

# **LAPORAN**

## **PEMROGRAMAN WEB PRAKTIK**

**Dosen Pengampu:**  
**Aditya Dimas Dewanto, S.T., M.T.**



**Dibuat oleh:**

**Nama : Muhammad Zaki As Shidiqi**  
**NPM : 5240411230**

**PROGRAM STUDI INFORMATIKA**  
**FAKULTAS SAINS DAN TEKNOLOGI**  
**UNIVERSITAS TEKNOLOGI YOGYAKARTA**  
**YOGYAKARTA**  
**2025**

## DAFTAR ISI

DAFTAR ISI.....	2
BAB I. VIRTUALENV DAN INSTALASI FLASK .....	3
1.1    Virtual Env .....	3
1.2    Instalasi Flask .....	4
BAB II. INTEGRASI FLASK DAN BOOTSTRAP .....	7
2.1    Integrasi Bootstrap.....	7
2.2    Tugas .....	10
KESIMPULAN.....	15

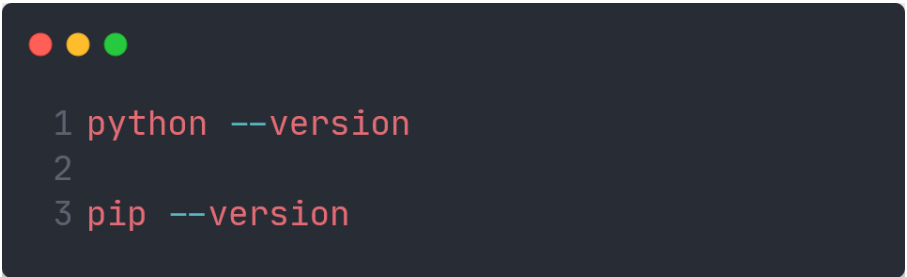
## BAB I. VIRTUALENV DAN INSTALASI FLASK

### 1.1 Virtual Env

Virtual env adalah sebuah package pada python yang memungkinkan kita untuk membuat virtual environment atau lingkungan virtual. Virtual env berfungsi untuk memisahkan lingkungan dari masing-masing proyek, jadi dengan virtual env dapat memudahkan kita untuk mengelola package-package yang terinstal pada masing-masing proyek.

a. Memeriksa python.

Sebelum melakukan penginstalan virtual env, pastikan python dan pip sudah terinstal pada perangkat yang digunakan. Biasanya bisa dicek menggunakan command pada terminal.



```
1 python --version
2
3 pip --version
```

b. Perintah pada CLI.

Jika muncul versi python dan pip artinya python dan pip sudah terinstal pada komputer kita. Maka kita bisa melanjutkan langkah berikutnya yaitu instalasi virtual env dengan mengetikkan command berikut pada terminal dan tunggu instalasi sampai selesai.



```
1 pip install virtualenv
```

c. Membuat environment.

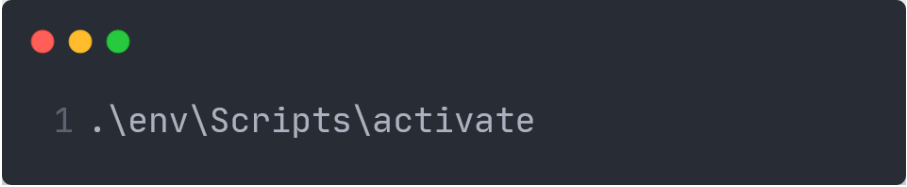
Setelah virtual env terinstal, langkah berikutnya adalah membuat sebuah environment untuk proyek kita. Untuk membuat sebuah environment pada sebuah proyek, kita dapat mengetikkan command berikut pada terminal dan tunggu prosesnya sampai selesai.



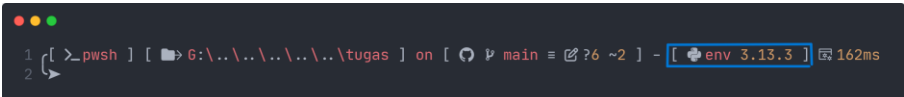
```
1 virtualenv env
```

d. Mengaktifkan environment.

Setelah muncul folder env pada proyek kita, maka pembuatan environment untuk proyek kita berhasil. Selanjutnya kita dapat mengaktifkan environment tersebut dengan mengetikkan command berikut pada terminal.



```
1 .\env\Scripts\activate
```



```
1 [ >_pwsh ] [ 6:\..\..\..\..\tugas ] on [ P main = ?6 ~2 ] - [ env 3.13.3 ] 162ms
2
```

Pastikan sudah muncul (env) biasanya di bagian depan sebelum directory. Jika sudah muncul artinya kita sudah berhasil mengaktifkan environment khusus untuk proyek kita. Maka dari itu, kita bisa melanjutkan ke langkah berikutnya yaitu instalasi Flask.

## 1.2 Instalasi Flask

Flask adalah sebuah micro framework yang memungkinkan kita memungkinkan kita membuat webserver menggunakan python. Prinsip kerjanya dengan me-return html pada setiap function yang kita berikan route diatasnya.

a. Perintah pada CLI.

Sebelum menginstal flask pastikan environment yang kita buat sebelumnya sudah aktif. Selanjutnya untuk menginstall flask kita dapat mengetikkan command berikut pada terminal dan tunggu proses instalasi sampai selesai.



```
1 pip install flask
```

b. Memeriksa package yang terinstal.

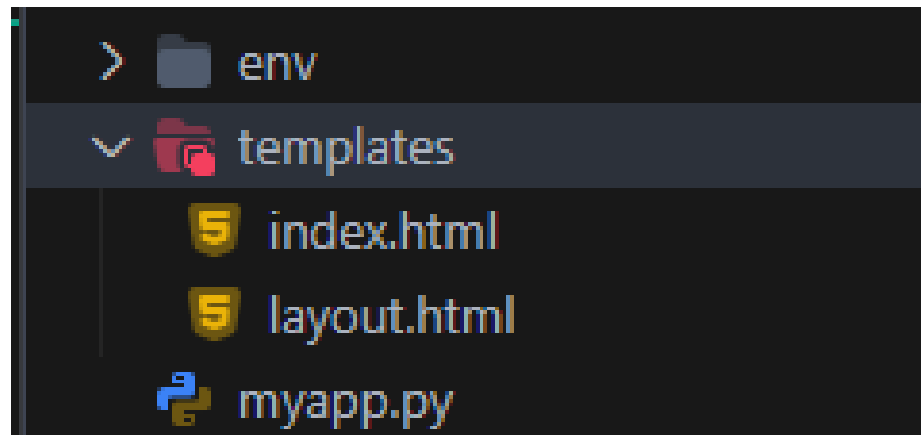
Setelah proses instalasi selesai langkah yang harus kita lakukan selanjutnya adalah memastikan bahwa flask benar benar sudah terinstal pada environment proyek kita. Untuk melakukan pengecekan kita dapat mengetikkan command berikut pada terminal dan pastikan terdapat Flask pada list package yang muncul.

```
1 pip list
```

```
1 Package          Version
2 -----
3 blinker           1.9.0
4 click             8.3.0
5 colorama          0.4.6
6 Flask             3.1.2
7 itsdangerous      2.2.0
8 Jinja2            3.1.6
9 MarkupSafe        3.0.3
10 pip              25.2
11 Werkzeug          3.1.3
```

c. Membuat sturktur folder.

Jika sudah ada Flask pada list package, maka kita bisa memulai proses setup proyek kita. Kita bisa memulai dengan membuat struktur folder terlebih dahulu. Berikut adalah struktur folder pada proyek flask ini.



- d. Menuliskan kode untuk menjalankan webserver.

Selanjutnya kita mulai menuliskan kode pada file `myapp.py`. Ini adalah file utama untuk membuat webserver menggunakan python. Pada file ini kita dapat membuat route, menangkap request misalnya dengan method GET dan POST.

```
1 from flask import Flask, render_template
2
3 app = Flask(__name__)
4
5 @app.route('/')
6 def home():
7     return render_template('index.html')
8
9 if __name__ == '__main__':
10     app.run(debug=True)
```

- e. Membuat layout untuk template / acuan halaman lainnya.

Setelah selesai menuliskan kode pada file `myapp.py` kita lanjutkan menuliskan kode html pada file `layout.html` yang berada di folder `templates`. File ini akan menjadi layout utama untuk halaman lainnya. Kode file ini berupa tag paling luar dari html yaitu `<html>`, `<head>`, dan `<body>`. Selain ada ketiga tag tersebut, terdapat 2 block yaitu `head` dan `body`, kedua block ini yang nantinya adalah bagian isi dari website kita.

```

1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6     {% block head %} {% endblock head %}
7   </head>
8   <body>
9     {% block body %} {% endblock body %}
10  </body>
11 </html>

```

f. Membuat isi untuk halaman indeks.

Sekarang kita mulai membuat sebuah halaman awal pada website kita pada file index.html yang berada di folder templates juga. Pada awal baris kode kita akan menuliskan sintaks `{% extends 'layout.html' %}` dari Jinja2 yang artinya mewariskan template awal dari layout.html yang sudah kita buat sebelumnya. Pada block head kita dapat mengisi dengan tag title untuk halaman ini, title ini yang nanti akan tampil sebagai nama tab pada browser. Bagian block body kita bisa mengisi dengan tag-tag lain pada html, bagian inilah yang nanti akan tampil sebagai isi halaman dari website kita.

```

1 {% extends "layout.html" %}
2
3 {% block head %}
4   <title>Home</title>
5 {% endblock head %}
6
7 {% block body %}
8   <h1>Welcome to My Web Page</h1>
9   <p>This is a simple web page using a base layout.</p>
10 {% endblock body %}

```

## BAB II. INTEGRASI FLASK DAN BOOTSTRAP

### 2.1 Integrasi Bootstrap

Bootstrap adalah framework HTML, CSS, dan JavaScript yang open source dan digunakan untuk pengembangan website yang responsif. Framework ini menyediakan berbagai komponen yang siap digunakan. Ada beberapa cara untuk melakukan integrasi proyek kita dengan Bootstrap, yang paling mudah adalah menggunakan CDN. Tetapi, kelemahan CDN ini adalah harus membutuhkan koneksi internet, jadi nanti akan ada 2 cara integrasi, yaitu CDN dan manual.

a. Integrasi menggunakan CDN

Untuk melakukan integrasi menggunakan CDN kita perlu membuka dokumentasi pada halaman website resmi Bootstrap <https://getbootstrap.com/docs/5.3/getting-started/introduction/>. Salin pada bagian tag <link> dan <script> pada dokumentasi Bootstrap.

```
1 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.8/dist/css/
  bootstrap.min.css" rel="stylesheet" integrity="sha384-
  sRIL4kxILFvY47J16cr9ZwB07vP4J8+LH7qKQnuqkuIAvNWLzeN8tE5YBujZqJLB"
  crossorigin="anonymous">
2
3 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.8/dist/js/
  bootstrap.bundle.min.js" integrity="sha384-
  FKyoEForCGLyvwx9Hj09JcYn3nv7wiPVLz7YYwJrWVcXK/BmnVDxM+D2scQbITxI"
  crossorigin="anonymous"></script>
```

Kemudian paste tak <link> sebelumnya pada bagian <head> dari file layout.html pada proyek kita. Untuk tag <script> paste pada bagian paling bawah body sebelum tag penutup.

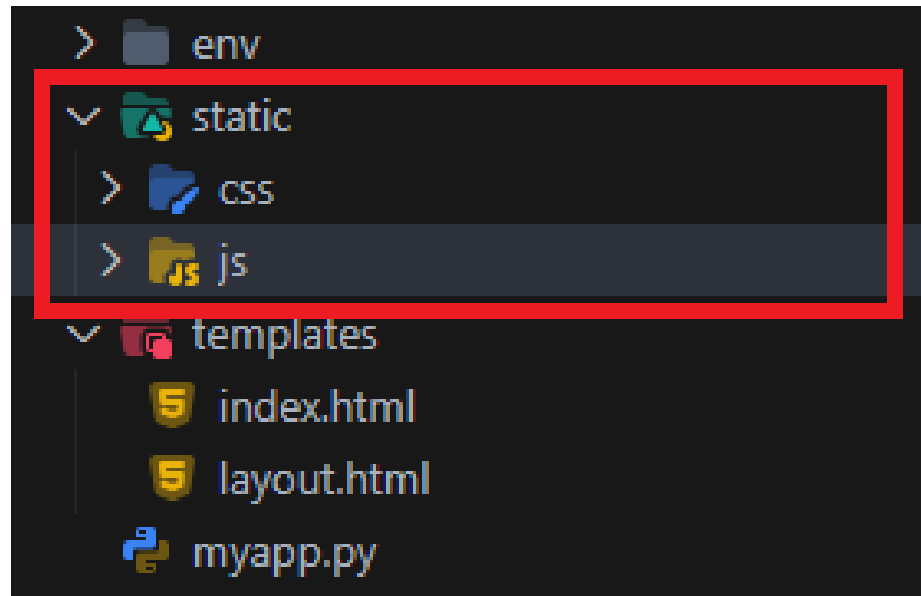
```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.8/dist/css/bootstrap.min.css"
7       rel="stylesheet"
8       integrity="sha384-sRIL4kxILFvY47J16cr9ZwB07vP4J8+LH7qKQnuqkuIAvNWLzeN8tE5YBujZqJLB"
9       crossorigin="anonymous">
10    {% block head %} {% endblock %}
11  </head>
12  <body>
13    {% block body %} {% endblock %}
14    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.8/dist/js/bootstrap.bundle.min.js"
15      integrity="sha384-FKyoEForCGLyvwx9Hj09JcYn3nv7wiPVLz7YYwJrWVcXK/BmnVDxM+D2scQbITxI"
16      crossorigin="anonymous">
17    </script>
18  </body>
19 </html>
```

Jika sudah seperti ini artinya kita sudah berhasil mengintegrasikan Bootstrap dengan proyek kita menggunakan CDN.

b. Integrasi manual

Seperti yang sudah disebutkan sebelumnya kelebihan dari instalasi manual adalah tetap bisa digunakan walaupun tidak ada internet dan ready production app. Untuk instalasi manual kita perlu mengunduh file zip dari halaman website resmi Bootstrap <https://getbootstrap.com/docs/5.3/getting-started/download/>. Klik pada tombol download atau unduh pada bagian *Compiled CSS and JS*. Setelah proses unduh selesai file zip teresbut kemudian diekstrak lalu salin folder css dan js ke folder static pada proyek kita.





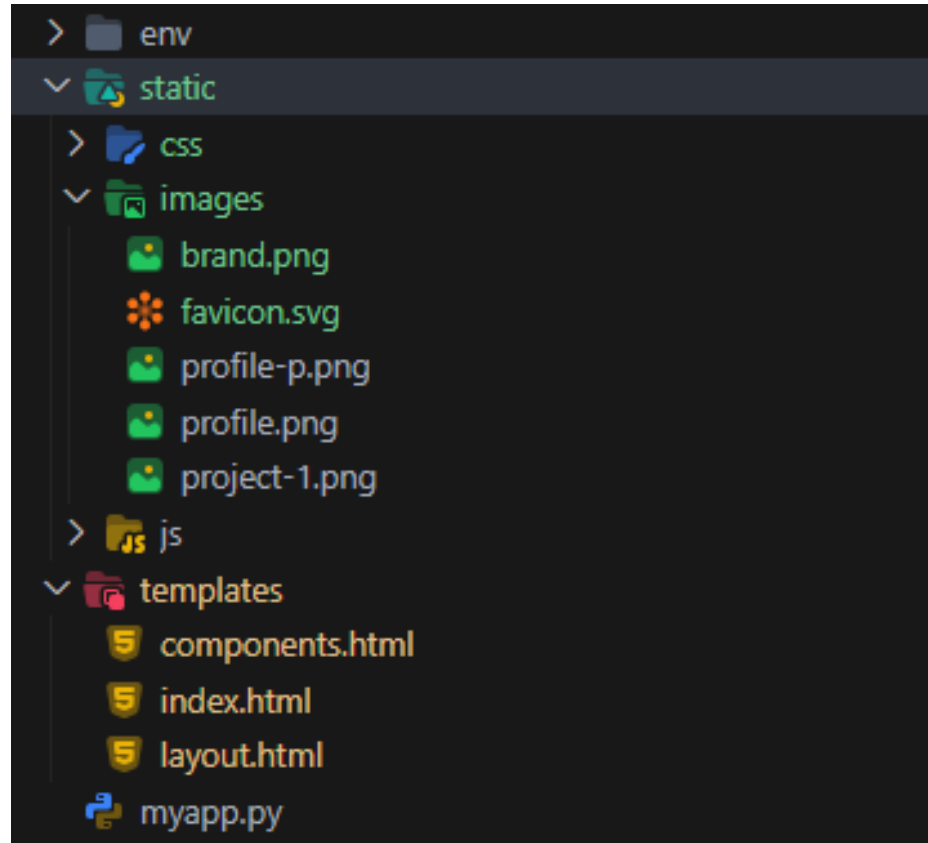
Setelah proses menyalin sudah selesai, kita dapat memulai mengintegrasikan proyek kita dengan Bootstrap. Caranya mirip seperti menggunakan CDN sebelumnya, hanya perlu menambahkan tag `<link>` dan `<script>`

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6     <link
7       rel="stylesheet"
8       href="{{ url_for('static', filename='css/bootstrap.min.css') }}"
9     />
10    {% block head %} {% endblock %}
11  </head>
12  <body>
13    {% block body %} {% endblock %}
14    <script src="{{ url_for('static', filename='js/bootstrap.bundle.min.js') }}"></script>
15  </body>
16 </html>
```

Jika sudah seperti ini maka proyek kita sudah terintegrasi dengan Bootstrap dengan cara manual.

## 2.2 Tugas

- a. Struktur folder



- b. Kode pada myapp.py

```
1 from flask import Flask, render_template
2
3 app = Flask(__name__)
4
5 @app.route('/')
6 def home():
7     return render_template('index.html')
8
9 if __name__ == '__main__':
10     app.run(debug=True)
```

### c. Kode pada layout.html

File ini berfungsi sebagai template untuk membuat halaman kedepannya. Ini akan memudahkan proses pembuatan website, kita tidak perlu membuat ulang layout untuk masing-masing halaman. Karena navbar akan tampil pada semua halaman maka meletakkan navbar pada layout.html membuat kita tidak membuat ulang kode yang sama pada tiap halaman.

```
1 {% import 'components.html' as c %}
2
3 <!DOCTYPE html>
4 <html lang="en">
5   <head>
6     <meta charset="UTF-8" />
7     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
8     <link rel="icon" type="image/svg" href="{{ url_for('static', filename='images/favicon.svg') }}" />
9     <link
10       rel="stylesheet"
11       href="{{ url_for('static', filename='css/bootstrap.min.css') }}"
12     />
13     <link href="https://unpkg.com/aos@2.3.4/dist/aos.css" rel="stylesheet">
14     <link
15       rel="stylesheet"
16       href="{{ url_for('static', filename='css/style.css') }}"
17     />
18     <link
19       rel="stylesheet"
20       href="https://cdnjs.cloudflare.com/ajax/libs/animate.css/4.1.1/animate.min.css"
21     />
22     <link href="https://fonts.googleapis.com/css2?family=Geist:wght@100..900&display=swap" rel="stylesheet">
23   {% block head %} {% endblock %}
24 </head>
25 <body class="bg-body-tertiary">
26   <nav class="navbar navbar-expand-lg fixed-top px-3 animate__animated animate__fadeInDown">
27     <div class="container-fluid">
28       <a class="navbar-brand" href="#">
29         
35       </a>
36
37       <div class="ms-auto gap-3 d-flex align-items-center">
38         <div
39           class="collapse navbar-collapse justify-content-center "
40           id="navbarNavAltMarkup"
41         >
42           <div class="navbar-nav">
43             <a class="nav-link" aria-current="page" href="#home">Home</a>
44             <a class="nav-link" aria-current="page" href="#about">About</a>
45           </div>
46         </div>
47         <a href="https://github.com/muzaaqi" target="_blank">
48           {{ c.Icon("github", size=24) }}
49         </a>
50         <button
51           class="navbar-toggler"
52           type="button"
53           data-bs-toggle="collapse"
54           data-bs-target="#navbarNavAltMarkup"
55           aria-controls="navbarNavAltMarkup"
56           aria-expanded="false"
57           aria-label="Toggle navigation"
58         >
59           <span class="navbar-toggler-icon"></span>
60         </button>
61       </div>
62     </div>
63   </nav>
64
65   {% block body %} {% endblock %}
66   <script src="{{ url_for('static', filename='js/bootstrap.bundle.min.js') }}"></script>
67 </html>
```

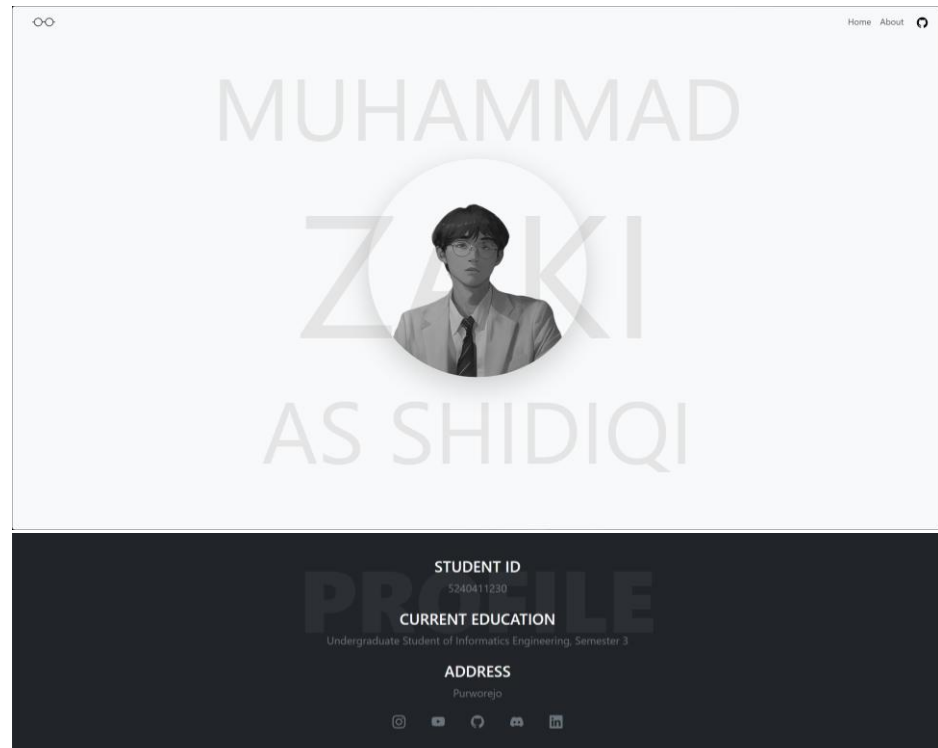
d. Kode pada index.html

File ini sebagai halaman awal yang pengguna lihat pada website. Pada halaman ini saya menampilkan foto, nama, npm, pendidikan yang ditempuh, alamat, dan beberapa icon sosial media.

```
1 {% import 'components.html' as c %} {% extends 'layout.html' %} {% block head %}
2
3 <title>MUZAAQI</title>
4
5 {% endblock %} {% block body %}
6 <div
7   id="home"
8   class="d-md-flex position-relative align-items-center justify-content-center vh-100 overflow-hidden"
9 >
10   <div
11     class="position-absolute top-50 start-50 translate-middle text-center text-uppercase"
12     style="z-index: 1; color: rgba(0, 0, 0, 0.08); font-weight: 900;"
13   >
14     <h1 class="animate__animated animate__bounceInLeft" style="font-size: 10rem;">Muhammad</h1>
15     <h1 class="animate__animated animate__bounceInRight" style="font-size: 20rem;">Zaki</h1>
16     <h1 class="animate__animated animate__bounceInLeft" style="font-size: 10rem;">As Shidiqi</h1>
17   </div>
18   <div class="position-relative z-2 animate__animated animate__bounceIn animate__delay-1s">
19     
25   </div>
26 </div>
27
28 <section id="about" class="bg-dark text-white py-5 position-relative">
29   <div class="container text-center">
30     <div class="d-flex flex-column align-items-center gap-4">
31       <div>
32         <h3 class="fw-semibold text-uppercase mb-2">Student ID</h3>
33         <p class="fs-5 text-secondary mb-0">5240411230</p>
34       </div>
35       <div>
36         <h3 class="fw-semibold text-uppercase mb-2">Current Education</h3>
37         <p class="fs-5 text-secondary mb-0">
38           Undergraduate Student of Informatics Engineering, Semester 3
39         </p>
40       </div>
41       <div>
42         <h3 class="fw-semibold text-uppercase mb-2">Address</h3>
43         <p class="fs-5 text-secondary mb-0">Purworejo</p>
44       </div>
45       <div class="d-flex gap-5">
46         {{ c.Icon("instagram", size=24, url="https://instagram.com/muzaaqi/") }}
47         {{ c.Icon("youtube", size=24, url="https://youtube.com/@muzaaqi.") }}
48         {{ c.Icon("github", size=24, url="https://github.com/muzaaqi") }}
49         {{ c.Icon("discord", size=24, url="https://discord.com/users/494420968492761089") }}
50         {{ c.Icon("linkedin", size=24, url="https://linkedin.com/in/muzaaqi/") }}
51       </div>
52     </div>
53   </div>
54   <div
55     class="position-absolute top-0 start-50 translate-middle-x opacity-10"
56     style="font-size: 10rem; font-weight: 800; color: rgba(255, 255, 255, 0.05); pointer-events: none;"
57   >
58     PROFILE
59   </div>
60 </div>
61 </section>
62 {% endblock %}
63
```

[illegible]

f. Hasil



**Kode selengkapnya:**

<https://github.com/muzaaqi/pemrograman-web-praktik/tree/main/meet-2/tugas>

## KESIMPULAN

Melalui praktikum ini, dapat disimpulkan bahwa Flask dan Bootstrap dapat dikombinasikan untuk membangun website yang modern responsif dan mudah. Flask sebagai micro framework memudahkan kita untuk mengelola proyek karena strukturnya yang sangat simpel, mempermudah dalam routing, dan integrasi dengan HTML melalui Jinja2 Engine. Bootstrap juga berperan penting sebagai UI framework yang memberikan component untuk menyusun UI pada website, apalagi component yang diberikan sudah responsif, jadi kita tidak perlu menyusun untuk tampilan mobile dan desktop dari awal.

Dari hasil praktikum, penerapan kedua teknologi ini terbukti dapat meningkatkan efisiensi dalam proses pengembangan web. Selain itu, kolaborasi antara Flask dan Bootstrap juga mendukung penerapan konsep *separation of concerns* yang memisahkan logika pemrograman dan tampilan dengan baik.