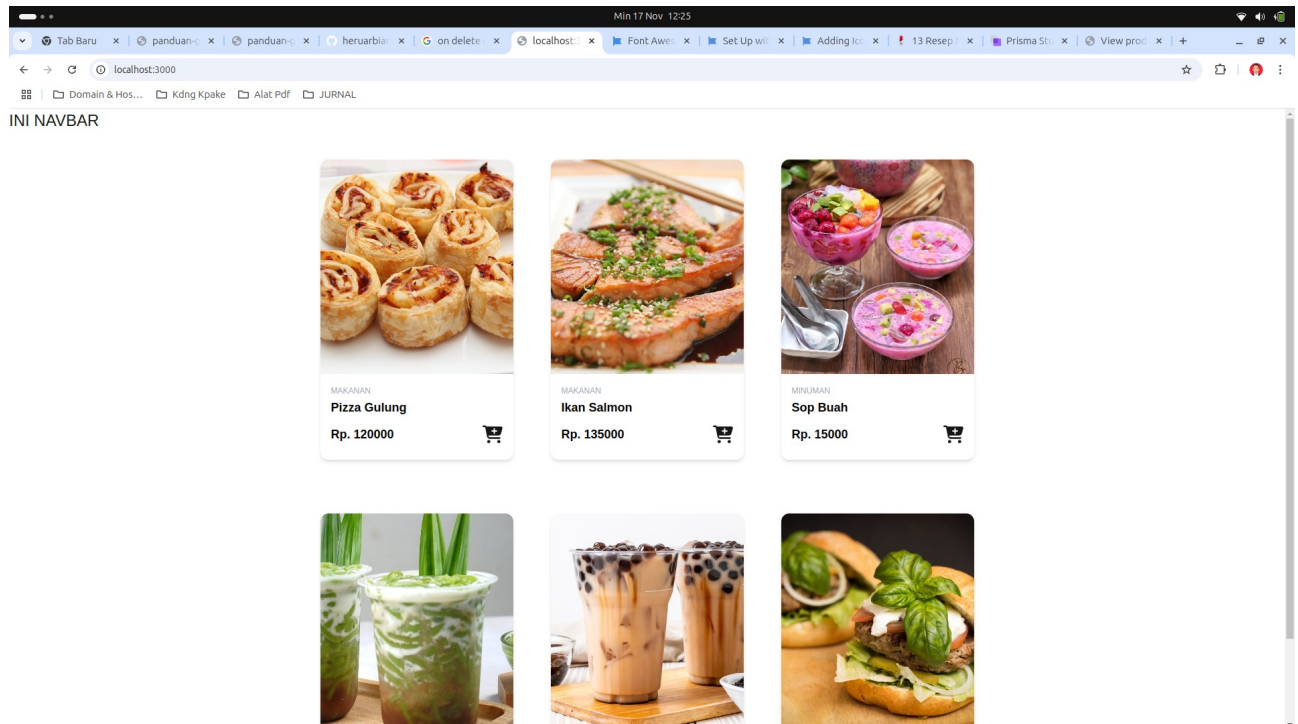
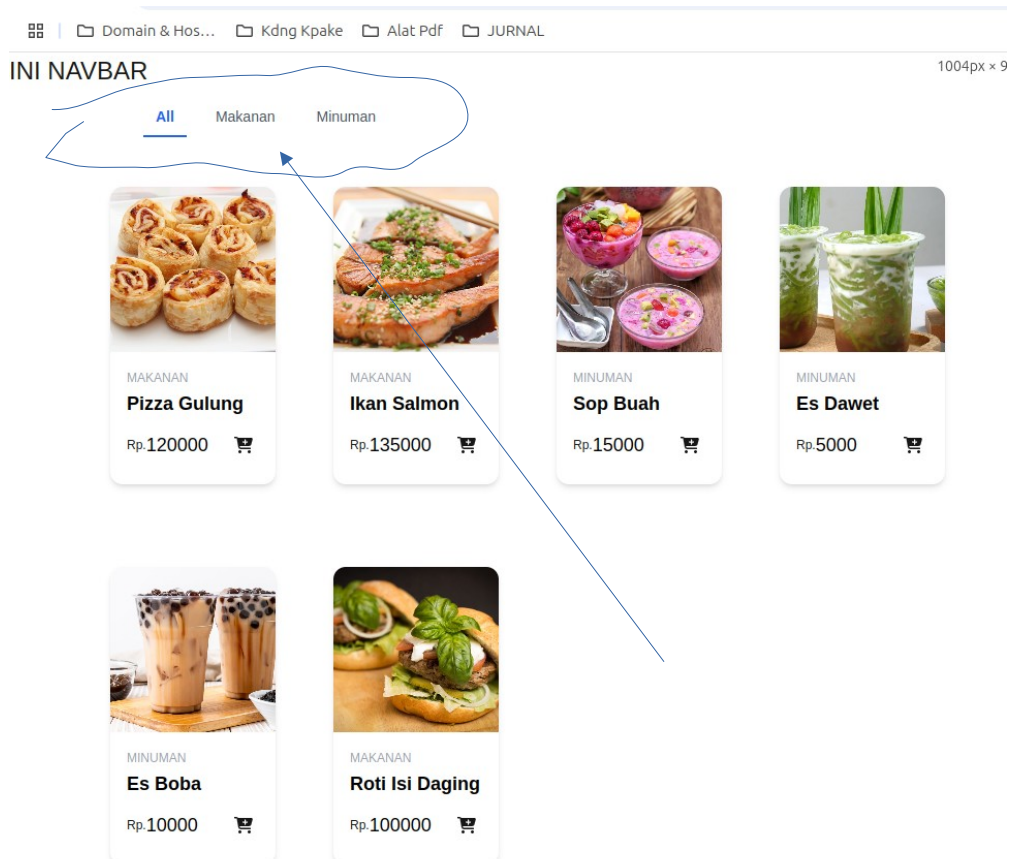


NAMA : HERUARBIANTO  
NPM : 22312004

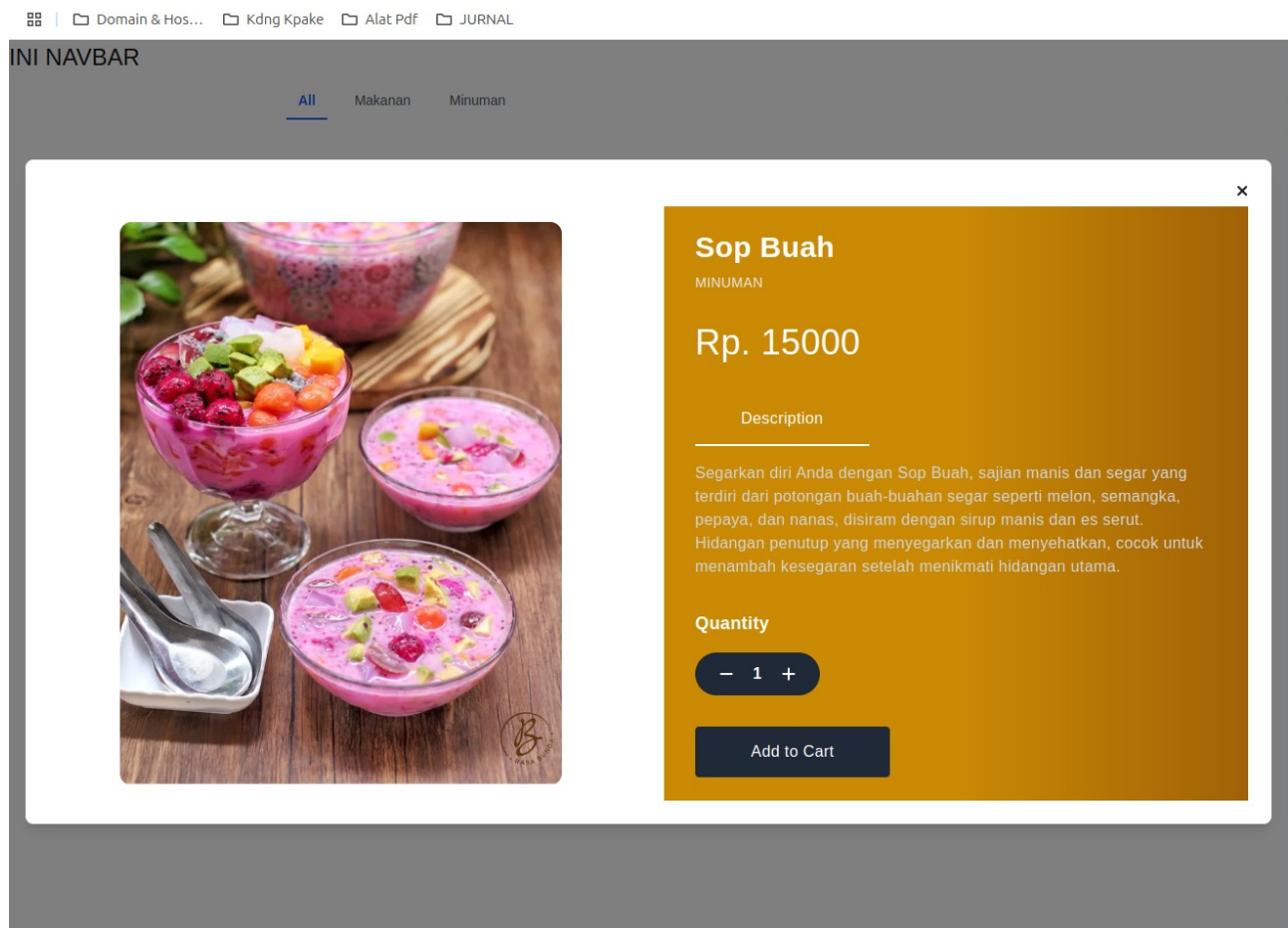
1. membuat database Menu Restoran (tb\_menu) mmenggunakan hook. Lalu menampilkan dengan grid layout responsive di halaman utama



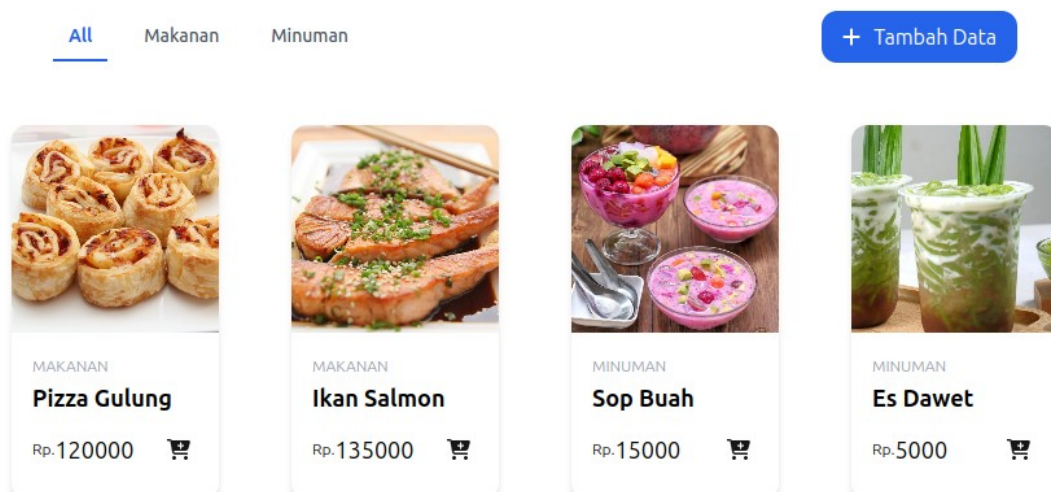
2. Membuat filter daftar menu berdasarkan kategori menggunakan hook

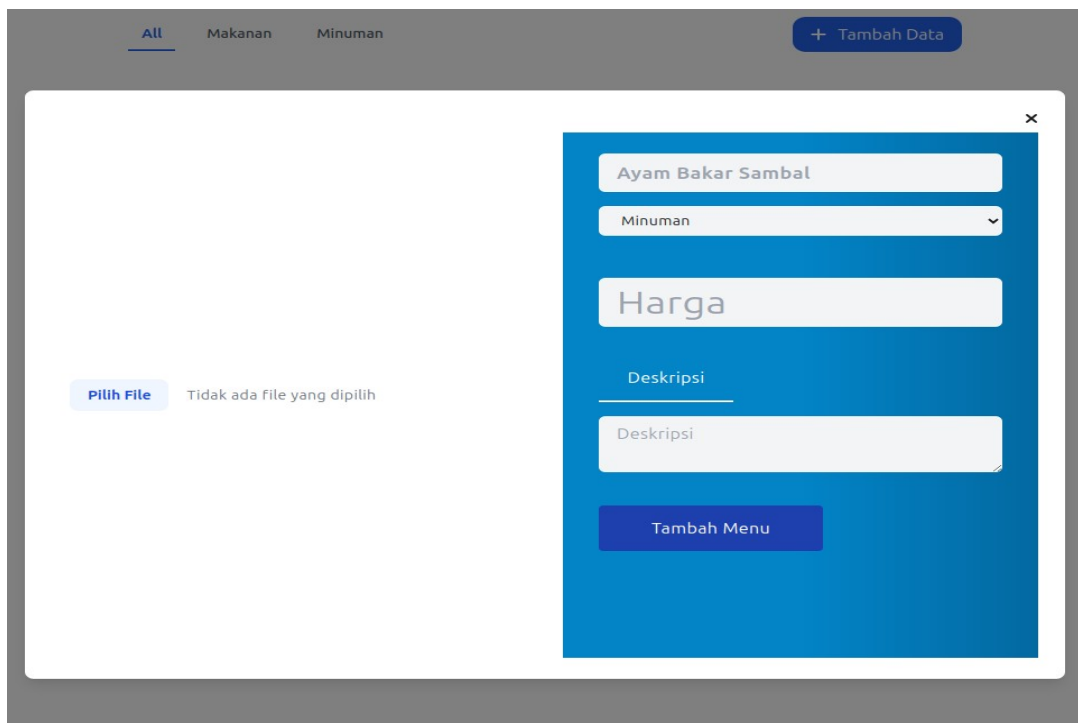


3. saya membuat Fitur **Modal Detail Menu** memungkinkan pengguna melihat detail lengkap sebuah menu dengan mengkliknya dari daftar menu. Ketika diklik, modal akan terbuka dan menampilkan informasi seperti gambar, nama, kategori, deskripsi, harga, serta pengaturan kuantitas dengan tombol tambah/kurang. Data menu diambil secara dinamis dari database menggunakan ID menu yang dikirim melalui fungsi menuDetail. Modal dapat ditutup dengan mengklik ikon tutup atau area luar modal untuk kemudahan navigasi.



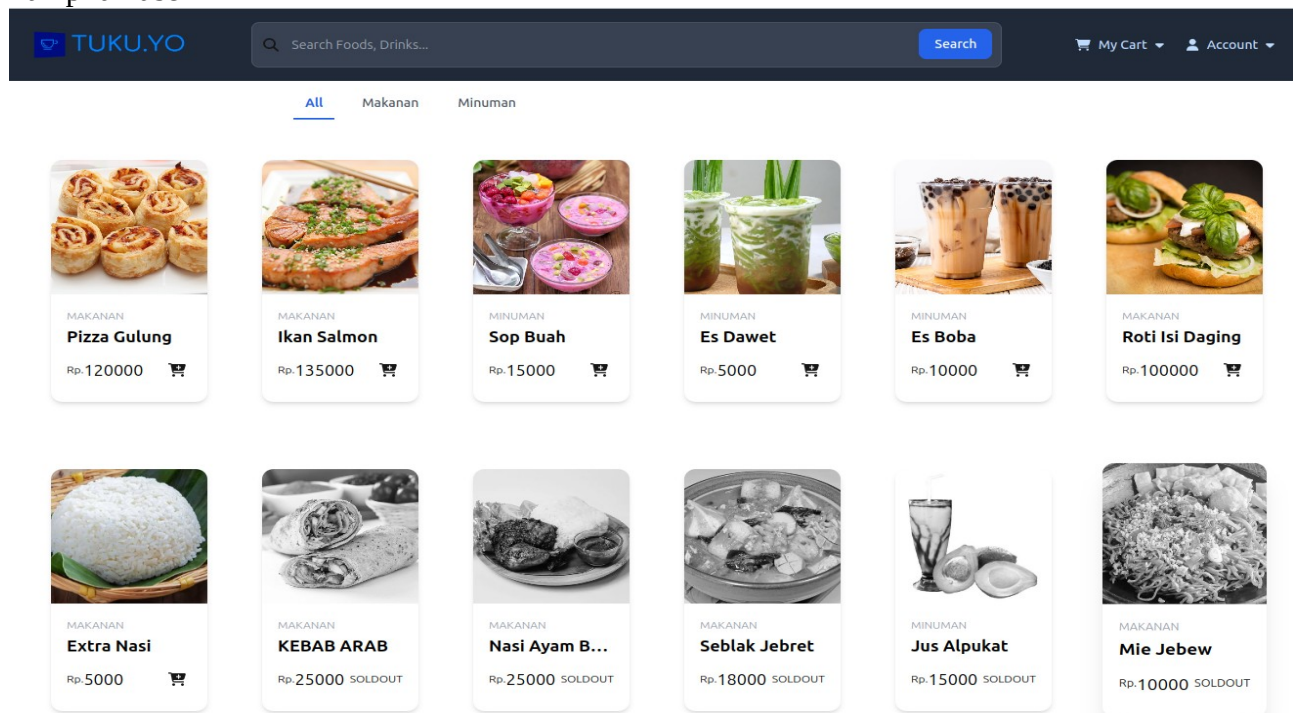
4. Saya membuat fitur untuk menambahkan data menggunakan modal di halaman **dashboardAdmin**. Proses unggah file gambar dilakukan dengan menyimpan gambar di direktori /ImageMenu/NamaFile.jpg dan kemudian path file tersebut disimpan ke dalam database. Untuk penamaan gambar, nama file diubah menggunakan nama menu yang diinputkan oleh pengguna, dengan spasi digantikan oleh tanda hubung (-). Di akhir nama file, ditambahkan timestamp waktu saat ini untuk memastikan setiap nama file unik.



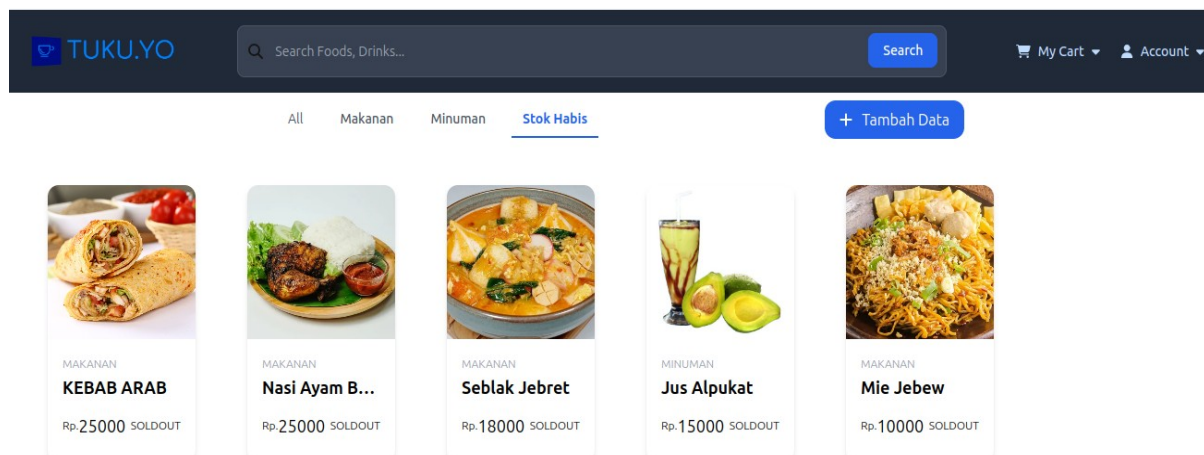


5. Saya menambahkan kolom “Ketersediaan” pada **tb\_menu** untuk mengontrol stok ready atau soldout. Yang dimana jika menu yang habis akan ditampilkan berwarna grayscale dan tidak dapat dipesan atau tidak dapat di klik apapun di halaman user/pelanggan.

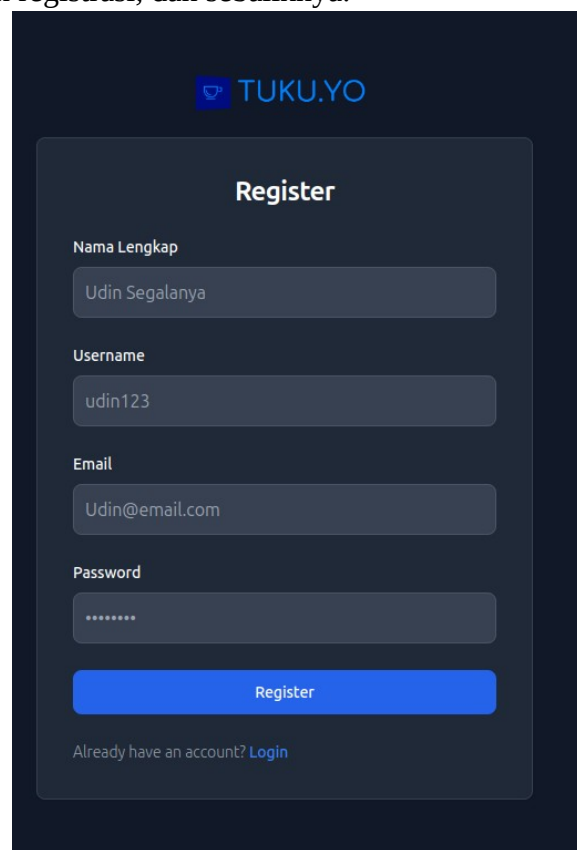
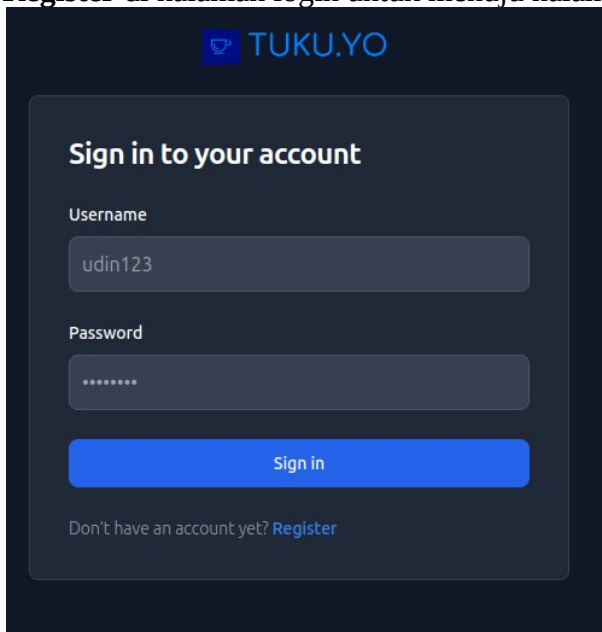
#### Tampilan user



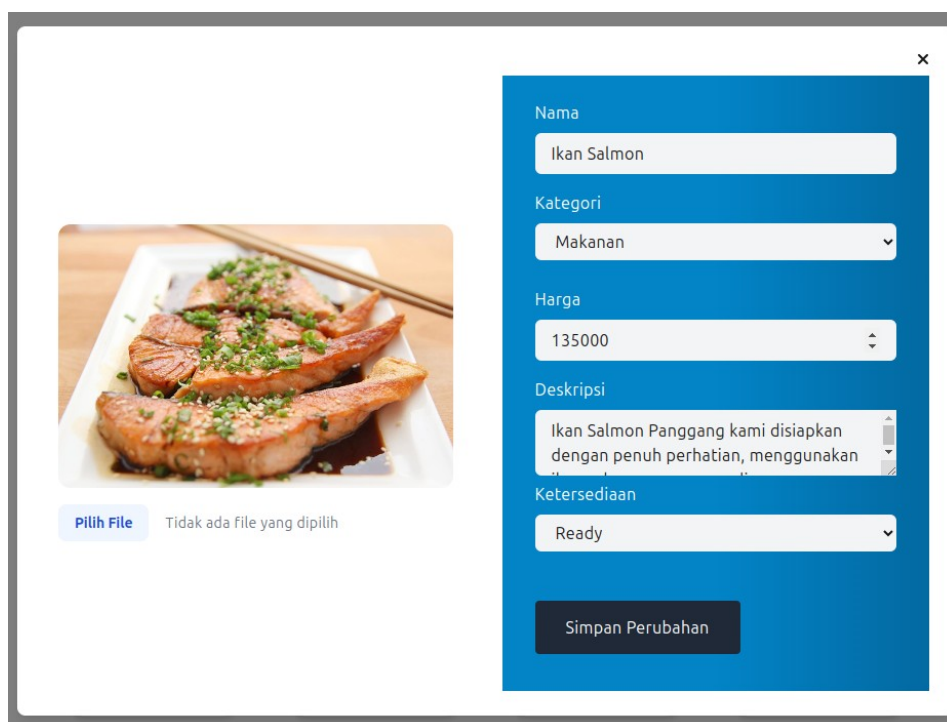
#### Tampilan Admin



6. Saya membuat halaman **Login** dan **Register** yang dapat berganti-ganti tampilan menggunakan tombol, tanpa memuat ulang halaman. Aplikasi ini memungkinkan pengguna untuk beralih antara formulir login dan registrasi secara dinamis melalui **state**. Pengguna dapat mengklik tombol **Register** di halaman login untuk menuju halaman registrasi, dan sebaliknya.



7. Menambah fitur untuk mengedit dan memperbarui data menu pada aplikasi, termasuk penanganan pengunggahan gambar baru dan penghapusan gambar lama. Pengguna dapat mengedit informasi seperti nama, kategori, harga, deskripsi, dan ketersediaan menu. Jika gambar baru diunggah, gambar tersebut akan diubah menjadi format base64 dan disimpan di server, menggantikan gambar lama yang ada. Fitur ini juga mencakup validasi input untuk memastikan data yang dimasukkan lengkap dan benar, serta memastikan respons yang baik saat pengeditan data berhasil dilakukan, dengan tampilan yang mudah dipahami oleh pengguna.





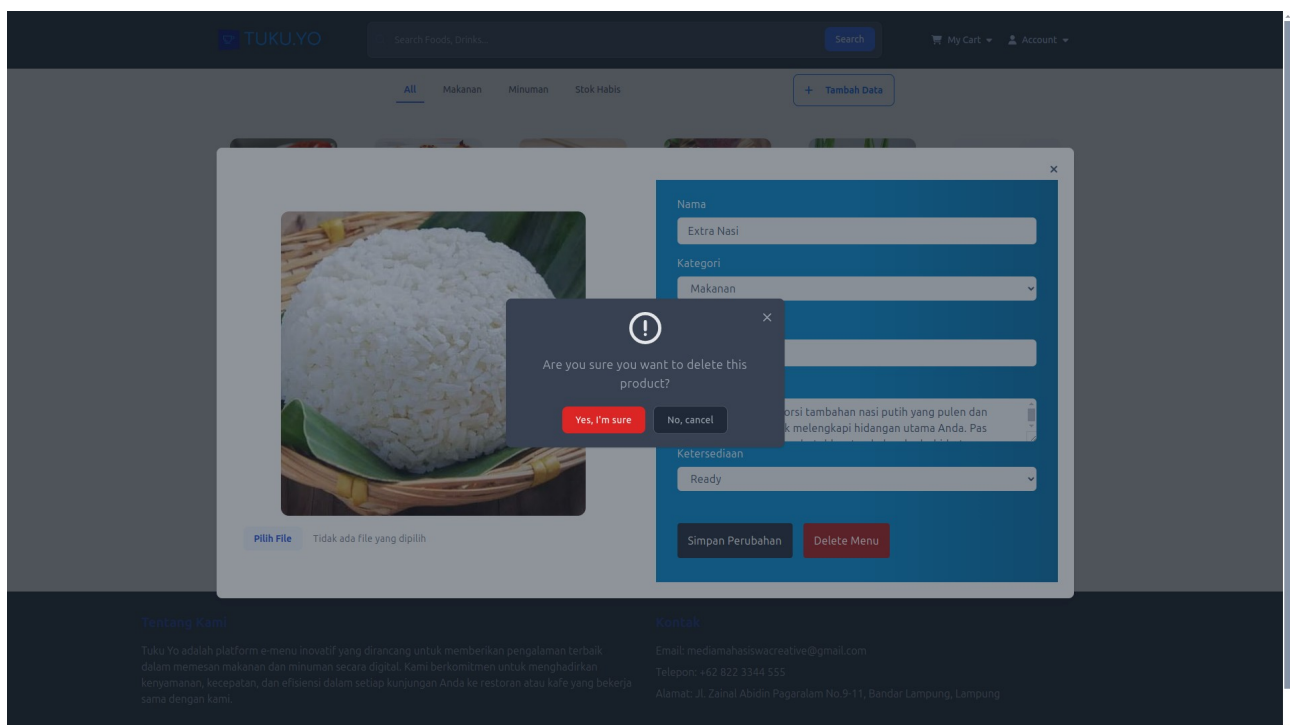
8. Saya mengimplementasikan logika pendaftaran pengguna yang memeriksa apakah username, email, atau nomor telepon yang dimasukkan sudah terdaftar di database. Jika salah satu dari data tersebut sudah ada, pendaftaran akan dibatalkan dan pesan kesalahan yang sesuai akan ditampilkan pada form. Hal ini memastikan bahwa setiap pengguna yang mendaftar memiliki informasi yang unik, mencegah duplikasi data dalam sistem.

The image displays three sequential screenshots of the 'TUKU.YO' registration form, illustrating validation feedback for different fields:

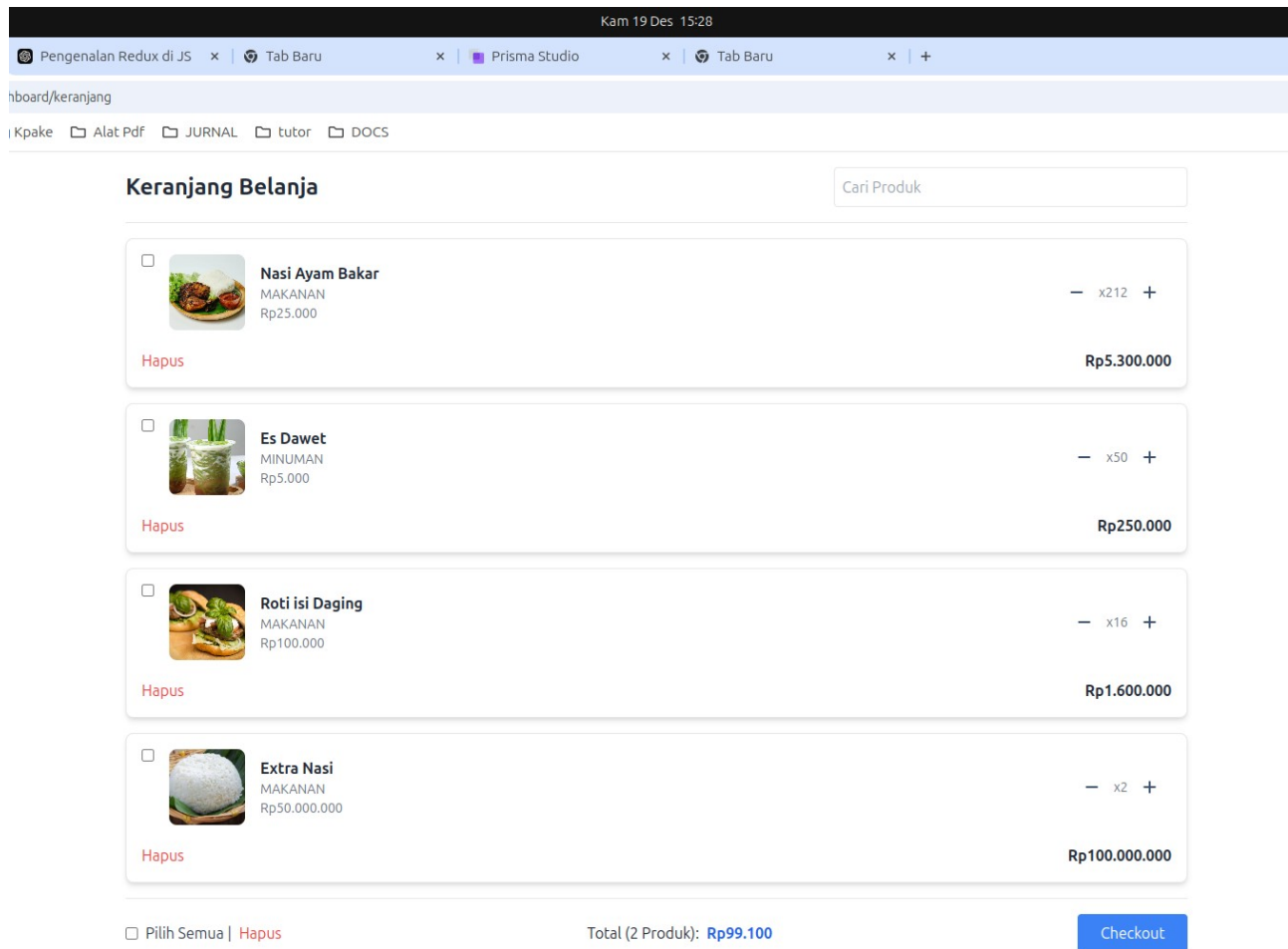
- First Screenshot:** The 'Username' field (22312004) is highlighted with a red border and a red error message 'Username Telah Digunakan' (Username Already Used).
- Second Screenshot:** The 'No. Telp' field (089612204603) is highlighted with a red border and a red error message 'No. Telp Telah Digunakan' (Phone Number Already Used).
- Third Screenshot:** The 'Email' field (heru\_arbianto@teknokrat.ac.id) is highlighted with a red border and a red error message 'Email Telah Digunakan' (Email Already Used).

Each form includes fields for: Nama Lengkap (HERU ARBIANTO), Username, No. Telp, Email, Password, and Alamat (Jl. Merpati 1, NO. 01, Pinang Jaya, Kemiling, Bandar). A 'Register' button and a 'Login' link are at the bottom of each form.

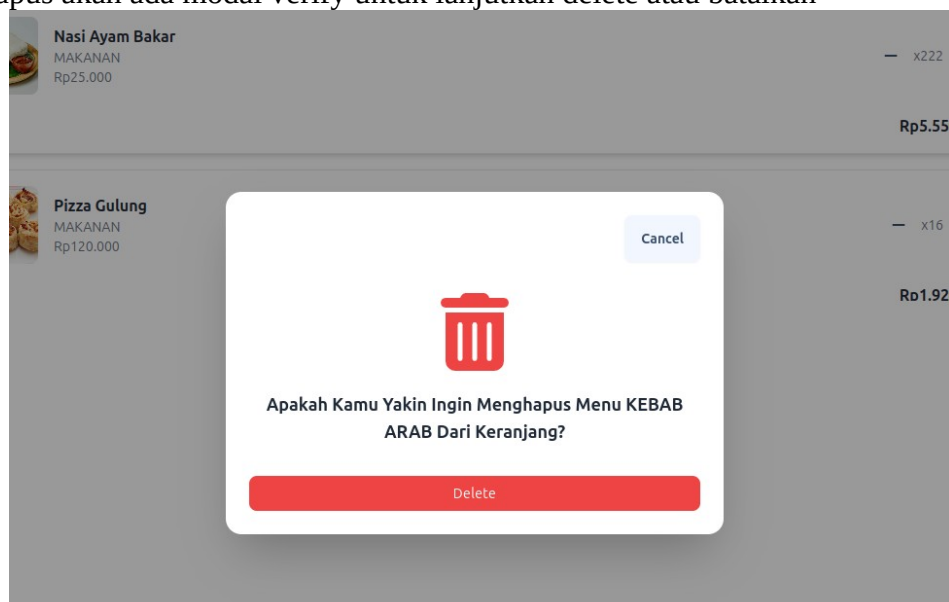
9. Saya Membuat Fitur delete di halaman DashboardAdmin, di bagian editMenu. Dengan tambahan Modal untuk Verify Delete



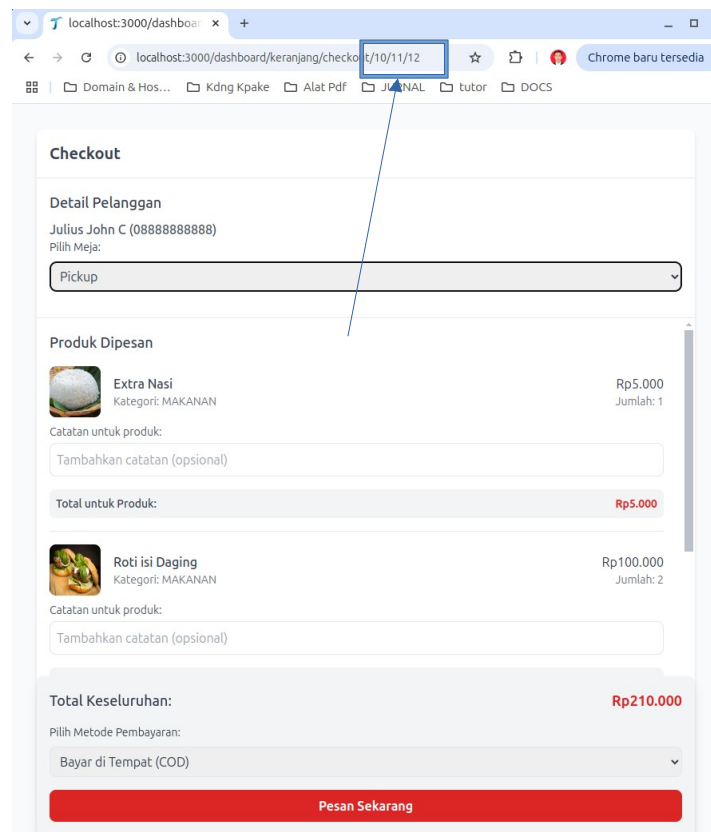
10. Saya membuat halaman keranjang pengguna. Di dalamnya, pengguna dapat menambah atau mengurangi jumlah produk yang ada di keranjang, dan setiap perubahan jumlah otomatis memperbarui total harga produk. Data produk diambil dari database dan ditampilkan dalam bentuk daftar dengan rincian gambar, nama, kategori, dan harga produk. Semua perubahan langsung tercermin pada tampilan dengan pembaruan state menggunakan useState dan useEffect.



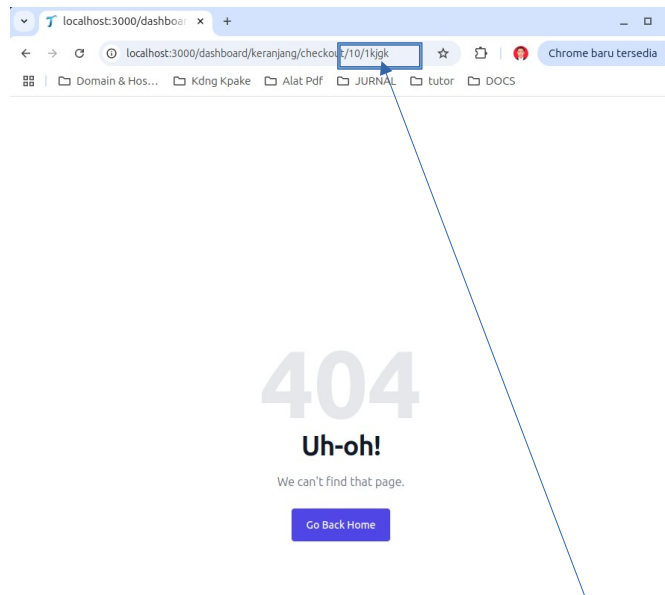
11. Saya membuat fitur delete apabila menu dikeranjang =1 dan user mencoba mengurangi quantity dengan menekan tombol minus(-), maka menu akan dihapus dari keranjang. Namun sebelum dihapus akan ada modal verify untuk lanjutkan delete atau batalkan



12. Membuat halaman checkout yang endpointnya menerima params berupa array yang berisi beberapa id keranjang yang dikirimkan dari halaman keranjang.



lalu untuk memastikan idkeranjang tidak di manipulasi (di coba coba oleh user) maka ketika user mencoba memasukan params dengan data selain array angka(idKeranjang) dan ketika id keranjang yang dimasukkan tidak sesuai dengan id user yang login maka return halaman forbidden 404



13. Saya Membuat fungsi untuk memproses pemesanan dari halaman checkout. Lalu pesanan yang berhasil dibuat akan dikirimkan ke halaman profile dan di tampilkan daftar orderan yang berhasil dibuat

**Julius John C**  
julius@teknokrat.ac.id  
[Edit Profile](#)

**Statistics**

- 2 Orders
- 3 Quantity
- 255.3rb Pengeluaran

**Recent Orders**

| Order ID | Date        | Status             | Total       |
|----------|-------------|--------------------|-------------|
| #1       | Jan 16 2025 | SELESAI            | Rp. 144.300 |
| #2       | Jan 16 2025 | DIBATALKAN         | Rp. 299.700 |
| #3       | Jan 16 2025 | SELESAI            | Rp. 111.000 |
| #4       | Jan 16 2025 | MENUNGGUPEMBAYARAN | Rp. 133.200 |

[Logout](#)

Lalu ketika orderan selesai, maka statistik orderan akan bertambah atau menyesuaikan berdasarkan orderan yang telah selesai seperti pada gambar diatas yang saya lingkarin

14. Saya Membuat modal detail pesanan/ invoice, modal akan terbuka ketika salah satu list pesanan yang ada di halaman profile di klik

ishboard/profile

ig Kpake | Alat Pdf | JURNAL | tutor | DOCS

**JK Julius John C**  
julius@teknokrat.ac.id  
[Edit Profile](#)

**Statistics**

- 255.3rb Pengeluaran

**Recent Orders**

| Order ID | Date        |
|----------|-------------|
| #1       | Jan 16 2025 |
| #2       | Jan 16 2025 |
| #3       | Jan 16 2025 |
| #4       | Jan 16 2025 |
| #5       | Jan 18 2025 |

**TUKU.YO Invoice**

Tuku.yo  
Jl. Za Pagar Alam, Bandar Lampung  
+62 896-1220-4603  
mediamahasiswacreative@gmail.com

Invoice Number: 3  
Date: Thu Jan 16 2025 15:20:18 GMT+0700 (Waktu Indonesia Barat)

**Bill To:**  
Julius John C  
08888888888  
julius@teknokrat.ac.id

| Description     | Qty | Rate    | Amount  |
|-----------------|-----|---------|---------|
| Roti isi Daging | 1   | 100.000 | 100.000 |

Subtotal: 100.000  
PPN (11%): 11.000  
**Total: 111.000**

[Logout](#)



15. membuat fungsi untuk membatalkan pesanan otomatis setelah 24jam tidak melakukan pembayaran, fungsi tersebut dipanggil setiap 30rb milidetik/30detik menggunakan Scheduler dengan **setInterval**

```
export const batalkanPesananOtomatis = async () => {
  try {
    // Hitung waktu 24 jam yang lalu
    const batasWaktu = new Date();
    batasWaktu.setHours(batasWaktu.getHours() - 24);

    // Cari pesanan yang memenuhi syarat
    const pesananYangDibatalkan = await prisma.tb_pemesanan.updateMany({
      where: {
        status: "MENUNGGUPEMBAYARAN",
        createdAt: {
          lt: batasWaktu,
        },
      },
      data: {
        status: "DIBATALKAN",
      },
    });

    console.log(
      `${pesananYangDibatalkan.count} pesanan telah dibatalkan secara otomatis.`
    );
  } catch (error: any) {
    console.error("Gagal membatalkan pesanan secara otomatis:", error.message);
  }
};

// Jalankan fungsi secara periodik setiap 30 detik
setInterval(batalkanPesananOtomatis, 30000);
```

16. Membuat statistik orderan di dashboard admin berdasarkan hari,minggu dan bulan



- Dashboard
- Kanban
- Inbox 3
- Users
- Products
- Logout
- Sign Up

Pro

Total Profit  
**294.1rb**

Daily Report  
**294.1rb**

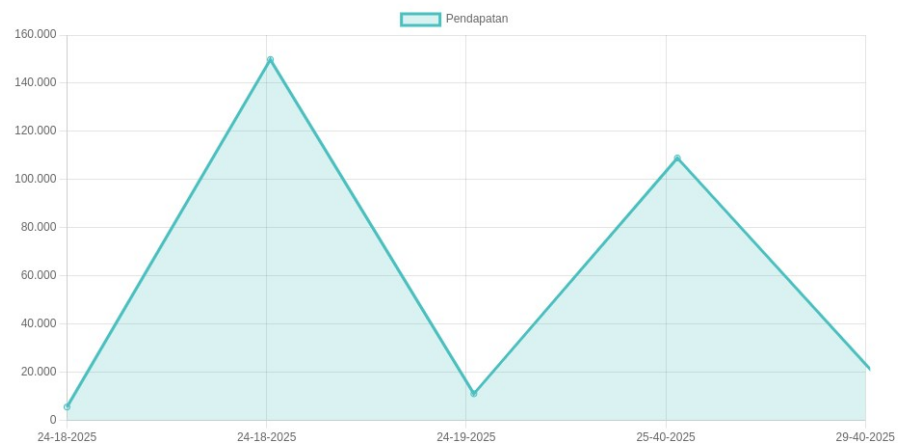
Weekly Report  
**294.1rb**

Monthly Report  
**294.1rb**

Total Orders  
**5**

Revenue Chart

Day Week Month



## 17. memberi warna pada design statistik orderan di dashboard admin

Domain & Hos... Kdng Kpake Alat Pdf JURNAL tutor DOCS



Dashboard

Kanban

Inbox

Users

Products

Logout

Sign Up

Pro

3

TOTAL PROFIT

294.1rb

DAILY REPORT

294.1rb

WEEKLY REPORT

294.1rb

MONTHLY REPORT

294.1rb

TOTAL ORDERS

5

Revenue Chart

Day

Week

Month

