telur untuk waktu yang akan datang.

4.2 Spesifikasi Kebutuhan Sistem Baru

Analisis sistem terdahulu dilakukan untuk menentukan arah perancangan sistem baru yang dapat menjadi solusi permasalahan dari sistem sebelumnya. Dari analisis sistem terdahulu, dapat dijadikan dasar untuk merancang kebutuhan piranti lunak serta perangkat keras untuk merancang aplikasi tersebut. Adapun beberapa spesifikasi kebutuhan sistem baru untuk sistem prediksi harga penjualan telur tersebut.

4.2.1 Fitur – Fitur

Analisis sistem terdahulu sudah dilakukan untuk menentukan permasalahan atau *bottleneck* yang terjadi pada proses sebelumnya sehingga perancangan sistem baru dapat dijalankan berdasarkan dari analisis sistem terdahulu. Dari analisis sistem terdahulu juga penulis dapat melihat solusi atas beberapa permasalahan yang ada dengan menuangkan permasalahan tersebut kedalam beberapa fitur yang akan dibuat pada sistem yang akan dikembangkan. Beberapa fitur yang akan dikembangkan antara lain sebagai berikut:

Tabel 4. 1 Fitur-fitur

No	Fitur	Keterangan	Fungsi
1	Administrator	Kelola User	<ul style="list-style-type: none"> • Admin dapat menambahkan <i>User</i> admin • Admin dapat menambahkan <i>User Sales</i> • Admin dapat menambahkan <i>User Warehouse</i>
		Data Prediksi	<ul style="list-style-type: none"> • Admin melihat prediksi harga telur selanjutnya
		Kelola penjualan	<ul style="list-style-type: none"> • Admin dapat meng-update atribut data penjualan • Admin dapat memperbaiki data komoditi • Admin dapat memasukan data penjualan • Admin dapat menghapus data penjualan
2	<i>Sales</i>	Kelola stok	<ul style="list-style-type: none"> • Admin dapat memperbaiki data stok • Admin dapat menambahkan data stok • Admin dapat menghapus data stok
		Kelola penjualan	<ul style="list-style-type: none"> • <i>Sales</i> dapat menginput data penjualan • <i>Sales</i> dapat memperbaiki data penjualan • <i>Sales</i> dapat menghapus data penjualan • <i>Sales</i> dapat melakukan penambahan komoditi penjualan
3	<i>Warehouse</i>	Kelola stok	<ul style="list-style-type: none"> • Bagian <i>warehouse</i> dapat menambahkan data stok • Bagian <i>warehouse</i> dapat melakukan pengecekan stok • Bagian <i>warehouse</i> dapat memperbaiki data stok

4.2.2 Spesifikasi Kebutuhan Input

Aplikasi prediksi harga telur pada penelitian ini, tentu diperlukan proses input agar aplikasi dapat berjalan dengan baik. Proses input yang dibutuhkan pada aplikasi ini sebagai berikut.

Input Harga

1. Input stok telur

Untuk menggunakan aplikasi ini, pengguna harus melakukan input stok telur.

4.2.3 Spesifikasi Kebutuhan Output

Hasil dari aplikasi prediksi harga telur berfokus hasil prediksi yang di dapat berdasarkan harga telur sebelumnya. Berikut merupakan spesifikasi output aplikasi.

1. Prediksi harga telur dimasa yang akan datang berdasarkan stok telur yang akan datang.

4.2.4 Spesifikasi Perangkat Lunak

Dalam pembangunan sistem ini digunakan perangkat lunak, dan perangkat kerasnya Spesifikasi kebutuhan perangkat lunak merupakan hal yang penting dalam mempersiapkan penelitian. Peneliti akan menjabarkan kebutuhan, mulai dari perangkat sampai aplikasi editor. Berikut merupakan spesifikasi kebutuhan perangkat lunak.

Tabel 4.2 Kebutuhan Perangkat Lunak

No	Kebutuhan software
1	Sistem Operasi Windows 10 (minimum)
2	Visual Studio Code (untuk coding)
3	Xampp (versi terbaru)
4	Web Browser
5	Internet

4.2.5 Spesifikasi Kebutuhan Perangkat Keras

Spesifikasi kebutuhan perangkat keras yang digunakan pada penelitian ini mencakup prosesor, harddisk dan memory. Terlihat spesifikasi kebutuhan perangkat keras tertera pada table berikut ini.

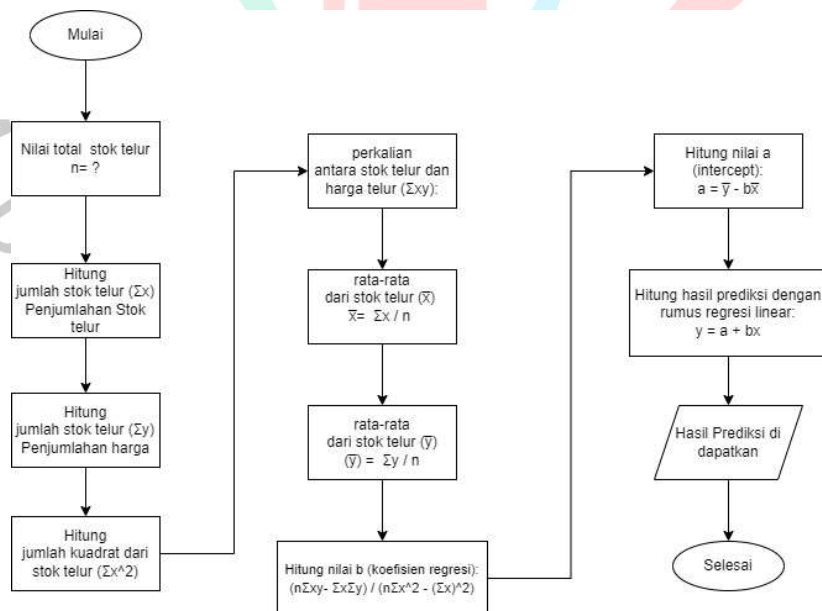
Tabel 4.3 Kebutuhan Perangkat Keras
Kebutuhan Hardware

Hardware	Spesifikasi
Processor	Intel Core I5 (4-Cores)
Ram	8 Gb
Storage	SSD 512Gb
Display	14 inc Display HD Resolution

4.3 Perancangan Sistem

Proses perancangan sistem prediksi penjualan telur ini menggunakan UML (Unified Modeling Language), yang terdiri dari lima bagian: *flowchart*, *Use Case Diagram*, *Use Case Narrative*, *Activity Diagram*, *Class Diagram*, dan *Sequence Diagram*.

4.3.1. Flowchart Aplikasi Prediksi Harga



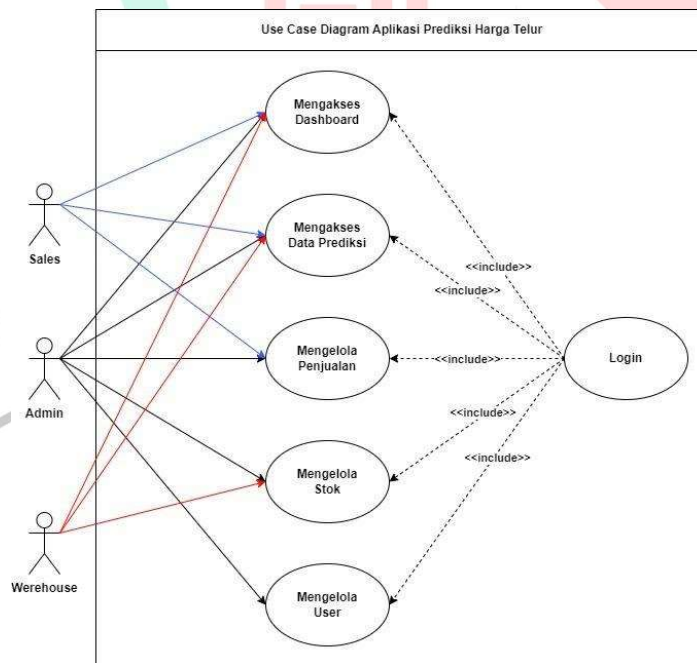
Gambar 4. 1 Flowchart Aplikasi Prediksi Harga

Pada gambar 4.1 merupakan gambaran tahapan proses algoritma regresi linear berjalan, pada tahap awal sistem akan menentukan nilai total stok telur dengan

symbol n , setelah itu menghitung jumlah stok telur dengan rumus $(\sum x)$, kemudian menghitung penjumlahan harga $\sum y$, lanjut pada menghitung jumlah kuadrat dari stok telur dengan rumus $(\sum x^2)$, lanjut pada tahap perkalian antara stok telur dan harga telur dengan rumus $(\sum xy)$, selanjutnya menghitung rata-rata dari stok telur dengan rumus $\bar{x} = \sum x / n$, lalu menghitung nilai b (koefisien regresi) dengan rumus $b = (n\sum xy - \sum x \sum y) / (n\sum x^2 - (\sum x)^2)$, lalu menghitung nilai a (intercept) dengan rumus $a = \bar{y} - b\bar{x}$, terakhir memasukan rumus $y = a + bX$, dengan memasukan nilai-nilai yang sudah di dapat sebelumnya, sehingga bisa mendapatkan prediksi harga

4.3.2 Use Case Diagram

Diagram Use Case menggambarkan interaksi antara aktor dan sistem dalam sistem prediksi penjualan telur. Aktor menggambarkan pengguna atau entitas yang terlibat dalam penggunaan sistem, sementara use case menggambarkan fungsionalitas yang akan diimplementasikan dalam perangkat lunak tersebut. Berikut adalah contoh use case diagram untuk sistem prediksi penjualan telur.



Gambar 4.2 Use case diagram prediksi penjualan telur

Dari gambar 4.2, dapat diketahui beberapa actor yang terlibat dalam sistem sehingga dapat mempengaruhi sistem. Actor tersebut berupa admin yang memiliki

fungsi penuh dalam sistem, bagian sales yang akan melakukan kontrol pada fitur sales dan bagian warehouse yang melakukan kontrol pada stok.

4.3.1 Use Case Narrative

Use case Narrative digunakan untuk melihat dokumentasi secara rinci alur dan skenario penggunaan dari setiap use case dalam sistem prediksi penjualan telur.

Berikut adalah Use case Narrative dari sistem yang sedang dikembangkan:

A. Admin

1. Use Case Narrative Login

Tabel 4. 4 Use Case Narrative login

<i>Use case name</i>	<i>Login</i>	
<i>Actor</i>	<i>Admin</i>	
<i>Description</i>	Use case ini menggambarkan langkah-langkah yang harus dilakukan oleh seorang admin untuk melakukan login ke dalam sistem.	
<i>Precondition</i>	Halaman masih menunjukkan halaman beranda	
<i>Typical Course of Event</i>	<i>Action</i>	<i>System Response</i>
	<ol style="list-style-type: none"> 1. User admin melakukan input username dan password 2. User admin melakukan klik login 	<ol style="list-style-type: none"> 1.1 Sistem memeriksa data username dan password 2.1 Setelah data username dan password cocok dengan data yang ada dalam database, sistem akan menampilkan halaman beranda yang berisi informasi dan fitur-fitur yang terkait dengan status login admin yang berhasil.
<i>Alternate Course</i>	Jika username dan password yang dimasukkan tidak cocok dengan data yang ada dalam database, sistem akan menampilkan pesan error.	
<i>Post Condition</i>	User admin sukses melakukan login	

2. Use Case Narrative Dashboard Admin

Tabel 4. 5 Use Case Narrative Dashboard

<i>Use case name</i>	<i>Dashboard Admin</i>	
<i>Actor</i>	<i>Admin</i>	
<i>Description</i>	Use case ini menjelaskan tentang bagaimana admin mengakses dashboard admin	
<i>Precondition</i>	Halaman masih menunjukkan halaman beranda	
<i>Typical Course of Event</i>	<i>Action</i>	<i>System Response</i>
	1. <i>User admin</i> melakukan input <i>username</i> dan <i>password</i> 2. <i>User admin</i> melakukan klik login	1.1 Sistem memeriksa data <i>username</i> dan <i>password</i> 2.1 Apabila data sesuai maka sistem akan menampilkan halaman <i>dashboard admin</i> dengan tampilan jumlah penjualan dan barang keluar.
<i>Alternate Course</i>	Apabila kombinasi <i>username</i> dan <i>password</i> yang diinput tidak cocok dengan data yang terdapat dalam database, sistem akan menampilkan pesan error.	
<i>Post Condition</i>	<i>User admin</i> sukses melakukan akses admin dashboard	

3. Use Case Narrative Kelola User

Tabel 4. 6 Use Case Narrative kelola user

<i>Use case name</i>	<i>Kelola User</i>	
<i>Actor</i>	<i>Admin</i>	
<i>Description</i>	Use case ini menjelaskan tentang bagaimana admin melakukan Kelola user	
<i>Precondition</i>	Data user yang tertera masih dengan menggunakan data user lama	
	<i>Action</i>	<i>System Response</i>

<i>Typical Course of Event</i>	<ol style="list-style-type: none"> 1. <i>User admin</i> melakukan input <i>username</i> dan <i>password</i> 2. <i>User admin</i> melakukan klik login 3. <i>User admin</i> melakukan klik pada <i>Kelola user</i> 4. <i>User admin</i> mengubah data <i>user</i> dan mengklik tombol simpan 	<ol style="list-style-type: none"> 1.1 Sistem memeriksa data <i>username</i> dan <i>password</i> 2.1 Apabila data sesuai maka sistem akan menampilkan halaman <i>dashboard admin</i> 3.1 Sistem menampilkan halaman <i>Kelola user</i> 4.1 Data diperbarui dan disimpan kedalam sistem
<i>Alternate Course</i>	4. Jika <i>User admin</i> melakukan klik tombol batal maka data tidak tersimpan dan halaman dikembalikan ke <i>Kelola user</i>	
<i>Post Condition</i>	<i>User admin</i> sukses melakukan perubahan data <i>user</i>	

4. Use Case Narrative Data Prediksi

Tabel 4. 7 Usecase Narrative Data Prediksi

<i>Use case name</i>	<i>Data Prediksi</i>	
<i>Actor</i>	<i>Admin</i>	
<i>Description</i>	Admin melakukan data prediksi pada sistem dengan mengakses fitur prediksi, memasukkan data, memilih metode prediksi, dan memperoleh hasil prediksi penjualan telur dalam bentuk laporan atau grafik.	
<i>Precondition</i>	Data prediksi belum ter-update atau bahkan belum ada sama sekali	
<i>Typical Course of Event</i>	<i>Action</i>	<i>System Response</i>
	<ol style="list-style-type: none"> 1. <i>User admin</i> melakukan login 2. <i>User admin</i> melakukan klik pada penjualan 3. <i>User admin</i> melakukan klik penjualan 4. <i>User admin</i> melakukan input data sales dan melakukan klik simpan 5. <i>User admin</i> melakukan klik perbarui data dan klik simpan 	<ol style="list-style-type: none"> 1.1 Sistem menampilkan halaman dashboard admin 2.1 Sistem menampilkan halaman <i>Kelola sales</i> 3.1 Sistem menampilkan halaman tambah sales 4.1 Data sales ditambahkan kedalam sistem. 5.1 Data sales sukses diperbarui dari sistem

	6. User admin melakukan klik hapus data dan klik ok 7. Data prediksi secara otomatis muncul pada halaman data prediksi	6.1 Data sales sukses dihapus dari sistem 7.1 Sistem menampilkan data prediksi terupdate
<i>Alternate Course</i>	4. Jika <i>User admin</i> melakukan klik tombol batal maka data tidak tersimpan dan halaman dikembalikan ke halaman penjualan	
<i>Post Condition</i>	<i>User admin</i> sukses melakukan prediksi data	

5. Use Case Narrative Kelola Sales

Tabel 4. 8 Use Case Narrative kelola sales

<i>Use case name</i>	<i>Kelola Sales</i>	
<i>Actor</i>	<i>Admin</i>	
<i>Description</i>	Use case ini menjelaskan tentang bagaimana admin melakukan Kelola Sales	
<i>Precondition</i>	Data sales masih menunjukkan data sales yang lama sehingga akan mempengaruhi data prediksi	
<i>Typical Course of Event</i>	<i>Action</i>	<i>System Response</i>
	1. <i>User admin</i> melakukan login	1.2 Sistem menampilkan halaman dashboard admin
	2. <i>User admin</i> melakukan klik pada Kelola Sales	2.2 Sistem menampilkan halaman Kelola sales
	3. <i>User admin</i> melakukan klik tambah sales	3.2 Sistem menampilkan halaman tambah sales
	4. <i>User admin</i> melakukan input data sales dan melakukan klik simpan	4.2 Data sales ditambahkan kedalam sistem. 5.2 Data sales sukses diperbarui dari sistem
	5. <i>User admin</i> melakukan klik perbarui data dan klik simpan	
	6. <i>User admin</i> melakukan klik hapus data dan klik ok	6.2 Data sales sukses dihapus dari sistem

<i>Alternate Course</i>	5. Jika <i>User admin</i> melakukan klik tombol batal maka data tidak tersimpan dan halaman dikembalikan ke Kelola sales
<i>Post Condition</i>	<i>User admin</i> sukses melakukan perubahan data sales

6. Use Case Narrative Kelola Stok

Tabel 4. 9 Use Case Narrative kelola stok Use Case Narrative kelola stok

<i>Use case name</i>	<i>Kelola Stok</i>	
<i>Actor</i>	<i>Admin</i>	
<i>Description</i>	<i>Use case</i> ini menjelaskan tentang bagaimana admin melakukan <i>Kelola Stok</i>	
<i>Precondition</i>	Data sales masih menunjukkan data stok yang lama dan belum <i>ter-update</i> sehingga belum mengetahui stok saat ini	
<i>Typical Course of Event</i>	<i>Action</i>	<i>System Response</i>
	1. <i>User admin</i> melakukan login	1.1 Sistem menampilkan halaman dashboard admin
	2. <i>User admin</i> melakukan klik pada Kelola Stok	2.1 Sistem menampilkan halaman Kelola Stok
	3. <i>User admin</i> melakukan klik tambah Stok	3.1 Sistem menampilkan form tambah stok
	4. <i>User admin</i> melakukan input data stok dan melakukan klik simpan	4.1 Data stok ditambahkan kedalam sistem.
	5. <i>User admin</i> melakukan klik perbarui data dan klik simpan	5.1 Data stok sukses diperbarui dari sistem
	6. <i>User admin</i> melakukan input stok keluar	6.1 Data stok keluar sukses disimpan dan stok diperbarui
<i>Alternate Course</i>	4. Jika <i>User admin</i> melakukan klik tombol batal maka data tidak tersimpan dan halaman dikembalikan ke Kelola Stok	
<i>Post Condition</i>	<i>User admin</i> sukses melakukan perubahan data stok	

7. Use Case Narrative Profile

Tabel 4. 10 Use Case Narrative profile

<i>Use case name</i>	<i>Profile</i>	
<i>Actor</i>	<i>Admin</i>	
<i>Description</i>	Use case ini menjelaskan tentang bagaimana admin melakukan perubahan data profile	
<i>Precondition</i>	Data profile user masih menggunakan data yang lama	
<i>Typical Course of Event</i>	<i>Action</i>	<i>System Response</i>
	1. <i>User admin</i> melakukan login 2. <i>User admin</i> melakukan klik pada Profile 3. <i>User admin</i> input password baru 4. <i>User admin</i> melakukan klik simpan	1.1 Sistem menampilkan halaman dashboard admin 2.1 Sistem menampilkan Profile user 3.1 Sistem menunjukan <i>field</i> yang terisi 4.1 Data <i>password</i> berhasil diubah
<i>Alternate Course</i>	4. Jika <i>User admin</i> melakukan klik tombol batal maka data tidak tersimpan dan halaman dikembalikan profile	
<i>Post Condition</i>	<i>User admin</i> sukses melakukan perubahan data profile	

B. Sales

1. Use Case Narrative Sales Login

Tabel 4. 11 Use Case Narrative Sales Login

<i>Use case name</i>	<i>Login</i>	
<i>Actor</i>	<i>Sales</i>	
<i>Description</i>	User Sales dapat melakukan login ke sistem dengan memasukkan username dan password yang telah terdaftar, sehingga dapat mengakses fitur dan informasi.	
<i>Precondition</i>	Halaman masih menunjukan halaman beranda	
<i>Typical Course of Event</i>	<i>Action</i>	<i>System Response</i>
	1. <i>User Sales</i> melakukan input username dan password	1.1 Sistem memeriksa data <i>username</i> dan <i>password</i> 2.1 Apabila data sesuai maka sistem akan

	2. <i>User Sales</i> melakukan klik login	menampilkan halaman beranda dengan status login yang sudah ada.
<i>Alternate Course</i>	Apabila username dan password yang dimasukkan oleh <i>User Sales</i> tidak cocok dengan data yang ada di database, sistem akan memberikan pesan error.	
<i>Post Condition</i>	<i>User Sales</i> sukses melakukan login	

2. Use Case Narrative Sales Kelola Sales

Tabel 4. 12 Use Case Narrative Sales Kelola Sales

<i>Use case name</i>	<i>Kelola Sales</i>	
<i>Actor</i>	<i>Sales</i>	
<i>Description</i>	Use case ini menjelaskan tentang bagaimana <i>user sales</i> melakukan <i>Kelola Sales</i>	
<i>Precondition</i>	Data <i>sales</i> masih menunjukkan data <i>sales</i> yang lama sehingga akan mempengaruhi data prediksi	
<i>Typical Course of Event</i>	<i>Action</i>	<i>System Response</i>
	1. <i>User Sales</i> melakukan login	1.1 Sistem menampilkan halaman dashboard admin
	2. <i>User Sales</i> melakukan klik pada <i>Kelola Sales</i>	2.1 Sistem menampilkan halaman <i>Kelola sales</i>
	3. <i>User sales</i> melakukan klik tambah sales	3.1 Sistem menampilkan halaman tambah sales
	4. <i>User sales</i> melakukan input data sales dan melakukan klik simpan	4.1 Data sales ditambahkan kedalam sistem.
	5. <i>User sales</i> melakukan klik perbarui data dan klik simpan	5.1 Data sales sukses diperbarui dari sistem
	6. <i>User sales</i> melakukan klik hapus data dan klik ok	6.1 Data sales sukses dihapus dari sistem
<i>Alternate Course</i>	8. Jika <i>User sales</i> melakukan klik tombol batal maka data tidak tersimpan dan halaman dikembalikan ke <i>Kelola sales</i>	
<i>Post Condition</i>	<i>User sales</i> sukses melakukan perubahan data sales	

3. Use Case Narrative Sales Profile

Tabel 4. 13 Use Case Narrative sales profile

<i>Use case name</i>	<i>Profile</i>	
<i>Actor</i>	<i>Sales</i>	
<i>Description</i>	Use case ini menggambarkan langkah-langkah yang dilakukan oleh user sales dalam melakukan perubahan data profil mereka dalam sistem.	
<i>Precondition</i>	Data profile user masih menggunakan data yang lama	
<i>Typical Course of Event</i>	<i>Action</i>	<i>System Response</i>
	1. <i>User admin</i> melakukan login 2. <i>User admin</i> melakukan klik pada Profile 3. <i>User admin</i> input password baru 4. <i>User admin</i> melakukan klik simpan	1.1 Sistem menampilkan halaman dashboard admin 2.1 Sistem menampilkan Profile user 3.1 Sistem menunjukan <i>field</i> yang terisi 4.1 Data <i>password</i> berhasil diubah
<i>Alternate Course</i>	4. Jika <i>User sales</i> melakukan klik tombol batal maka data tidak tersimpan dan halaman dikembalikan profile	
<i>Post Condition</i>	<i>User sales</i> sukses melakukan perubahan data profile	

C. Warehouse

1. Use Case Narrative Warehouse Login

Tabel 4. 14 Use Case Narrative Warehouse Login

<i>Use case name</i>	<i>Login</i>	
<i>Actor</i>	<i>Warehouse</i>	
<i>Description</i>	Use case ini menggambarkan proses login yang dilakukan oleh User Warehouse dengan memasukkan username dan password yang terdaftar dalam sistem.	
<i>Precondition</i>	Halaman masih menunjukan halaman beranda	
<i>Typical Course of Event</i>	<i>Action</i>	<i>System Response</i>
	1. <i>User Warehouse</i> melakukan input username dan password	1.1 Sistem memeriksa data <i>username</i> dan <i>password</i>

	2. <i>User Warehouse</i> melakukan klik login	2.1 Apabila data sesuai maka sistem akan menampilkan halaman beranda dengan status login yang sudah ada.
<i>Alternate Course</i>	Apabila username dan password yang dimasukkan oleh <i>User Warehouse</i> tidak sesuai dengan data yang ada dalam database, sistem akan menampilkan pesan error.	
<i>Post Condition</i>	<i>User Warehouse</i> sukses melakukan login	

2. Use Case Narrative Warehouse Kelola Stok

Tabel 4. 15 Use Case Narrative Warehouse Kelola Stok

<i>Use case name</i>	<i>Kelola Stok</i>	
<i>Actor</i>	<i>Warehouse</i>	
<i>Description</i>	<i>Use case</i> ini menjelaskan tentang bagaimana <i>User Warehouse</i> melakukan <i>Kelola Stok</i>	
<i>Precondition</i>	Data Stok masih menunjukkan data stok yang lama dan belum <i>ter-update</i> sehingga belum mengetahui stok saat ini	
<i>Typical Course of Event</i>	<i>Action</i>	<i>System Response</i>
	1. <i>User Warehouse</i> melakukan login	1.1 Sistem menampilkan halaman utama
	2. <i>User Warehouse</i> melakukan klik pada <i>Kelola Stok</i>	2.1 Sistem menampilkan halaman <i>Kelola Stok</i>
	3. <i>User Warehouse</i> melakukan klik tambah <i>Stok</i>	3.2 Sistem menampilkan form tambah stok
	4. <i>User Warehouse</i> melakukan input data stok dan melakukan klik <i>simpan</i>	4.2 Data stok ditambahkan kedalam sistem.
	5. <i>User Warehouse</i> melakukan klik <i>perbarui data</i> dan klik <i>simpan</i>	5.2 Data stok sukses diperbarui dari sistem
	6. <i>User Warehouse</i> melakukan input stok keluar	6.2 Data stok keluar sukses disimpan dan stok diperbarui

<i>Alternate Course</i>	5. Jika <i>User Warehouse</i> melakukan klik tombol batal maka data tidak tersimpan dan halaman dikembalikan ke Kelola Stok
<i>Post Condition</i>	<i>User Warehouse</i> sukses melakukan perubahan data stok

3. Use Case Narrative Warehouse Profile

Tabel 4. 16 Use Case Narrative Warehouse profile

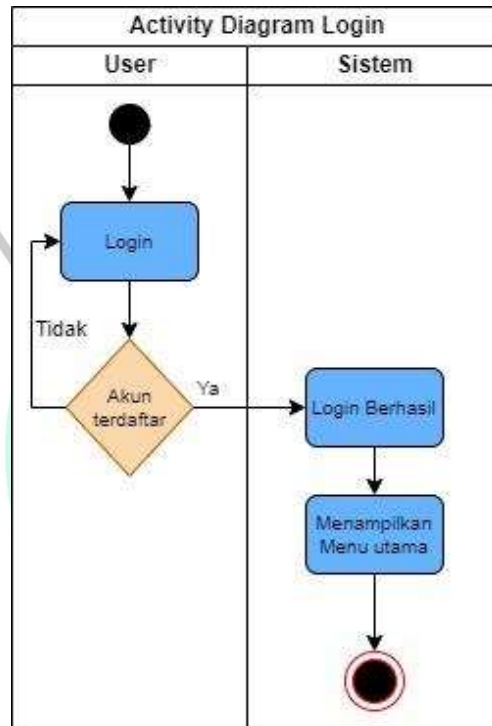
<i>Use case name</i>	<i>Profile</i>	
<i>Actor</i>	<i>Warehouse</i>	
<i>Description</i>	Use case ini menggambarkan bagaimana User Warehouse dapat melakukan perubahan pada data profilnya dalam sistem.	
<i>Precondition</i>	Data profile user masih menggunakan data yang lama	
<i>Typical Course of Event</i>	<i>Action</i>	<i>System Response</i>
	<ol style="list-style-type: none"> 1. <i>User warehouse</i> melakukan login 2. <i>User warehouse</i> melakukan klik pada Profile 3. <i>User warehouse</i> input password baru 4. <i>User warehouse</i> melakukan klik simpan 	<ol style="list-style-type: none"> 1.1 Sistem menampilkan halaman dashboard admin 2.1 Sistem menampilkan Profile user 3.1 Sistem menunjukan <i>field</i> yang terisi 4.1 Data <i>password</i> berhasil diubah
<i>Alternate Course</i>	4. Jika <i>User warehouse</i> melakukan klik tombol batal maka data tidak tersimpan dan halaman dikembalikan profile	
<i>Post Condition</i>	<i>User warehouse</i> sukses melakukan perubahan data profile	

4.3.2 Activity Diagram

Activity diagram adalah sebuah representasi visual yang digunakan dalam pengembangan aplikasi untuk menggambarkan urutan dan alur aktivitas dalam sebuah proses. Diagram ini berfungsi untuk memodelkan bagaimana objek dalam sistem berinteraksi dan melakukan tindakan-tindakan yang diperlukan. Berikut adalah contoh activity diagram pada perancangan aplikasi prediksi harga telur:

1. Activity Diagram Login

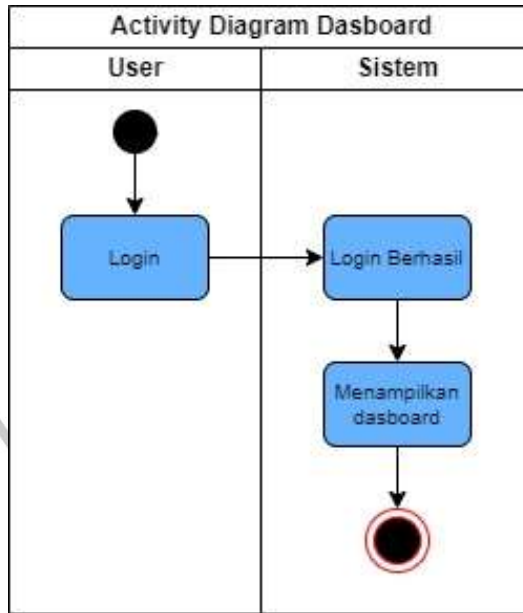
Activity Diagram Login Admin merupakan diagram aktivitas yang menjelaskan proses admin melakukan login pada sistem sehingga menghasilkan kondisi admin sudah terlogin kedalam sistem.



Gambar 4. 3 Activity Diagram login

2. Activity Diagram Dashboard

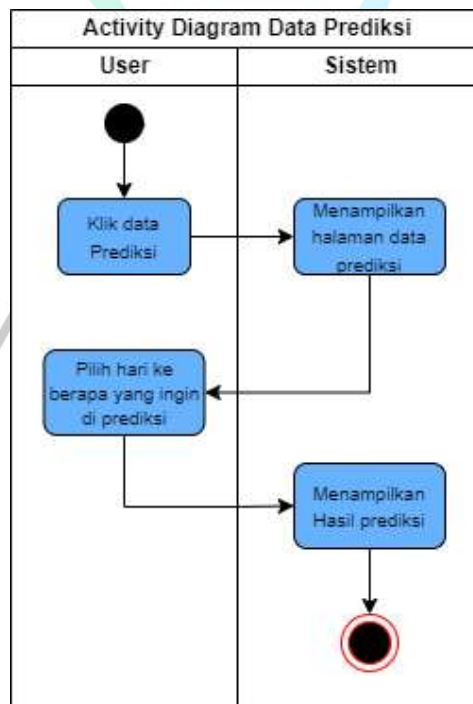
Activity Diagram Dashboard merupakan diagram yang menjelaskan alur bagaimana proses dalam mengakses halaman dashboard pada sistem.



Gambar 4. 4 activity diagram dashboard

3. Activity Diagram Prediksi Harga

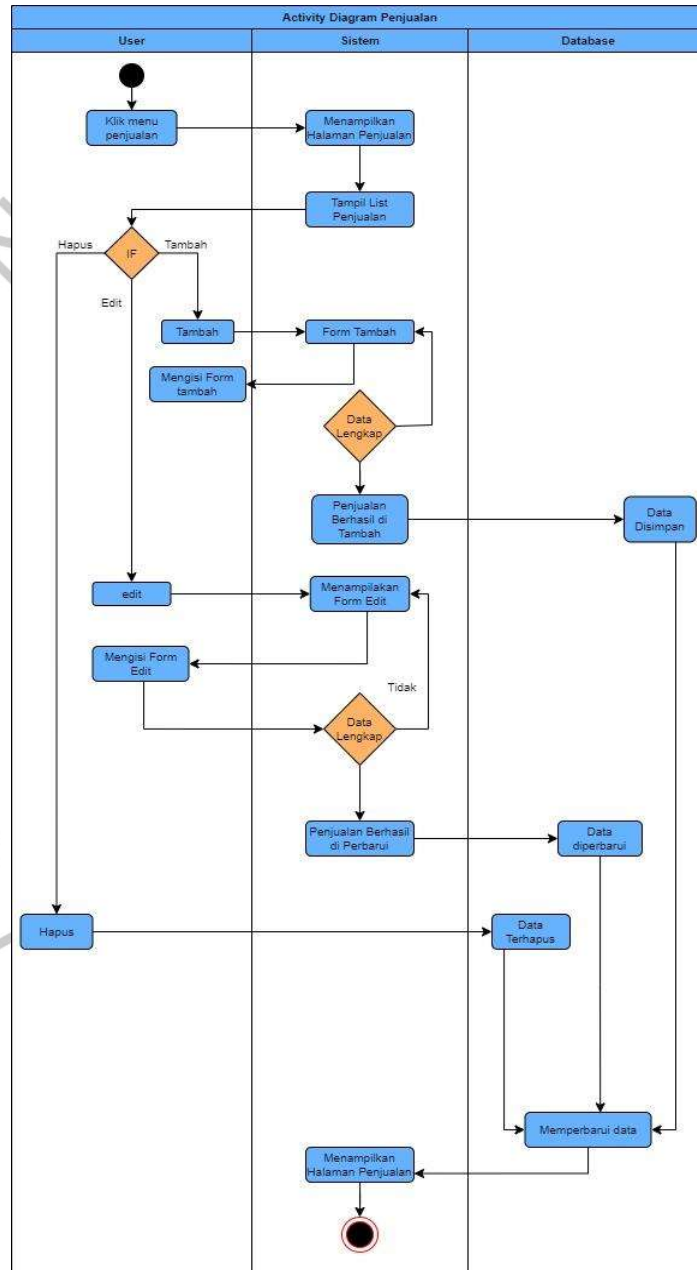
Activitu diagram prediksi harga merupakan diagram yang melihatkan prediksi harga yang akan datang



Gambar 4. 5 Acitivity Diagram Prediksi Harga

4. Activity Diagram Kelola User

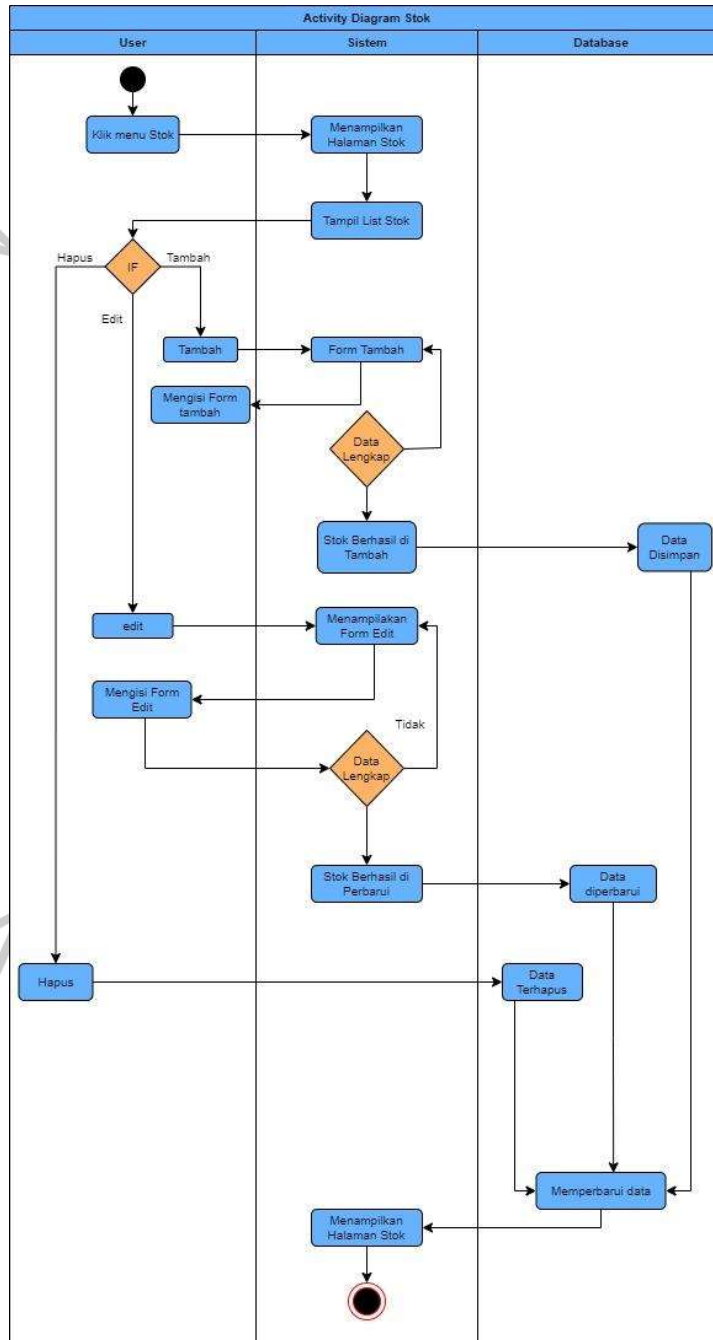
Activity diagram penjualan merupakan diagram yang menggambarkan aktivitas user penjualan melakukan pengelolaan data sales mulai dari melakukan penambahan, perbaruan hingga melakukan hapus data sales.



Gambar 4. 6 Acitivity Diagram Kelola User

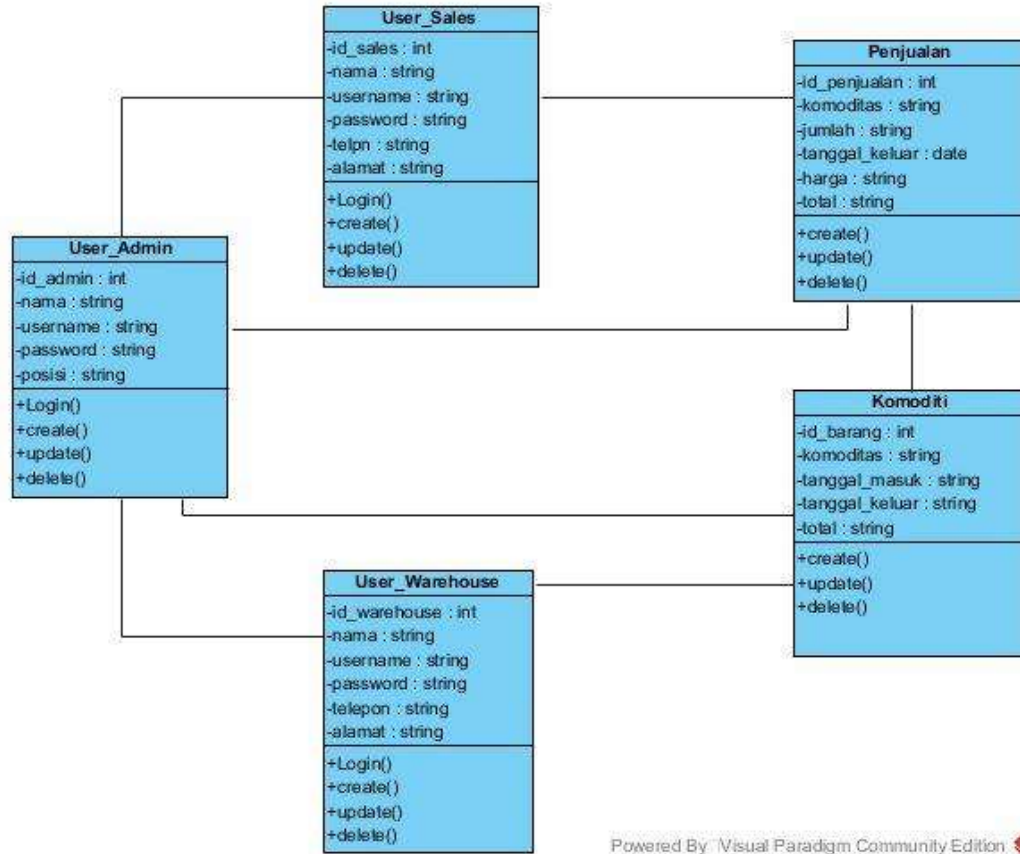
5. Activity Diagram Kelola Stok

Activity Diagram Warehouse Kelola Stok merupakan diagram yang menggambarkan aktivitas user Warehouse dalam melakukan pengelolaan Stok mulai dari penambahan, memperbarui hingga melakukan input data stok keluar.



Gambar 4. 7 Activity Diagram Kelola Sales

4.3.3 Class Diagram



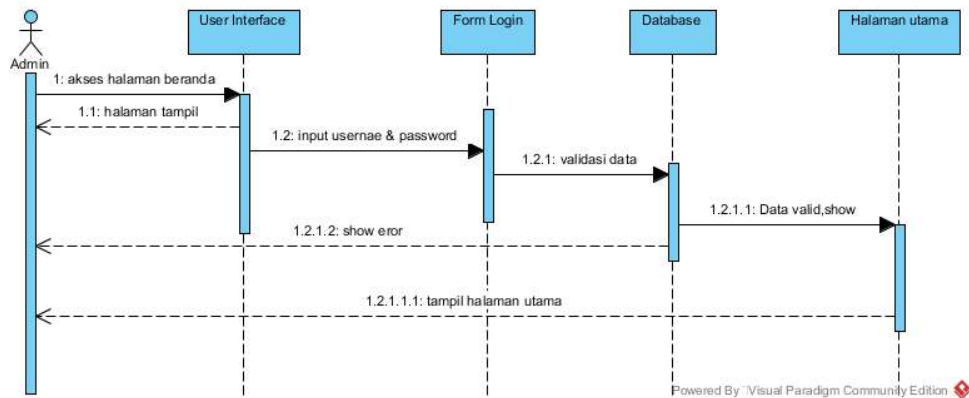
Gambar 4. 8 Class Diagram

4.3.4 Sequence Diagram

Berikut adalah rancangan sequence diagram untuk sistem prediksi harga penjualan telur.

1. Sequence Diagram Login

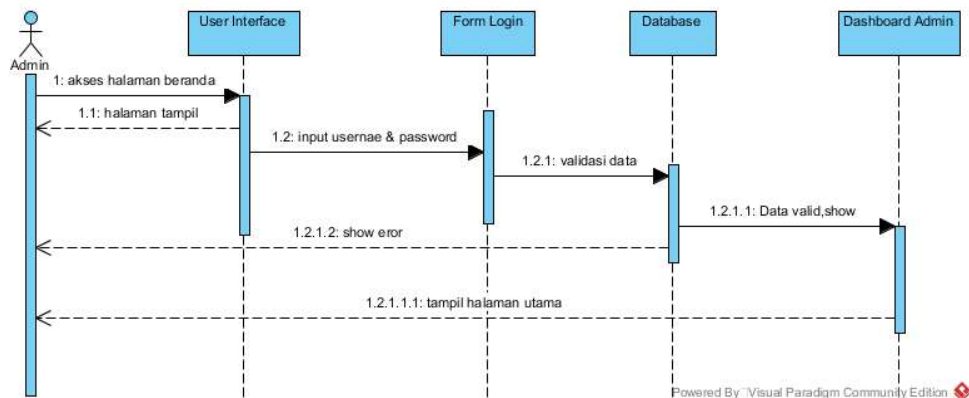
sequence diagram yang menjelaskan proses login pengguna. Pengguna memulai dengan membuka halaman login dan memasukkan username dan password. Sistem kemudian melakukan validasi terhadap data yang dimasukkan. Jika username atau password tidak cocok, sistem akan menghasilkan pesan error. Jika data valid, pengguna akan diarahkan ke halaman home admin.



Gambar 4. 9 Sequence Diagram Login

2. Sequence Diagram Mengakses Dashboard

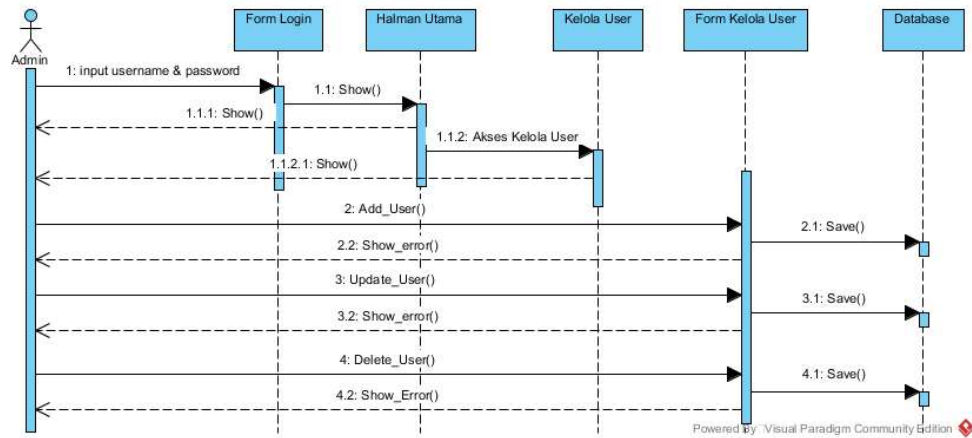
sequence diagram yang menjelaskan proses pengguna masuk ke dashboard admin. Pengguna memulai dengan memasukkan email dan password. Sistem kemudian melakukan validasi apakah email dan password tersebut terdaftar. Jika tidak terdaftar, sistem akan menghasilkan pesan error. Jika email dan password terdaftar, pengguna akan diarahkan ke halaman dashboard admin.



Gambar 4. 10 Sequence Diagram Dashboard Admin

3. Sequence Diagram Mengelola User

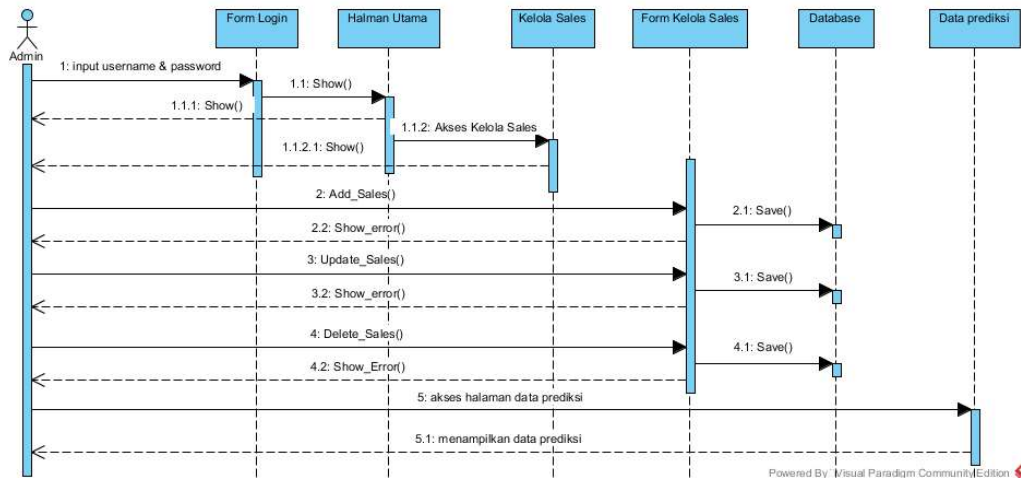
sequence diagram yang menjelaskan proses pengguna masuk ke menu "Kelola User". Pengguna memulai dengan memasukkan email dan password. Sistem kemudian melakukan validasi apakah email dan password tersebut terdaftar. Jika tidak terdaftar, sistem akan menghasilkan pesan error. Jika email dan password terdaftar, pengguna akan diarahkan ke halaman utama admin. Selanjutnya, pengguna mengklik menu "Kelola User" dan masuk ke halaman "Kelola User".



Gambar 4. 11 Sequence Diagram Kelola User

4. Sequence Diagram Mengakses Data Prediksi

Sequence diagram dibawah ini menjelaskan pengguna yang akan masuk pada menu data prediksi. Pengguna melakukan input penjualan setelah penjualan tersimpan maka sistem akan otomatis melakukan perubahan data pada data prediksi.

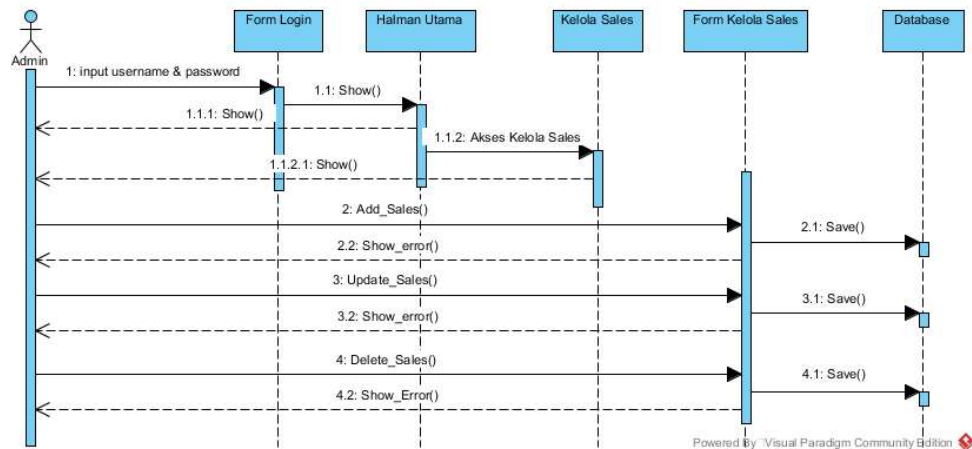


Gambar 4. 12 Sequence diagram data prediksi

5. Sequence Diagram Kelola Sales

sequence diagram yang menjelaskan proses pengguna masuk ke menu "Kelola Sales". Pengguna memulai dengan memasukkan email dan password. Sistem kemudian melakukan validasi apakah email dan password tersebut terdaftar. Jika tidak terdaftar, sistem akan menghasilkan pesan error. Jika email dan password

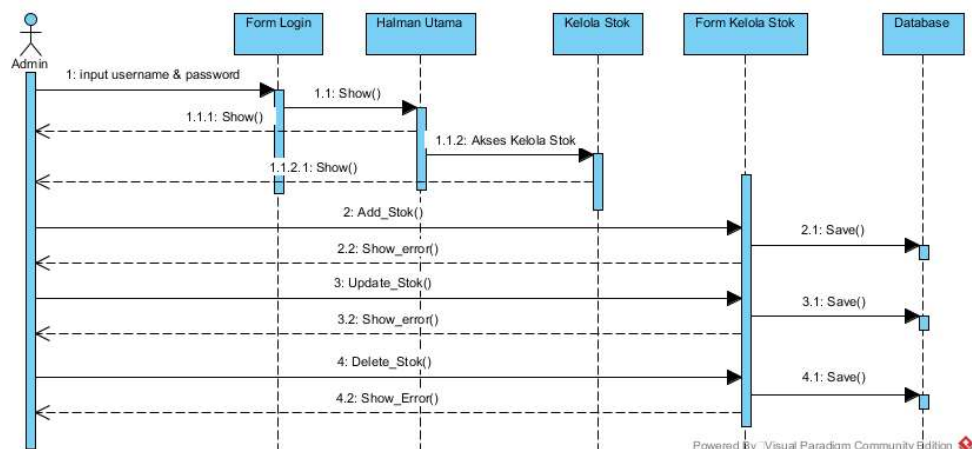
terdaftar, pengguna akan diarahkan ke halaman utama admin. Selanjutnya, pengguna mengklik menu "Kelola Sales" dan masuk ke halaman "Kelola Sales".



Gambar 4. 13 Sequence Diagram Kelola Sales

6. Sequence Diagram Kelola Stok

sequence diagram yang menjelaskan proses pengguna masuk ke menu "Kelola Stok". Pengguna memulai dengan memasukkan email dan password. Sistem kemudian melakukan validasi apakah email dan password tersebut terdaftar. Jika tidak terdaftar, sistem akan menghasilkan pesan error. Jika email dan password terdaftar, pengguna akan diarahkan ke halaman utama admin. Selanjutnya, pengguna mengklik menu "Kelola Stok" dan masuk ke halaman "Kelola Stok".



Gambar 4. 14 Sequence Diagram Kelola Stok

4.3.5 Perancangan Basis Data

Perancangan basis data merupakan proses rancangan dari sebuah basis data yang akan digunakan untuk menyimpan data dalam sebuah sistem. Pada aplikasi prediksi harga telur, peneliti membuat basis data yang diperlukan untuk menyimpan dan mengolah data. Berikut adalah rancangan basis data pada aplikasi prediksi harga telur.

Tabel 4. 17 basisdata user

No	Field	Type	Length	Keterangan
1	Id	Bigint	20	id user
2	Name	Varchar	255	nama user
3	Email	Varchar		Email user
4	email_verified_at	Timestamp		Verifikasi email
5	password	Varchar		Password user
6	remember_token	Timestamp		Lupa token
7	Created_at	TimeStamp		Menambah user
8	Update_at	Timestamp		Mengupdate data user

Tabel 4. 18 Perancangan Basis data Penjualan

No	Field	Type	Length	Keterangan
1	Id	Int	11	id penjualan
2	Id komoditi	int	11	Id komoditi
3	Nama_komoditi	Text		Nama komoditi
4	Harga_komoditi	Text		Harga komoditi
5	Qty	Text		Jumlah komoditi
6	Total harga	Text		Total harga
7	Created_at	Text		Tambah komoditi
8	Update_at	Text		Update komoditi
9	Catatan	Text		Catatan komoditi

Tabel 4. 19 Perancangan Basis Data reset Password

No	Field	Type	Length	Keterangan
1	Email	Varchar	255	Email user
2	Token	Varchar	255	Koneksi email
3	Created	Timestamp	100	Membuat password baru

Tabel 4. 20 Perancangan Basis Data Prediksi

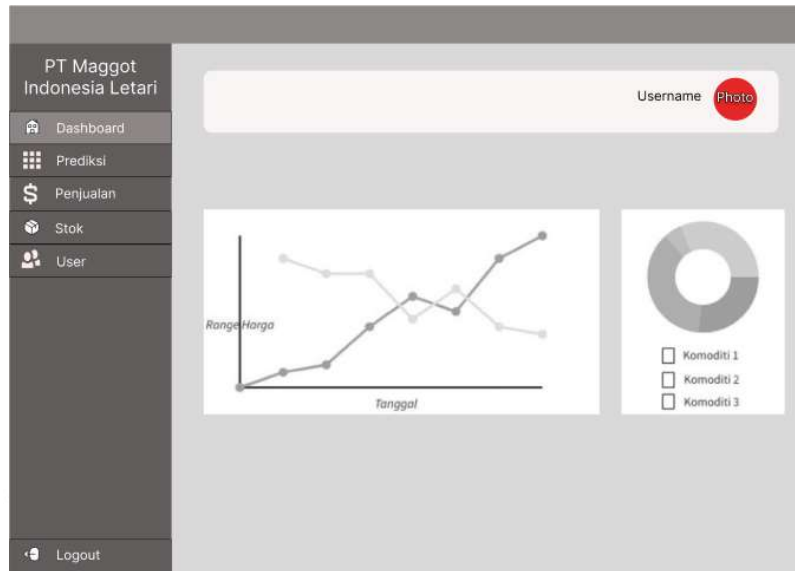
No	Field	Type	Length	Keterangan
1	Id	Int	11	Id
2	Date	Text		Tanggal
3	Price	Bigint	20	Harga
4	Created_at	Datetime		Tambah
5	upload	Datetime		Upload

4.3.7 Perancangan Antarmuka

Perancangan antarmuka adalah proses yang bertujuan untuk merancang tampilan visual dan interaksi antara pengguna dan sistem. Selain itu, perancangan antarmuka juga membantu memudahkan programmer dalam mendesain aplikasi yang diinginkan. Berikut merupakan perancangan antarmuka aplikasi prediksi harga telur.

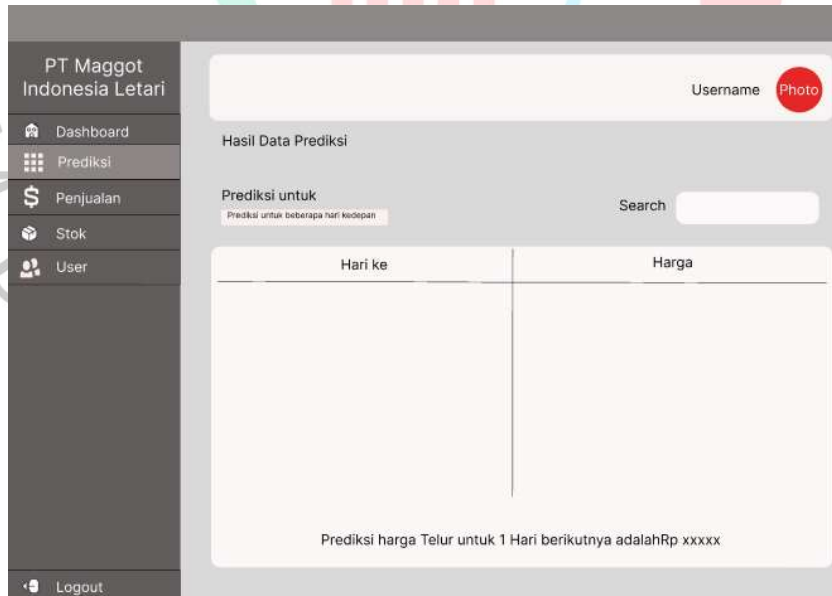
Gambar 4. 15 Tampilan menu login

Gambar 4.15 merupakan perancangan antarmuka untuk halaman login. Pada halaman login, pada halaman login terdapat memasukan username dan password



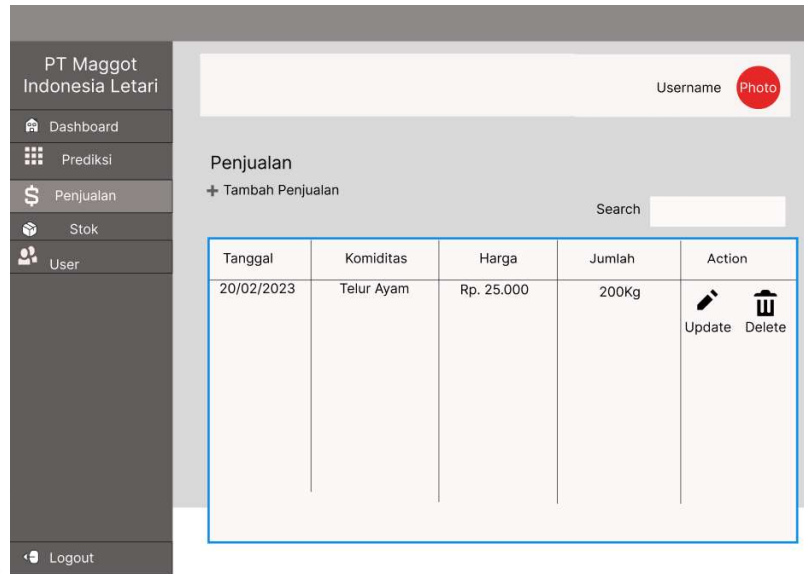
Gambar 4. 16 Tampilan Dashboard

Gambar 4.16 diatas merupakan perancangan antarmuka untuk halaman dashboard, pada halaman dashboard ini terdapat grafik harga dan data komoditi



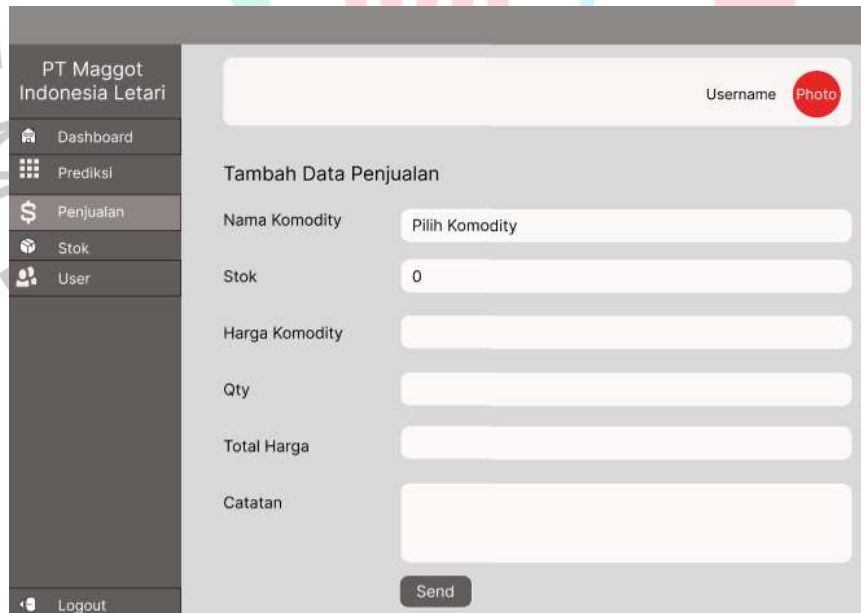
Gambar 4. 17 Tampilan Prediksi harga

Gambar 4.17 diatas merupakan rancangan antar muka halaman prediksi, pengguna dapat melihat prediksi harga telur 1 hari berikutnya.



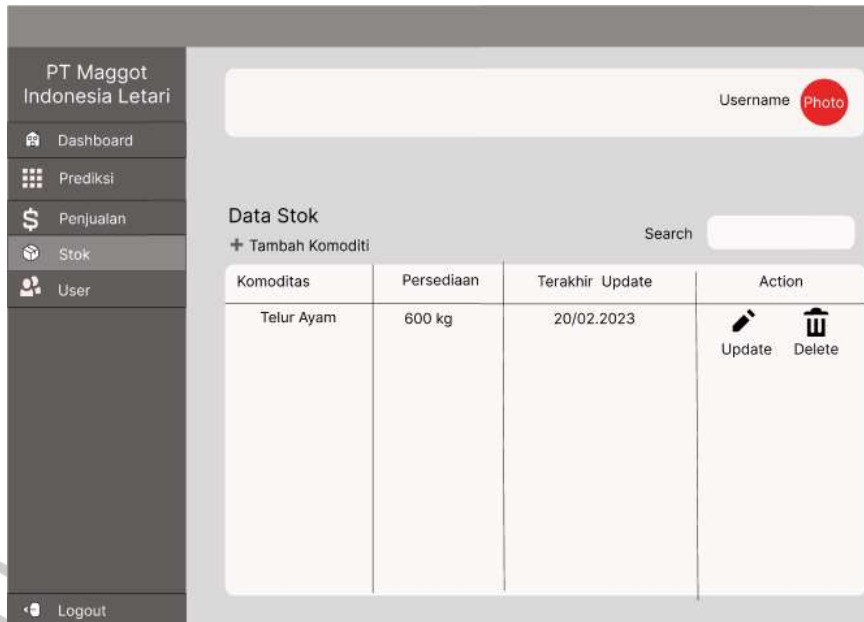
Gambar 4. 18 Tampilan Penjualan

Gambar 4.18 diatas merupakan rancangan antar muka halaman penjualan, pengguna dapat menambah,update,delete pada kolom penjualan dan dapat melakukan search.



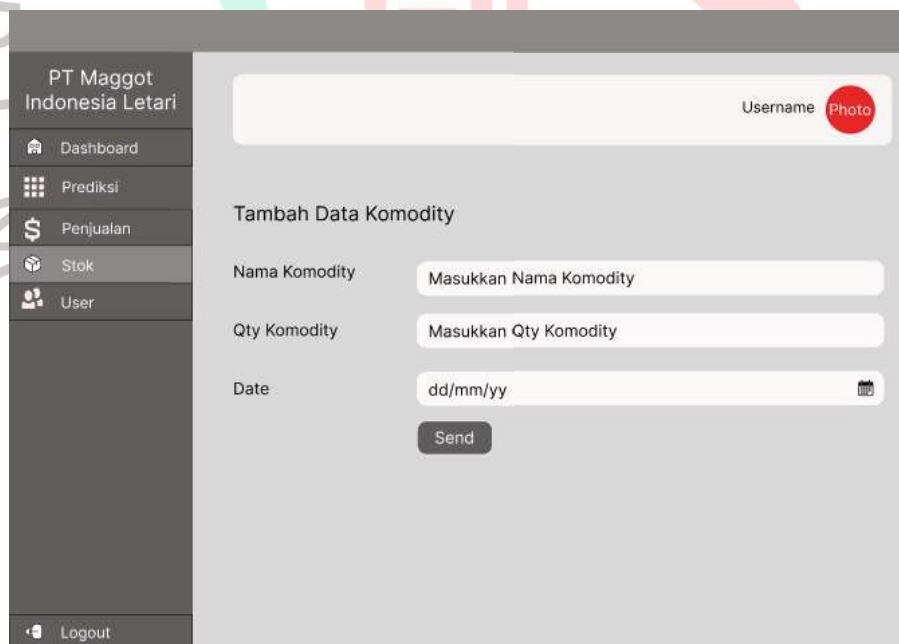
Gambar 4. 19 Tampilan Tambah Penjualan

Gambar 4.19 merupakan rancangan antar muka halaman tambah penjualan.



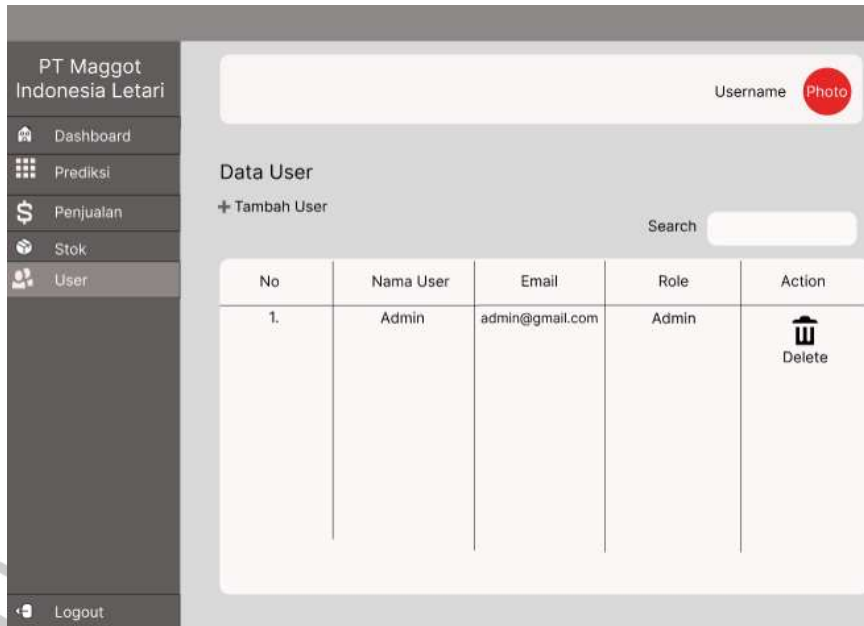
Gambar 4. 20 Tampilan Stok

Gambar 4.20 merupakan rancangan antar muka menu stok , pengguna dapat menambah,update,delete pada kolom stok dan dapat melakukan search



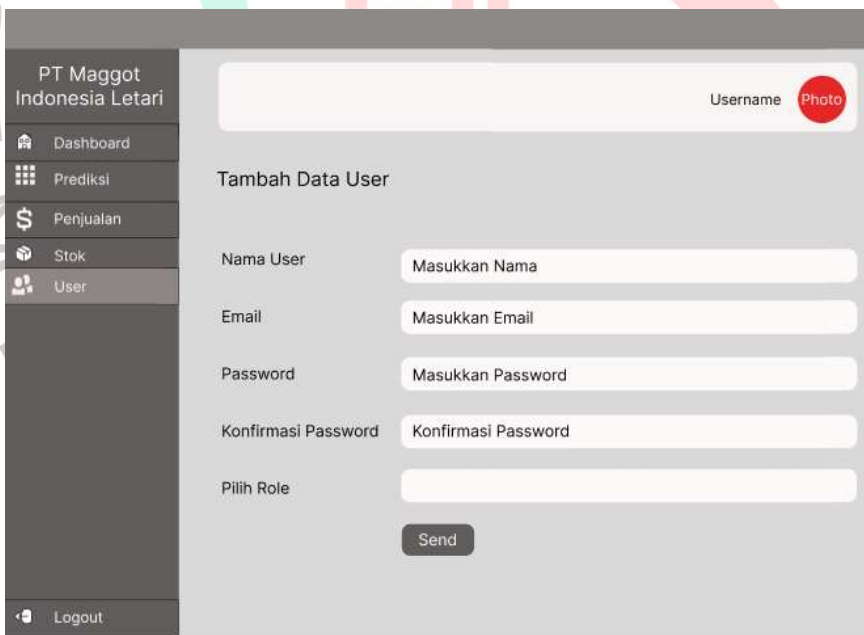
Gambar 4. 21 Tampilan Tambah Stok

Gambar 4.21 merupakan rancangan antar muka menu tambah stok , pengguna dapat menambah stok.



Gambar 4. 22 Tampilan Menu User

Gambar 4.22 diatas merupakan rancangan antar muka halaman user, pengguna dapat menambah,delete pada kolom user dan dapat melakukan search.



Gambar 4. 23 Tampilan Menu Tambah User

Gambar 4.23 merupakan rancangan antar muka menu tambah stok , pengguna dapat menambah stok.

4.3.8 Perancangan Pengujian

Perancangan pengujian adalah proses merancang dan mempersiapkan rencana pengujian untuk menguji perangkat lunak atau sistem secara menyeluruh. Perancangan pengujian pada penelitian ini menggunakan pengujian *white box* dan *black box*.

4.3.8.1 Perancangan Pengujian *Black Box*

Perancangan pengujian *black box* merupakan perancangan pengujian yang diterapkan pada semua fitur sebuah aplikasi untuk menentukan apakah output atau hasil yang diperoleh berkaitan dengan input yang diberikan. Berikut merupakan tabel skenario, input, dan hasil pengujian *black box*.

Tabel 4. 21 Rancangan pengujian balck box Web

No	Skenario Pengujian	Input	Hasil Yang Diharapkan
1	Login	Pengguna memasukan username dan password	Berhasil masuk ke tampilan <i>website</i> .
2	Dasbord	Pengguna dapat melihat menu dasbord yang ada apa aplikasi	Berhasil masuk dan dapat melihat menu dasbord
3	<i>Data Prediksi</i>	Pengguna mengklik menu data prediksi	Berhasil masuk pada menu data prediksi dan terlihat harga telur dan harga prediksi
4	<i>Penjualan</i>	Pengguna mengklik menu penjualan	Berhasil masuk pada menu penjualan dan dapat melihat penjualan telur
5	<i>Tambah data Penjualan</i>	Pengguna mengklik ikon tambah penjualan, maka akan muncul pilihan nama komodity, stok, harga komodity, qty, total harga, catatan, lalu mengisi data data yang tersedia di menu tersebut	Berhasil menambah data penjualan
5	<i>Edit data penjualan</i>	Pengguna mengklik ikon edit penjualan, maka akan muncul pilihan nama komodity, stok, harga komodity, qty, total harga, catatan, lalu mengisi data data yang tersedia di menu tersebut	Berhasil mengedit data penjualan
6	Hapus data penjualan	Pengguna mengklik ikon hapus penjualan, maka penjualan yang dipilih di hapus akan hilang	Berhasil menghapus salah satu data penjualan
7	Stok	Pengguna mengklik menu stok	Berhasil masuk pada menu stok dan dapat melihat data stok yang ada

8	Tambah data Komodity stok	Pengguna mengklik ikon tambah data komodity, maka akan muncul Nama komodity,Qty komodity, date	Berhasil menambah data komodity
9	Edit data stok	Pengguna mengklik ikon edit pada menu stok, maka akan muncul pilihan, nama komodity,qty komodity,date	Berhasil menedit data yang ada pada stok
10	Hapus data stok	Pengguna mengklik ikon hapus pada menu stok, maka stok yang dipilih di hapus akan hilang	Berhasil menghapus salah satu data yang ada pada kolom stok
11	Logout	Pengguna mengklik <i>button logout</i> .	Berhasil keluar dari aplikasi

4.3.8.2 Perancangan Pengujian *White Box*

Perancangan pengujian *white box* merupakan perancangan untuk merancang aplikasi, karena dengan melakukan pengujian ini, dapat memastikan bahwa aplikasi berjalan dengan baik sesuai dengan skenario yang diinginkan.

Tabel 4. 22 Pengujian White box web

No	Source Code	Hasil Yang Diharapkan
1	<pre> div class="card-body"> <form method="POST" action="{{ route('login') }}"> @csrf <div class="row mb-3"> <label for="email" class="col-md-4 col-form-label text-md-end">{{ _('Email Address') }}</label> <div class="col-md-6"> <input id="email" type="email" class="form-control @error('email') is-invalid @enderror" name="email" value="{{ old('email') }}" required autocomplete="email" autofocus> @error('email') {{ \$message }} @enderror </div> </div> </pre>	Login
2	<pre> function drawChart() { var data = google.visualization.arrayToDataTable(<?php echo \$sql ?>); var options = { title: 'Data Penjualan', curveType: 'function', legend: { position: 'bottom' } }; var chart = new google.visualization.LineChart(document.getElementById('curve_chart')); chart.draw(data, options); </pre>	Dashboard

3	<pre> <tr> </tr> </tbody> </table> <pre> <?php \$x = 0; \$jumlah_x = 0; \$jumlah_y = 0; \$jumlah_xx = 0; \$jumlah_xy = 0; foreach (\$GetDataPenjualan as \$key => \$data) { \$xi = \$key+1; \$jumlah_x += \$key; \$jumlah_y += \$data->harga_komoditi; \$jumlah_xx += (\$key * \$key); \$jumlah_xy += (\$key * \$data->harga_komoditi); } ?> </tr> </pre>	Data Prediksi
4	<pre> <tbody class="table table-border-bottom-0"> @foreach (\$GetDataPenjualan as \$key => \$data) <tr> <td>{{ \$key+1 }}</td> <td>{{ \$data->catatan }}</td> <td>{{ \$data->created_at }}</td> <td>@currency2(\$data->harga_komoditi)</td> <td>{{ \$key }}</td> <td>@currency2(\$data->harga_komoditi)</td> <td>{{ \$key * \$key }}</td> <td> @currency2(\$key * \$data->harga_komoditi) </td> </tr> @endforeach </tbody> <tfoot style="font-weight:bold;"> </tfoot> </pre>	Menu Penjualan
56	<pre> <div class="col-lg-12"> <h4 class="fw-bold py-3 mb-4">Penjualan Data Penjualan</h4> <div class="row"> <div class="col-md-12"> <ul class="nav nav-pills flex-column flex-md-row mb-3"> <li class="nav-item"> <i class="bx bx-plus-circle me-1"></i> Tambah Data Penjualan </pre>	Tambah data penjualan
6	<pre> <div class="col-lg-12"> <h4 class="fw-bold py-3 mb-4">Penjualan Edit Data Penjualan</h4> @if (\$message = Session::get('success')) <div class="alert alert-success2 alert-dismissible" role="alert"> {{ \$message }} <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button> </div> @endif </pre>	Edit data penjualan

7	<pre><form class="debb" action="{{url('penjualan/delete/'.\$data->id)}}" method="POST" id="deleteDataKomoditi-{{ \$data->id }}"> @csrf </form> </td> </tr> <tr> <td="" data-bbox="354 499 407 884">8</tr>></pre>	<pre></div> <div class="card mb-4"> <div class="card-body"> <table id="datatable" class="table"> <thead> <tr> <th>NO</th> <th>Nama Komoditi</th> <th>Harga Komoditi</th> <th>Qty Komoditi</th> <th>Date Komoditi</th> <th>Action</th> </tr> </thead> <tbody></pre>	Hapus data penjualan
9	<pre>div class="col-lg-12"> <h4 class="fy-bold py-3 mb-4">Komoditi Tambah Data Komoditi</h4> <div class="card mb-4"> <div class="card-body"> <form method="POST" action="{{route('store.komoditi')}}"> @csrf</pre>	Stok	
10	<pre>tbody> @foreach (\$GetDataKomoditi as \$key => \$data) <tr> <td>{{ \$key+1 }}</td> <td>{{ \$data->nama_komoditi }}</td> <td>{{ \$data->qty }}</td> <td>{{ \$data->date_in }}</td> </tr> id)}}"> <button class="btn btn-sm btn-primary"> EDIT </button> </pre>	Tambah data Komodity stok	
11	<pre><button class="btn btn-sm btn-danger" onclick="deleteItem('{{ \$data->id }}')"> HAPUS </button> <form class="debb" action="{{url('komoditi/delete/'.\$data->id)}}" method="POST" id="deleteDataKomoditi-{{ \$data->id }}"> @csrf </form> </td> </tr> <tr> <td="" data-bbox="354 1583 407 1749"></tr>></pre>		Edit data stok
		Hapus data stok	

12	<pre> <div class="card mb-4"> <div class="card-body"> <table id="datatable" class="table"> <thead> <tr> <th>NO</th> <th>Nama User</th> <th>Email</th> <th>Role</th> <th>Action</th> </tr> </thead> <tbody> </pre>	User
13	<pre> div class="col-ig-12"> <h4 class="fw-bold py-3 mb-4">User Data User</h4> <div class="row"> <div class="col-md-12"> <ul class="nav nav-pills flex-column flex-md-row mb-3"> <li class="nav-item"> <i class="bx bxs-plus-circle me-1"></i> Tambah User </pre>	Tambah user
14	<pre> button class="btn btn-sm btn-danger" onclick="deleteltem('{{ \$data->id }}')"> HAPUS </button> </pre>	Hapus user

Tabel 4.23 Perancangan Pengujian White Box python

No	Source Code	Hasil Yang Diharapkan
1	<pre> from google.colab import drive drive.mount('/content/drive') </pre>	untuk menghubungkan dan mengakses Google Drive ke dalam Colab Notebook.
2	<pre> import pandas as pd df= pd.read_excel('/content/drive/MyDrive/Colab Notebooks/Dataset-Penjualan.xlsx') df </pre>	Memuat dataset
3	<pre> df.head() </pre>	Menampilkan 5 dataset teratas
4	<pre> df.shape </pre>	Untuk menginformasikan jumlah baris dan jumlah kolom
5	<pre> df.info() </pre>	Menampilkan informasi tentang struktur dataframe
6	<pre> df.describe() </pre>	Untuk menghitung statistik seperti jumlah data, rata-rata, standar deviasi, nilai minimum, kuartil, dan nilai maksimum untuk setiap kolom numerik dalam DataFrame.
7	<pre> df.isnull().sum() </pre>	untuk mendeteksi nilai yang hilang (NaN/null) dalam DataFrame.

8	<pre>import pandas as pd import matplotlib.pyplot as plt f = plt.figure(figsize=(12,4)) f.add_subplot(1,2,1) df['harga'].plot(kind='kde') f.add_subplot(1,2,2) plt.boxplot(df['harga']) plt.show()</pre>	<p>tampilan visualisasi distribusi data pada kolom 'harga' menggunakan plot KDE (Kernel Density Estimation) dan box plot.</p>
9	<pre>import pandas as pd import matplotlib.pyplot as plt f = plt.figure(figsize=(12,4)) f.add_subplot(1,2,1) df['stok telur'].plot(kind='kde') f.add_subplot(1,2,2) plt.boxplot(df['stok telur']) plt.show()</pre>	<p>tampilan visualisasi distribusi data pada kolom 'stok telur' menggunakan plot KDE (Kernel Density Estimation) dan box plot. Plot KDE membantu dalam memperoleh pemahaman visual tentang bagaimana data 'stok telur' tersebar dan berdistribusi di sepanjang rentang nilainya.</p>
10	<pre>import pandas as pd import matplotlib.pyplot as plt plt.scatter(df['harga'], df['stok telur']) plt.xlabel('harga') plt.ylabel('stok telur') plt.title('Scatter Plot Stok Telur vs harga') plt.show()</pre>	<p>Untuk menampilkan scatter plot, sehingga dapat melihat apakah ada hubungan linier atau pola tertentu antara harga dan stok telur</p>
10	<pre>df.corr()</pre>	<p>Untuk menghitung korelasi antara setiap pasangan kolom dalam DataFrame df. Korelasi adalah ukuran statistik yang menggambarkan hubungan linier antara dua variabel. Hasilnya adalah sebuah DataFrame yang berisi koefisien korelasi antara setiap pasangan kolom.</p>
11	<pre>import pandas as pd import matplotlib.pyplot as plt from sklearn.linear_model import LinearRegression from sklearn.model_selection import train_test_split x = df['stok telur'].values.reshape(-1,1) y = df['harga'].values.reshape(-1,1) x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2) lin_reg = LinearRegression() lin_reg.fit(x_train, y_train) print(lin_reg.coef_) print(lin_reg.intercept_) lin_reg.score(x_test, y_test)</pre>	<p>Code tersebut melakukan regresi linear sederhana untuk memprediksi harga berdasarkan stok telur. Model regresi linear akan menghasilkan koefisien dan intercept yang dapat digunakan untuk membuat prediksi harga. Skor R-squared digunakan untuk mengevaluasi seberapa baik model tersebut cocok dengan data uji.</p>
12	<pre>y_prediksi = lin_reg.predict(x_test) plt.scatter(x_test, y_test) plt.plot(x_test, y_prediksi, c='r') plt.xlabel('stok telur') plt.ylabel('harga') plt.title('Plot stok telur vs Harga')</pre>	<p>Menampilkan plot yang memvisualisasikan hubungan antara stok telur dan harga menggunakan scatter plot dan garis regresi linear. Scatter plot menunjukkan sebaran data aktual (stok telur vs harga), sementara garis regresi linear menunjukkan prediksi harga berdasarkan stok telur.</p>

13	<code>lin_reg.predict([[500]])</code>	Untuk melakukan prediksi
----	---------------------------------------	--------------------------

