**Dokumen Teknis**

Rancang Bangun Website Toko Tambunan Pupuk

# Tugas Proyek Akhir Semester

**Mata Kuliah: Pengembangan Aplikasi Terdistribusi**

# Dipersiapkan oleh:

|  |  |
| --- | --- |
| NIM 11322010 | Richard Paulus Aritonang |
| NIM 11322026 | Aqustin Angel Tambunan |
| NIM 11322040 | Cristian Nicolas Tambunan |
| NIM 11322056 | Nita Herlinda Kurnyawati Simangunsong |
|  |

**Untuk: Institut Teknologi Del**

# 2024



**PROYEK PENGEMBANGAN APLIKASI TERDISTRIBUSI INSTITUT TEKNOLOGI DEL 2022**

*Nomor dokumen: DT-PASTI-10 Tanggal: 2022-05-15 Jumlah Halaman: 19*

# DAFTAR ISI

[Tugas Proyek Akhir Semester 1](#_Toc166747238)

[Dipersiapkan oleh: 1](#_Toc166747239)

[2024 1](#_Toc166747240)

[DAFTAR ISI 2](#_Toc166747241)

[DAFTAR GAMBAR 3](#_Toc166747242)

[DAFTAR TABEL 4](#_Toc166747243)

[1 Pendahuluan 1](#_Toc166747244)

[1.1 Deskripsi Umum *Website* 1](#_Toc166747245)

[1.2 Perbedaan *Monolith* dan *Microservice* 3](#_Toc166747246)

[1.3 Karakteristik Pengguna Website 4](#_Toc166747247)

[1.4 Fungsi pada *Website* 5](#_Toc166747248)

[2 Desain Rancangan *Website* 8](#_Toc166747249)

[2.1 Use Case Diagram 8](#_Toc166747250)

[2.2 Business Process Modeling Notation 9](#_Toc166747251)

[2.2.1 Proses Bisnis Registrasi 10](#_Toc166747252)

[2.2.2 Proses Bisnis Login 10](#_Toc166747253)

[2.2.3 Proses Bisnis Melihat Produk 11](#_Toc166747254)

[2.2.4 Proses Bisnis Melihat Detail Produk 12](#_Toc166747255)

[2.2.5 Proses Bisnis Menambah Produk ke Keranjang 12](#_Toc166747256)

[2.2.6 Proses Bisnis Melihat Halaman Keranjang 13](#_Toc166747257)

[2.2.7 Proses Bisnis Menghapus Produk dari Keranjang 14](#_Toc166747258)

[2.2.8 Proses Bisnis Melihat Produk 14](#_Toc166747259)

[2.2.9 Proses Bisnis Membuat Produk 15](#_Toc166747260)

[2.2.10 Proses Bisnis Mengedit Produk 16](#_Toc166747261)

[2.2.11 Proses Bisnis Menghapus Produk 16](#_Toc166747262)

[2.2.12 Proses Bisnis Melihat Kategori 17](#_Toc166747263)

[2.2.13 Proses Bisnis Membuat Kategori 18](#_Toc166747264)

[2.2.14 Proses Bisnis Mengedit Kategori 18](#_Toc166747265)

[2.2.15 Proses Bisnis Menghapus Kategori 19](#_Toc166747266)

[2.2.16 Proses Bisnis Melihat Galeri 20](#_Toc166747267)

[2.2.17 Proses Bisnis Membuat Galeri 20](#_Toc166747268)

[2.2.18 Proses Bisnis Mengedit Galeri 21](#_Toc166747269)

[2.2.19 Proses Bisnis Menghapus Galeri 21](#_Toc166747270)

[3 Tampilan Website 23](#_Toc166747271)

[4 Pengujian *Website* 35](#_Toc166747272)

# DAFTAR GAMBAR

[Gambar 1 Arsitektur Microservice 1](#_Toc166746783)

[Gambar 2 Perbedaan Arsitektur Microservice dan Monolith 3](#_Toc166746784)

[Gambar 3 Use case Diagram 8](#_Toc166746785)

[Gambar 4 BPMN Registrasi 10](#_Toc166746786)

[Gambar 5 BPMN Login 10](#_Toc166746787)

[Gambar 6 BPMN Melihat Produk 11](#_Toc166746788)

[Gambar 7 BPMN Melihat Detail Produk 12](#_Toc166746789)

[Gambar 8 Menambah Produk Ke Keranjang 13](#_Toc166746790)

[Gambar 9 Melihat Halaman Keranjang 13](#_Toc166746791)

[Gambar 10 Menghapus Produk dari Keranjang 14](#_Toc166746792)

[Gambar 11 Melihat Produk 15](#_Toc166746793)

[Gambar 12 Membuat Produk 15](#_Toc166746794)

[Gambar 13 Mengedit Produk 16](#_Toc166746795)

[Gambar 14 Menghapus Produk 17](#_Toc166746796)

[Gambar 15 Melihat Kategori 17](#_Toc166746797)

[Gambar 16 Membuat Kategori 18](#_Toc166746798)

[Gambar 17 Mengedit Kategori 19](#_Toc166746799)

[Gambar 18 Menghapus Kategori 19](#_Toc166746800)

[Gambar 19 Melihat Galeri 20](#_Toc166746801)

[Gambar 20 Membuat Galeri 21](#_Toc166746802)

[Gambar 21 Mengedit Galeri 21](#_Toc166746803)

[Gambar 22 Menghapus Galeri 22](#_Toc166746804)

[Gambar 23 Tampilan Register 23](#_Toc166746805)

[Gambar 24 Tampilan Login 24](#_Toc166746806)

[Gambar 25 Tampilan Melihat Produk 25](#_Toc166746807)

[Gambar 26 Tampilan Detail Produk 25](#_Toc166746808)

[Gambar 27 Tampilan Halaman Keranjang 26](#_Toc166746809)

[Gambar 28 Tampilan Menambah Produk 26](#_Toc166746810)

[Gambar 29 Tampilan Melihat Produk 27](#_Toc166746811)

[Gambar 30 Tampilan Mengedit Produk 28](#_Toc166746812)

[Gambar 31 Tampilan Menghapus Produk 29](#_Toc166746813)

[Gambar 32 Tampilan Menambah Kategori 29](#_Toc166746814)

[Gambar 33 Tampilan Melihat Kategori 30](#_Toc166746815)

[Gambar 34 Tampilan Mengedit Kategori 30](#_Toc166746816)

[Gambar 35 Tampilan Menghapus Kategori 31](#_Toc166746817)

[Gambar 36 Tampilan Menambah Galeri 32](#_Toc166746818)

[Gambar 37 Tampilan Melihat Galeri 32](#_Toc166746819)

[Gambar 38 Tampilan Mengedit Galeri 33](#_Toc166746820)

[Gambar 39 Tampilan Menghapus Galeri 34](#_Toc166746821)

[Gambar 40 Tampilan Logout 34](#_Toc166746822)

# DAFTAR TABEL

[Tabel 1 Karakteristik Pengguna Website 3](#_Toc166744997)

[Tabel 2 Pengujian Website Service Product Down 33](#_Toc166744998)

[Tabel 3 Pengujian Website Service Kategori Down 35](#_Toc166744999)

[Tabel 4 Pengujian Website Service Product Down 36](#_Toc166745000)

# Pendahuluan

Pada bab 1 berisi tentang deskripsi umum *website*, karakteristik pengguna *website*, dan fungsi pada *website*.

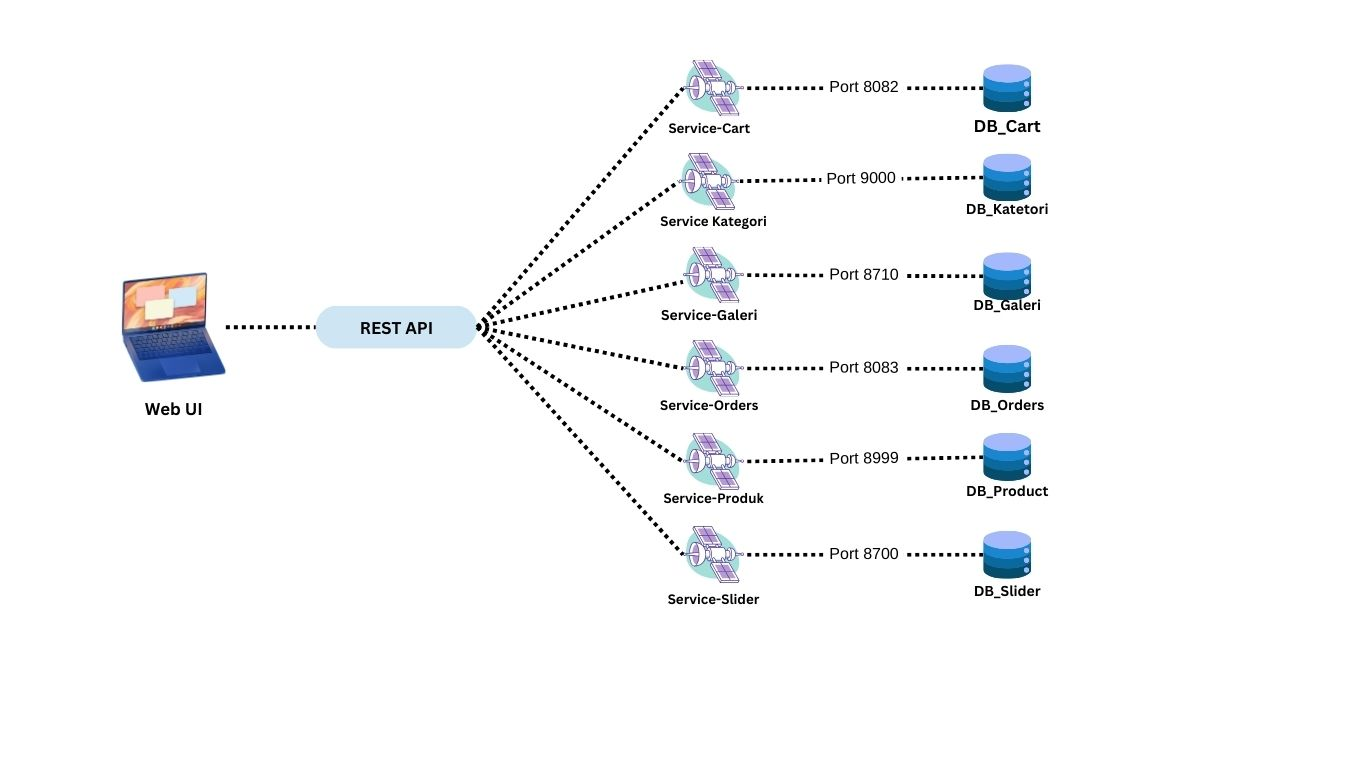
# Deskripsi Umum *Website*

Pembangunan *website* berbasis web ini ditujukan untuk sistem layanan jual beli *online* Toko Tambunan Pupuk*. Website* ini dapat digunakan oleh *admin* dan pelanggan untuk menjual dan membeli barang. Pada *website* ini, *admin* akan menambahkan produk yang dijual, sedangkan pelanggan dapat melakukan pembelian terhadap produk tersebut.

*Website* ini dibangun dengan menggunakan 2 bahasa pemrograman. Bagian *back-end* dan *front-end* dibangun dengan bahasa yang berbeda. Untuk bagian *back-end* digunakan Bahasa *Go* dan *framework Laravel* dan untuk bagian *front-end* menggunakan Bahasa *PHP,* khususnya *framework Laravel.*

*Website* ini menggunakan arsitektur *microservice.* Sesuai dengan namanya, arsitektur ini dirancang pada sebuah sistem untuk membagi *service* menjadi *service* yang lebih kecil. Dengan ini, maka setiap *service* akan memiliki *database* masing-masing. *Service* yang ada pada *website* adalah *Service\_cart, Service\_Category, Service\_Product, Service\_Galeri, Service\_Slider,* dan *Service\_Orders.* Setiap *service* akan berjalan secara *independent* sehingga satu *service* tidak akan memengaruhi *service* lainnya. Namun, meskipun berdiri sendiri, setiap *service* ini tetap saling terhubung antara satu dengan yang lainnya. Antarservice ini akan saling berkomunikasi dengan HTTP *request* berstandar REST API.

Setiap *database* pada masing-masing *service* memiliki *port* yang sama. Namun, *port* yang digunakan pada setiap *service* tersebut akan berbeda. Hal ini ditujukan sebagai pemisah antarservice sehingga saat satu *service* tersebut akan berbeda. Hal ini ditujukan sebagai pemisah antarservice sehingga saat satu *service* sedang tidak dapat dijalankan, maka *service* yang lainnya masih tetap dapat berjalan.



Gambar 1 Arsitektur Microservice

Arsitektur Microservice yang ditunjukkan pada gambar di atas menggambarkan bagaimana berbagai layanan (service) dalam *website* Toko Tambunan Pupuk verfungsi secara independent namun tetap terhubung melalui REST API. Berikut penjelasan lebih rinci tentang elemen-elemen arsutektur ini :

* + 1. Web UI:

Ini adalah antarmuka pengguna *(User Interface)* yang berinteraksi dengan pengguna *website.* Pengguna dapat mengakses berbagai fitur untuk membeli dan menjual barang melalui antarmuka ini.

* + 1. REST API:

*REST API* berfungsi sebagai penghubung antara *Web UI* dan layanan-layanan (service) di *back-end*. API ini menangani permintaan *HTTP* dari *Web UI* dan meneruskannya ke layanan yang sesuai. API ini juga mengembalikan respon dari layanan back-end ke *Web UI*.

* + 1. *Service Cart*:

Layanan ini menangani semua fungsi terkait keranjang belanja. Semua data dan logika terkait keranjang belanja disimpan dalam layanan ini. *Service-Cart* berkomunikasi dengan database *DB\_Cart* dan menggunakan port 8082.

* + 1. Service-Kategori :

Layanan ini mengelola kategori produk yang ada di website. Semua data kategori dan operasinya ditangani di sini. Service-Kategori berkomunikasi dengan database DB\_Kategori dan menggunakan port 9000.

* + 1. Service-Galeri :

Layanan ini bertanggung jawab untuk mengelola galeri gambar produk. Semua data gambar produk dan operasi terkait ditangani oleh layanan ini. Service-Galeri berkomunikasi dengan database DB\_Galeri dan menggunakan port 8710.

* + 1. Service- Orders:

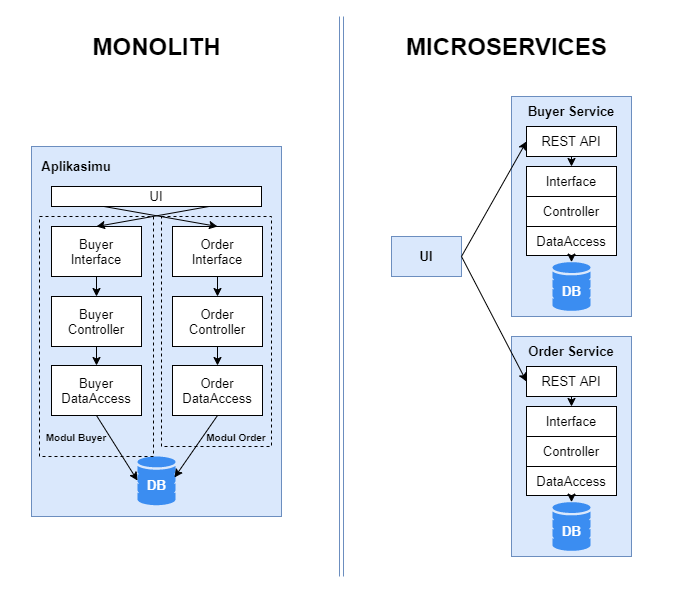
Layanan ini menangani semua fungsi terkait pesanan yang dibuat oleh pelanggan. Semua data dan logika terkait pesanan disimpan di sini. Service-Orders berkomunikasi dengan database DB\_Orders dan menggunakan port 8083.

* + 1. Service -Produk:

Layanan ini mengelola data produk yang dijual di *website*. Semua informasi produk dan operasi terkait ditangani oleh layanan ini. Service-Produk berkomunikasi dengan database *DB\_Product* dan menggunakan port 8999.

# Perbedaan *Monolith* dan *Microservice*

**Monolith** adalah arsitektur dimana keseluruhan kode akan dikompilasi menjadi satu aplikasi (biasanya menjadi satu *binary* atau *artifact)* dimana aplikasi tersebut menjalankan seluruh proses yang dibutuhkan. Kebutuhan akan komunikasi kepada aplikasi atau service lain bisa jadi tidak ada, karena aplikasi ini telah mencakup seluruh kode yang dibutuhkan, sedangkan **Microservice** adalah arsitektur yang menekankan kepada banyak service atau aplikasi yang tidak terkait erat satu dengan yang lainnya *(loosely coupled).* Service ini memiliki tujuan yang sangat terfokus dan memiliki metode komunikasi antar service yang tidak berat.



Gambar 2 Perbedaan Arsitektur Microservice dan Monolith

Pada arsitektur **monolitik**, seluruh komponen aplikasi seperti antarmuka pengguna (UI) terintegrasi dalam satu kesatuan aplikasi dan berbagi satu basis data tunggal. Semua bagian saling bergantung satu sama lain, sehingga jika ada satu bagian yang gagal, bisa mempengaruhi keseluruhan sistem.

Sementara pada arsitektur **microservices**, aplikasi dipecah menjadi beberapa layanan kecil yang independen, masing-masing dengan fungsi spesifik. Setiap layanan memiliki antarmuka, controller, dan akses data sendiri serta dapat memiliki basis data terpisah. Layanan-layanan ini berkomunikasi satu sama lain melalui API (REST API). Pendekatan ini memungkinkan fleksibilitas yang lebih besar, skalabilitas, dan ketahanan yang lebih baik, karena kegagalan pada satu layanan tidak akan mempengaruhi layanan lainnya.

# Karakteristik Pengguna Website

Pada *website* Toko Tambunan Pupuk ini, terdapat 2 kategori. Kategori pengguna pada *website* adalah *admin* dan pelanggan. Karakteristik pengguna dilampiarkan pada Tabel 1.

Tabel 1 Karakteristik Pengguna Website

|  |  |  |
| --- | --- | --- |
| **Kategori Pengguna** | **Fungsi** | **Hak Akses ke Aplikasi** |
| *Admin* | Mengelola data dalam *website* | 1. Akses ke menu registrasi 2. Akses ke menu *login* 3. Akses ke fungsi penambahan, pengubahan, dan penghapusan data produk 4. Akses untuk keluar sistem melalui *logout* |
| Pelanggan | Melakukan pemesanan sampai pembayaranmelalui *website* | 1. Akses ke menu registrasi 2. Akses ke menu *login* 3. Akses ke pemesanan produk 4. Akses untuk keluar sistem melalui   *logout* |

# Fungsi pada *Website*

Adapun fungsi yang terdapat pada *website* Toko Tambunan Pupuk ini, yaitu:

* + 1. Fungsi registrasi

Fungsi registrasi digunakan oleh *user* untuk mendaftarkan akunnya sebagai *admin* atau

pelangganpada *website*.

* + 1. Fungsi *login*

Fungsi *login* digunakan oleh *admin* dan pelangganuntuk dapat masuk dan mengakses *website.*

* + 1. Fungsi Melihat Produk

Fungsi melihat produk digunakan oleh pelangganuntuk melihat produk yang terdapat pada *website*.

* + 1. Fungsi Melihat Detail Produk

Fungsi melihat detail produk digunakan oleh pelangganuntuk melihat detail dari produk yang tersedia pada *website*.

* + 1. Fungsi Menambah Produk ke Keranjang

Fungsi menambah produk ke keranjang digunakan oleh pelanggan untuk menambahkan produk yang akan dibeli ke dalam keranjang yang terdapat pada sistem.

* + 1. Fungsi Melihat Halaman Keranjang

Fungsi melihat keranjang digunakan oleh pelanggan untuk melihat pesanan apa saja yang terdapat di keranjang.

* + 1. Fungsi Menghapus Produk dari Keranjang

Fungsi menghapus produk dari keranjang digunakan oleh pelanggan untuk menghapus produk yang dibatalkan untuk dibeli dari keranjang.

* + 1. Fungsi Melihat Produk

Fungsi melihat produk ini digunakan oleh *admin* untuk melihat produk yang ada pada sistem.

* + 1. Fungsi Membuat Produk

Fungsi membuat produk ini digunakan oleh *admin* untuk membuat produk baru yang ingin ditambahkan ke dalam sistem.

* + 1. Fungsi Mengedit Produk

Fungsi mengedit produk ini digunakan oleh *admin* untuk mengedit produk yang sudah ada pada sistem.

* + 1. Fungsi Menghapus Produk

Fungsi menghapus produk ini digunakan oleh *admin* untuk menghapus produk yang sudah tidak tersedia.

* + 1. Fungsi Melihat Kategori

Fungsi melihat kategori ini digunakan oleh *admin* untuk melihat kategori yang ada pada sistem.

* + 1. Fungsi Membuat Kategori

Fungsi membuat kategori ini digunakan oleh *admin* untuk membuat kategori baru yang ingin ditambahkan ke dalam sistem.

* + 1. Fungsi Mengedit Kategori

Fungsi mengedit kategori ini digunakan oleh *admin* untuk mengedit kategori yang sebelumnya sudah ada pada sistem.

* + 1. Fungsi Menghapus Kategori

Fungsi menghapus kategori ini digunakan oleh *admin* untuk menghapus kategori yang sudah tidak tersedia dalam sistem.

* + 1. Fungsi Melihat Galeri

Fungsi melihat galeri ini digunakan oleh *admin* untuk melihat galeri yang ada pada sistem.

* + 1. Fungsi Membuat Galeri

Fungsi membuat galeri ini digunakan oleh *admin* untuk membuat galeri baru yang ingin ditambahkan ke dalam sistem.

* + 1. Fungsi Mengedit Galeri

Fungsi mengedit galeri ini digunakan oleh *admin* untuk mengedit galeri yang sebelumnya sudah ada pada sistem.

* + 1. Fungsi Menghapus Galeri

Fungsi menghapus galeri ini digunakan oleh *admin* untuk menghapus galeri yang sudah tidak tersedia dalam sistem.

* + 1. Fungsi Melihat Daftar Pesanan

Fungsi melihat daftar pesanan ini digunakan oleh *admin* untuk melihat daftar pesanan pelanggan yang sudah masuk ke dalam sistem.

* + 1. Fungsi *logout*

Fungsi *logout* digunakan oleh *seller* dan *customer* untuk dapat keluar dari aplikasi*.*

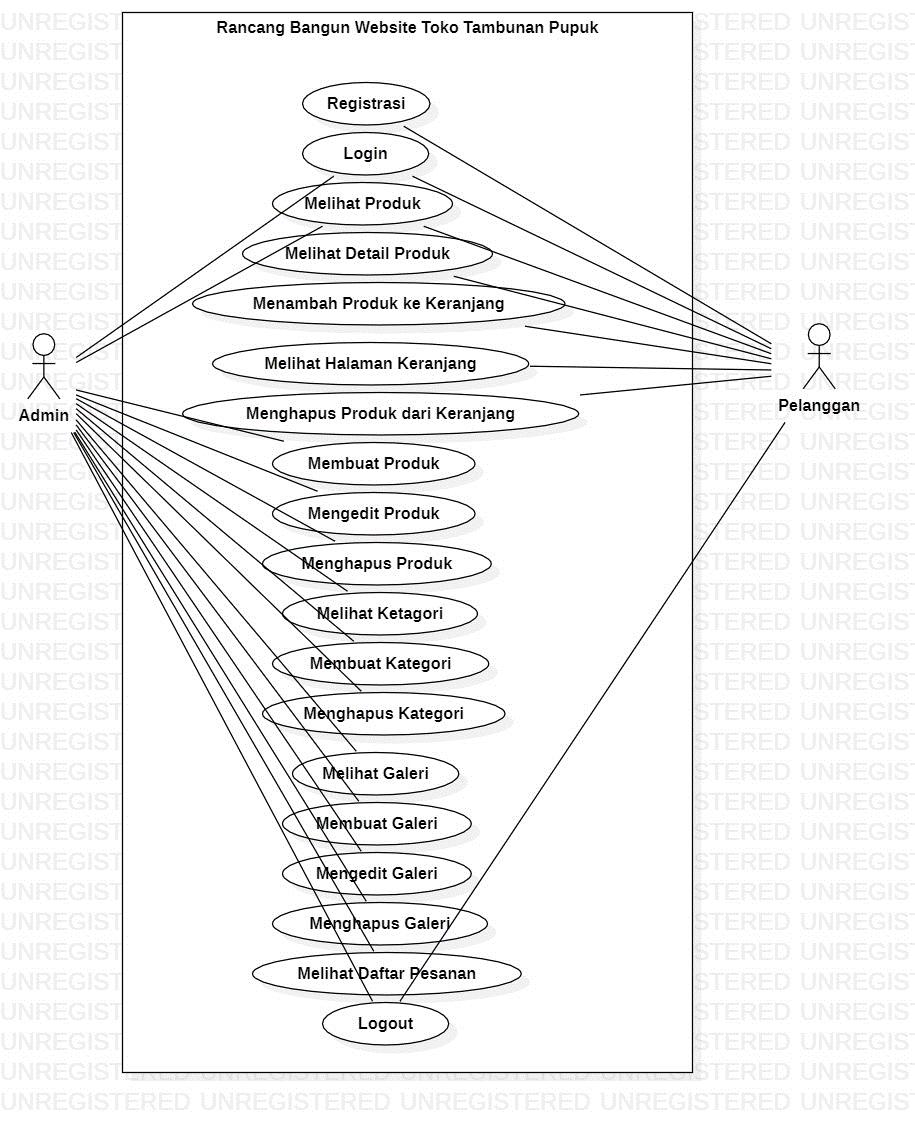
# Desain Rancangan *Website*

Pada bab 2 berisi tentang desain rancangan *website*, yaitu *use case* diagram dan *business process modeling notation* (BPMN).

# Use Case Diagram

*Use case* diagram pada Gambar 2 memuat setiap fungsi yang dapat dijalankan pada *website*. *Website* Toko Tambunan Pupuk ini dapat digunakan oleh 2 *role,* yaitu sebagai *admin* dan pelanggan. Kedua *role* harus melakukan login terlebih dahulu agar dapatmengakses halaman website. Yang menjadi pembeda dari kedua *role* tersebut terletak pada layanan yang dapat diakses keduanya setelah *login* ke dalam *website*. *Admin* dapat melakukan CRUD product, kategori, galeri dan melihat daftar pesanan, sedangkan pelanggan dapat melihat produk, melihat detail produk, menambah produk ke keranjang, melihat halaman keranjang, dan menghapus produk dari keranjang.

*Use case* diagram *website* Toko Tambunan Pupuk terlampir pada Gambar 2.



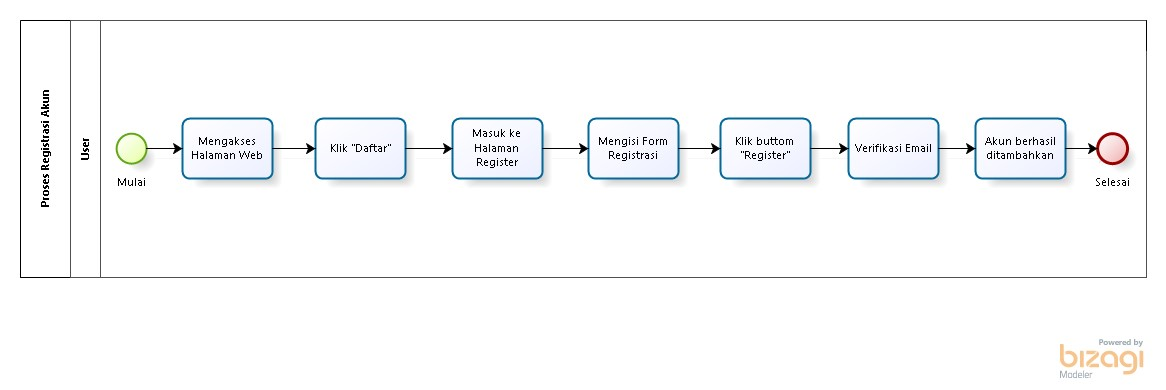
Gambar 3 Use case Diagram

# Business Process Modeling Notation

Setiap fungsi pada *website* dapat dijalankan dengan proses yang tertera pada setiap *business process modeling notation* berikut.

# Proses Bisnis Registrasi

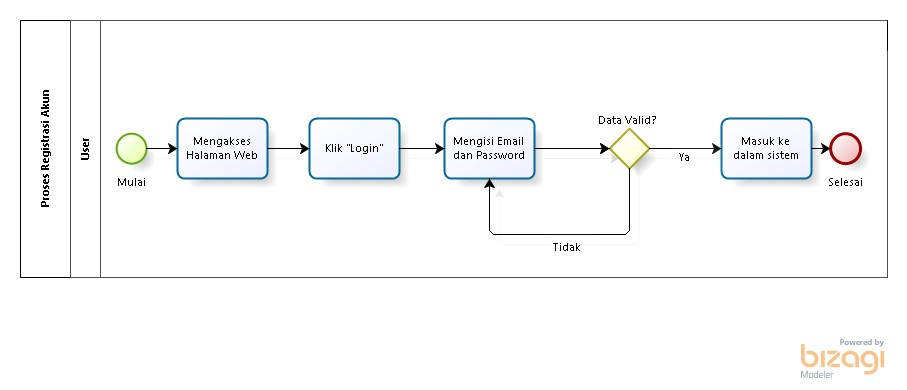
Fungsi registrasi digunakan oleh *user* untuk mendaftarkan akunnya sebagai *admin* atau pelanggan*.* pada *website*. *User* dapat mendaftarkan akunnya dengan mengisi data isian pada formulir lalu mengirim data tersebut. Jika registrasi berhasil dilakukan, maka akun *user* berhasil tertambah. Selanjutnya, *user* akan dapat *login* sesuai dengan *role* yang mereka daftarkan sebelumnya. Proses bisnis registrasi pada *website* terlampir pada Gambar 2.



Gambar 4 BPMN Registrasi

# Proses Bisnis Login

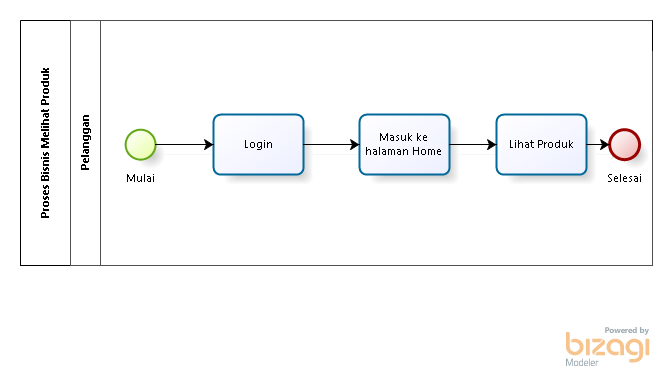
Fungsi *login* digunakan oleh *user* (*admin* dan pelanggan) untuk dapat masuk dan mengakses *website. User* diminta untuk memasukkan email dan password yang telah terdaftar sebelumnya. Jika data yang dimasukkan sudah valid, maka *user* akan berhasil masuk ke dalam *website*. Proses bisnis *login* pada *website* terlampir pada Gambar 3.



Gambar 5 BPMN Login

# Proses Bisnis Melihat Produk

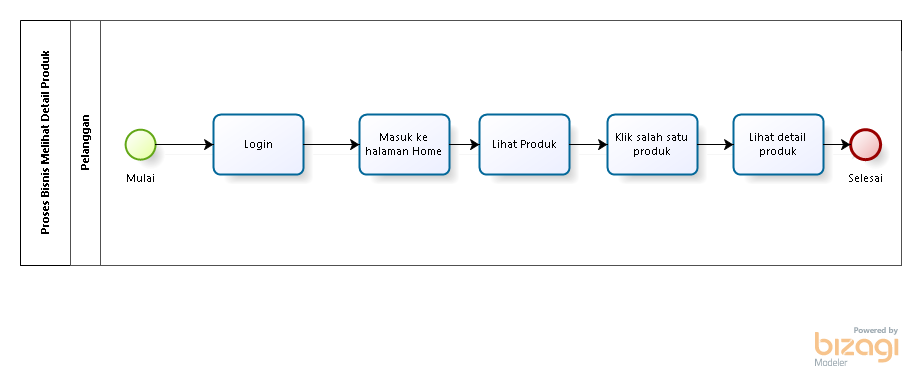
Fungsi mengelola produk digunakan oleh pelangganuntuk melihat produk. Dimulai dari pelanggan yang melakukan *login,* kemudian masuk ke halaman *Home.* Di halaman *home* akan menampilkan produk dan pelanggan dapat melihat produk.



Gambar 6 BPMN Melihat Produk

# Proses Bisnis Melihat Detail Produk

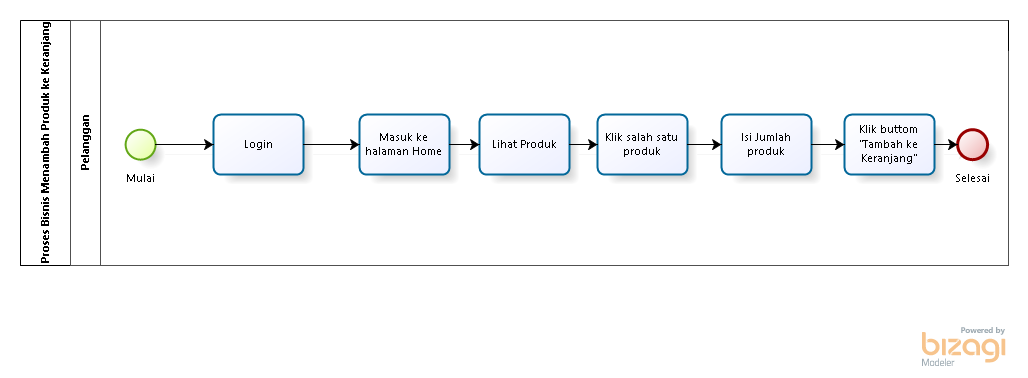
Fungsi melihat detail produk digunakan oleh pelanggan untuk melihat detail dari sebuah produk yang tersedia dalam sistem. Dimulai dari pelanggan yang melakukan *login,* kemudian masuk ke halaman *home.* Di halaman *home* terdapat tampilan produk. Untuk melihat detail dari produk, klik salah satu produk, kemudian akan menampilkan detail produk. Proses bisnis melihat produk pada *website* terlampir pada Gambar 5.



Gambar 7 BPMN Melihat Detail Produk

# Proses Bisnis Menambah Produk ke Keranjang

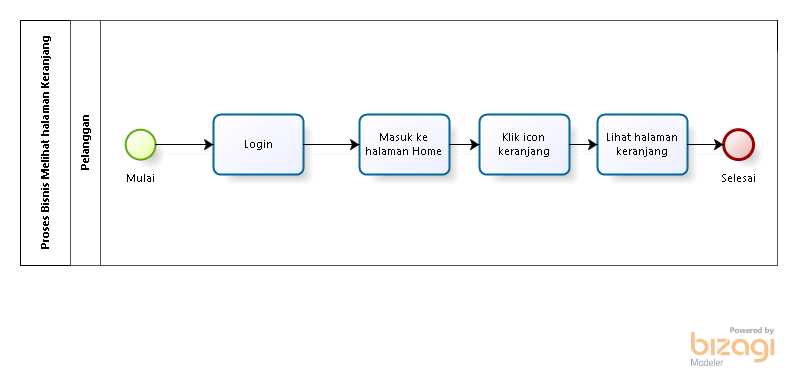
Fungsi menambah produk ke keranjang digunakan pelanggan untuk dapat menambahkan sebuah produk ke dalam keranjang dalam sistem. Dimulai dari pelanggan yang melakukan *login,* kemudian masuk ke halaman *home.* Di halaman *home* akan menampilkan produk. Lalu, pilih salah satu produk dan isi jumlah produk yang ingin dimasukkan ke dalam keranjang. Kemudian, klik *button* “Tambah ke Keranjang”. Proses bisnis menambah produk ke keranjang pada *website* terlampir pada Gambar 6.



Gambar 8 Menambah Produk Ke Keranjang

# Proses Bisnis Melihat Halaman Keranjang

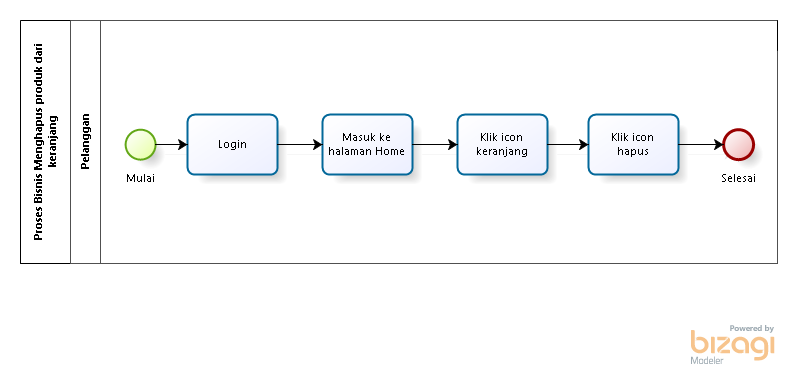
Fungsi melihat halaman keranjang digunakan oleh pelanggan. Dimulai dengan pelanggan yang melakukan *login,* kemudian masuk ke halaman *home.* Pada halaman *home,* klik *icon* keranjang, dan akan menampilkan halaman keranjang. Proses bisnis melihat halaman keranjangpada *website* terlampir pada Gambar .7



Gambar 9 Melihat Halaman Keranjang

# Proses Bisnis Menghapus Produk dari Keranjang

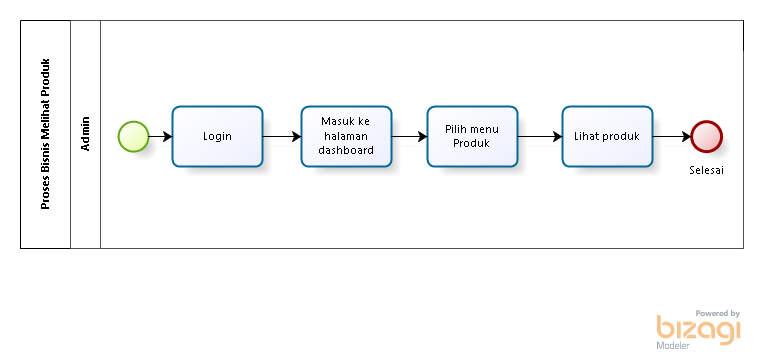
Fungsi menghapus produk dari keranjang digunakan oleh pelanggan ketika ingin menghapus produk yang sudah dimasukkan ke keranjang. Dimulai dari pelanggan yang melakukan *login,* kemudian masuk ke halaman *home.* Di halaman *home,* klik *icon* keranjang, kemudian klik *icon* hapus. Proses bisnis menghapus produk dari keranjangpada *website* terlampir pada Gambar 8.



Gambar 10 Menghapus Produk dari Keranjang

# Proses Bisnis Melihat Produk

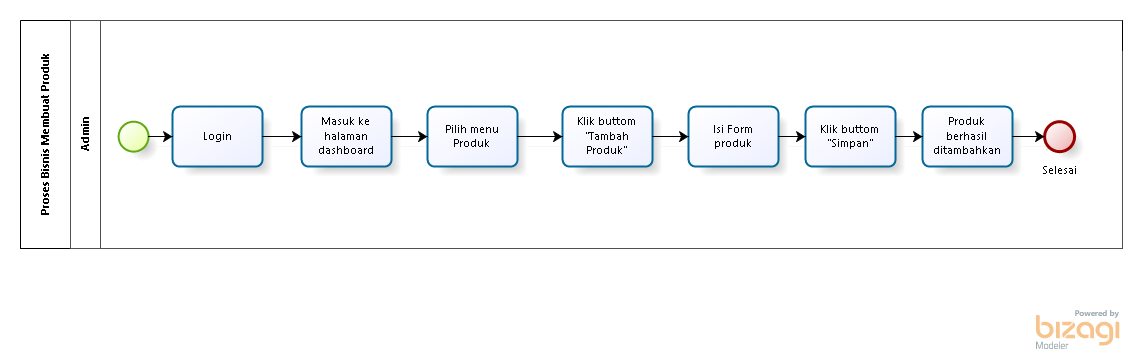
Fungsi melihat produk digunakan oleh *admin* untuk melihat produk yang ada pada sistem. Proses bisnis melihat produkpada *website* terlampir pada Gambar 9.



Gambar 11 Melihat Produk

# Proses Bisnis Membuat Produk

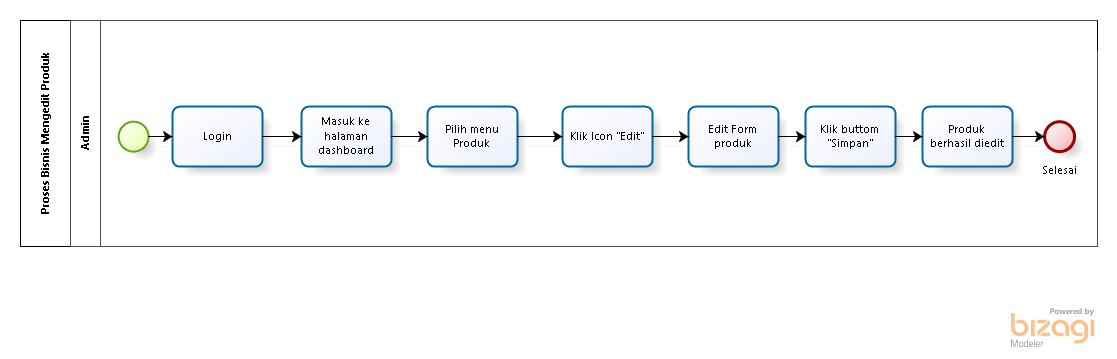
Fungsi membuat produk digunakan oleh *admin* untuk membuat produk baru ke dalam sistem. Dimulai dari *admin* yang melakukan *login* kemudian masuk ke halaman *dashboard.* Setelah itu, pilih menu produk, dan klik *button* “Tambah Produk”. Setelah itu, isi form produk, dan klik *button* “Simpan”. Setelah itu produk berhasil ditambahkan. Proses bisnis membuat produkpada *website* terlampir pada Gambar 10.



Gambar 12 Membuat Produk

# Proses Bisnis Mengedit Produk

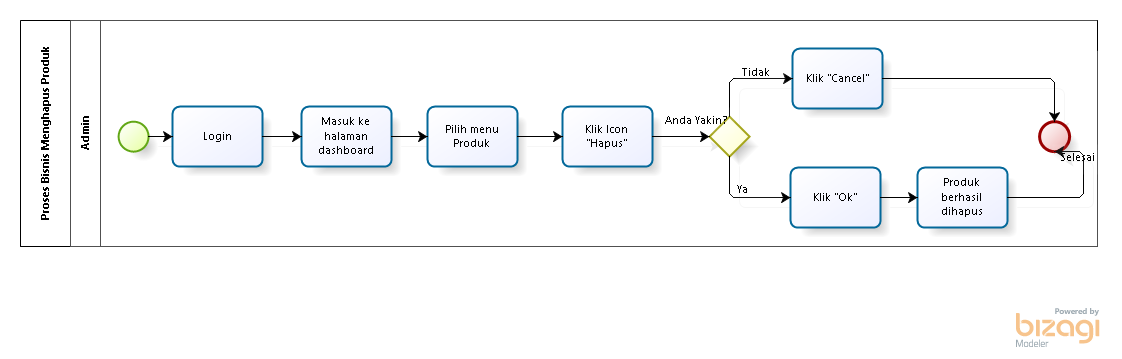
Fungsi mengedit produk digunakan oleh *admin* untuk mengedit produk yang sudah ada sebelumnya pada sistem. Dimulai dengan *admin* yang melakukan *login,* kemudian masuk ke halaman *dashboard* dan pilih menu produk. Pada menu produk, klik *icon* “Edit”, lalu edit form produk yang ingin di-*update.* Setelah itu, klik *button* “simpan” dan produk berhasil diedit. Proses bisnis mengedit produkpada *website* terlampir pada Gambar 11.



Gambar 13 Mengedit Produk

# Proses Bisnis Menghapus Produk

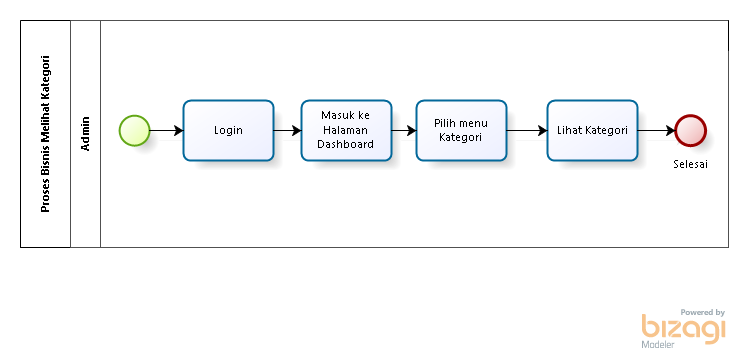
Fungsi menghapus produk ini digunakan oleh *admin* untuk menghapus produk yang sudah tidak tersedia lagi pada sistem. Dimulai dengan *admin* yang melakukan *login,* kemudian masuk ke halaman *dashboard.* Pada halaman *dashboard,* pilih menu produk. Setelah memilih menu produk, klik *icon* “Hapus”. Kemudian sistem akan mengkonfirmasi, apakah anda yakin untuk menghapus produk tersebut. Jika yakin, klik *button* “OK”, lalu produk berhasil dihapus.  Proses bisnis menghapus produkpada *website* terlampir pada Gambar 12.



Gambar 14 Menghapus Produk

# Proses Bisnis Melihat Kategori

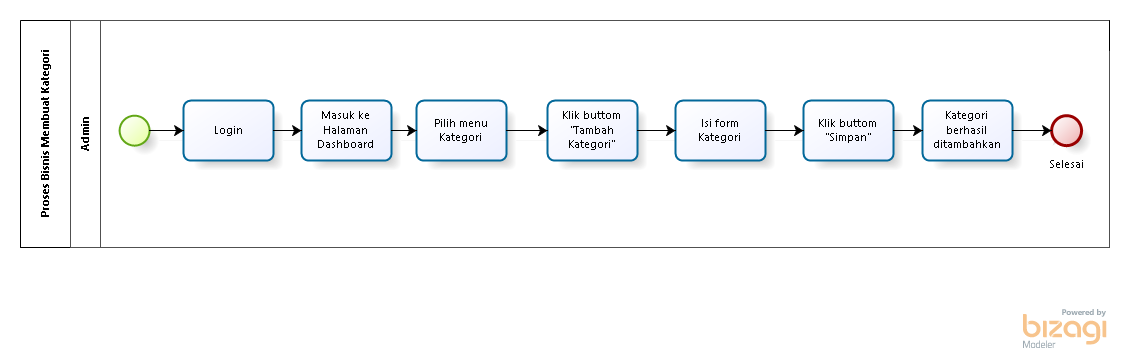
Fungsi melihat kategori digunakan oleh *admin* untuk melihat kategori yang ada pada sistem. Proses bisnis melihat kategoripada *website* terlampir pada Gambar 13.



Gambar 15 Melihat Kategori

# Proses Bisnis Membuat Kategori

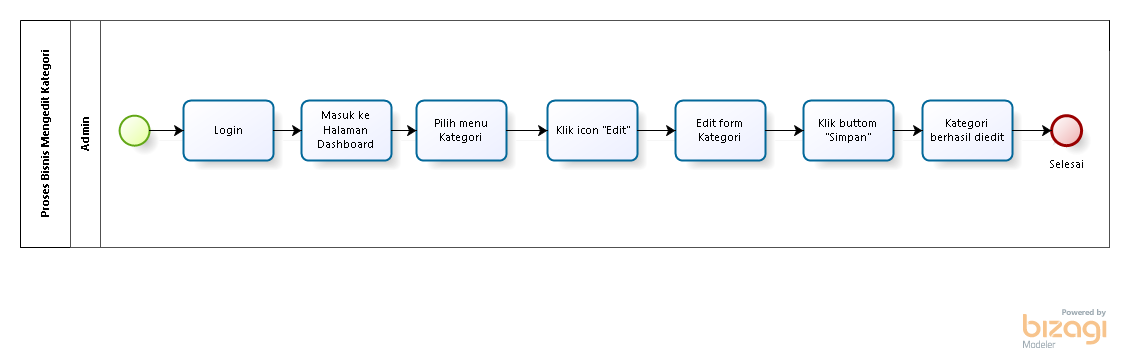
Fungsi membuat kategori digunakan oleh *admin* untuk membuat kategori baru ke dalam sistem. Dimulai dari *admin* yang melakukan *login* kemudian masuk ke halaman *dashboard.* Setelah itu, pilih menu kategori, dan klik *button* “Tambah Kategori”. Setelah itu, isi form kategori, dan klik *button* “Simpan”. Setelah itu kategori berhasil ditambahkan. Proses bisnis membuat kategoripada *website* terlampir pada Gambar 14.



Gambar 16 Membuat Kategori

# Proses Bisnis Mengedit Kategori

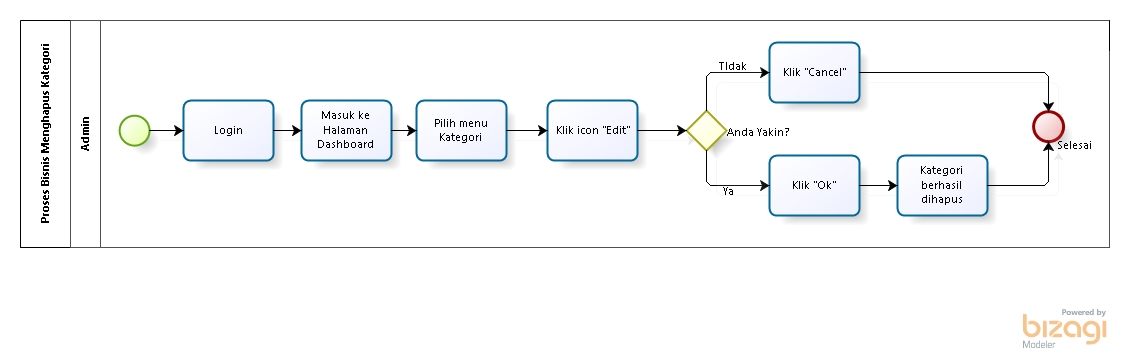
Fungsi mengedit kategori digunakan oleh *admin* untuk mengedit kategori yang sudah ada sebelumnya pada sistem. Dimulai dengan *admin* yang melakukan *login,* kemudian masuk ke halaman *dashboard* dan pilih menu kategori. Pada menu kategori, klik *icon* “Edit”, lalu edit form kategori yang ingin di-*update.* Setelah itu, klik *button* “simpan” dan kategori berhasil diedit. Proses bisnis mengedit kategoripada *website* terlampir pada Gambar 15.



Gambar 17 Mengedit Kategori

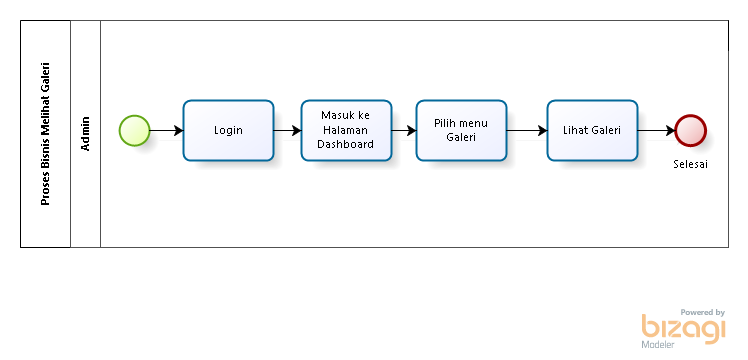
# Proses Bisnis Menghapus Kategori

Fungsi menghapus kategori ini digunakan oleh *admin* untuk menghapus kategori yang sudah tidak tersedia lagi pada sistem. Dimulai dengan *admin* yang melakukan *login,* kemudian masuk ke halaman *dashboard.* Pada halaman *dashboard,* pilih menu kategori. Setelah memilih menu kategori, klik *icon* “Hapus”. Kemudian sistem akan mengkonfirmasi, apakah anda yakin untuk menghapus kategori tersebut. Jika yakin, klik *button* “OK”, lalu kategori berhasil dihapus.  Proses bisnis menghapus kategoripada *website* terlampir pada Gambar 16.



Gambar 18 Menghapus Kategori

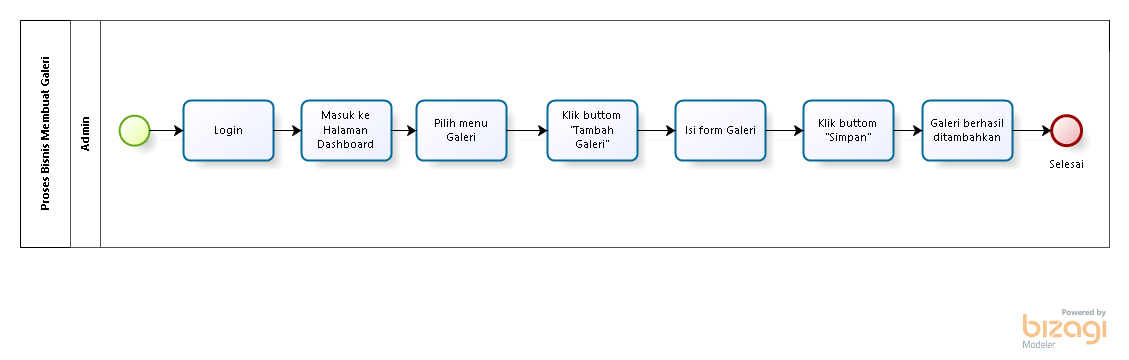
# Proses Bisnis Melihat Galeri

Fungsi melihat galeri digunakan oleh *admin* untuk melihat galeri yang ada pada sistem. Proses bisnis melihat galeripada *website* terlampir pada Gambar 

Gambar 19 Melihat Galeri

# Proses Bisnis Membuat Galeri

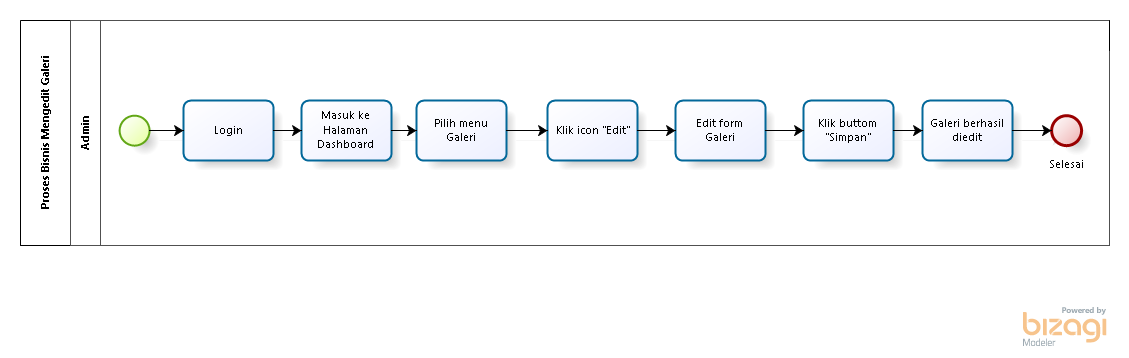
Fungsi membuat galeri digunakan oleh *admin* untuk membuat galeri baru ke dalam sistem. Dimulai dari *admin* yang melakukan *login* kemudian masuk ke halaman *dashboard.* Setelah itu, pilih menu galeri, dan klik *button* “Tambah Galeri”. Setelah itu, isi form galeri, dan klik *button* “Simpan”. Setelah itu kategori berhasil ditambahkan. Proses bisnis membuat galeri pada *website* terlampir pada Gambar 18.



Gambar 20 Membuat Galeri

# Proses Bisnis Mengedit Galeri

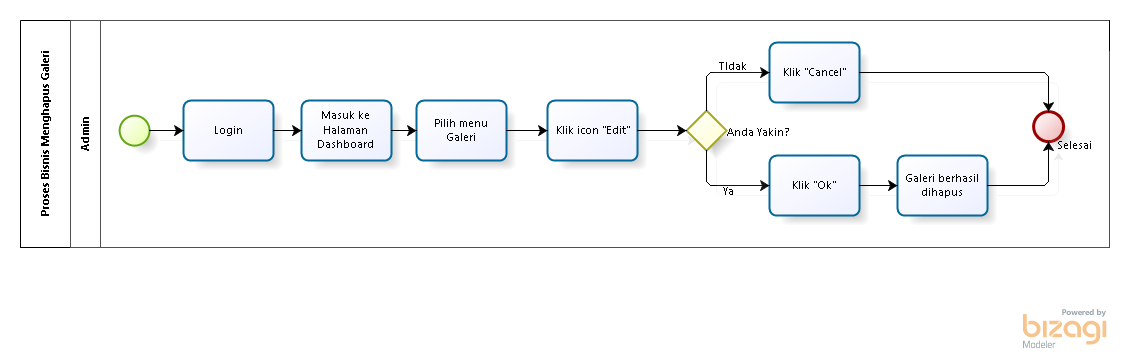
Fungsi mengedit galeri digunakan oleh *admin* untuk mengedit galeri yang sudah ada sebelumnya pada sistem. Dimulai dengan *admin* yang melakukan *login,* kemudian masuk ke halaman *dashboard* dan pilih menu galeri. Pada menu galeri, klik *icon* “Edit”, lalu edit form galeri yang ingin di-*update.* Setelah itu, klik *button* “simpan” dan galeri berhasil diedit. Proses bisnis mengedit galeri pada *website* terlampir pada Gambar 19.



Gambar 21 Mengedit Galeri

# Proses Bisnis Menghapus Galeri

Fungsi menghapus galeri ini digunakan oleh *admin* untuk menghapus galeri yang sudah tidak tersedia lagi pada sistem. Dimulai dengan *admin* yang melakukan *login,* kemudian masuk ke halaman *dashboard.* Pada halaman *dashboard,* pilih menu galeri. Setelah memilih menu galeri, klik *icon* “Hapus”. Kemudian sistem akan mengkonfirmasi, apakah anda yakin untuk menghapus galeri tersebut. Jika yakin, klik *button* “OK”, lalu galeri berhasil dihapus.  Proses bisnis menghapus galeri pada *website* terlampir pada Gambar 20.

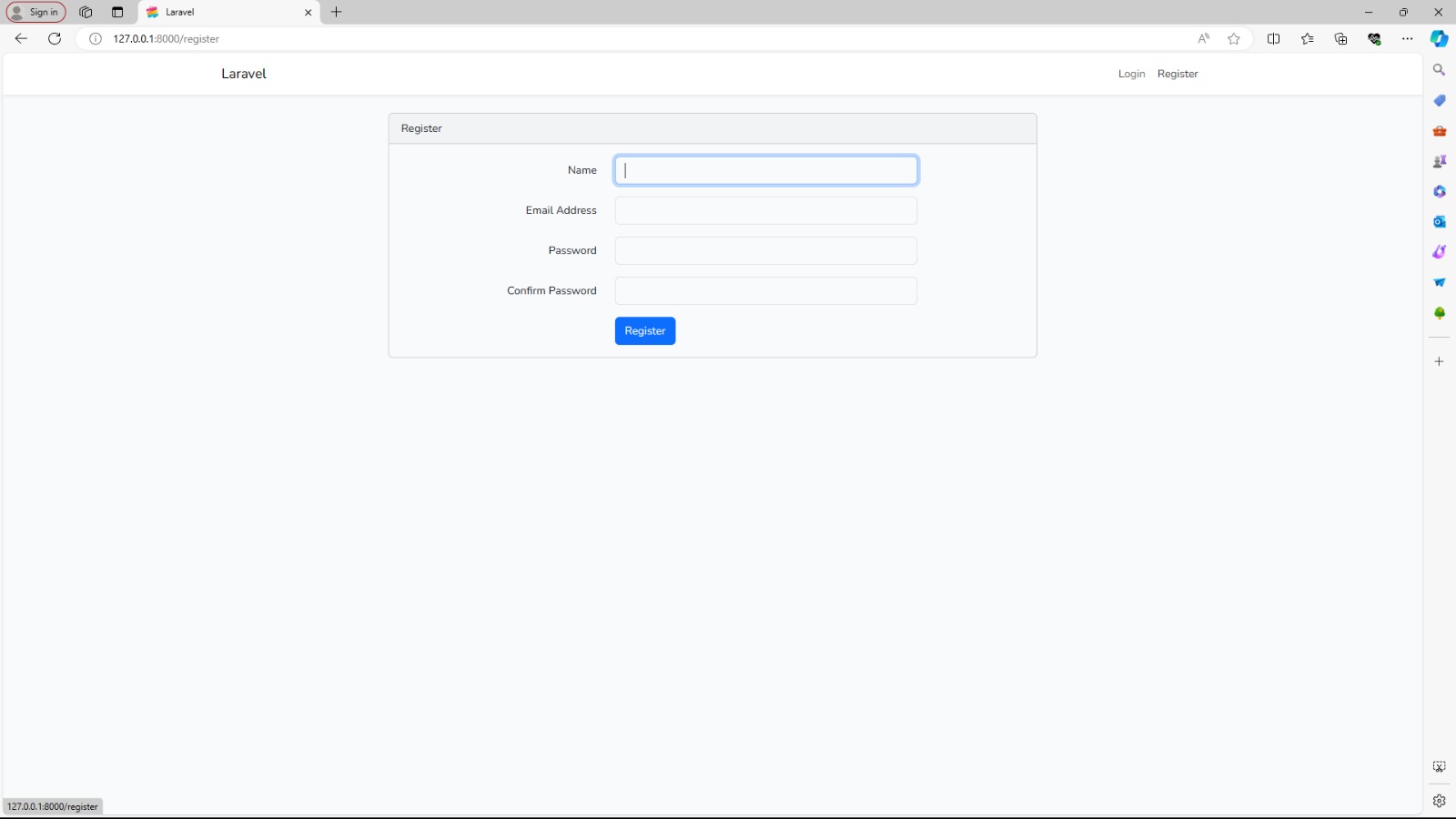


Gambar 22 Menghapus Galeri

# Tampilan Website

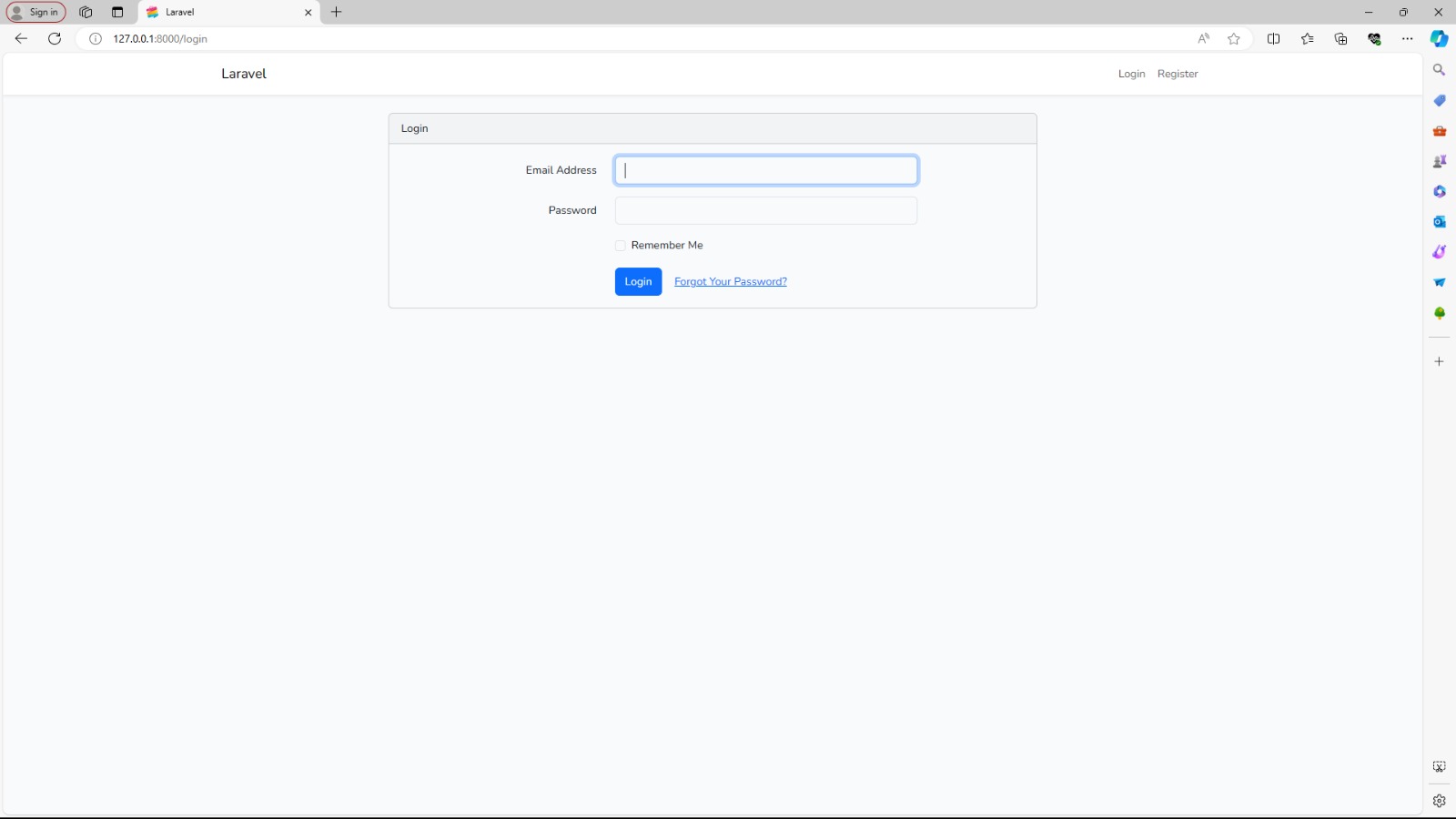
Pada bab 3 berisi tentang tampilan website yang telah dibangun.

* + 1. Tampilan Registrasi



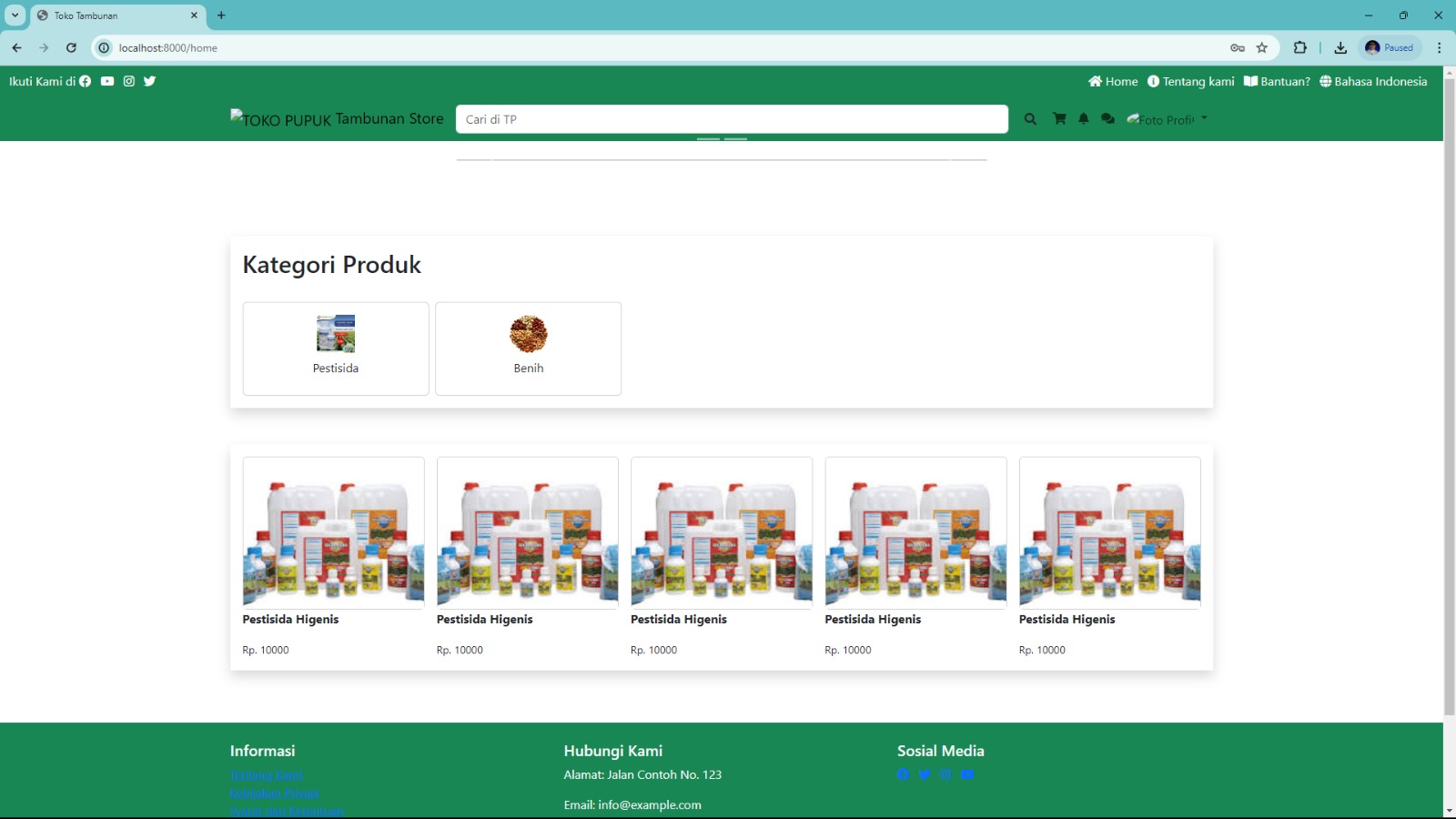
Gambar 23 Tampilan Register

* + 1. Tampilan *Login*



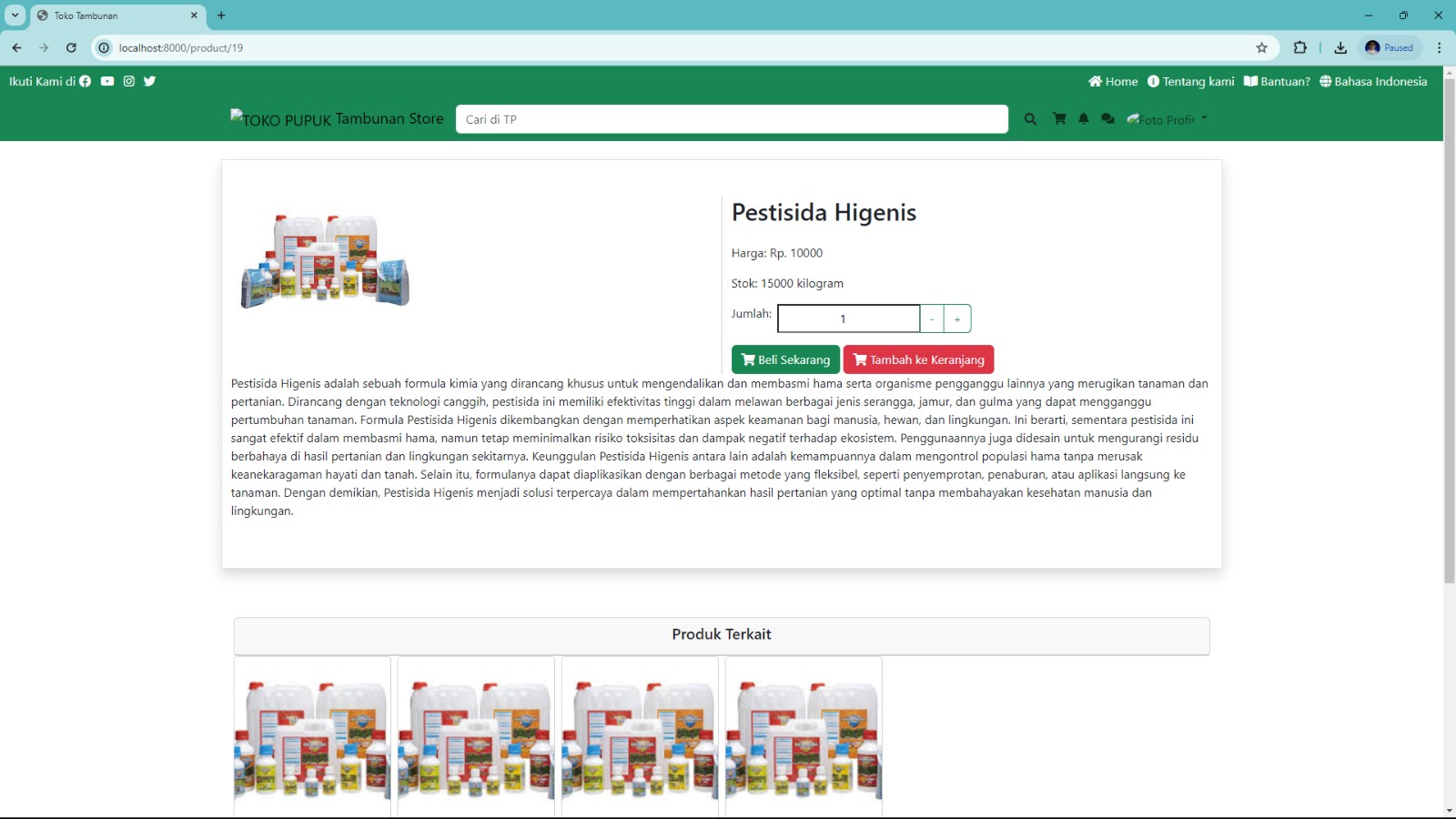
Gambar 24 Tampilan Login

* + 1. Tampilan Melihat Produk



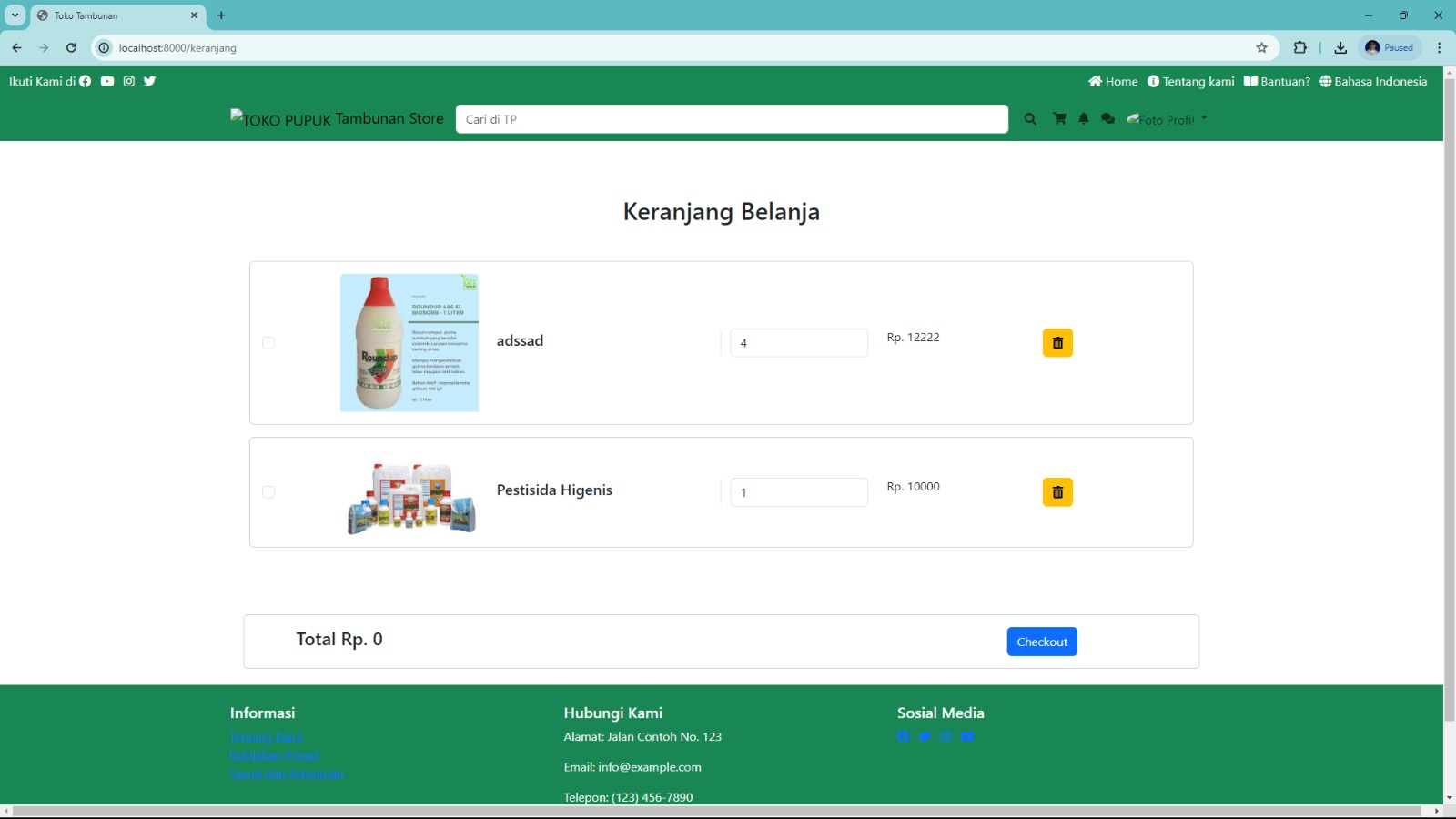
Gambar 25 Tampilan Melihat Produk

* + 1. Tampilan Detail Produk

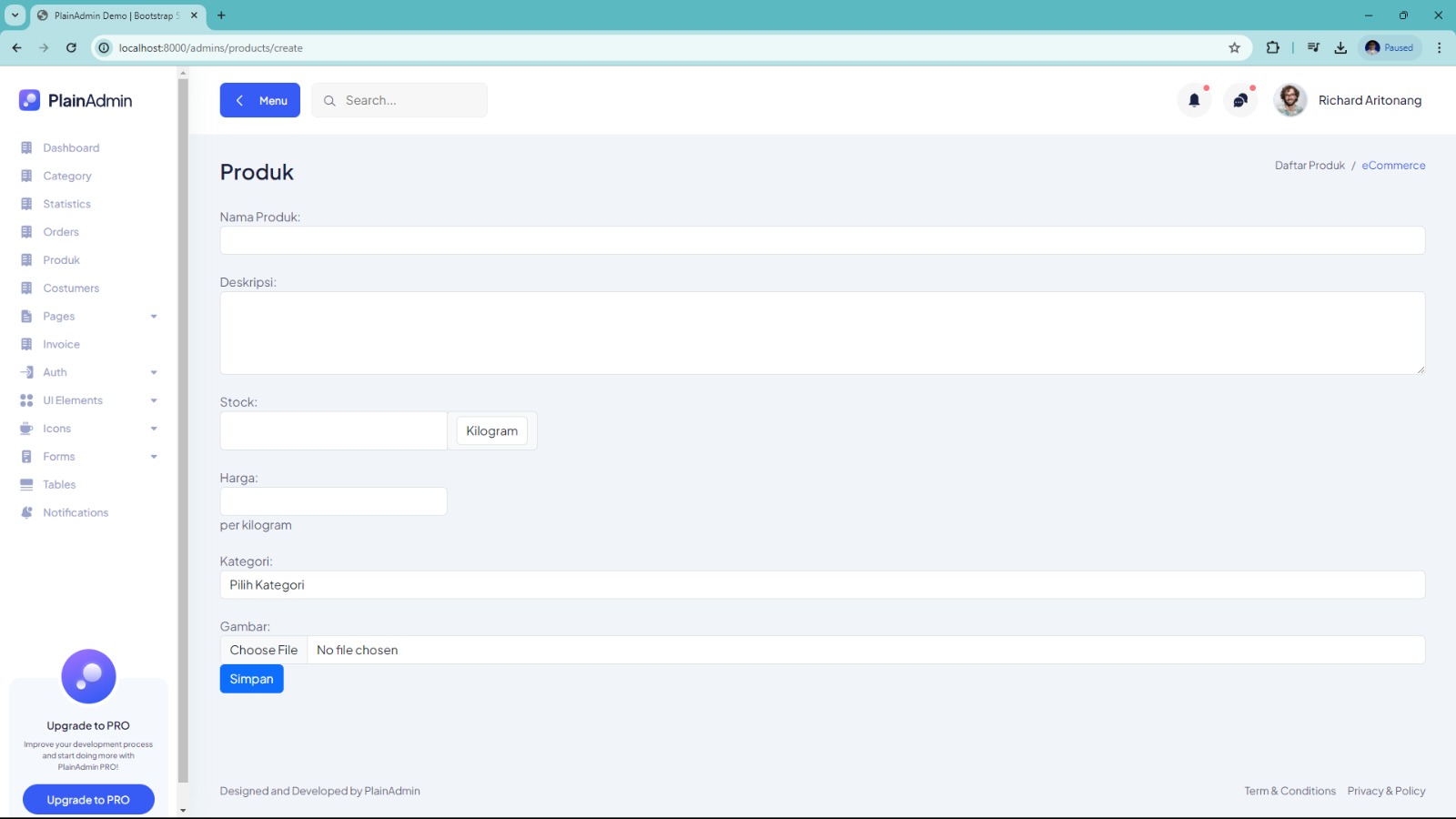


Gambar 26 Tampilan Detail Produk

* + 1. Tampilan Halaman Keranjang

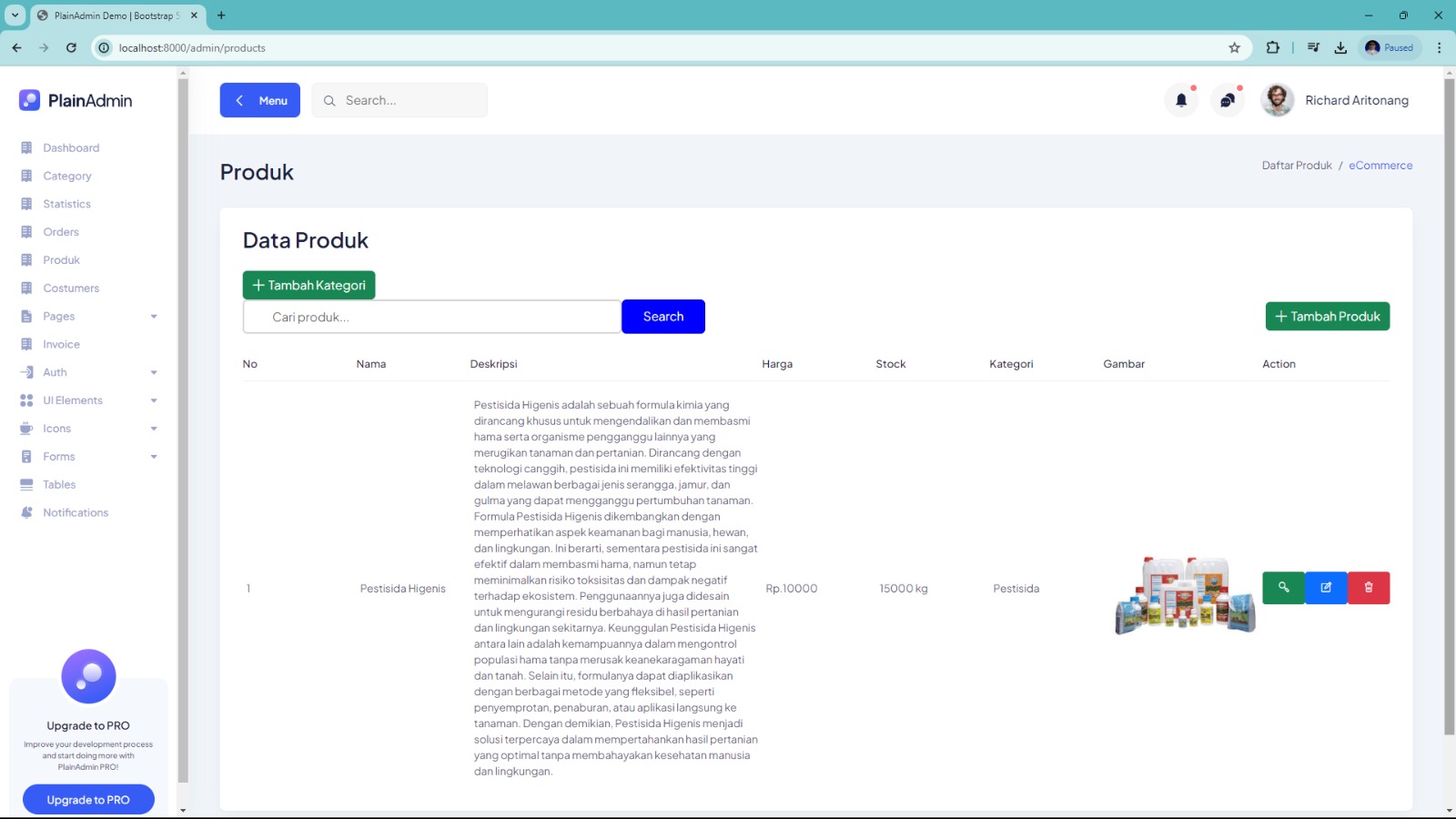


Gambar 27 Tampilan Halaman Keranjang



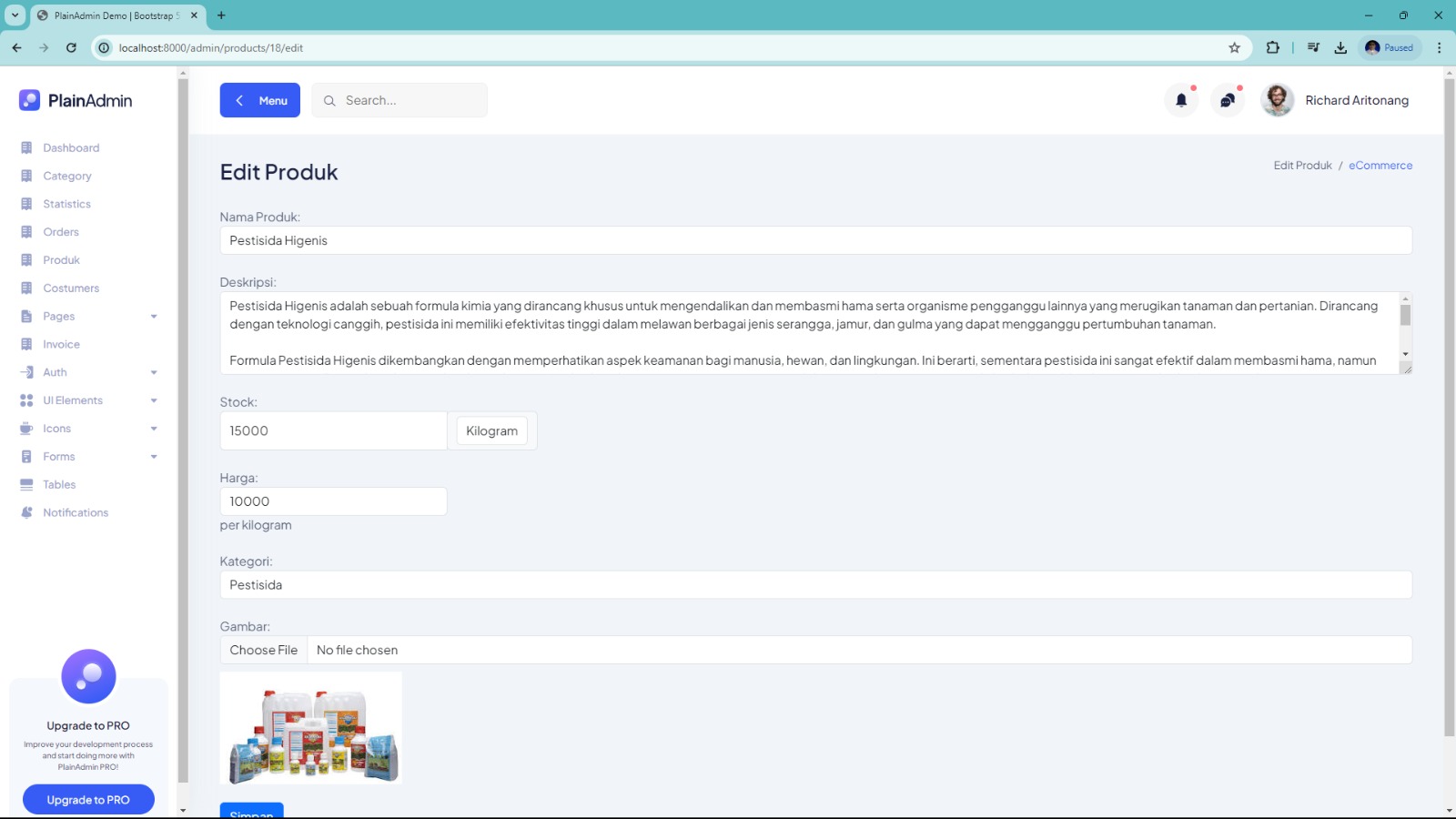
Gambar 28 Tampilan Menambah Produk

* + 1. Tampilan Melihat Produk



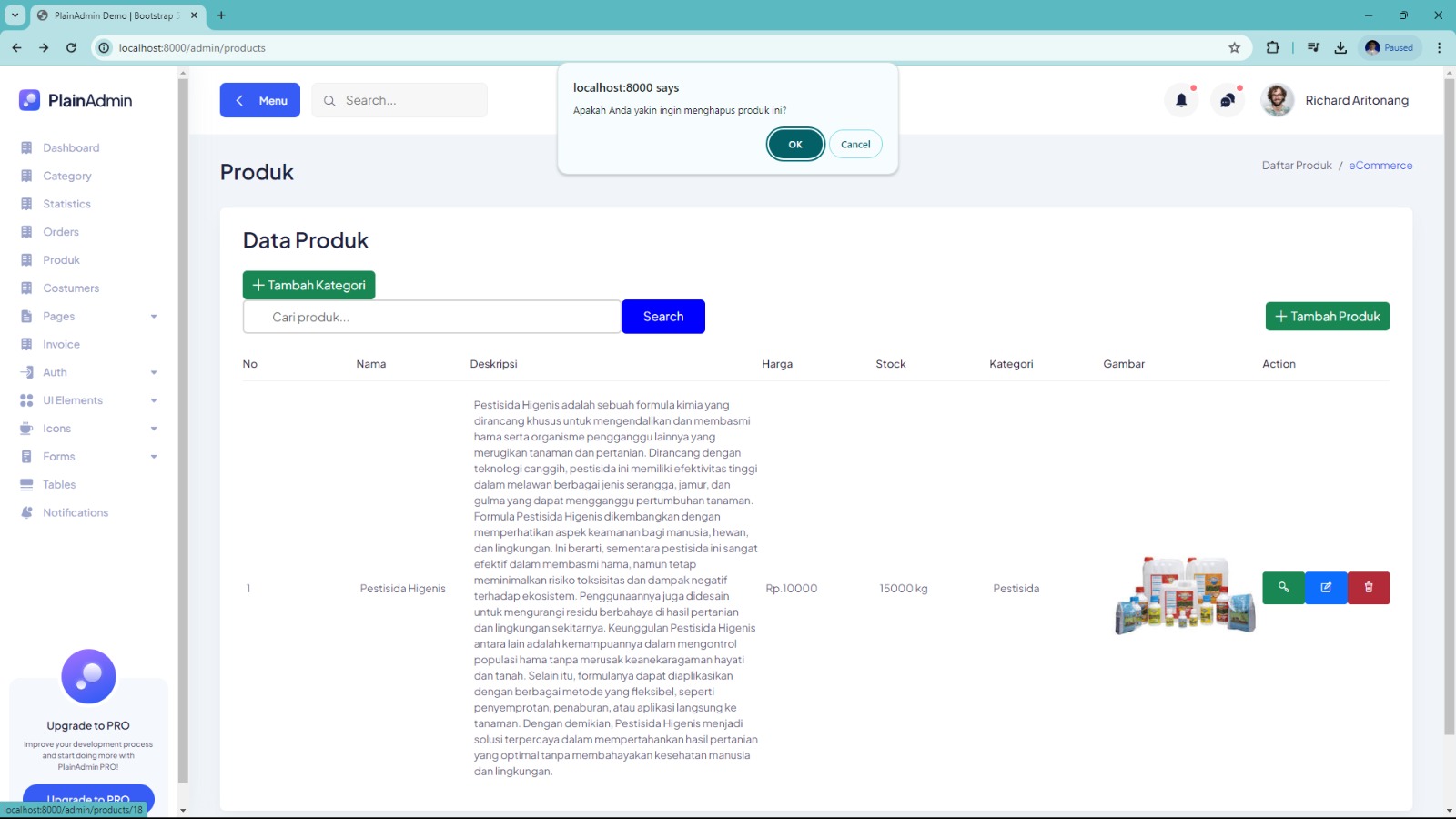
Gambar 29 Tampilan Melihat Produk

* + 1. Tampilan Mengedit Produk



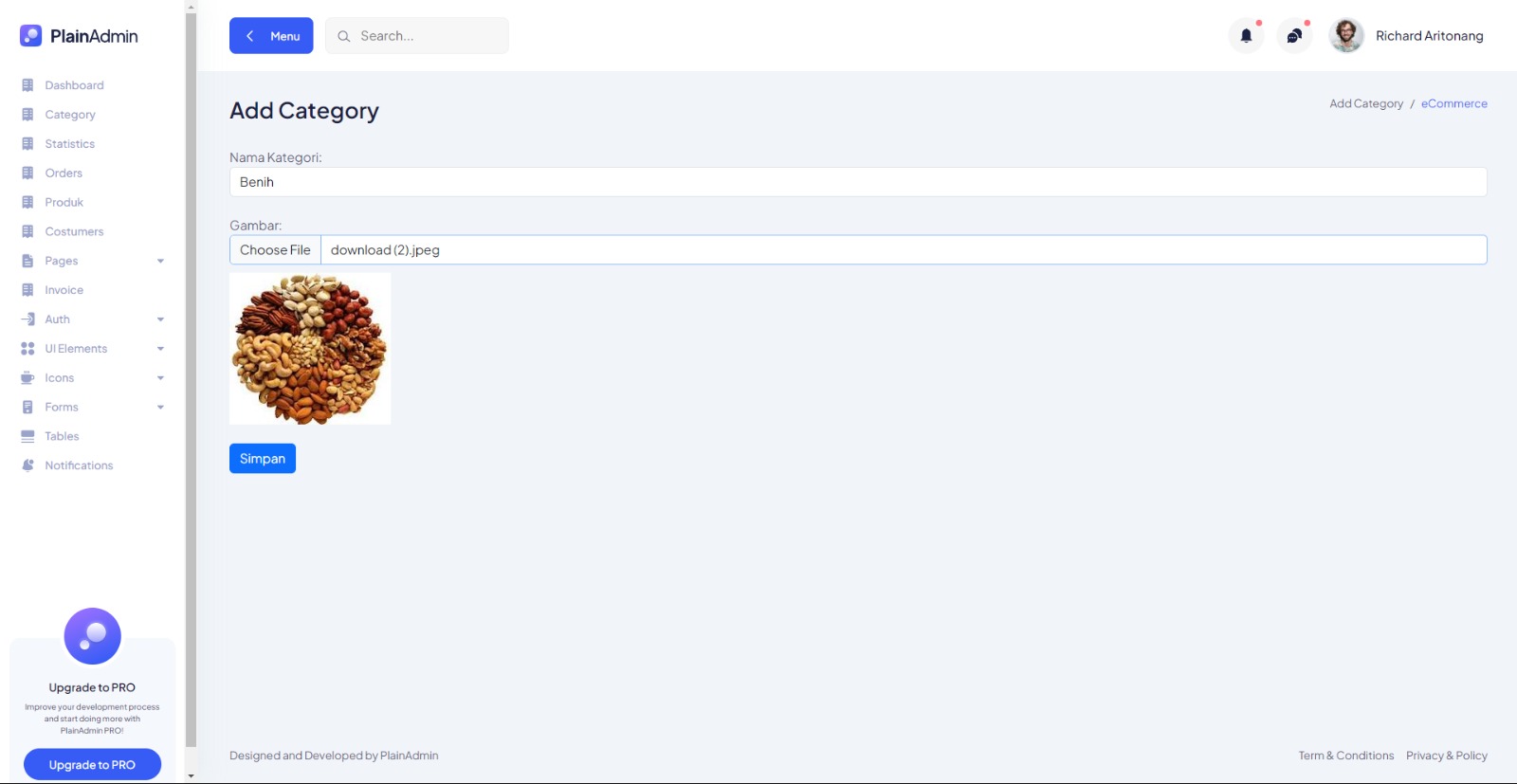
Gambar 30 Tampilan Mengedit Produk

* + 1. Tampilan Menghapus Produk



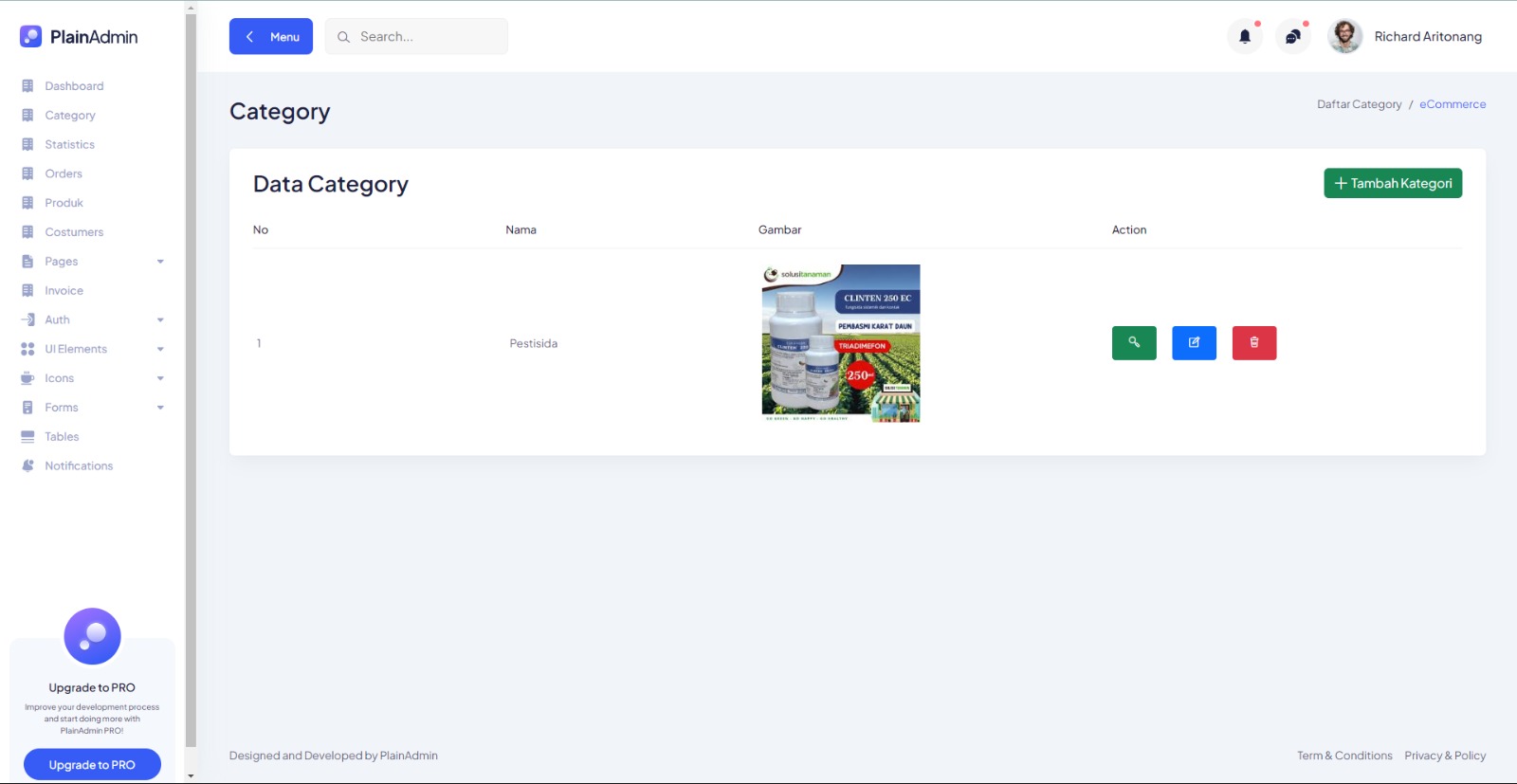
Gambar 31 Tampilan Menghapus Produk

* + 1. Tampilan Menambah Kategori



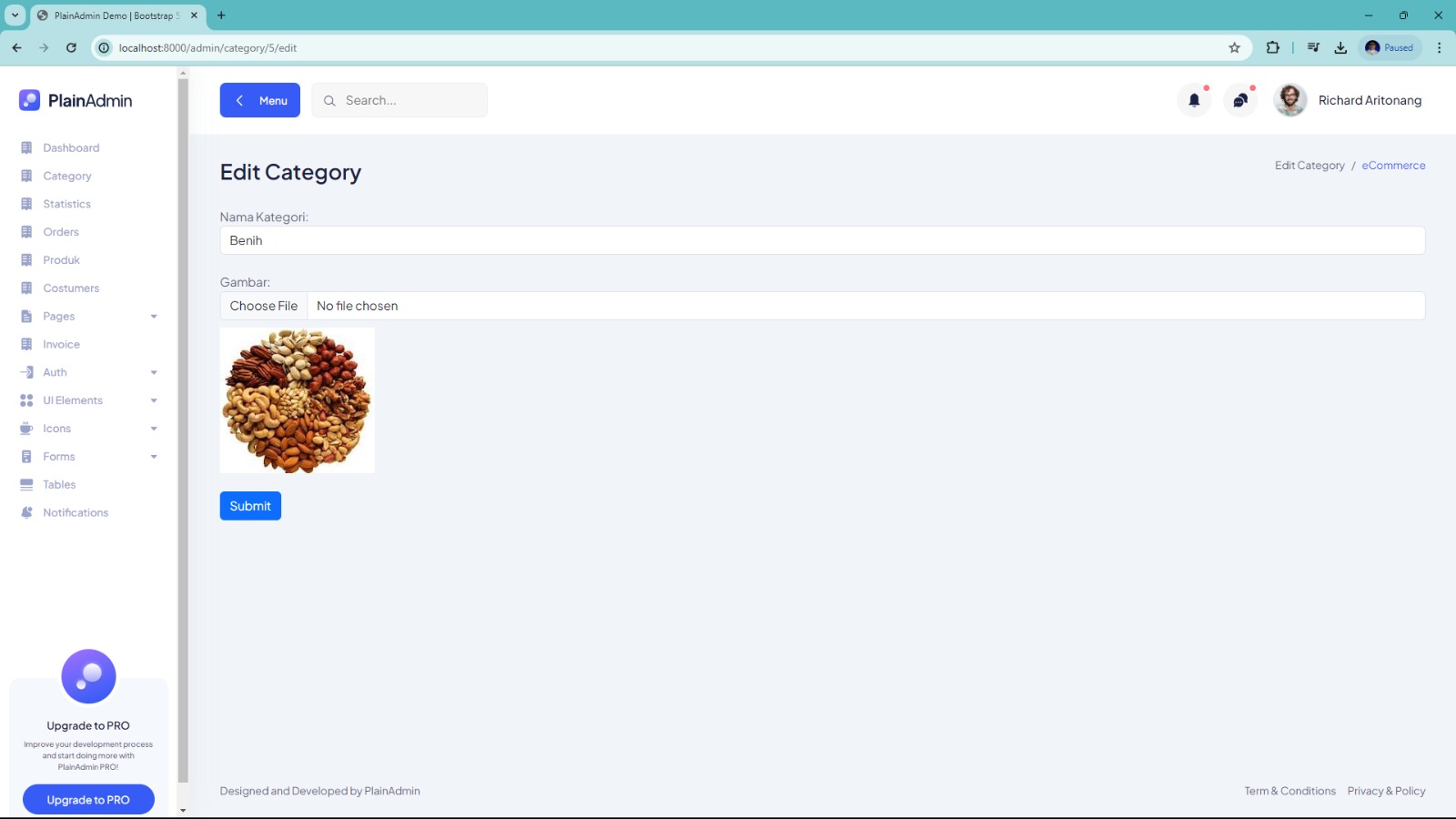
Gambar 32 Tampilan Menambah Kategori

* + 1. Tampilan Melihat Kategori



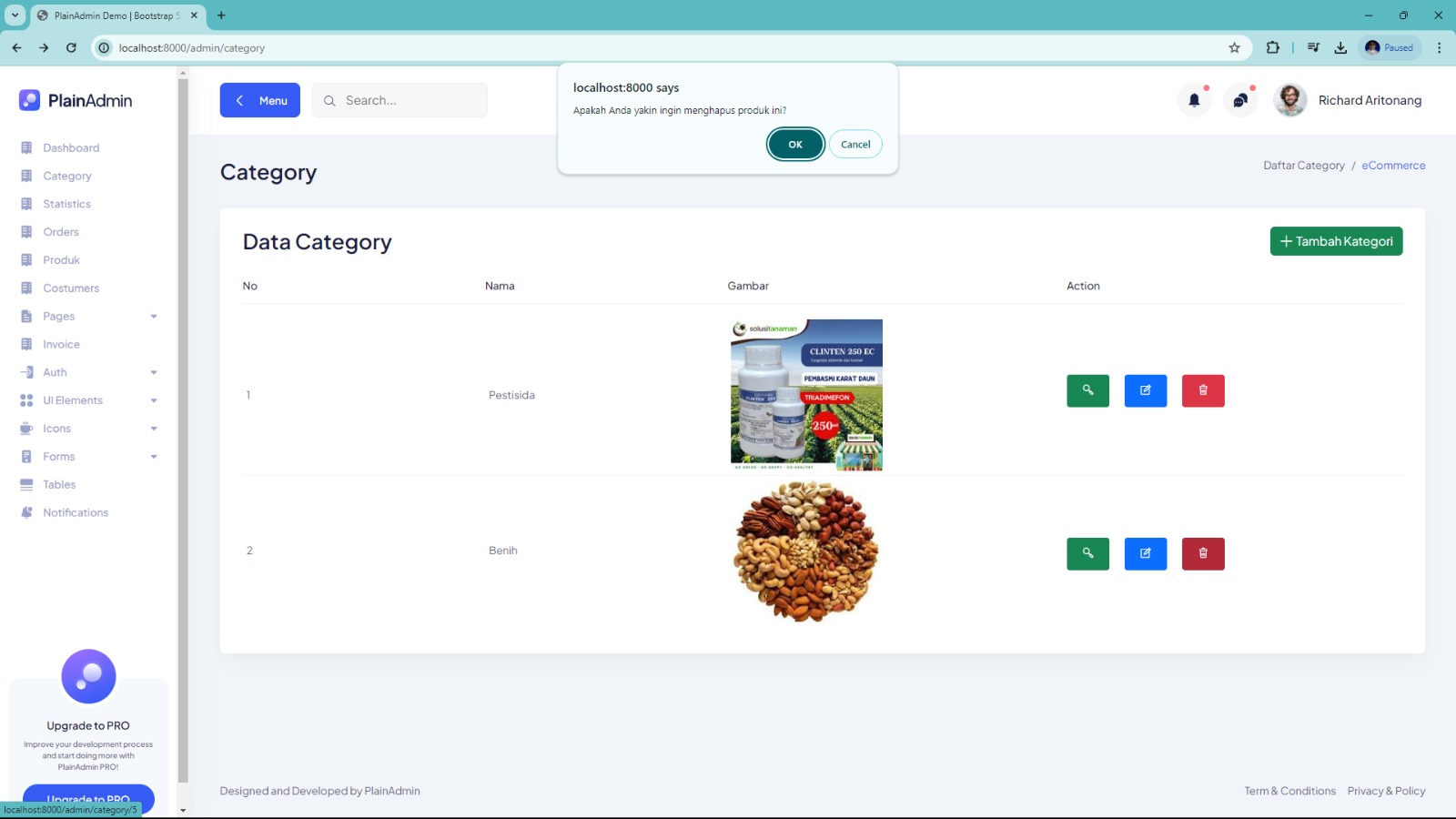
Gambar 33 Tampilan Melihat Kategori

* + 1. Tampilan Mengedit Kategori



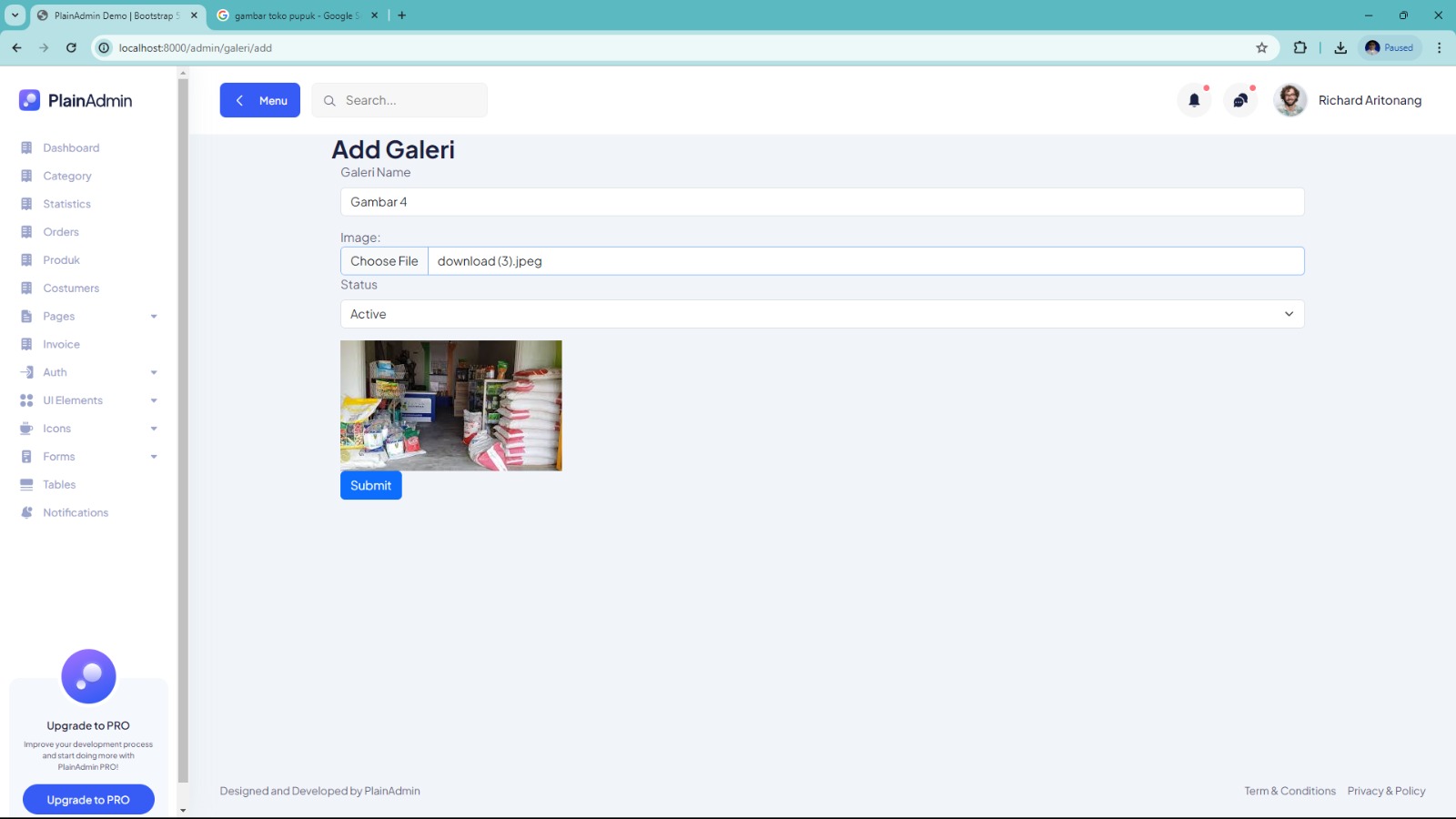
Gambar 34 Tampilan Mengedit Kategori

* + 1. Tampilan Menghapus Kategori



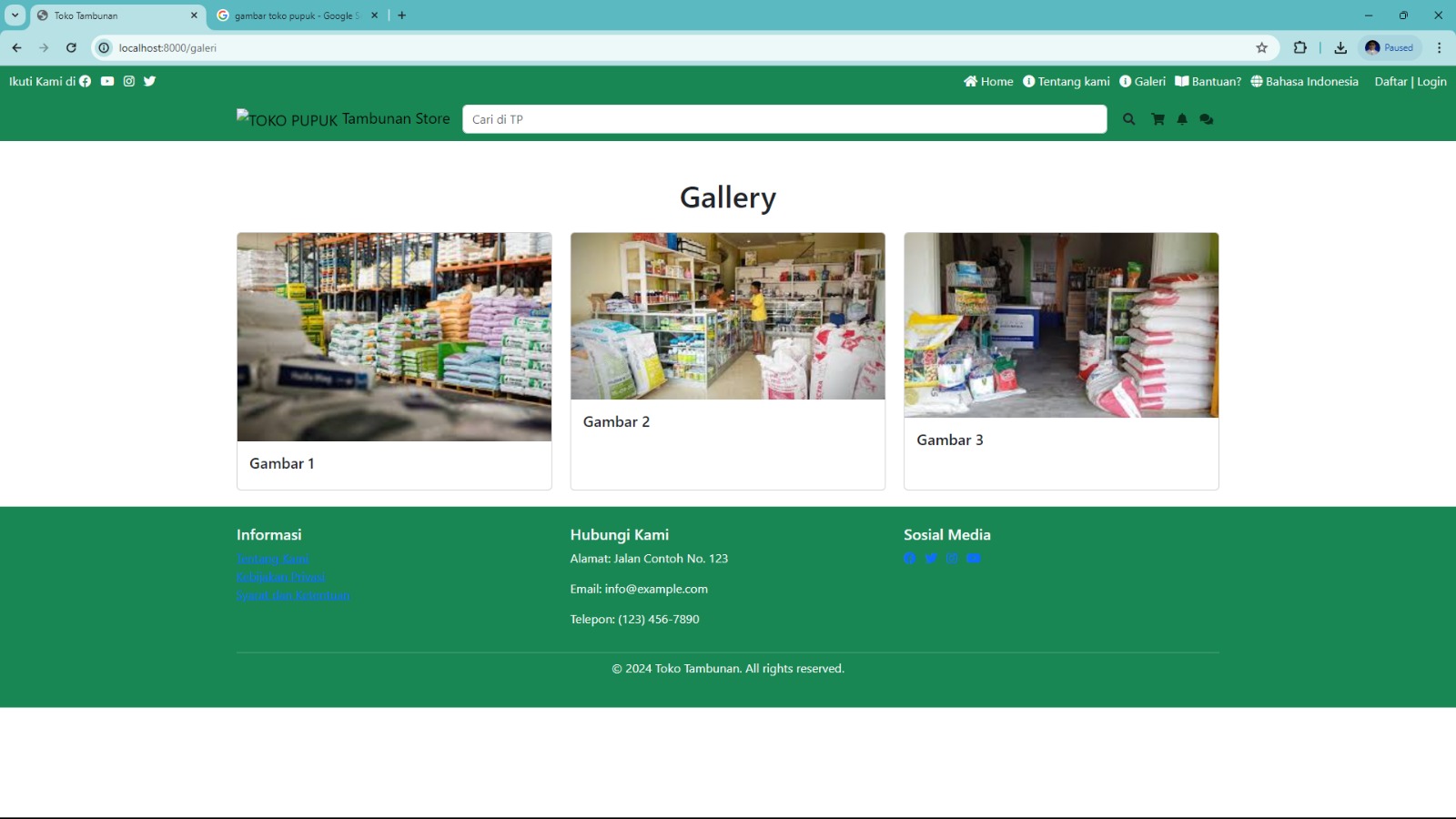
Gambar 35 Tampilan Menghapus Kategori

* + 1. Tampilan Menambah Galeri



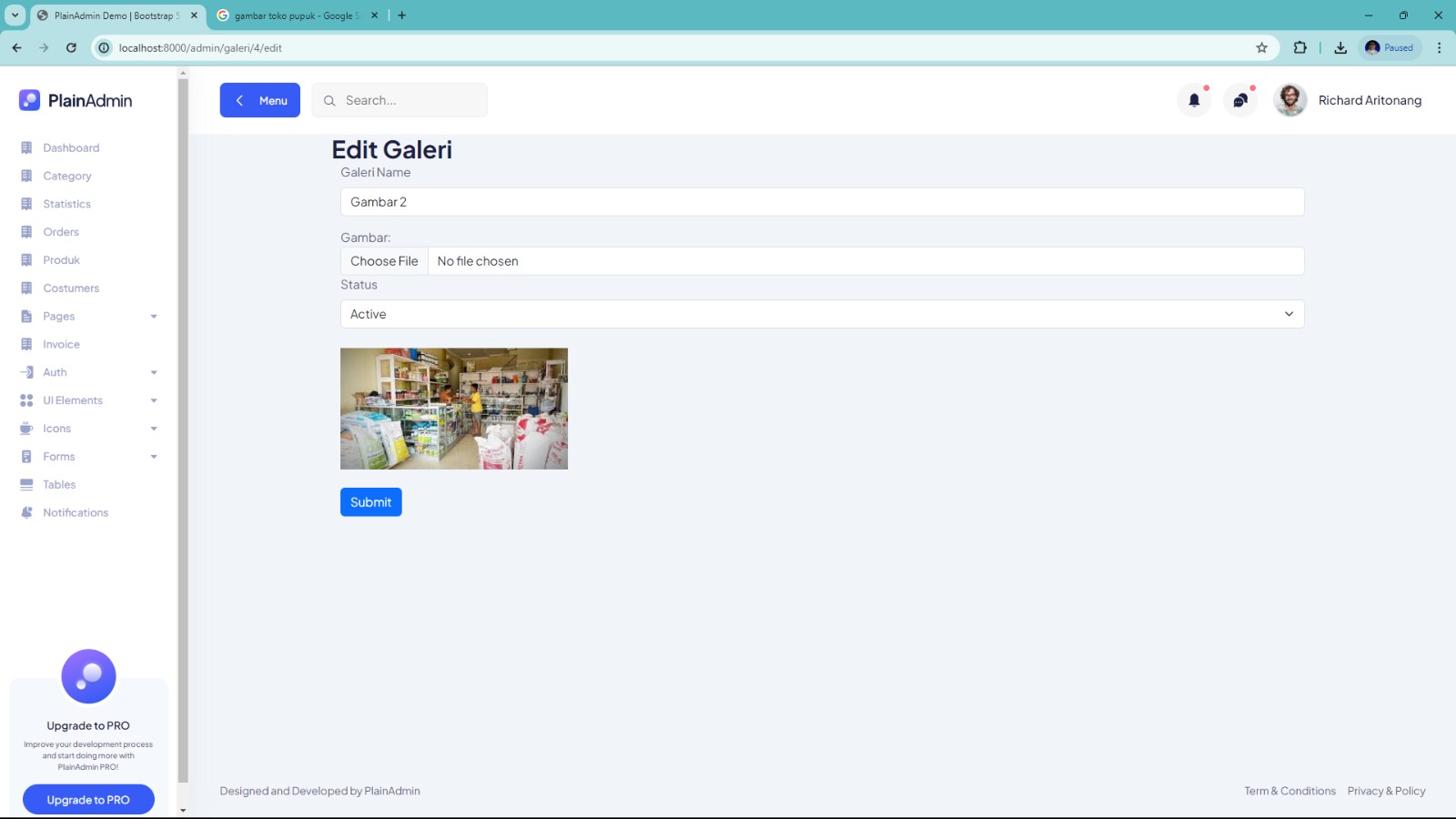
Gambar 36 Tampilan Menambah Galeri

* + 1. Tampilan Melihat Galeri



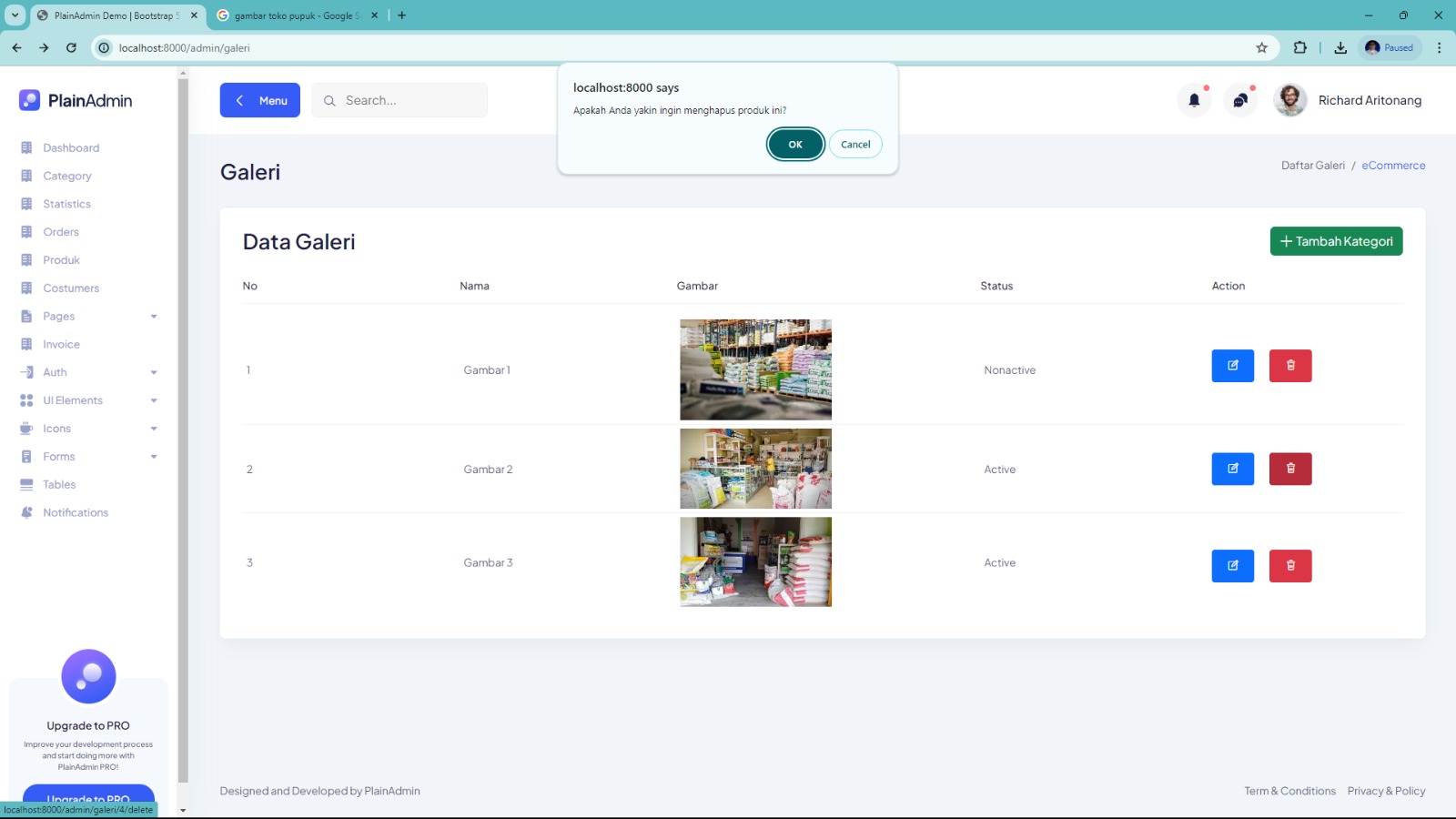
Gambar 37 Tampilan Melihat Galeri

* + 1. Tampilan Mengedit Galeri



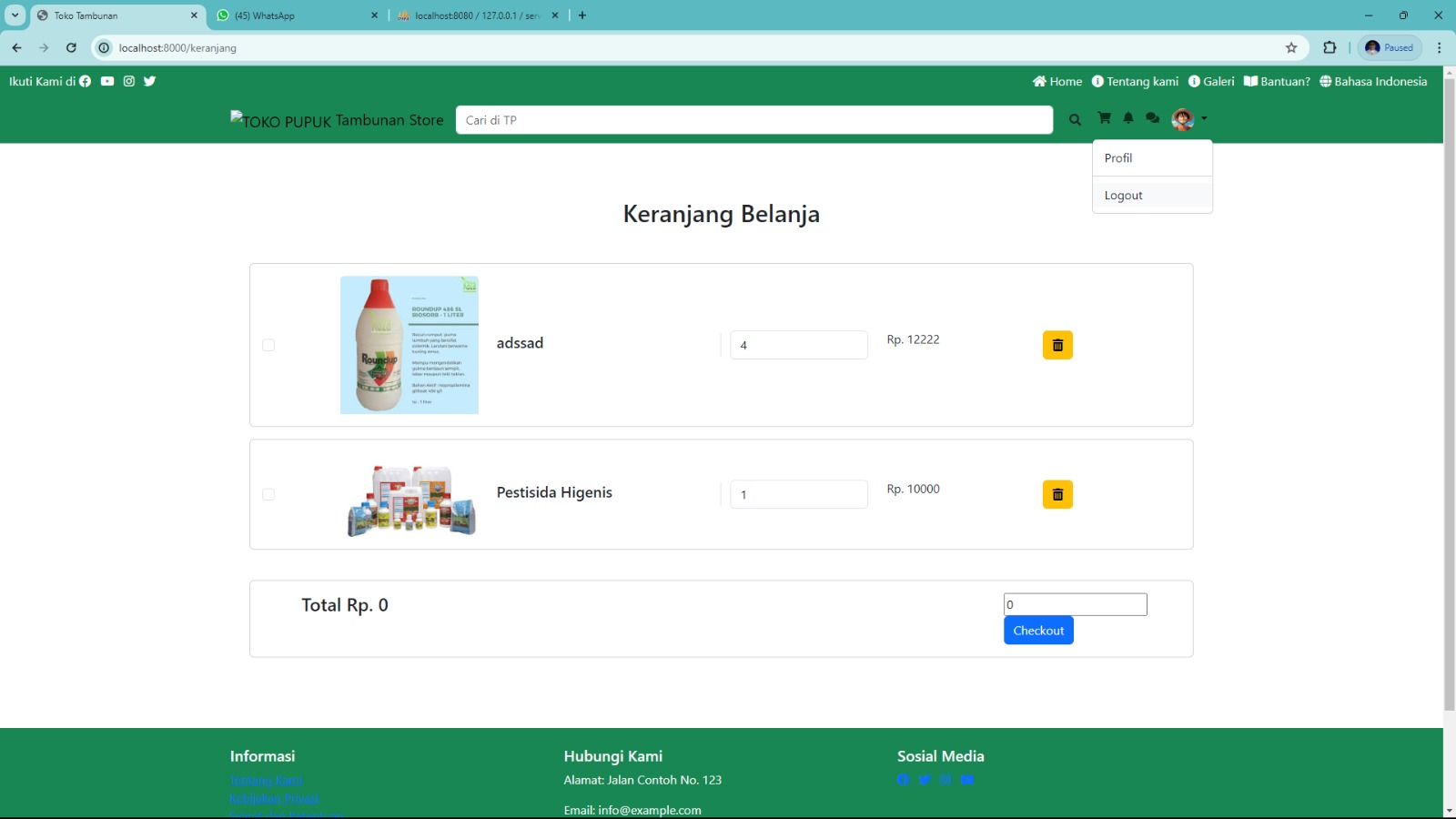
Gambar 38 Tampilan Mengedit Galeri

* + 1. Tampilan Menghapus Galeri



Gambar 39 Tampilan Menghapus Galeri

* + 1. Tampilan *Logout*



Gambar 40 Tampilan Logout

# Pengujian *Website*

Pada pengujian ini, kami berfokus pada *microservice* yang sedang *down.* Service produk *down*. Saat *service* produk *down* maka semua data produk dari database tidak akan ditampilkan pada *website*. Penggunaan *Get, Update, Post, Delete* juga tidak akan bisa dilakukan pada produk saat servicenya sedang *down*. Namun saat service ini *down*, *user* masih bisa melakukan *login*. Hanya saja setelah *login* maka halaman beranda akan kosong karena data produk gagal untuk ditampilkan dari database.

Tabel 2 Pengujian Website Service Product Down

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Butir Uji** | Pengujian *service* produk *down* | | |
| **Tujuan** | Untuk mengetahui apakah *service* lain berjalan saat *service* produk *down* | | |
| **Tanggal Pengujian** | 15/05/2024 | | |
| **Penguji** | Kelompok 10 | | |
| **Skenario Pengujian** | | | |
| * Menonaktifkan service produk * Menjalankan *website* | | | |
| **Kriteria Evaluasi Hasil** | | | |
| Berhasil menjalankan *service category*, meskipun *service* produk sedang *down* | | | |
| **Kasus dan Hasil Pengujian** | | | |
| **Data Masukan** | **Yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| *Service* produk  dinonaktifkan | Service *category* berjalan | Sesuai yang diharapkan | [X] diterima  [ ] ditolak |
| **Catatan** | | | |
|  | | | |

Tabel 3 Pengujian Website Service Kategori Down

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Butir Uji** | Pengujian service *kategori down* | | |
| **Tujuan** | Untuk mengetahui apakah *service* lain berjalan saat *service category down* | | |
| **Tanggal Pengujian** | 15/05/2024 | | |
| **Penguji** | Kelompok 10 | | |
| **Skenario Pengujian** | | | |
| * Menonaktifkan service *kategori produk* * Menjalankan *website* | | | |
| **Kriteria Evaluasi Hasil** | | | |
| *Guest* berhasil melihat produk | | | |
| **Kasus dan Hasil Pengujian** | | | |
| **Data Masukan** | **Yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| *Service Category*  dinonaktifkan | Service produk berjalan | Sesuai yang diharapkan | [X] diterima  [ ] ditolak |
| **Catatan** | | | |
|  | | | |

Tabel 4 Pengujian Website Service Product Down

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Butir Uji** | Pengujian *service product down* | | |
| **Tujuan** | Untuk mengetahui apakah *service* lain berjalan saat *service product down* | | |
| **Tanggal Pengujian** | 15/05/2024 | | |
| **Penguji** | Kelompok 10 | | |
| **Skenario Pengujian** | | | |
| * Menonaktifkan service *product* * Menjalankan *website* | | | |
| **Kriteria Evaluasi Hasil** | | | |
| Berhasil menjalankan *service cart* meskipun service *order* sedang *down* | | | |
| **Kasus dan Hasil Pengujian** | | | |
| **Data Masukan** | **Yang diharapkan** | **Pengamatan** | **Kesimpulan** |
| *Service product*  dinonaktifkan | Service *cart* berjalan | Sesuai yang diharapkan | [X] diterima [ ] ditolak |
| **Catatan** | | | |
|  | | | |

**KESIMPULAN:** Dengan menggunakan *microservice,* setiap *service* pada *website* yang dibangun dapat berjalan secara independent. Artinya, ketika service lain sedang *down*, hal ini tidak akan memengaruhi service lainnya.