

**APLIKASI PEMESANAN MENU KEDAI FMR BERBASIS PYTHON  
MENGUNAKAN TKINTER**



**Dosen Pengampu:**

Hasanuddin Al-Habib, S.Si., M.Si.

Dr. Heri Purnawan, S.Si., M.Si.

**Nama Anggota :**

Jauzah Rifda (25031554007)

Naima'ul Iza Fadhilah (25031554034)

Fahrani Zalfa Zakiya (25031554170)

**KELAS 2025G**

**MATA KULIAH PEMROGRAMAN DASAR**

**PROGRAM STUDI S1 SAINS DATA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS NEGERI SURABAYA**

**DESEMBER 2025**

## DAFTAR ISI

DAFTAR ISI .....	II
BAB 1 .....	1
PENDAHULUAN.....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan .....	2
BAB 2 .....	3
ANALISIS DAN PERANCANGAN .....	3
2.1 Analisis Kebutuhan Aplikasi.....	3
2.1.1 Kebutuhan Fungsional .....	3
2.1.2 Kebutuhan Non-fungsional.....	3
2.2 Sketsa Desain Antarmuka .....	3
2.3 Diagram Alur .....	4
BAB 3 .....	5
IMPLEMENTASI .....	5
3.1 Penjelasan Kode Program.....	5
3.2 Cara Penggunaan .....	7
3.3 Tampilan Aplikasi .....	8
LAMPIRAN .....	11
DAFTAR PUSTAKA.....	39

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Seiring kemajuan zaman, perkembangan teknologi informasi dan digitalisasi telah memberikan dampak besar terhadap berbagai aspek kehidupan, salah satunya dalam bidang pelayanan dan bisnis, khususnya pada industri restoran, karena pemanfaatan pemanfaatan teknologi ini menjadi strategi penting bagi pemilik usaha untuk meningkatkan efisiensi operasional sekaligus kualitas pelayanan kepada pelanggan. salah satu bentuk dari penerapan teknologi tersebut adalah penggunaan sistem menu digital sebagai pengganti menu fisik sehingga lebih efisien.

Pada umumnya, restoran masih banyak yang menggunakan menu cetak sebagai media utama untuk menyampaikan informasi menu makanan dan minuman kepada pelanggan, Namun, menu fisik banyak memiliki keterbatasan, seperti mudah rusak, kurang higienis, memerlukan biaya cetak yang tidak sedikit, serta tidak fleksibel dalam merubah menu. Maka hal ini tentu dapat menghambat efektivitas pelayanan dan menambah beban operasional restoran.

Kemudian seiring dengan meningkatnya penggunaan perangkat digital seperti smartphone dan tablet, menu digital dapat menjadi solusi yang relevan dan juga efisien. Menu digital ini memungkinkan pelanggan dapat mengakses daftar menu secara mandiri dengan tampilan yang lebih jelas, informatif, dan menarik. Selain itu dengan adanya menu digital ini dapat mempercepat proses pelayanan, mengurangi kesalahpahaman antara penjual dan pembeli dalam pencatatan pesanan, sehingga dapat meningkatkan kepuasan pelanggan.

Bagi pihak pengelola restoran, sistem menu digital dapat mempermudah untuk mengubah harga menu secara real time tanpa harus mencetak ulang menu sebelumnya. Dengan demikian, sistem menu digital ini tidak hanya membantu meningkatkan kualitas layanan kepada pelanggan saja, tetapi juga mendukung efisiensi manajemen restoran. Oleh karena itu, pengembangan sistem menu digital restoran menjadi langkah strategis untuk menjawab kebutuhan terhadap layanan yang modern, efektif, dan adaptif terhadap perkembangan teknologi.

## **1.2 Rumusan Masalah**

Dari latar belakang dapat dirumuskan pada rumusan masalah berikut:

- a. Bagaimana merancang aplikasi pemesanan makanan berbasis desktop menggunakan Python?
- b. Bagaimana merancang sistem login agar pengguna dapat memilih peran sebagai Customer atau Admin sesuai dengan hak aksesnya?
- c. Bagaimana menampilkan menu dan menghitung total pesanan secara otomatis?
- d. Bagaimana menghasilkan struk pemesanan dalam bentuk PDF?
- e. Bagaimana menyediakan fitur admin untuk mengelola menu?

## **1.3 Tujuan**

Tujuan dari pembuatan aplikasi ini adalah:

- a. Membuat aplikasi pemesanan menu berbasis desktop.
- b. Memudahkan proses pemesanan bagi pengguna.
- c. Memudahkan admin atau pengelola restaurant dalam memperbarui harga menu.
- d. Mengurangi kesalahan pencatatan dan perhitungan harga.
- e. Menyediakan fitur pembuatan struk secara otomatis.

## **BAB 2**

### **ANALISIS DAN PERANCANGAN**

#### **2.1 Analisis Kebutuhan Aplikasi**

##### **2.1.1 Kebutuhan Fungsional**

1. Menampilkan daftar menu makanan dan minuman beserta harganya.
2. Menampilkan gambar menu yang dipilih.
3. Menambahkan dan menghapus menu dari daftar pesanan.
4. Menghitung total harga pesanan secara otomatis.
5. Menghasilkan Order ID secara acak dan berbeda-beda.
6. Menyimpan untuk pemesanan dalam bentuk file PDF.
7. Menyediakan login sebagai customer dan admin.
8. Admin dapat mengedit harga menu dan menyimpannya ke file JSON.

##### **2.1.2 Kebutuhan Non-fungsional**

1. Aplikasi mudah digunakan.
2. Berjalan pada sistem operasi yang mengandung Python.
3. Tampilan antarmuka sederhana dan informatif.
4. Data menu tersimpan secara lokal dalam file JSON.

#### **2.2 Sketsa Desain Antarmuka**

Desain antarmuka aplikasi terdiri dari beberapa bagian utama:

a. Halaman Login

Berisi pilihan login sebagai Customer atau Admin.

b. Menu Panel (Kiri)

Menampilkan daftar menu makan dan minuman beserta harga.

c. Display Panel (Tengah)

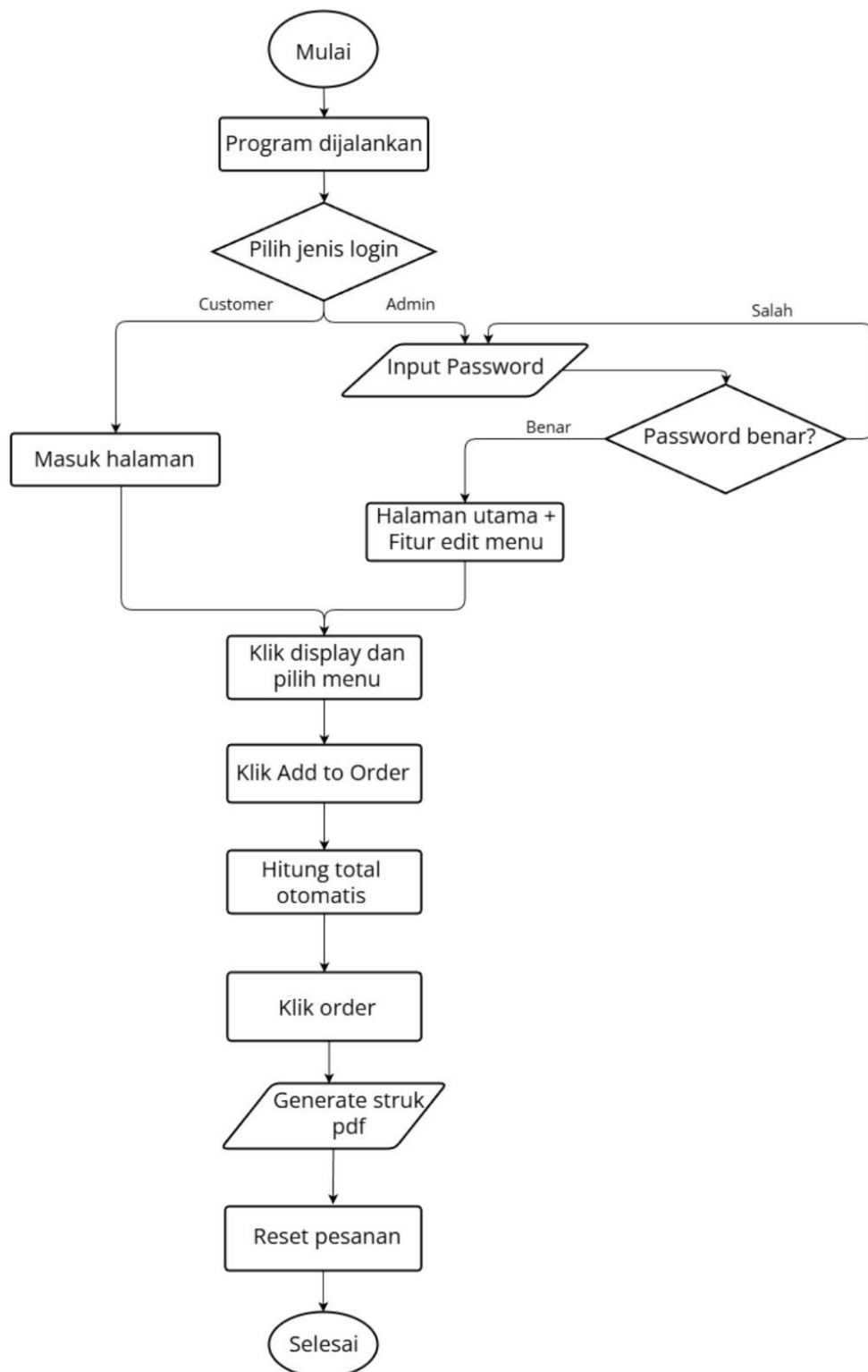
Menampilkan gambar dan nama menu yang dipilih.

d. Order Panel (Kanan)

Menampilkan Order ID, daftar pesanan, total harga, serta tombol Order dan Edit Menu (untuk admin).

Desain antarmuka dibuat menggunakan library *Tkinter* dan *ttk* dengan tujuan agar tampilan sederhana, rapi, dan mudah dipahami pengguna.

## 2.3 Diagram Alur



Gambar 1. Diagram Alur

## BAB 3

### IMPLEMENTASI

#### 3.1 Penjelasan Kode Program

Aplikasi Pemesanan Menu Kedai FMR dikembangkan menggunakan bahasa pemrograman Python dengan library Tkinter sebagai antarmuka grafis (GUI). Selain itu, beberapa library pendukung seperti PIL untuk pengolahan gambar, JSON untuk penyimpanan data menu, dan ReportLab untuk pembuatan struk dalam bentuk PDF.

Berikut adalah penjelasan singkat bagian-bagian penting dari kode program:

a. Sistem Login (Customer dan Admin)

Program ini diawali dengan tampilan login yang memungkinkan pengguna dapat memilih peran sebagai *Customer* atau *Admin*. Jika memilih admin, pengguna diwajibkan memasukkan password sebelum dapat mengakses fitur pengeditan menu.

```
def login_window():  
    ...  
    def select_customer():  
        ...  
    def select_admin():  
        password_entry = ttk.Entry(password_window, show="*")  
        password_entry.pack()
```

b. Penyimpanan dan Pengolahan Data Menu

Data menu dan harga disimpan dalam file menu.json. Hal ini memungkinkan admin untuk mengubah harga menu tanpa harus mengubah kode program secara langsung.

```
MENU_FILE = "menu.json"
```

Jika file tidak ditemukan, sistem akan memuat menu default yang sudah ditentukan di dalam program.

c. Fitur Edit Harga oleh Admin

Fitur edit harga hanya dapat diakses oleh pengguna yang login sebagai Admin.

Setelah Admin berhasil login, tombol edit harga akan muncul pada panel pesanan.

Tombol ini digunakan untuk membuka jendela khusus untuk pengeditan menu. Pada jendela edit harga, admin dapat melihat daftar seluruh menu, mengubah harga setiap menu, dan juga menyimpan perubahan harga.

Perubahan harga akan disimpan ke dalam file menu.json sehingga data tetap tersimpan meskipun aplikasi ditutup. Cuplikan kode berikut menunjukkan proses perubahan harga menu:

```
def save_menu(menu):  
    with open(MENU_FILE, 'w') as f:  
        json.dump(menu, f)
```

Setelah data disimpan, sistem akan memuat ulang menu agar harga terbaru langsung ditampilkan pada aplikasi.

d. Tampilan Menu dan Gambar

Setiap menu ditampilkan dalam bentuk tabel dan tombol Display. Saat tombol ditekan, gambar menu akan ditampilkan di panel bagian tengah aplikasi.

```
def displaysteak_ayam():  
    steak_ayamDishframe.configure(relief="sunken", style="SelectedDish.TFrame")  
    nasi_gorengDishframe.configure(style="DishFrame.TFrame")
```

```
displayLabel.configure(  
    image=steak_ayamImage,  
    text="Steak Ayam",  
    font=('Helvetica', 14, "bold"),  
    foreground="white",  
    compound="bottom",  
    padding=(5, 5, 5, 5),  
)
```

e. Proses Pemesanan

Pengguna dapat menambahkan menu ke dalam daftar pesanan dengan menekan tombol *Add to Order*. Sistem akan mencatat pesanan langsung, memperbarui total harga secara otomatis.

```
def add():  
    update_total = int(order_total) + prices[displayLabel.cget("text")]
```

Selain itu, pengguna juga dapat menghapus pesanan yang telah ditambahkan.

f. Perhitungan Total Harga

Total harga dihitung secara otomatis setiap kali terjadi penambahan atau penghapusan menu, sehingga meminimalkan kesalahan perhitungan.



g. Pembuatan Struk PDF

Setelah tombol *Order* ditekan, sistem akan menghasilkan struk pemesanan dalam bentuk file PDF menggunakan library ReportLab.

```
def generate_pdf(order_id, transaction_list, total, order_day, order_time):  
    filename = f"{order_id}.pdf"  
    doc = SimpleDocTemplate(filename, pagesize=letter)  
    styles = getSampleStyleSheet()  
    story = []
```

Struk berisi informasi Order ID, tanggal, waktu, daftar pesanan, dan total pembayaran.

```
def ORDER_ID():  
    numbers = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]  
    letters = ['A', 'B', 'C', 'D', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T',  
    'U', 'V', 'W', 'X', 'Y', 'Z']  
    order_id = "FMR_"  
    random_letters = ""  
    random_digits = ""  
    for i in range(0, 3):  
        random_letters += random.choice(letters)  
        random_digits += str(random.choice(numbers))  
  
    order_id += random_letters + random_digits  
    return order_id
```

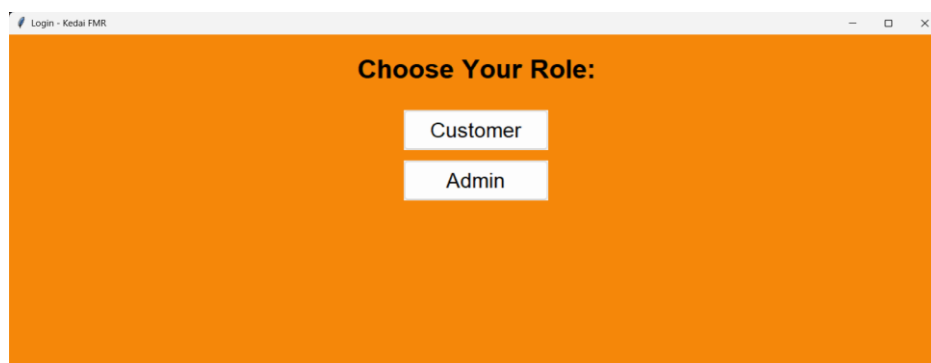
### 3.2 Cara Penggunaan

Berikut alur atau langkah-langkah penggunaan Aplikasi Pemesanan Menu Kedai FMR:

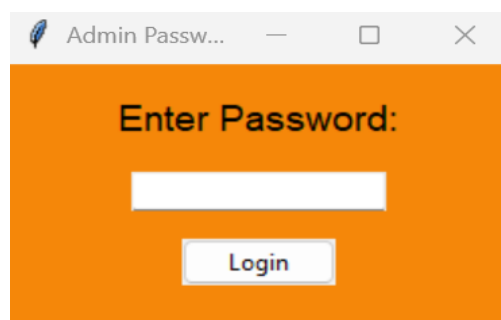
1. Program dijalankan oleh pengguna.
2. Sistem menampilkan halaman login.
3. Pengguna memilih jenis login sebagai Customer atau Admin.
4. Jika memilih Customer, sistem langsung menampilkan halaman utama aplikasi.
5. Jika memilih Admin, sistem meminta pengguna memasukkan password.

6. Sistem melakukan pengecekan password Admin:
  - a. Jika password benar, admin diarahkan ke halaman utama aplikasi yang dilengkapi fitur edit dan pengelolaan menu.
  - b. Jika Password salah, sistem mengembalikan admin ke halaman login.
7. Pada halaman utama, pengguna (Customer maupun Admin) dapat melihat dan memilih menu yang tersedia.
8. Admin dapat melakukan pengeditan menu seperti mengubah harga menu.
9. Pengguna memilih menu dan menampilkan detail gambar menu dengan menekan tombol display.
10. Pengguna menambahkan menu ke dalam daftar menu dengan menekan tombol *Add to Order*.
11. Sistem secara otomatis menghitung total harga pesanan.
12. Pengguna menekan tombol *Order* untuk mengkonfirmasi pesanan.
13. Sistem menghasilkan struk pemesanan secara otomatis dalam bentuk file PDF.
14. Sistem mereset data pesanan setelah transaksi selesai.
15. Proses pemesanan selesai dan aplikasi siap digunakan untuk pesanan berikutnya.

### 3.3 Tampilan Aplikasi



Gambar 2. Halaman Login



Gambar 3. Input Password



Gambar 4. Display Admin



Gambar 5. Display Customer



Gambar 6. Display Menu

Admin Edit Menu

Edit Menu Items

Steak Ayam	28000
Nasi Goreng	15000
Ayam Goreng	18000
Ayam Bakar	22000
Soto Ayam	16000
Rawon	27000
Mendoan	10000
Tahu Isi	10000
Risol	12000
Lumpia	12000
Es Teh	5000
Es Jeruk	6000
Lemon Tea	6000
Es Coklat	10000

Save Changes

Gambar 7. Display Edit Menu

## LAMPIRAN

```
import tkinter as tk
from tkinter import ttk
from tkinter import *
from PIL import Image, ImageTk
import random
from datetime import date
from datetime import datetime
from reportlab.pdfgen import canvas
from reportlab.lib.pagesizes import letter
from reportlab.lib.styles import getSampleStyleSheet
from reportlab.platypus import SimpleDocTemplate, Paragraph, Spacer
import tkinter.messagebox as messagebox
import json
import os

# File untuk menyimpan menu
MENU_FILE = "menu.json"

# Fungsi untuk load menu dari file
def load_menu():
    if os.path.exists(MENU_FILE):
        with open(MENU_FILE, 'r') as f:
            return json.load(f)
    else:
        # Menu awal
        return {
            "Steak Ayam": 28000,
            "Nasi Goreng": 15000,
            "Ayam Goreng": 18000,
            "Ayam Bakar": 22000,
            "Soto Ayam": 16000,
            "Rawon": 27000,
```

```

        "Mendoan": 10000,
        "Tahu Isi": 10000,
        "Risol": 12000,
        "Lumpia": 12000,
        "Es Teh": 5000,
        "Es Jeruk": 6000,
        "Lemon Tea": 6000,
        "Es Coklat": 10000,
    }

# Fungsi untuk save menu ke file
def save_menu(menu):
    with open(MENU_FILE, 'w') as f:
        json.dump(menu, f)

prices = load_menu()

# Fungsi untuk membuat PDF struk
def generate_pdf(order_id, transaction_list, total, order_day, order_time):
    filename = f"{order_id}.pdf"
    doc = SimpleDocTemplate(filename, pagesize=letter)
    styles = getSampleStyleSheet()
    story = []

    # Header
    story.append(Paragraph("Kedai FMR", styles['Title']))
    story.append(Spacer(1, 12))
    story.append(Paragraph("_____",
styles['Normal']))
    story.append(Spacer(1, 12))
    story.append(Paragraph(f"Order ID: {order_id}", styles['Normal']))
    story.append(Paragraph(f"Date: {order_day.strftime('%x')}", styles['Normal']))
    story.append(Paragraph(f"Time: {order_time.strftime('%X')}", styles['Normal']))
    story.append(Spacer(1, 12))

```

```

# Items
for item in transaction_list:
    story.append(Paragraph(item, styles['Normal']))
    story.append(Spacer(1, 12))
    story.append(Paragraph(total, styles['Normal']))

doc.build(story)

messagebox.showinfo("Success", f"Receipt saved as {filename}")

# Kode Pemesanan
def ORDER_ID():
    numbers = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
    letters = ['A', 'B', 'C', 'D', 'F', 'G', 'H', 'T', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S',
'T', 'U', 'V', 'W', 'X', 'Y', 'Z']
    order_id = "FMR_"
    random_letters = ""
    random_digits = ""
    for i in range(0, 3):
        random_letters += random.choice(letters)
        random_digits += str(random.choice(numbers))

    order_id += random_letters + random_digits
    return order_id

def main_app(is_admin=False):
    root = Tk()
    root.title("Kedai FMR")

    # Fungsi untuk reload menu (untuk update setelah edit admin)
    def reload_menu():
        global prices
        prices = load_menu()
        # Update labels harga di menu

```

```

steak_ayamDishLabel.config(text=f"Steak Ayam .... Rp {prices['Steak Ayam']:,}")
nasi_gorengDishLabel.config(text=f"Nasi Goreng .... Rp {prices['Nasi
Goreng']:,}")
ayam_gorengDishLabel.config(text=f"Ayam Goreng .... Rp {prices['Ayam
Goreng']:,}")
ayam_bakarDishLabel.config(text=f"Ayam Bakar .... Rp {prices['Ayam
Bakar']:,}")
sotoDishLabel.config(text=f"Soto Ayam .... Rp {prices['Soto Ayam']:,}")
rawonDishLabel.config(text=f"Rawon .... Rp {prices['Rawon']:,}")
mendoanDishLabel.config(text=f"Mendoan .... Rp {prices['Mendoan']:,}")
tahu_isiDishLabel.config(text=f"Tahu Isi .... Rp {prices['Tahu Isi']:,}")
risolDishLabel.config(text=f"Risol .... Rp {prices['Risol']:,}")
lumpiaDishLabel.config(text=f"Lumpia .... Rp {prices['Lumpia']:,}")
es_tehDishLabel.config(text=f"Es Teh .... Rp {prices['Es Teh']:,}")
es_jerukDishLabel.config(text=f"Es Jeruk .... Rp {prices['Es Jeruk']:,}")
lemon_teaDishLabel.config(text=f"Lemon Tea .... Rp {prices['Lemon Tea']:,}")
es_coklatDishLabel.config(text=f"Es Coklat .... Rp {prices['Es Coklat']:,}")

# Fungsi untuk tampilan edit admin
def admin_edit():
    edit_win = Toplevel(root)
    edit_win.title("Admin Edit Menu")
    edit_win.geometry("400x600")

    ttk.Label(edit_win, text="Edit Menu Items").pack(pady=10)

    entries = {}
    for item, price in prices.items():
        frame = ttk.Frame(edit_win)
        frame.pack(fill='x', padx=10, pady=2)
        ttk.Label(frame, text=item, width=15).pack(side='left')
        entry = ttk.Entry(frame)
        entry.insert(0, str(price))
        entry.pack(side='right')

```



```

entries[item] = entry

def save_changes():
    try:
        new_prices = {item: int(entry.get()) for item, entry in entries.items()}
        save_menu(new_prices)
        reload_menu()
        messagebox.showinfo("Success", "Menu updated successfully!")
        edit_win.destroy()
    except ValueError:
        messagebox.showerror("Error", "Please enter valid numbers for prices.")

    ttk.Button(edit_win, text="Save Changes",
command=save_changes).pack(pady=10)

#bagian tombol untuk menambah pesanan
def add():
    current_order = orderTransaction.cget("text")
    added_dish = displayLabel.cget("text") + " Rp" +
str(prices[displayLabel.cget("text")])
    if current_order:
        updated_order = current_order + "\n" + added_dish
    else:
        updated_order = added_dish
    orderTransaction.configure(text=updated_order)

# pembaruan total harga pesanan
order_total = orderTotalLabel.cget("text").replace("TOTAL : ", "")
order_total = order_total.replace("Rp ", "")
update_total = int(order_total) + prices[displayLabel.cget("text")]
orderTotalLabel.configure(text="TOTAL : Rp " + str(update_total))

#bagian tombol untuk menghapus pesanan
def remove():

```

```

dish_to_remove = displayLabel.cget("text") + " Rp" +
str(prices[displayLabel.cget("text")])
current_order = orderTransaction.cget("text")
transaction_list = current_order.split("\n") if current_order else []

if dish_to_remove in transaction_list:
    transaction_list.remove(dish_to_remove)
    updated_order = "\n".join(transaction_list) if transaction_list else ""
    orderTransaction.configure(text=updated_order)

# pembaruan total harga pesanan
order_total = orderTotalLabel.cget("text").replace("TOTAL : ",
""),replace("Rp ", "")
update_total = int(order_total) - prices[displayLabel.cget("text")]
orderTotalLabel.configure(text="TOTAL : Rp " + str(update_total))

#display sections
def displaysteak_ayam():
    steak_ayamDishframe.configure(relief="sunken", style="SelectedDish.TFrame")
    nasi_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_bakarDishframe.configure(style="DishFrame.TFrame")
    sotoDishframe.configure(style="DishFrame.TFrame")
    rawonDishframe.configure(style="DishFrame.TFrame")
    mendoanDishframe.configure(style="DishFrame.TFrame")
    tahu_isiDishframe.configure(style="DishFrame.TFrame")
    risolDishframe.configure(style="DishFrame.TFrame")
    lumpiaDishframe.configure(style="DishFrame.TFrame")
    es_tehDishframe.configure(style="DishFrame.TFrame")
    es_jerukDishframe.configure(style="DishFrame.TFrame")
    lemon_teaDishframe.configure(style="DishFrame.TFrame")
    es_coklatDishframe.configure(style="DishFrame.TFrame")

displayLabel.configure(

```

```

        image=steak_ayamImage,
        text="Steak Ayam",
        font=('Helvetica', 14, "bold"),
        foreground="white",
        compound="bottom",
        padding=(5, 5, 5, 5),
    )

```

```

def displaynasi_goreng():

```

```

    nasi_gorengDishframe.configure(relief="sunken", style="SelectedDish.TFrame")
    steak_ayamDishframe.configure(style="DishFrame.TFrame")
    ayam_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_bakarDishframe.configure(style="DishFrame.TFrame")
    sotoDishframe.configure(style="DishFrame.TFrame")
    rawonDishframe.configure(style="DishFrame.TFrame")
    mendoanDishframe.configure(style="DishFrame.TFrame")
    tahu_isiDishframe.configure(style="DishFrame.TFrame")
    risolDishframe.configure(style="DishFrame.TFrame")
    lumpiaDishframe.configure(style="DishFrame.TFrame")
    es_tehDishframe.configure(style="DishFrame.TFrame")
    es_jerukDishframe.configure(style="DishFrame.TFrame")
    lemon_teaDishframe.configure(style="DishFrame.TFrame")
    es_coklatDishframe.configure(style="DishFrame.TFrame")

```

```

displayLabel.configure(
    image=nasi_gorengImage,
    text="Nasi Goreng",
    font=('Helvetica', 14, "bold"),
    foreground="white",
    compound="bottom",
    padding=(5, 5, 5, 5),
)

```

```

def displayayam_goreng():

```

```

ayam_gorengDishframe.configure(relief="sunken", style="SelectedDish.TFrame")
steak_ayamDishframe.configure(style="DishFrame.TFrame")
nasi_gorengDishframe.configure(style="DishFrame.TFrame")
ayam_bakarDishframe.configure(style="DishFrame.TFrame")
sotoDishframe.configure(style="DishFrame.TFrame")
rawonDishframe.configure(style="DishFrame.TFrame")
mendoanDishframe.configure(style="DishFrame.TFrame")
tahu_isiDishframe.configure(style="DishFrame.TFrame")
risolDishframe.configure(style="DishFrame.TFrame")
lumpiaDishframe.configure(style="DishFrame.TFrame")
es_tehDishframe.configure(style="DishFrame.TFrame")
es_jerukDishframe.configure(style="DishFrame.TFrame")
lemon_teaDishframe.configure(style="DishFrame.TFrame")
es_coklatDishframe.configure(style="DishFrame.TFrame")

```

```

displayLabel.configure(
    image=ayam_gorengImage,
    text="Ayam Goreng",
    font=('Helvetica', 14, "bold"),
    foreground="white",
    compound="bottom",
    padding=(5, 5, 5, 5),
)

```

```

def displayayam_bakar():

```

```

    ayam_bakarDishframe.configure(relief="sunken", style="SelectedDish.TFrame")
    steak_ayamDishframe.configure(style="DishFrame.TFrame")
    nasi_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_gorengDishframe.configure(style="DishFrame.TFrame")
    sotoDishframe.configure(style="DishFrame.TFrame")
    rawonDishframe.configure(style="DishFrame.TFrame")
    mendoanDishframe.configure(style="DishFrame.TFrame")
    tahu_isiDishframe.configure(style="DishFrame.TFrame")
    risolDishframe.configure(style="DishFrame.TFrame")

```

```

lumpiaDishframe.configure(style="DishFrame.TFrame")
es_tehDishframe.configure(style="DishFrame.TFrame")
es_jerukDishframe.configure(style="DishFrame.TFrame")
lemon_teaDishframe.configure(style="DishFrame.TFrame")
es_coklatDishframe.configure(style="DishFrame.TFrame")

```

```

displayLabel.configure(
    image=ayam_bakarImage,
    text="Ayam Bakar",
    font=('Helvetica', 14, "bold"),
    foreground="white",
    compound="bottom",
    padding=(5, 5, 5, 5),
)

```

```

def displaysoto():

```

```

    sotoDishframe.configure(relief="sunken", style="SelectedDish.TFrame")
    steak_ayamDishframe.configure(style="DishFrame.TFrame")
    nasi_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_bakarDishframe.configure(style="DishFrame.TFrame")
    rawonDishframe.configure(style="DishFrame.TFrame")
    mendoanDishframe.configure(style="DishFrame.TFrame")
    tahu_isiDishframe.configure(style="DishFrame.TFrame")
    risolDishframe.configure(style="DishFrame.TFrame")
    lumpiaDishframe.configure(style="DishFrame.TFrame")
    es_tehDishframe.configure(style="DishFrame.TFrame")
    es_jerukDishframe.configure(style="DishFrame.TFrame")
    lemon_teaDishframe.configure(style="DishFrame.TFrame")
    es_coklatDishframe.configure(style="DishFrame.TFrame")

```

```

displayLabel.configure(
    image=sotoImage,
    text="Soto Ayam",

```

```
font=('Helvetica', 14, "bold"),
foreground="white",
compound="bottom",
padding=(5, 5, 5, 5),
)
```

```
def displayrawon():
```

```
    rawonDishframe.configure(relief="sunken", style="SelectedDish.TFrame")
    steak_ayamDishframe.configure(style="DishFrame.TFrame")
    nasi_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_bakarDishframe.configure(style="DishFrame.TFrame")
    sotoDishframe.configure(style="DishFrame.TFrame")
    mendoanDishframe.configure(style="DishFrame.TFrame")
    tahu_isiDishframe.configure(style="DishFrame.TFrame")
    risolDishframe.configure(style="DishFrame.TFrame")
    lumpiaDishframe.configure(style="DishFrame.TFrame")
    es_tehDishframe.configure(style="DishFrame.TFrame")
    es_jerukDishframe.configure(style="DishFrame.TFrame")
    lemon_teaDishframe.configure(style="DishFrame.TFrame")
    es_coklatDishframe.configure(style="DishFrame.TFrame")
```

```
displayLabel.configure(
    image=rawonImage,
    text="Rawon",
    font=('Helvetica', 14, "bold"),
    foreground="white",
    compound="bottom",
    padding=(5, 5, 5, 5),
)
```

```
def displaymendoan():
```

```
    mendoanDishframe.configure(relief="sunken", style="SelectedDish.TFrame")
    steak_ayamDishframe.configure(style="DishFrame.TFrame")
```

```

nasi_gorengDishframe.configure(style="DishFrame.TFrame")
ayam_gorengDishframe.configure(style="DishFrame.TFrame")
ayam_bakarDishframe.configure(style="DishFrame.TFrame")
sotoDishframe.configure(style="DishFrame.TFrame")
rawonDishframe.configure(style="DishFrame.TFrame")
tahu_isiDishframe.configure(style="DishFrame.TFrame")
risolDishframe.configure(style="DishFrame.TFrame")
lumpiaDishframe.configure(style="DishFrame.TFrame")
es_tehDishframe.configure(style="DishFrame.TFrame")
es_jerukDishframe.configure(style="DishFrame.TFrame")
lemon_teaDishframe.configure(style="DishFrame.TFrame")
es_coklatDishframe.configure(style="DishFrame.TFrame")

```

```

displayLabel.configure(
    image=mendoanImage,
    text="Mendoan",
    font=('Helvetica', 14, "bold"),
    foreground="white",
    compound="bottom",
    padding=(5, 5, 5, 5),
)

```

```

def displaytahu_isi():
    tahu_isiDishframe.configure(relief="sunken", style="SelectedDish.TFrame")
    steak_ayamDishframe.configure(style="DishFrame.TFrame")
    nasi_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_bakarDishframe.configure(style="DishFrame.TFrame")
    sotoDishframe.configure(style="DishFrame.TFrame")
    rawonDishframe.configure(style="DishFrame.TFrame")
    mendoanDishframe.configure(style="DishFrame.TFrame")
    risolDishframe.configure(style="DishFrame.TFrame")
    lumpiaDishframe.configure(style="DishFrame.TFrame")
    es_tehDishframe.configure(style="DishFrame.TFrame")

```

```
es_jerukDishframe.configure(style="DishFrame.TFrame")
lemon_teaDishframe.configure(style="DishFrame.TFrame")
es_coklatDishframe.configure(style="DishFrame.TFrame")
```

```
displayLabel.configure(
    image=tahu_isiImage,
    text="Tahu Isi",
    font=('Helvetica', 14, "bold"),
    foreground="white",
    compound="bottom",
    padding=(5, 5, 5, 5),
)
```

```
def displayrisol():
```

```
    risolDishframe.configure(relief="sunken", style="SelectedDish.TFrame")
    steak_ayamDishframe.configure(style="DishFrame.TFrame")
    nasi_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_bakarDishframe.configure(style="DishFrame.TFrame")
    sotoDishframe.configure(style="DishFrame.TFrame")
    rawonDishframe.configure(style="DishFrame.TFrame")
    mendoanDishframe.configure(style="DishFrame.TFrame")
    tahu_isiDishframe.configure(style="DishFrame.TFrame")
    lumpiaDishframe.configure(style="DishFrame.TFrame")
    es_tehDishframe.configure(style="DishFrame.TFrame")
    es_jerukDishframe.configure(style="DishFrame.TFrame")
    lemon_teaDishframe.configure(style="DishFrame.TFrame")
    es_coklatDishframe.configure(style="DishFrame.TFrame")
```

```
displayLabel.configure(
    image=risolImage,
    text="Risol",
    font=('Helvetica', 14, "bold"),
    foreground="white",
```



```

        compound="bottom",
        padding=(5, 5, 5, 5),
    )

```

```

def displaylumpia():

```

```

    lumpiaDishframe.configure(relief="sunken", style="SelectedDish.TFrame")
    steak_ayamDishframe.configure(style="DishFrame.TFrame")
    nasi_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_bakarDishframe.configure(style="DishFrame.TFrame")
    sotoDishframe.configure(style="DishFrame.TFrame")
    rawonDishframe.configure(style="DishFrame.TFrame")
    mendoanDishframe.configure(style="DishFrame.TFrame")
    tahu_isiDishframe.configure(style="DishFrame.TFrame")
    risolDishframe.configure(style="DishFrame.TFrame")
    es_tehDishframe.configure(style="DishFrame.TFrame")
    es_jerukDishframe.configure(style="DishFrame.TFrame")
    lemon_teaDishframe.configure(style="DishFrame.TFrame")
    es_coklatDishframe.configure(style="DishFrame.TFrame")

```

```

displayLabel.configure(
    image=lumpiaImage,
    text="Lumpia",
    font=('Helvetica', 14, "bold"),
    foreground="white",
    compound="bottom",
    padding=(5, 5, 5, 5),
)

```

```

def displayes_teh():

```

```

    es_tehDishframe.configure(relief="sunken", style="SelectedDish.TFrame")
    steak_ayamDishframe.configure(style="DishFrame.TFrame")
    nasi_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_gorengDishframe.configure(style="DishFrame.TFrame")

```

```

ayam_bakarDishframe.configure(style="DishFrame.TFrame")
sotoDishframe.configure(style="DishFrame.TFrame")
rawonDishframe.configure(style="DishFrame.TFrame")
mendoanDishframe.configure(style="DishFrame.TFrame")
tahu_isiDishframe.configure(style="DishFrame.TFrame")
risolDishframe.configure(style="DishFrame.TFrame")
lumpiaDishframe.configure(style="DishFrame.TFrame")
es_jerukDishframe.configure(style="DishFrame.TFrame")
lemon_teaDishframe.configure(style="DishFrame.TFrame")
es_coklatDishframe.configure(style="DishFrame.TFrame")

```

```

displayLabel.configure(
    image=es_tehImage,
    text="Es Teh",
    font=('Helvetica', 14, "bold"),
    foreground="white",
    compound="bottom",
    padding=(5, 5, 5, 5),
)

```

```

def displayes_jeruk():

```

```

    es_jerukDishframe.configure(relief="sunken", style="SelectedDish.TFrame")
    steak_ayamDishframe.configure(style="DishFrame.TFrame")
    nasi_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_bakarDishframe.configure(style="DishFrame.TFrame")
    sotoDishframe.configure(style="DishFrame.TFrame")
    rawonDishframe.configure(style="DishFrame.TFrame")
    mendoanDishframe.configure(style="DishFrame.TFrame")
    tahu_isiDishframe.configure(style="DishFrame.TFrame")
    risolDishframe.configure(style="DishFrame.TFrame")
    lumpiaDishframe.configure(style="DishFrame.TFrame")
    es_tehDishframe.configure(style="DishFrame.TFrame")
    lemon_teaDishframe.configure(style="DishFrame.TFrame")

```

```
es_coklatDishframe.configure(style="DishFrame.TFrame")
```

```
displayLabel.configure(  
    image=es_jerukImage,  
    text="Es Jeruk",  
    font=('Helvetica', 14, "bold"),  
    foreground="white",  
    compound="bottom",  
    padding=(5, 5, 5, 5),  
)
```

```
def displaylemon_tea():
```

```
    lemon_teaDishframe.configure(relief="sunken", style="SelectedDish.TFrame")  
    steak_ayamDishframe.configure(style="DishFrame.TFrame")  
    nasi_gorengDishframe.configure(style="DishFrame.TFrame")  
    ayam_gorengDishframe.configure(style="DishFrame.TFrame")  
    ayam_bakarDishframe.configure(style="DishFrame.TFrame")  
    sotoDishframe.configure(style="DishFrame.TFrame")  
    rawonDishframe.configure(style="DishFrame.TFrame")  
    mendoanDishframe.configure(style="DishFrame.TFrame")  
    tahu_isiDishframe.configure(style="DishFrame.TFrame")  
    risolDishframe.configure(style="DishFrame.TFrame")  
    lumpiaDishframe.configure(style="DishFrame.TFrame")  
    es_tehDishframe.configure(style="DishFrame.TFrame")  
    es_jerukDishframe.configure(style="DishFrame.TFrame")  
    es_coklatDishframe.configure(style="DishFrame.TFrame")
```

```
displayLabel.configure(  
    image=lemon_teaImage,  
    text="Lemon Tea",  
    font=('Helvetica', 14, "bold"),  
    foreground="white",  
    compound="bottom",  
    padding=(5, 5, 5, 5),
```

```

)

def displayes_coklat():
    es_coklatDishframe.configure(relief="sunken", style="SelectedDish.TFrame")
    steak_ayamDishframe.configure(style="DishFrame.TFrame")
    nasi_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_gorengDishframe.configure(style="DishFrame.TFrame")
    ayam_bakarDishframe.configure(style="DishFrame.TFrame")
    sotoDishframe.configure(style="DishFrame.TFrame")
    rawonDishframe.configure(style="DishFrame.TFrame")
    mendoanDishframe.configure(style="DishFrame.TFrame")
    tahu_isiDishframe.configure(style="DishFrame.TFrame")
    risolDishframe.configure(style="DishFrame.TFrame")
    lumpiaDishframe.configure(style="DishFrame.TFrame")
    es_tehDishframe.configure(style="DishFrame.TFrame")
    es_jerukDishframe.configure(style="DishFrame.TFrame")
    lemon_teaDishframe.configure(style="DishFrame.TFrame")

    displayLabel.configure(
        image=es_coklatImage,
        text="Es Coklat",
        font=('Helvetica', 14, "bold"),
        foreground="white",
        compound="bottom",
        padding=(5, 5, 5, 5),
    )

def order():
    new_receipt = orderIDLabel.cget("text").replace("ORDER ID : ", "")
    transaction_list = orderTransaction.cget("text").split("\n") if
orderTransaction.cget("text") else []

    order_day = date.today()
    order_time = datetime.now()

```

```

total = orderTotalLabel.cget("text")

generate_pdf(new_receipt, transaction_list, total, order_day, order_time)

# pengaturan ulang
orderTotalLabel.configure(text="TOTAL : Rp 0")
orderIDLabel.configure(text="ORDER ID : " + ORDER_ID())
orderTransaction.configure(text="")

#Styling and Image
s = ttk.Style()
s.configure('MainFrame.TFrame', background="#2B2B28")
s.configure('MenuFrame.TFrame', background="#6B2301")
s.configure('DisplayFrame.TFrame', background="#0F1110")
s.configure('OrderFrame.TFrame', background="#F58709")
s.configure('DishFrame.TFrame', background="#F58709", relief="raised")
s.configure('SelectedDish.TFrame', background="#F58709", relief="sunken")
s.configure('MenuLabel.TLabel',
            background="#6B2301",
            font=("Arial", 13, "italic"),
            foreground="white",
            padding=(2, 2, 2, 2),
            width=25
            )
s.configure('orderTotalLabel.TLabel',
            background="#0F1110",
            font=("Arial", 10, "bold"),
            foreground="white",
            padding=(2, 2, 2, 2),
            anchor="w")
s.configure('orderTransaction.TLabel',
            background="#4A4A48",
            font=("Helvetica", 12),

```

```

        foreground="white",
        wraplength=170,
        anchor="nw",
        padding=(3, 3, 3, 3),
    )

#region images
# top banner images
LogoImageObject = Image.open("images/logo.png").resize((130, 130))
LogoImage = ImageTk.PhotoImage(LogoImageObject)

TopBannerImageObject = Image.open("images/top banner.png").resize((1145, 130))
TopBannerImage = ImageTk.PhotoImage(TopBannerImageObject)

#menu images
displayDefaultImageObject = Image.open("images/menu makanan.png").resize((650,
360))
displayDefaultImage = ImageTk.PhotoImage(displayDefaultImageObject)

steak_ayamImageObject = Image.open("menu/steak ayam.png").resize((350, 334))
steak_ayamImage = ImageTk.PhotoImage(steak_ayamImageObject)

nasi_gorengImageObject = Image.open("menu/nasi goreng.png").resize((350, 334))
nasi_gorengImage = ImageTk.PhotoImage(nasi_gorengImageObject)

ayam_gorengImageObject = Image.open("menu/ayam goreng.png").resize((350,
334))
ayam_gorengImage = ImageTk.PhotoImage(ayam_gorengImageObject)

ayam_bakarImageObject = Image.open("menu/ayam bakar.png").resize((350, 334))
ayam_bakarImage = ImageTk.PhotoImage(ayam_bakarImageObject)

sotoImageObject = Image.open("menu/soto ayam.png").resize((350, 334))
sotoImage = ImageTk.PhotoImage(sotoImageObject)

```

```

rawonImageObject = Image.open("menu/rawon.png").resize((350, 334))
rawonImage = ImageTk.PhotoImage(rawonImageObject)

mendoanImageObject = Image.open("menu/mendoan.png").resize((350, 334))
mendoanImage = ImageTk.PhotoImage(mendoanImageObject)

tahu_isiImageObject = Image.open("menu/tahu isi.png").resize((350, 334))
tahu_isiImage = ImageTk.PhotoImage(tahu_isiImageObject)

risolImageObject = Image.open("menu/risol.png").resize((350, 334))
risolImage = ImageTk.PhotoImage(risolImageObject)

lumpiaImageObject = Image.open("menu/lumpia.png").resize((350, 334))
lumpiaImage = ImageTk.PhotoImage(lumpiaImageObject)

es_tehImageObject = Image.open("menu/es teh.png").resize((350, 334))
es_tehImage = ImageTk.PhotoImage(es_tehImageObject)

es_jerukImageObject = Image.open("menu/es jeruk.png").resize((350, 334))
es_jerukImage = ImageTk.PhotoImage(es_jerukImageObject)

lemon_teaImageObject = Image.open("menu/lemon tea.png").resize((350, 334))
lemon_teaImage = ImageTk.PhotoImage(lemon_teaImageObject)

es_coklatImageObject = Image.open("menu/es coklat.png").resize((350, 334))
es_coklatImage = ImageTk.PhotoImage(es_coklatImageObject)

#Section Frame
mainFrame = ttk.Frame(root, width=800, height=700, style='MainFrame.TFrame')
mainFrame.grid(row=0, column=0, sticky="NSEW")

topBannerFrame = ttk.Frame(mainFrame)
topBannerFrame.grid(row=0, column=0, sticky="NSEW", columnspan=3)

```

```

menuFrame = ttk.Frame(mainFrame, style='MenuFrame.TFrame')
menuFrame.grid(row=1, column=0, padx=3, pady=2, sticky="NSEW")

displayFrame = ttk.Frame(mainFrame, style='MenuFrame.TFrame')
displayFrame.grid(row=1, column=1, padx=3, pady=2, sticky="NSEW")

orderFrame = ttk.Frame(mainFrame, style="OrderFrame.TFrame")
orderFrame.grid(row=1, column=2, padx=3, pady=2, sticky="NSEW")

#dish frame
steak_ayamDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
steak_ayamDishframe.grid(row=1, column=0, sticky="NSEW")

nasi_gorengDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
nasi_gorengDishframe.grid(row=2, column=0, sticky="NSEW")

ayam_gorengDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
ayam_gorengDishframe.grid(row=3, column=0, sticky="NSEW")

ayam_bakarDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
ayam_bakarDishframe.grid(row=4, column=0, sticky="NSEW")

sotoDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
sotoDishframe.grid(row=5, column=0, sticky="NSEW")

rawonDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
rawonDishframe.grid(row=6, column=0, sticky="NSEW")

mendoanDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
mendoanDishframe.grid(row=7, column=0, sticky="NSEW")

tahu_isiDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
tahu_isiDishframe.grid(row=8, column=0, sticky="NSEW")

```



```

risolDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
risolDishframe.grid(row=9, column=0, sticky="NSEW")

lumpiaDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
lumpiaDishframe.grid(row=10, column=0, sticky="NSEW")

es_tehDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
es_tehDishframe.grid(row=11, column=0, sticky="NSEW")

es_jerukDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
es_jerukDishframe.grid(row=12, column=0, sticky="NSEW")

lemon_teaDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
lemon_teaDishframe.grid(row=13, column=0, sticky="NSEW")

es_coklatDishframe = ttk.Frame(menuFrame, style="DishFrame.TFrame")
es_coklatDishframe.grid(row=14, column=0, sticky="NSEW")

#top banner section
LogoLabel = ttk.Label(topBannerFrame, image=LogoImage,
background="#FFFCFC")
LogoLabel.grid(row=0, column=0, sticky="W")

RestaurantBannerLabel = ttk.Label(topBannerFrame, image=TopBannerImage,
background="#080808")
RestaurantBannerLabel.grid(row=0, column=1, sticky="NSEW")

#menu section
MainMenulabel = ttk.Label(menuFrame, text="MENU",
style="MenuLabel.TLabel")
MainMenulabel.grid(row=0, column=0, sticky="WE")
MainMenulabel.configure(
    anchor="center",

```

```

font=("Helvetica", 14, "bold")
)

steak_ayamDishLabel = ttk.Label(steak_ayamDishframe, text=f"Steak Ayam ..... Rp
{prices['Steak Ayam'];,}", style="MenuLabel.TLabel")
steak_ayamDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

nasi_gorengDishLabel = ttk.Label(nasi_gorengDishframe, text=f"Nasi Goreng ..... Rp
{prices['Nasi Goreng'];,}", style="MenuLabel.TLabel")
nasi_gorengDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

ayam_gorengDishLabel = ttk.Label(ayam_gorengDishframe, text=f"Ayam Goreng
..... Rp {prices['Ayam Goreng'];,}", style="MenuLabel.TLabel")
ayam_gorengDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

ayam_bakarDishLabel = ttk.Label(ayam_bakarDishframe, text=f"Ayam Bakar .....
Rp {prices['Ayam Bakar'];,}", style="MenuLabel.TLabel")
ayam_bakarDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

sotoDishLabel = ttk.Label(sotoDishframe, text=f"Soto Ayam ..... Rp {prices['Soto
Ayam'];,}", style="MenuLabel.TLabel")
sotoDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

rawonDishLabel = ttk.Label(rawonDishframe, text=f"Rawon ..... Rp
{prices['Rawon'];,}", style="MenuLabel.TLabel")
rawonDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

mendoanDishLabel = ttk.Label(mendoanDishframe, text=f"Mendoan ..... Rp
{prices['Mendoan'];,}", style="MenuLabel.TLabel")
mendoanDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

tahu_isiDishLabel = ttk.Label(tahu_isiDishframe, text=f"Tahu Isi ..... Rp
{prices['Tahu Isi'];,}", style="MenuLabel.TLabel")
tahu_isiDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

```

```

lumpiaDishLabel = ttk.Label(lumpiaDishframe, text=f"Lumpia ..... Rp
{prices['Lumpia']:,}", style="MenuLabel.TLabel")
lumpiaDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

risolDishLabel = ttk.Label(risolDishframe, text=f"Risol ..... Rp {prices['Risol']:,}",
style="MenuLabel.TLabel")
risolDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

es_tehDishLabel = ttk.Label(es_tehDishframe, text=f"Es Teh ..... Rp {prices['Es
Teh']:,}", style="MenuLabel.TLabel")
es_tehDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

es_jerukDishLabel = ttk.Label(es_jerukDishframe, text=f"Es Jeruk ..... Rp
{prices['Es Jeruk']:,}", style="MenuLabel.TLabel")
es_jerukDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

lemon_teaDishLabel = ttk.Label(lemon_teaDishframe, text=f"Lemon Tea ..... Rp
{prices['Lemon Tea']:,}", style="MenuLabel.TLabel")
lemon_teaDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

es_coklatDishLabel = ttk.Label(es_coklatDishframe, text=f"Es Coklat ..... Rp
{prices['Es Coklat']:,}", style="MenuLabel.TLabel")
es_coklatDishLabel.grid(row=0, column=0, padx=10, pady=2, sticky="W")

#Tombol
steak_ayamDisplayButton = ttk.Button(steak_ayamDishframe, text="Display",
command=displaysteak_ayam)
steak_ayamDisplayButton.grid(row=0, column=1, padx=10)

nasi_gorengDisplayButton = ttk.Button(nasi_gorengDishframe, text="Display",
command=displaynasi_goreng)
nasi_gorengDisplayButton.grid(row=0, column=1, padx=10)

```

```

    ayam_gorengDisplayButton = ttk.Button(ayam_gorengDishframe, text="Display",
command=displayayam_goreng)
    ayam_gorengDisplayButton.grid(row=0, column=1, padx=10)

    ayam_bakarDisplayButton = ttk.Button(ayam_bakarDishframe, text="Display",
command=displayayam_bakar)
    ayam_bakarDisplayButton.grid(row=0, column=1, padx=10)

    sotoDisplayButton = ttk.Button(sotoDishframe, text="Display",
command=displaysoto)
    sotoDisplayButton.grid(row=0, column=1, padx=10)

    rawonDisplayButton = ttk.Button(rawonDishframe, text="Display",
command=displayrawon)
    rawonDisplayButton.grid(row=0, column=1, padx=10)

    mendoanDisplayButton = ttk.Button(mendoanDishframe, text="Display",
command=displaymendoan)
    mendoanDisplayButton.grid(row=0, column=1, padx=10)

    tahu_isiDisplayButton = ttk.Button(tahu_isiDishframe, text="Display",
command=displaytahu_isi)
    tahu_isiDisplayButton.grid(row=0, column=1, padx=10)

    risolDisplayButton = ttk.Button(risolDishframe, text="Display",
command=displayrisol)
    risolDisplayButton.grid(row=0, column=1, padx=10)

    lumpiaDisplayButton = ttk.Button(lumpiaDishframe, text="Display",
command=displaylumpia)
    lumpiaDisplayButton.grid(row=0, column=1, padx=10)

    es_tehDisplayButton = ttk.Button(es_tehDishframe, text="Display",
command=displayes_teh)

```

```

es_tehDisplayButton.grid(row=0, column=1, padx=10)

es_jerukDisplayButton = ttk.Button(es_jerukDishframe, text="Display",
command=displayes_jeruk)
es_jerukDisplayButton.grid(row=0, column=1, padx=10)

lemon_teaDisplayButton = ttk.Button(lemon_teaDishframe, text="Display",
command=displaylemon_tea)
lemon_teaDisplayButton.grid(row=0, column=1, padx=10)

es_coklatDisplayButton = ttk.Button(es_coklatDishframe, text="Display",
command=displayes_coklat)
es_coklatDisplayButton.grid(row=0, column=1, padx=10)

#Order Section
orderTitleLabel = ttk.Label(orderFrame, text="ORDER")
orderTitleLabel.configure(
    foreground="white", background="black",
    font=("Helvetica", 14, "bold"), anchor="center",
    padding=(5, 5, 5, 5),
)
orderTitleLabel.grid(row=0, column=0, sticky="EW")

orderIDLabel = ttk.Label(orderFrame, text="ORDER ID : " + ORDER_ID())
orderIDLabel.configure(
    background="black",
    foreground="white",
    font=("Helvetica", 11, "italic"),
    anchor="center",
)
orderIDLabel.grid(row=1, column=0, sticky="EW")

orderTransaction = ttk.Label(orderFrame, style='orderTransaction.TLabel')
orderTransaction.grid(row=2, column=0, sticky="NSEW")

```

```

    orderTotalLabel = ttk.Label(orderFrame, text="TOTAL : Rp 0",
style='orderTotalLabel.TLabel')
    orderTotalLabel.grid(row=3, column=0, sticky="EW")

    orderButton = ttk.Button(orderFrame, text="ORDER", command=order)
    orderButton.grid(row=4, column=0, sticky="EW")

    if is_admin:
        adminButton = ttk.Button(orderFrame, text="EDIT MENU",
command=admin_edit)
        adminButton.grid(row=5, column=0, sticky="EW")

#Display Section
displayLabel = ttk.Label(displayFrame, image=displayDefaultImage)
displayLabel.grid(row=0, column=0, sticky="NSEW", columnspan=2)
displayLabel.configure(background="#0F1110")

addOrderButton = ttk.Button(displayFrame, text="ADD TO ORDER",
command=add)
addOrderButton.grid(row=1, column=0, padx=2, sticky="NSEW")

removeOrderButton = ttk.Button(displayFrame, text="REMOVE",
command=remove)
removeOrderButton.grid(row=1, column=1, padx=2, sticky="NSEW")

#grid configuration
mainFrame.columnconfigure(2, weight=1)
mainFrame.rowconfigure(1, weight=1)
menuFrame.columnconfigure(0, weight=1)
menuFrame.rowconfigure(1, weight=1)
menuFrame.rowconfigure(2, weight=1)
menuFrame.rowconfigure(3, weight=1)
menuFrame.rowconfigure(4, weight=1)

```

```

menuFrame.rowconfigure(5, weight=1)
menuFrame.rowconfigure(6, weight=1)
orderFrame.columnconfigure(0, weight=1)
orderFrame.rowconfigure(2, weight=1)

root.mainloop()

# Fungsi untuk Login
def login_window():
    login_root = Tk()
    login_root.title("Login - Kedai FMR")
    login_root.geometry("1210x590")
    login_root.configure(bg="#F58709")

    ttk.Label(
        login_root,
        text="Choose Your Role:",
        font=("Arial", 26, "bold"),
        foreground="#010000",
        background="#F58709"
    ).pack(pady=25)

    def select_customer():
        login_root.destroy()
        main_app(is_admin=False)

    def select_admin():
        password_window = Toplevel(login_root)
        password_window.title("Admin Password")
        password_window.geometry("250x180")
        password_window.configure(bg="#F58709")

        pw_label = Label(
            password_window,

```

```

        text="Enter Password:",
        font=("Arial", 14),
        fg="black",
        bg="#F58709"
    )
    pw_label.pack(pady=15)

    password_entry = ttk.Entry(password_window, show="*")
    password_entry.pack()

    def check_password():
        if password_entry.get() == "maulunyil":
            password_window.destroy()
            login_root.destroy()
            main_app(is_admin=True)
        else:
            messagebox.showerror("Error", "Incorrect Password")

    # Tombol login
    ttk.Button(password_window, text="Login",
command=check_password).pack(pady=15)

    ttk.Button(login_root, text="Customer", command=select_customer,
style="Big.TButton").pack(pady=7)
    ttk.Button(login_root, text="Admin", command=select_admin,
style="Big.TButton").pack(pady=7)

    style = ttk.Style()
    style.configure("Big.TButton", font=("Arial", 20), padding=7)

    login_root.mainloop()

if __name__ == "__main__":
    login_window()

```



## DAFTAR PUSTAKA

- Putri, A. L. (2024). Pengaruh Sistem Informasi Terhadap Inovasi Digital, Transformasi Digital, dan Customer Relationship Management (CRM. *Dinasti Information and Technology*.
- Rezeki, P. M. (2025). Perancangan Aplikasi Pemesanan Menu Berbasis Website Pada Tanggul Kopi Banda Aceh. *Directory Of Serambi Engineering Journals*.
- Thinking Through Code. (2022, 4 Oktober). *Python Tkinter Project - Restaurant Order System - Part 1 - GUI Frontend* [Video]. YouTube.  
<https://www.youtube.com/watch?v=h59nSmwewO4>
- Thinking Through Code. (2022, 4 Oktober). *Python Tkinter Project - Restaurant Order System - Functionality Part2* [Video]. YouTube.  
<https://www.youtube.com/watch?v=BhPP2wj5b0&t=2105s>