

Caesar Coffee website



[HOME](#)

[MENU](#)

[DELIVERY](#)

[APP](#)

[ABOUT](#)

[CONTACT](#)

WELCOME TO

Caesar Coffee

FREE COFFE IS A
TAP AWAY

[Join Now](#)



KATA PENGANTAR

Puji syukur kami panjatkan kepada Tuhan Yang Maha Esa, karena berkat rahmat dan karunia-Nya, modul pembelajaran ini dapat siselesaikan dengan baik. Modul ini dirancang untuk membantu siswa dan praktisi dalam memahami dan mengembangkan aplikasi berbasis web menggunakan Flask dan SQLite. Dengan adanya modul ini, diharapkan pembaca dapat memperoleh pemahaman mendalam tentang cara membangun website sederhana hingga kompleks dengan menggunakan teknologi yang tersedia.


Pembuatan website menggunakan Flask dapat dilakukan dengan beberapa Langkah utama. Pertama, pastikan lingkungan pengembangan telah dipersiapkan dengan menginstal Python dan Flask di PyCharm menggunakan fitur Python Packages di PyCharm dengan mencari dan menginstal Flask melalui menu Python Packages. Setelah itu, kita Membuat struktur proyek yang terdiri dari file utama app.py , direktori templates/ untuk menyimpan file HTML, serta direktori static/ untuk menyimpan file CSS dan JavaScript.

Setelah sturktur proyek siap, kita mulai dengan Membuat rute (routing) di dalam app.py menggunakan decorator @app.route(' / ') yang akan menangani permintaan HTTP dari klien. Kita dapat menggunakan render_template untuk menampilkan halaman web denga HTML yang telah disusun di dalam folder templates/ . Selanjutnya, kita dapat menghubungkan aplikasi Flask dengan database SQLite untuk Menyimpan dan mengelola data. Hal ini dilakukan dengan menggunakan Pustaka sqlite3 dalam Python.

Setelah semua komponen selesai di buat, webite dapat dijalanka dengan perintah python app.py, dan dapat diakses melalui browser di alamat <http://127.0.0.1:5000/> . Dengan mengikuti Langkah-langkah tersebut, pembaca diharapkan mampu mengembangkan website berbasis Flask secara mandiri.

Kami mengucapkan teria kasih kepada semua pihak yang telah memberikan dukungan dalam penyusunan modul ini. Semoga modul ini dapat menjadi referensi yayng bermanfaat dan mendukung pembelajaran dalam dunia teknologi.

Malang, 25 Februari 2025

	Bidang Keahlian: Teknologi Informasi	Lab: SE - 209	Tanggal: MAR. 2025	TI	SE	04	VT	Page 1 - 26
	Program Keahlian: TI-SE-TE	Universal Internet of Things	Rev.: 05/03/2025	Dibuat oleh: EKO-Jaad-Rully				

BAB I PENDAHULUAN

1.1. Latar Belakang

Dalam era digital saat ini, pembuatan website menjadi keterampilan yang sangat penting. Website tidak hanya digunakan untuk keperluan bisnis, tetapi juga sebagai sarana komunikasi, edukasi, dan hiburan. Penggunaan framework web mempermudah proses pengembangan website, baik untuk pemula maupun profesional. Flask merupakan salah satu framework berbasis Python yang ringan dan fleksibel, sehingga cocok digunakan untuk mengembangkan website dengan cepat dan efisien. Dengan Flask, pengembang dapat membuat aplikasi web sederhana maupun kompleks dengan struktur kode yang mudah dipahami.

Selain itu, database juga memegang peranan penting dalam pengolahan data dalam sebuah website. Salah satu database yang sering digunakan bersama Flask adalah SQLite. SQLite merupakan database yang ringan dan mudah digunakan, sehingga sangat cocok untuk proyek kecil hingga menengah. Dengan kombinasi Flask dan SQLite, pengembang dapat membuat website yang memiliki performa baik dengan pengolahan data yang efektif.

Modul ini disusun untuk memberikan panduan Langkah demi Langkah dalam membangun website menggunakan Flask DAN SQLite. Modul ini mencakup berbagai aspek, mulai dari instalasi Flask, pembuatan struktur proyek, pembuatan tampilan dengan HTML dan CSS, hingga pengelolaan data dengan SQLite.

1.2. Tujuan

Tujuan dari penyusunan modul ini adalah:

1. Memberikan pemahaman dasar tentang Flask dan cara penggunaannya dalam pembuatan website.
2. Menjelaskan struktur proyek Flask dan fungsinya masing-masing.
3. Menajarkan cara menghubungkan Flask dengan database SQLite.
4. Membantu siswa dan praktisi dalam membuat website yang dapat berjalan secara local maupun online.



1.3. Manfaat

Dengan mempelajari modul ini, pembaca diharapkan dapat:

1. Memahami dasar-dasar pengembangan web menggunakan Flask.
2. Mampu membuat proyek web sederhana dengan struktur yang baik.
3. Mengelola database menggunakan SQLite dalam website Flask.
4. Mengembangkan website yang interaktif dan dinamis.

1.4. Sistematika Modul

Modul ini disusun menjadi beberapa bab dengan sistematika sebagai berikut:

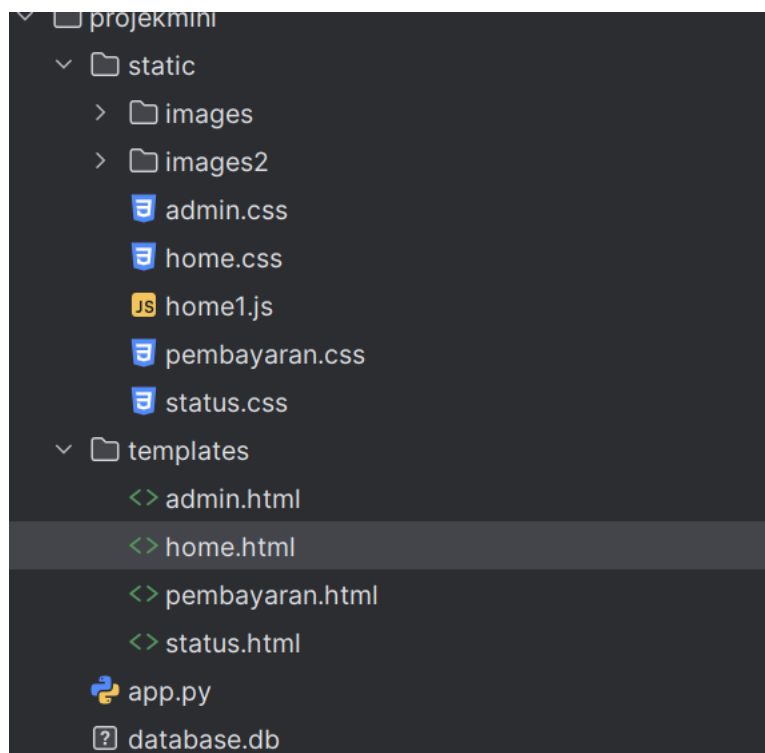
1. **Bab I: Pendahuluan** - Membahas latar belakang, tujuan, manfaat, dan sistematika penulisan modul.
2. **Bab II: Struktur Proyek Flask** - Menguraikan struktur direktori proyek Flask dan Desain sistem.
3. **Bab III: Backend dengan MICROFRAMEWORK FLASK PYTHON** - Membahas cara mengelola database dengan SQLite dalam aplikasi Flask.
4. **Bab IV: Frontend dengan HTML CSS dan JS** - Membahas cara mengelola Frontend .

BAB II

STRUKTUR PROYEK FLASK

2.1. Struktur Direktori Proyek

Saat membangun website dengan Flask, kita perlu memiliki struktur proyek yang rapi agar mempermudah pengelolaan file dan pemanggilan fungsi. Struktur direktori proyek Flask umumnya terlihat seperti berikut:

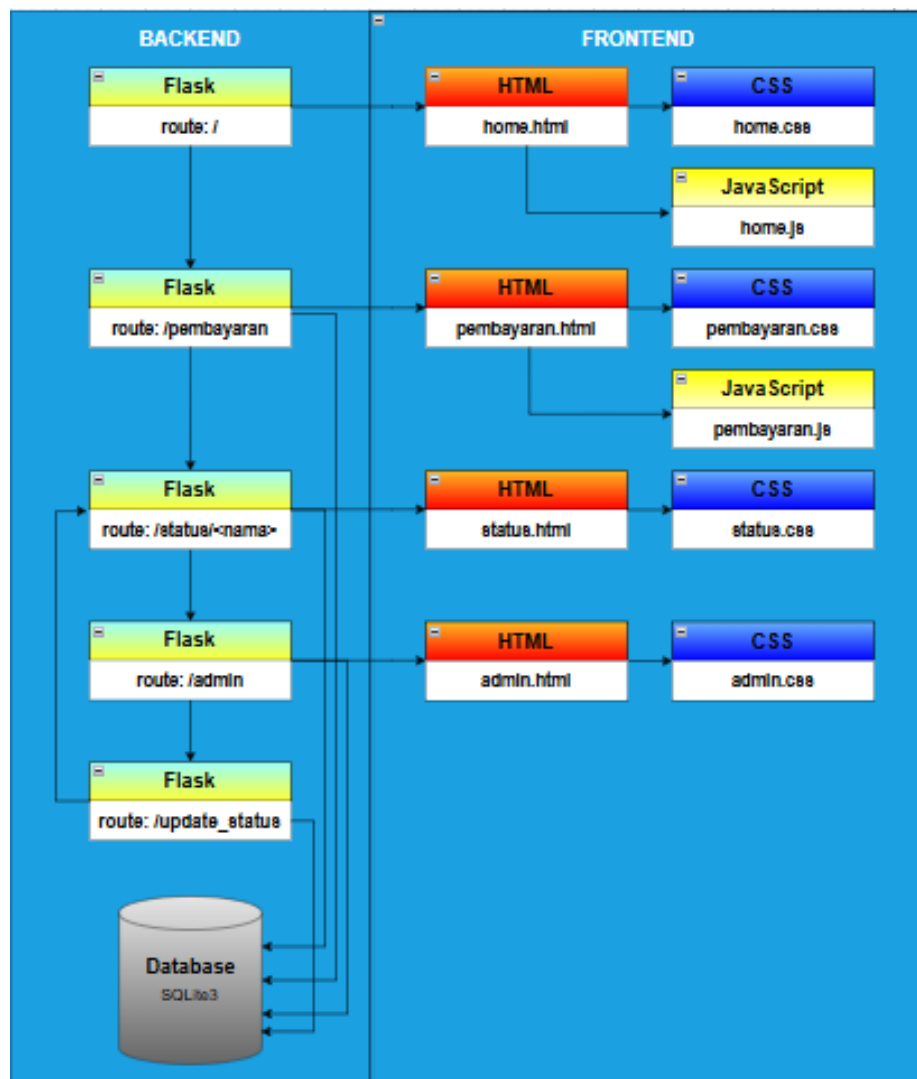


Setiap folder memiliki fungsinya masing-masing:

- static/ Folder ini Menyimpan file static seperti CSS, JavaScript, dan gambar.
- templates/ Folder yang menyimpan file HTML yang akan dirender oleh Flask.
- App.py File utama yang menjalankan website Flask, menangani request, dan menghubungkan ke database.
- Database.db Database SQLite untuk Menyimpan data pesanan.

2.2. Desain Sistem

Cara kerja website dapat dilihat pada flowchart berikut.



*Keterangan:

Flask akan bekerja sebagai Backend yang menangani input data dan disimpan ke database;
HTML, CSS, dan JavaScript akan menjadi tampilan utama website;
SQLite3 berperan sebagai database, yang menyimpan data-data yang telah dimasukkan oleh pengguna.

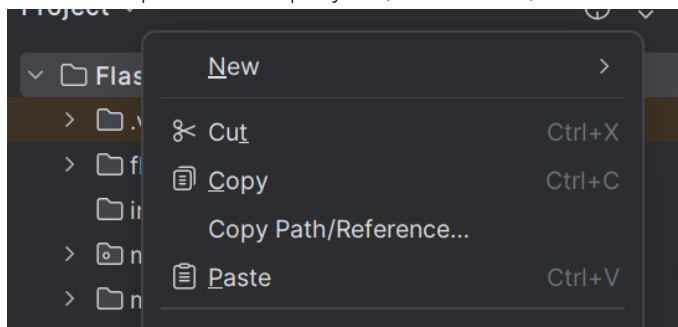
BAB III

BAKEND DENGAN MICROFRAMEWORK FLASK PYTHON

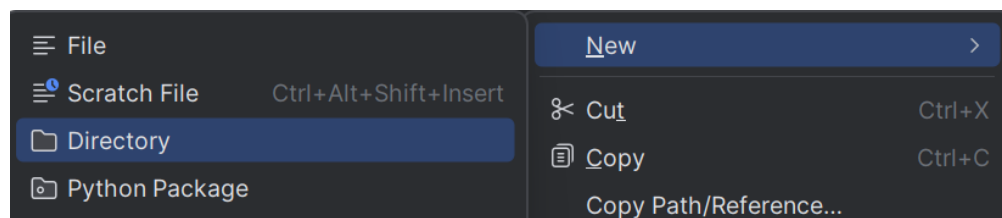
3.1.Membuat File Di Pycharm

Untuk Membuat setiap file di **Pycharm**, lakukan Langkah – Langkah berikut:

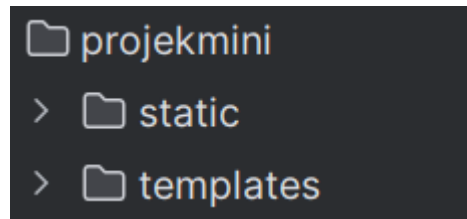
1. Klik kanan pada folder proyek (folder anda).



2. Terus klik **new** => **Directory** buat nama terserah kalian (projekmini).

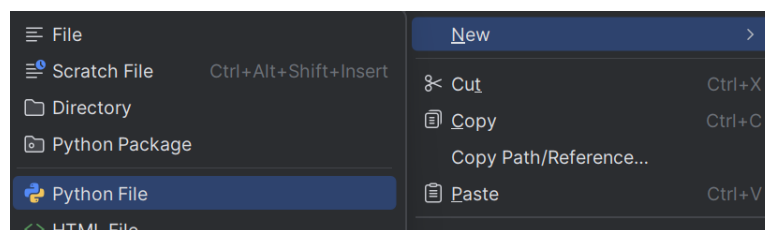


3. Terus klik kanan yang directory yang anda buat.
 - Buat nama folder " templates " untuk file HTML.
 - Buat nama folder " static " untuk file CSS, JAVASCRIPT, dan Gambar.

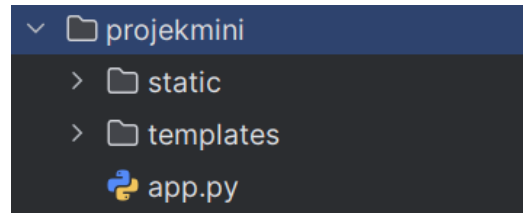


4. Untuk Membuat file app.py (file utama)

- Klik kanan ke directory yang anda buat
- Klik **new** => **Python File**



5. Ketik nama file : " app.py ", lalu klik **enter** .



3.2.Membangun Backend dengan Flask Python

Berikut adalah cara memulai membangun Backend Website dengan Flask :

1. Ketik kode di bawah ini

```
from flask import Flask, render_template, request, redirect, url_for, jsonify
import sqlite3

app = Flask(__name__)
```

Kode tersebut mengimpor beberapa modul dan fungsi dari Flask, modul sqlite3 di Python, serta inialisasi aplikasi Flask.

2. Koneksi ke database


```
def connect_db(): 5 usages
    return sqlite3.connect('database.db')
```

Dengan fungsi connect_db(), kalian dapat dengan mudah mengelola koneksi ke database SQLite dan menjalankan operasi database dalam aplikasi Flask.

3. Buat tabel jika belum tersedia

```
def create_table(): 1 usage
    conn = connect_db()
    cursor = conn.cursor()
    cursor.execute('''CREATE TABLE IF NOT EXISTS orders (
        id INTEGER PRIMARY KEY AUTOINCREMENT,
        nama TEXT,
        menu TEXT,
        harga REAL,
        jumlah INTEGER,
        total REAL,
        status TEXT DEFAULT "Menunggu")''')
    conn.commit()
    conn.close()

create_table()
```

Tabel ini akan terkoneksi dengan DB Browser, yang mana data-data seperti nama, menu, dan lainnya akan tersimpan di dalamnya.

4. Buat route untuk halaman utama

```
@app.route('/')
def home_page():
    return render_template('home.html')
```

Kode di atas adalah contoh penggunaan route dalam Flask. Pada fungsi home_page, route akan mengembalikan file HTML yaitu home.html.

5. Halaman pembayaran pelanggan

```
@app.route(rule: '/pembayaran', methods=['GET', 'POST'])
def pembayaran():
    if request.method == 'POST':
        data = request.json
        nama = data['nama']
        menu = data['menu']
        harga = float(data['harga'].replace("$", ""))
        jumlah = int(data['jumlah'])
        total = harga * jumlah

        conn = connect_db()
        cursor = conn.cursor()
        cursor.execute(sql: "INSERT INTO orders (nama, menu, harga, jumlah, total, status) VALUES (?, ?, ?, ?, ?, ?)",
            parameters: (nama, menu, harga, jumlah, total, "Menunggu"))
        conn.commit()
        conn.close()

        return jsonify({'success': True, "message": "Pesanan berhasil disimpan!"})

    return render_template('pembayaran.html')
```

Pada route ini akan berisi data-data dalam database yang memuat nama, menu, harga, jumlah, dan total harga, serta menampilkan file pembayaran.html. Pelanggan juga dapat melakukan pemesanan pada halaman ini.

1. Menjalankan program

```
if __name__ == '__main__':
    app.run(debug=True, host='0.0.0.0')
```

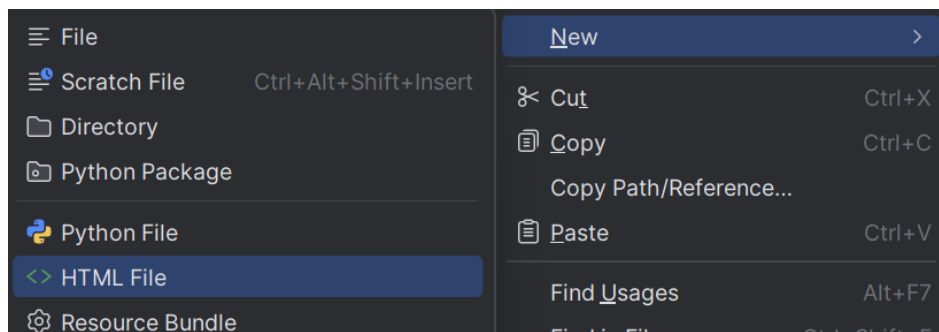
Kode di atas berfungsi untuk menjalankan program yang telah dibuat sebagai auto-reload, debugger jika terjadi error, serta akses dari jaringan eksternal.

BAB IV

FRONTEND DENGAN HTML CSS DAN JS

4.1. Membuat Struktur Halaman HTML

Buat file " home.html " di dalam templates/ untuk menampilkan halaman utama website



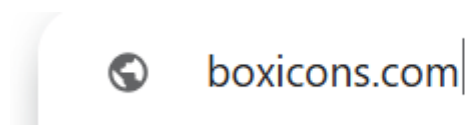
4.2. Code home.html

Ikuti yang ada di dalam foto tersebut



4.3. Menghubungkan ikon dari Boxicons.com

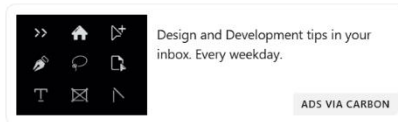
Kalian cari di google atau crome search " boxicons.com "



Dan kalian akan di menuju ke website " boxicons.com "

High Quality Web Icons

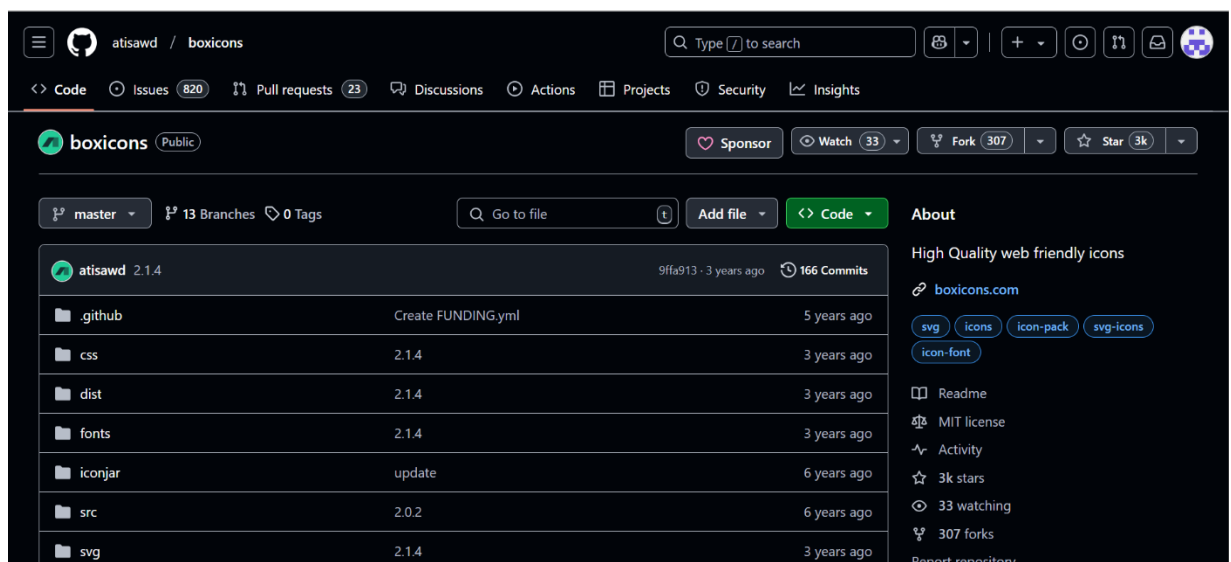
Simple Open Source icons carefully crafted for designers & developers



Sekarang kalian klik pojok kanan atas " Github "

[Github](#)
[Usage](#)
[Download v2.1.4](#)

Kalian akan menuju ke akun github website tersebut dan harus mempunyai akun github masing-masing



Kalian scroll kebawah sampai menemukan " Usage "

Usage

Using via CSS

1. Include the stylesheet on your document's `<head>`

```
<head>
  <link rel="stylesheet" href="boxicons.min.css">
</head>
```

Instead of installing you may use the remote version

```
<head>
  <link rel="stylesheet"
    href="https://cdn.jsdelivr.net/npm/boxicons@latest/css/boxicons.min.css">
  <!-- or -->
  <link rel="stylesheet"
    href="https://unpkg.com/boxicons@latest/css/boxicons.min.css">
</head>
```

Terus kalian bertuju ke text "intead of installing you may use the remote version"

```
<link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/boxicons@latest/css/boxicons.min.css">
```

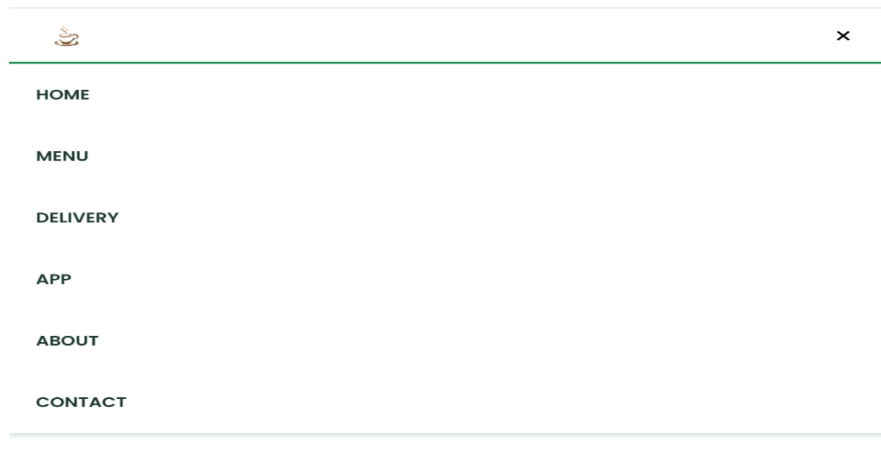
Terus kalian copy paste ke html yang kalian buat di dalam "head"

```
<head>
  <meta charset="UTF-8">
  <meta http-equiv="x-ua-compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- or -->
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/boxicons@latest/css/boxicons.min.css">
  <link rel="stylesheet" href="https://unpkg.com/boxicons@latest/css/boxicons.min.css">
</head>
```

Sekarang saya tunjukkan hasil navbar saya, setelah itu kita membuat navbar yang sesuai dengan saya tunjukkan dibawah hasil navbar ada code HTML,CSS dan JS. kalian ikuti code dari foto dibawah ini :

- Output navbar





HTML

```
11  ✓ <body>
12    <!--navbar-->
13  ✓ <header>
14      <a href="#" class="logo"></a>
15      <div class="bx bx-menu" id="menu-icon"></div>
16
17  ✓   <ul class="navbar">
18       <li><a href="#home">Home</a></li>
19       <li><a href="#menu">Menu</a></li>
20       <li><a href="#delivery">Delivery</a></li>
21       <li><a href="#app">App</a></li>
22       <li><a href="#about">About</a></li>
23       <li><a href="#contact">Contact</a></li>
24   </ul>
25 </header>
26 </body>
27 <script src="../static/home.js"></script>
28 </html>
```

CSS

```

1  @import url('https://fonts.googleapis.com/css2?');
2  *{
3      margin: 0;
4      padding: 0;
5      box-sizing: border-box;
6      scroll-padding-top: 2rem;
7      text-decoration: none;
8      list-style: none;
9      scroll-behavior: smooth;
10     font-family: "Poppins", serif;
11 }
12
13 :root{
14     --main-color: #008148;
15     --second-color: #1e3932;
16 }
17 section{
18     padding: 50px 10%;
19 }
20 ::selection{
21     color: #fff;
22     background: var(--main-color);
23 }
24
25 img{
26     width: 100%;
27 }
28 header{
29     position: fixed;
30     width: 100%;
31     top: 0;
32     right: 0;
33     z-index: 1000;
34     display: flex;
35     align-items: center;
36     justify-content: space-between;
37     background: #fff;
38     box-shadow: 0 4px 41px rgb(14 55 54 / 14%);
39     padding: 15px 10%;
40     transition: 0.2s;
41 }
42 .logo{
43     display: flex;
44     align-items: center;
45 }
46 .logo img{
47     width: 40px;
48 }
49
50 .navbar{
51     display: flex;
52 }
53 .navbar a{
54     font-size: 1rem;
55     padding: 11px 20px;
56     color: var(--second-color);
57     font-weight: 600;
58     text-transform: uppercase;
59 }
60 .navbar a:hover{
61     color: var(--main-color);
62 }
63 #menu-icon{
64     font-size: 24px;
65     cursor: pointer;
66     z-index: 1001;
67     display: none;
68 }
69 @media (max-width: 768px){
70     header{
71         padding: 11px 4%;
72     }
73     #menu-icon{
74         display: initial;
75     }
76     header .navbar{
77         position: absolute;
78         top: -500px;
79         left: 0;
80         right: 0;
81         display: flex;
82         flex-direction: column;
83         background: #fff;
84         box-shadow: 0 4px 4px rgb(14 55 54 / 14%);
85         border-top: 2px solid var(--main-color);
86         transition: 0.2s;
87         text-align: left;
88     }
89     .navbar.active{
90         top: 100%;
91     }
92     .navbar a{
93         padding: 1.5rem;
94         display: block;
95         color: var(--second-color);
96     }

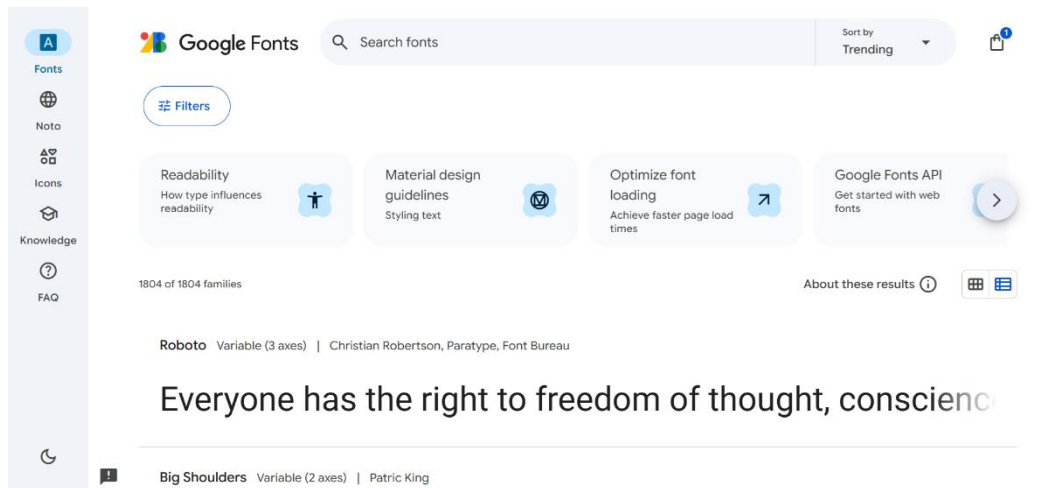
```


JAVASCRIPT

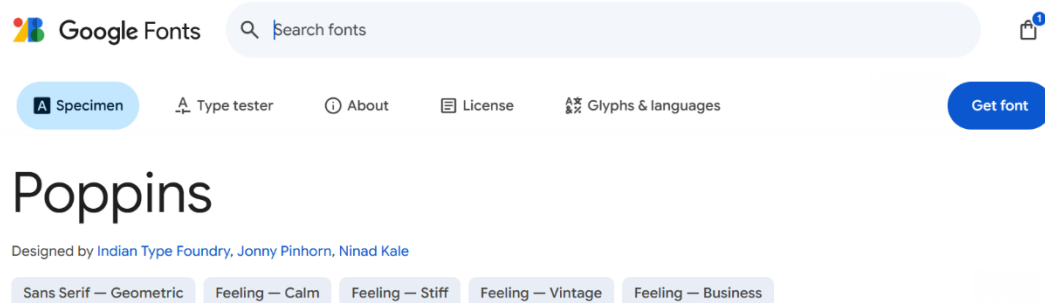
```
1 let menu = document.querySelector('#menu-icon');
2 let navbar = document.querySelector('.navbar');
3
4 menu.onclick = () => {
5     menu.classList.toggle('bx-x');
6     navbar.classList.toggle('active');
7 }
8 window.onscroll = () => {
9     menu.classList.remove('bx-x');
10    navbar.classList.remove('active');
11 }
```

4.4. Menghubungkan font " Poppions "

- Kalau sudah kalian buka di google search " <https://fonts.google.com/> "



Kalau sudah di website google font kalian kalian klik " Search fonts " ketik " Poppions "



Kalian klik " get font " di kanan atas tersebut

1 font family selected

Share Remove all

Poppins Static

Download all (1)

Ang lahat ng tao'y isinilang na

Get embed code

Download all (1)

See how to use

Kalau sudah di halaman tersebut kalian klik " Get embed code "

← Embed code

Poppins Static

Whereas recognition

Thin 100

Thin 100 Italic

Web

Android

iOS

Flutter

<link> @import

Embed code in the <head> of your html

```
<style>
@import url('https://fonts.googleapis.com/css2?family=Poppins:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,700;1,800;1,900&display=swap');
</style>
```

Copy code

Kalau sudah di halaman tersebut kita copy code yang ada di pojok kanan bawah tersebut

<style>

```
@import url('https://fonts.googleapis.com/css2?family=Poppins:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,700;1,800;1,900&display=swap');
```

</style>

Copy code

Paste code tersebut ke CSS yang kalian buat

```
@import url('https://fonts.googleapis.com/css2?family=Poppins:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,700;1,800;1,900&display=swap');
```

4.5. pembayaran.html

- Output pembayaran.html

Masukan Menu Anda

Nama :

Nama Menu:

Harga:

Jumlah:

Pesan



SHOP COFFEE

MENU



★★★★☆

Macchiato

\$ 11



★★★★☆

Cappuccino

\$ 10



★★★★☆

Americano

\$ 12

HTML

```
<!-- Menu -->
<section class="shop">
  <div class="heading">
    <span>Shop Coffee</span>
    <h1>Menu</h1>
  </div>
  <div class="shop-container">
    <div class="box" onclick="isiForm('Macchiato', '11')">
      <div class="box-img">
        
      </div>
      <div class="stars">
        <i class='bx bxs-star'></i>
        <i class='bx bxs-star'></i>
        <i class='bx bxs-star'></i>
        <i class='bx bxs-star'></i>
        <i class='bx bxs-star-half'></i>
      </div>
      <h2>Macchiato</h2>
      <span>$ 11</span>
    </div>

    <div class="box" onclick="isiForm('Cappuccino', '10')">
      <div class="box-img">
        
      </div>
      <div class="stars">
        <i class='bx bxs-star'></i>
        <i class='bx bxs-star'></i>
        <i class='bx bxs-star'></i>
        <i class='bx bxs-star'></i>
        <i class='bx bxs-star-half'></i>
      </div>
      <h2>Cappuccino</h2>
      <span>$ 10</span>
    </div>
  </div>
</section>
```

```

<!-- Form -->
<div class="container">
  <h2>Masukan Menu Anda</h2>
  <form id="pembayaran">
    <label for="nama">Nama :</label>
    <input type="text" id="nama" placeholder="Masukan Nama Anda" required>

    <label for="menu">Nama Menu: </label>
    <input type="text" id="menu" readonly>

    <label for="harga">Harga: </label>
    <input type="text" id="harga" readonly>

    <label for="jumlah">Jumlah: </label>
    <input type="number" id="jumlah" min="1" max="5" placeholder="Masukkan jumlah barang" required>

    <div class="shop-btn-container">
      <button type="button" class="btn" onclick="redirectToStatus()">Pesan</button>
    </div>
  </form>
</div>

```

CSS

```

1  * {
2    margin: 0;
3    padding: 0;
4    box-sizing: border-box;
5    scroll-padding-top: 2rem;
6    text-decoration: none;
7    list-style: none;
8    scroll-behavior: smooth;
9    font-family: "Poppins", serif;
10 }
11
12 :root {
13   --main-color: #008148;
14   --second-color: #1e3932;
15 }
16
17 section {
18   padding: 50px 10%;
19 }
20
21 *::selection {
22   color: #fff;
23   background: var(--main-color);
24 }
25
26 img {
27   width: 100%;
28 }
29
30 header {
31   position: fixed;
32   width: 100%;
33   top: 0;
34   right: 0;
35   z-index: 1000;
36   display: flex;
37   align-items: center;
38   justify-content: space-between;
39   background: #fff;
40   box-shadow: 0 4px 41px rgb(14 55 54 / 14%);
41   padding: 15px 10%;
42   transition: 0.2s;
43 }
44
45 .logo {
46   display: flex;
47   align-items: center;
48 }

```

Kode lebih lengkapnya ada di GitHub

<https://github.com/Zedrise/tempaltes-css-website.git>

JavaScript

```
function redirectToStatus() {  
    const nama = document.getElementById("nama").value;  
    const menu = document.getElementById("menu").value;  
    const harga = document.getElementById("harga").value;  
    const jumlah = document.getElementById("jumlah").value;  
  
    if (!nama || !menu || !harga || !jumlah) {  
        alert("Harap isi semua field!");  
        return;  
    }  
}
```

```
fetch('/pembayaran', {  
    method: 'POST',  
    headers: {  
        'Content-Type': 'application/json',  
    },  
    body: JSON.stringify({  
        nama: nama,  
        menu: menu,  
        harga: harga,  
        jumlah: jumlah,  
    }),  
})  
    .then(response => response.json())  
    .then(data => {  
        if (data.success) {  
            window.location.href = `/status/${nama}`;  
        } else {  
            alert("Gagal menyimpan data!");  
        }  
    })  
};
```



```
function isiForm(namaMenu, hargaMenu) {
    document.getElementById("menu").value = namaMenu;
    document.getElementById("harga").value = `$$${hargaMenu}`;
    document.getElementById("jumlah").focus();
}
```

DAFTAR PUSTAKA

- [1] <https://github.com/Zedrise/tempaltes-css-website.git>
- [2] <https://zedrise.pythonanywhere.com/>
- [3] <https://boxicons.com/>
- [4] <https://fonts.google.com/>
- [5] <https://www.w3schools.com/>
- [6] <https://youtu.be/hsSs2qpLq9U?si=RRVWKO44fwOOXD-L>
- [7] https://youtu.be/Qjy0k4h_nPo?si=UuHem0IVVHN Az49w

PENYUSUN



Eko Subiyantoro adalah widyaiswara di BBPPMPV BOE-VEDC Malang di Departemen Teknologi Informasi. Lahir di Banyuwangi pada tahun 1975. Pendidikan SD, SMP, dan STM (sekarang SMK) diselesaikan di kota kelahirannya. Selanjutnya ia menyelesaikan S1 di Program Studi Pendidikan Teknik Elektronika FPTK IKIP Yogyakarta (sekarang UNY) pada tahun 1998. Sarjana Sains Terapan (SST) diperolehnya melalui program beasiswa Dikmenjur (sekarang Direktorat Pembinaan Sekolah Menengah Kejuruan) di PENS ITS Surabaya pada tahun 2002 pada Prodi Teknologi Informasi dengan predikat cum laude. Magister Teknik diperolehnya melalui program beasiswa unggulan BPKLN di ITS Surabaya Program Studi Teknik Elektro konsentrasi Jaringan Cerdas Multimedia pada tahun 2009 juga dengan predikat cum laude. Doktor diperolehnya juga melalui beasiswa unggulan bagi pegawai Kemdikbud di Prodi S3 Ilmu Komputer FMIPA Universitas Gadjah Mada.

Pengalaman luar negeri diperolehnya melalui program Inwent dari pemerintah Jerman pada tahun 2003 dalam program Advance Automation Training. Selain itu, pada tahun 2006 mengikuti Expert Program Maintenance and Repair di Abha College Kingdom of Saudi Arabia (KSA). Selanjutnya pada tahun 2013 melalui IGI kerja sama Indonesia–Jerman diikutkan dalam program Modern Management School di Kassel Jerman.

Penulis memiliki pengalaman kerja di PPPPTK BOE/VEDC Malang sebagai Kepala Departemen Teknologi Informasi selama 2 periode 2010–2013 dan 2013–2015. Selain itu juga sebagai tim pengembang Java Education Network Indonesia (JENI), sebuah sistem yang diinisiasi BPKLN Kemdikbud dalam kurun waktu tahun 2007–2009. Sertifikasi yang diperoleh dalam bidang pemrograman adalah Sun Certified Java Programmer (SCJP) pada tahun 2008.

Eko Subiyantoro dapat dihubungi melalui alamat e-mail ekovedc@gmail.com