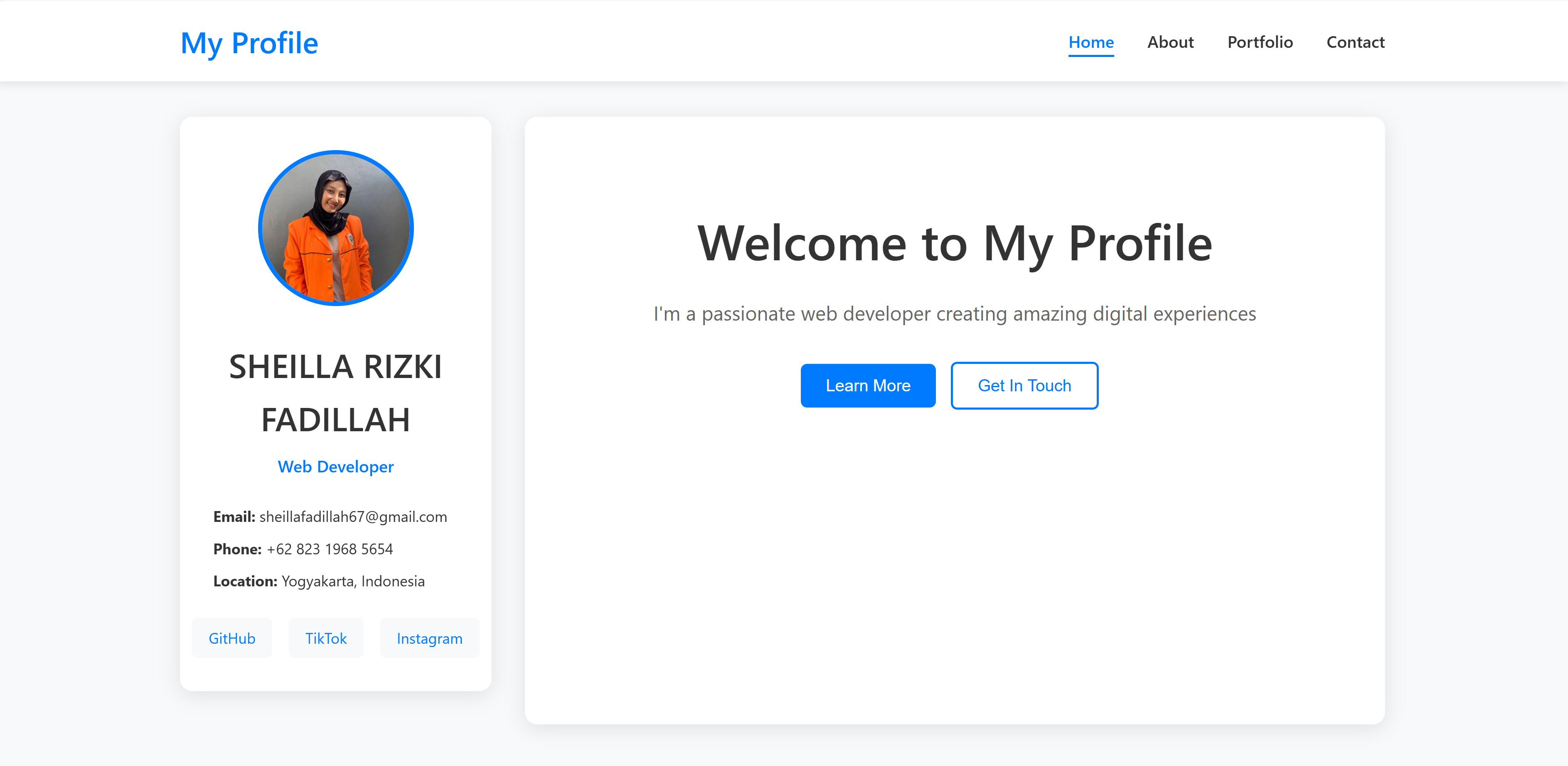
**Tugas Projek**

**Pemrograman Web Profil**

Disusun untuk memenuhi tugas projek Pemrograman web

Dosen Pengampu : Ali Tarmuji, S.T., M.Cs.



**Disusun oleh:**

**NAMA : Sheilla Rizki Fadillah**

**NIM : 2400018115**

**KELAS : B**

**Link Github :**

<https://github.com/Sheilla1812/Pweb-B-Sheilla-Rizki-Fadillah-2400018115->

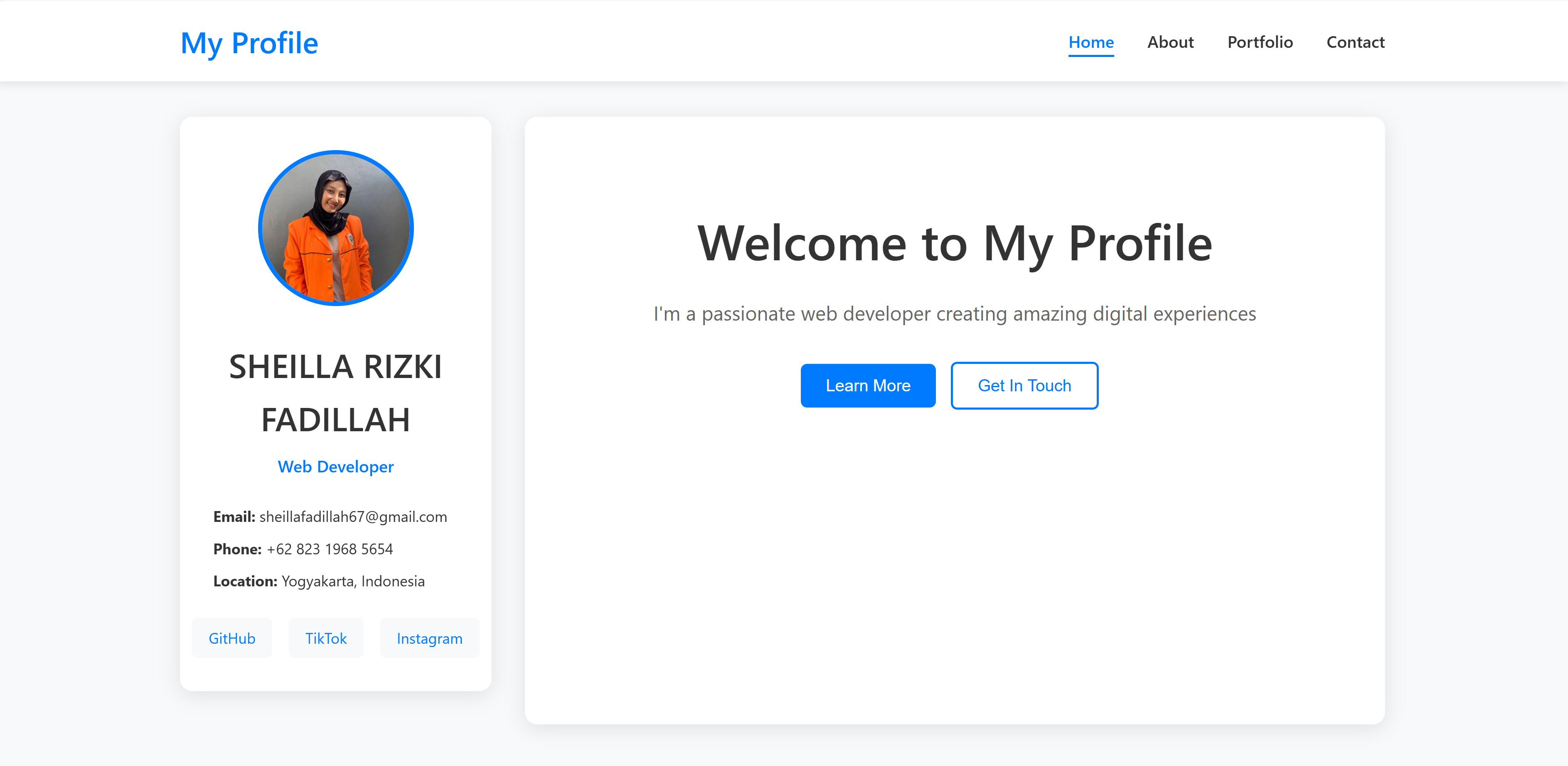
## PROGRAM STUDI S1 INFORMATIKA

**FAKULTAS TEKNOLOGI INDUSTRI**

**UNIVERSITAS AHMAD DAHLAN**

**2025/2026**

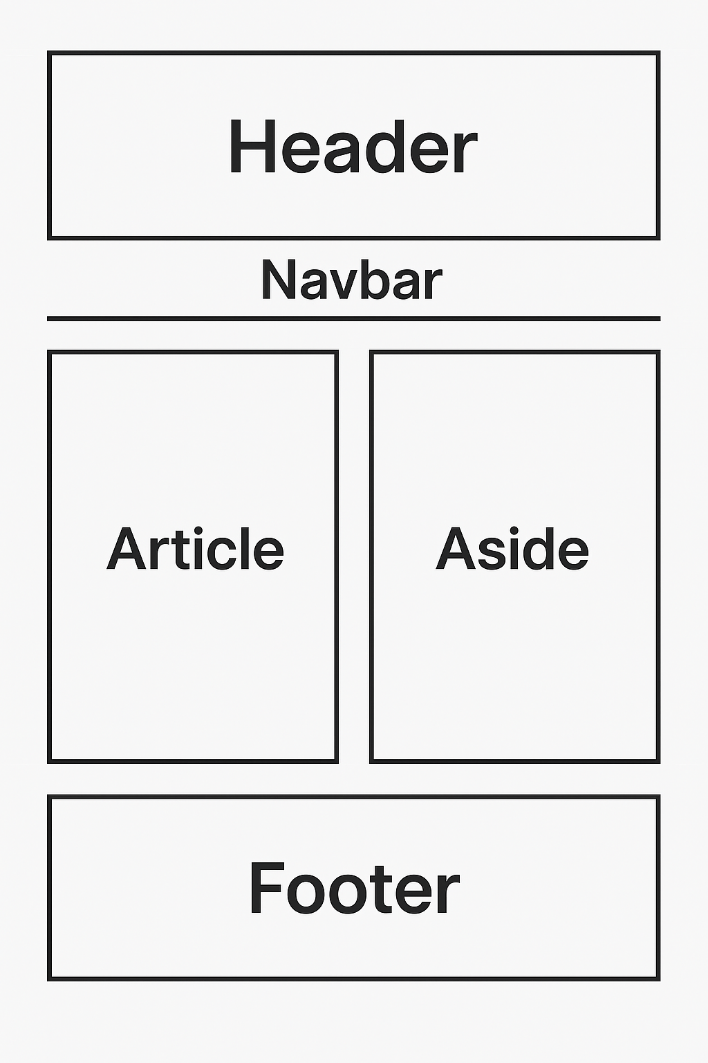
### A. TAMPILAN AWALAN WEB



#### B. DESKRIPSI PROJEK

Program ini adalah tampilan halaman profil yang memperkenalkan Sheilla Rizki Fadillah, seorang mahasiswa Universitas Ahmad Dahlan yang bercita cita menjadi web developer dan lulus tanpa skripsi. Halaman ini menampilkan deskripsi diri, projek yang sudah dibuat, Riwayat pendidikan dan sosial media. Terdapat bagian “Home” yang disertai foto diri, ada juga bagian “About” yang mendeskripsikan diri dari pengalaman, bagian “portofolio” yang menampilkan berapa jumlah projek yang sudah saya kerjakan dan yang terakhir bagian “contact” yang berisikan link untuk menuju ke sosial media.

##### C. RANCANGAN DESAIN WEB



##### D. PENJELASAN DESAIN WEB

* Header

Berada di bagian atas halaman, berfungsi sebagai tempat untuk menampilkan judul atau nama pemilik situs. Biasanya, header ini menonjol dengan ukuran font yang lebih besar dan warna yang mencolok untuk menarik perhatian pengguna.

* Navbar

Terletak tepat di bawah header, berisi menu navigasi untuk memudahkan pengguna berpindah antara halaman. Navbar ini biasanya menggunakan tautan (link) untuk menghubungkan berbagai bagian atau halaman lain di dalam situs.

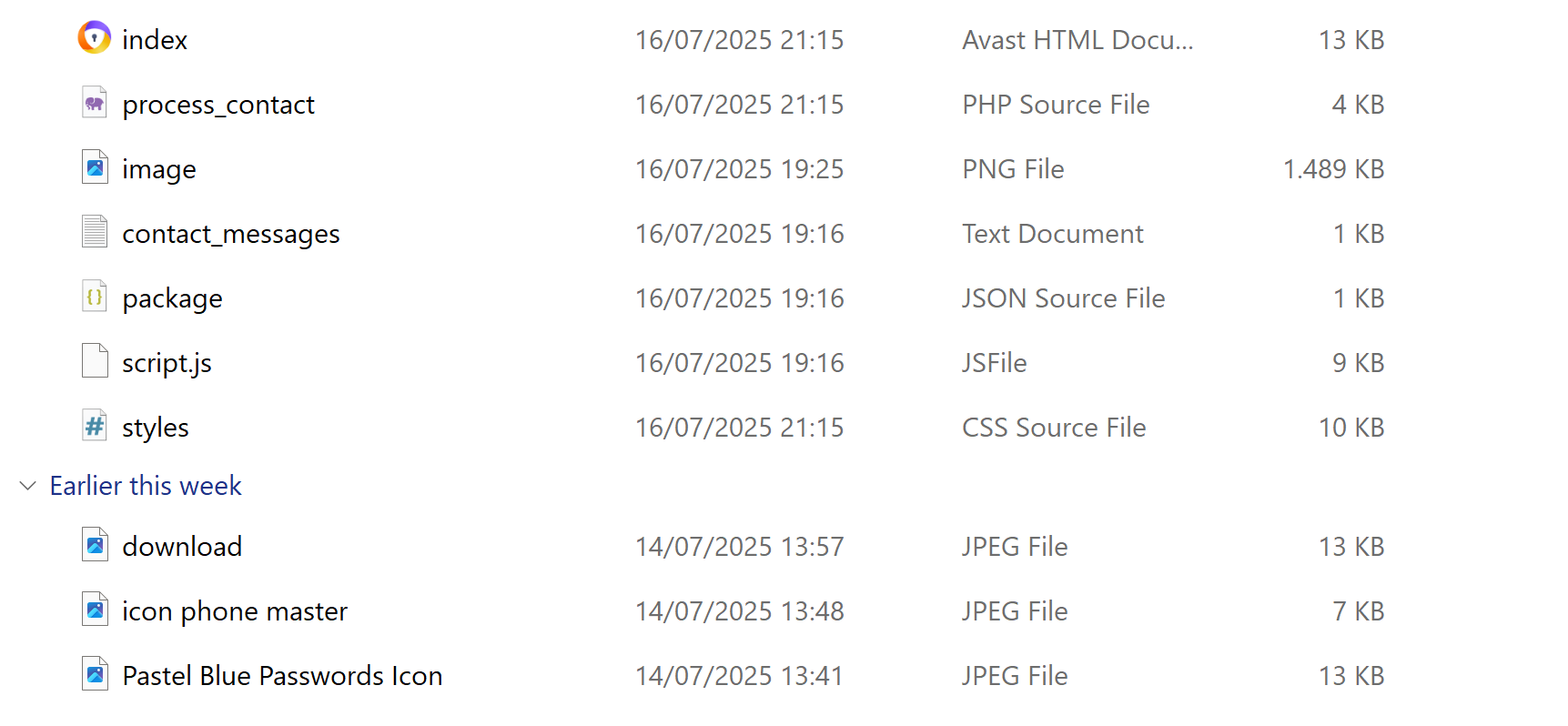
* Main Section

Dibagi menjadi dua kolom utama:

* Article: Area konten utama yang berfungsi untuk menampilkan teks atau informasi utama yang ingin disampaikan, seperti pengenalan diri, pengalaman, atau proyek.
* Aside: Bagian samping yang sering digunakan untuk menampilkan konten tambahan, seperti profil singkat, daftar artikel terbaru, atau kontak sosial media.
* Footer

Terletak di bagian paling bawah halaman, berfungsi untuk menampilkan informasi penutup, seperti hak cipta, kontak, atau pesan motivasi.

**E. SCRENSHOOT TAMPILAN FILE**

****

**F. PENJELASAN SCREENSHOT FILE**

***1. index***

Tipe: HTML File.

Fungsi: Halaman utama dari website (homepage).

Isinya kemungkinan ada <header>, <main>, <footer> dan terhubung ke styles.css dan script.js.

***2. process\_contact***

Tipe: PHP Source File

Fungsi: File backend buat proses form kontak.

Misalnya: ketika user kirim pesan, data dikirim ke sini lalu diproses (disimpan/dikirim via email).

***3. image***

Fungsi: Gambar pendukung web, bisa jadi logo, banner, atau ilustrasi.

Dipanggil di HTML seperti:

html

SalinEdit

<img src="image.png" alt="gambar">

***4. contact\_messages***

Tipe: Text Document (.txt)

Fungsi: Menyimpan data pesan dari form kontak secara lokal.

Format isi bisa berupa:

Nama: Sella

Email: sella@email.com

Pesan: Halo, ini pesan saya!

***5. package***

Tipe: JSON File

Nama lengkap mungkin package.json

Fungsi: Menyimpan metadata proyek (biasanya buat Node.js/NPM), seperti:

Nama proyek

Versi

Dependensi (library yang dipakai)

***7. script.js***

Tipe: JavaScript File

Fungsi: Nambahin interaktivitas ke web.

Contoh fitur:

Validasi form

Efek hover/klik

Notifikasi/alert

Disambungkan di HTML pakai:

html

SalinEdit

<script src="script.js"></script>

***8. styles***

Tipe: CSS File

Fungsi: Mengatur tampilan web seperti warna, font, layout.

Disambungkan ke HTML lewat:

<link rel="stylesheet" href="styles.css">

#### G. TEXT SOURCE CODE

* Index.html

|  |
| --- |
| <!DOCTYPE html>  <html lang="id">  <head>  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <meta name="description" content="Profil Personal - Portfolio dan Informasi Pribadi">  <meta name="keywords" content="profil, portfolio, personal, web developer">  <meta name="author" content="Your Name">  <title>Profil Personal - sella</title>  <link rel="stylesheet" href="styles.css">  <link rel="icon" type="image/x-icon" href="favicon.ico">  </head>  <body>  <!-- Header Section -->  <header class="header">  <div class="container">  <div class="logo">  <h1>My Profile</h1>  </div>  <nav class="navigation">  <ul>  <li><a href="#home" class="nav-link active">Home</a></li>  <li><a href="#about" class="nav-link">About</a></li>  <li><a href="#portfolio" class="nav-link">Portfolio</a></li>  <li><a href="#contact" class="nav-link">Contact</a></li>  </ul>  </nav>  <div class="mobile-menu-toggle">  <span></span>  <span></span>  <span></span>  </div>  </div>  </header>  <!-- Main Content -->  <main class="main-content">  <div class="container">  <!-- Sidebar -->  <aside class="sidebar">  <div class="profile-card">  <div class="profile-image">  <img src="image.png" alt="Profile Photo" id="profile-img">  </div>  <div class="profile-info">  <h2>SHEILLA RIZKI FADILLAH</h2>  <p class="title">Web Developer</p>  <div class="contact-info">  <p><strong>Email:</strong> sheillafadillah67@gmail.com</p>  <p><strong>Phone:</strong> +62 823 1968 5654</p>  <p><strong>Location:</strong> Yogyakarta, Indonesia</p>  </div>  <div class="social-links">  <a href="https://github.com/Sheilla1812" class="social-link" target="\_blank">GitHub</a>  <a href="https://www.tiktok.com/@helloshel\_12" class="social-link" target="\_blank">TikTok</a>  <a href="https://www.instagram.com/sheillarizkifdllh/" class="social-link" target="\_blank">Instagram</a>  </div>  </div>  </div>  </aside>  <!-- Article/Content -->  <article class="content">  <!-- Home Section -->  <section id="home" class="section active">  <div class="hero">  <h1 class="hero-title">Welcome to My Profile</h1>  <p class="hero-subtitle">I'm a passionate web developer creating amazing digital experiences</p>  <div class="hero-buttons">  <button class="btn btn-primary" onclick="scrollToSection('about')">Learn More</button>  <button class="btn btn-secondary" onclick="scrollToSection('contact')">Get In Touch</button>  </div>  </div>  </section>  <!-- About Section -->  <section id="about" class="section">  <h2 class="section-title">About Me</h2>  <div class="about-content">  <p>Hello! I am an informatics student who loves traveling and dreams of becoming a cool web developer.</p>    <div class="skills-grid">  <div class="skill-category">  <h3>skill</h3>  <ul>  <li>HTML</li>  <li>JavaScript</li>  <li>c++</li>  <li>css</li>  </ul>  </div>  <div class="skill-category">  <h3>Tools</h3>  <ul>  <li>GitHub</li>  <li>VS Code</li>  <li>Figma</li>  <li>dev c++</li>  </ul>  </div>  </div>  <div class="education">  <h3>Education</h3>  <div class="education-item">  <h4>Engineering department</h4>  <p>Universitas Ahmad Dahlan | 2024 - present</p>  </div>  </div>  </div>  </section>  <!-- Portfolio Section -->  <section id="portfolio" class="section">  <h2 class="section-title">Portfolio</h2>  <div class="portfolio-grid">  <div class="portfolio-item">  <div class="portfolio-image">  <img src="Pastel Blue Passwords Icon.jpeg" alt="Project 1">  </div>  <div class="portfolio-content">  <h3>Application login</h3>  <p>create an application login design using assembly</p>  <div class="portfolio-tags">  <a href="https://github.com/Sheilla1812/Program-Aplikasi-Login" target="\_blank">  <span class="tag">Go to file</span></a>  </div>  </div>  </div>  <div class="portfolio-item">  <div class="portfolio-image">  <img src="download.jpeg" alt="Project 2">  </div>  <div class="portfolio-content">  <h3>Minimarket shopping structure</h3>  <p>A printing minimarket shopping structure using c++</p>  <div class="portfolio-tags">  <a href="https://github.com/Sheilla1812/Tugas-akhir-DDP/blob/main/Tugas%20akhir%20DDP.cpp" target="\_blank">  <span class="tag">Go to file</span></a>  </div>  </div>  </div>  <div class="portfolio-item">  <div class="portfolio-image">  <img src="icon phone master.jpeg" alt="Project 3">  </div>  <div class="portfolio-content">  <h3>Profile Website</h3>  <p>create a personal website using</p>  <div class="portfolio-tags">  <a href="https://github.com/Sheilla1812/PWEB2025-B-2400018115-Sheilla-Rizki-Fadillah-Web-Profil" target="\_blank">  <span class="tag">Go to file</span></a>  </div>  </div>  </div>  </div>  </section>  <!-- Contact Section -->  <section id="contact" class="section">  <h2 class="section-title">Contact Me</h2>  <div class="contact-content">  <div class="contact-info-section">  <h3>Get In Touch</h3>  <p>I'm always interested in new opportunities and collaborations. Feel free to reach out!</p>    <div class="contact-methods">  <div class="contact-method">  <strong>Email:</strong> sheillafadillah67@gmail.com  </div>  <div class="contact-method">  <strong>Phone:</strong> +62 823 1968 5654  </div>  <div class="contact-method">  <strong>Address:</strong> Yogyakarta, Indonesia  </div>  </div>  </div>  <div class="contact-form-section">  <form id="contactForm" class="contact-form" action="process\_contact.php" method="POST">  <div class="form-group">  <label for="name">Name \*</label>  <input type="text" id="name" name="name" required>  <span class="error-message" id="name-error"></span>  </div>  <div class="form-group">  <label for="email">Email \*</label>  <input type="email" id="email" name="email" required>  <span class="error-message" id="email-error"></span>  </div>  <div class="form-group">  <label for="phone">Phone</label>  <input type="tel" id="phone" name="phone">  <span class="error-message" id="phone-error"></span>  </div>  <div class="form-group">  <label for="subject">Subject \*</label>  <input type="text" id="subject" name="subject" required>  <span class="error-message" id="subject-error"></span>  </div>  <div class="form-group">  <label for="message">Message \*</label>  <textarea id="message" name="message" rows="5" required></textarea>  <span class="error-message" id="message-error"></span>  </div>  <button type="submit" class="btn btn-primary">Send Message</button>  </form>  </div>  </div>  </section>  </article>  </div>  </main>  <!-- Footer -->  <footer class="footer">  <div class="container">  <div class="footer-content">  <div class="footer-section">  <h3>Sheilla Rizki Fadillah</h3>  <p>Web Developer passionate about creating amazing digital experiences.</p>  </div>  <div class="footer-section">  <h3>Quick Links</h3>  <ul>  <li><a href="#home">Home</a></li>  <li><a href="#about">About</a></li>  <li><a href="#portfolio">Portfolio</a></li>  <li><a href="#contact">Contact</a></li>  </ul>  </div>  <div class="footer-section">  <h3>Connect</h3>  <div class="social-links">  <a href="https://github.com/Sheilla1812" class="social-link" target="\_blank">GitHub</a>  <a href="https://www.tiktok.com/@helloshel\_12" class="social-link" target="\_blank">TikTok</a>  <a href="https://www.instagram.com/sheillarizkifdllh/" class="social-link" target="\_blank">Instagram</a>  </div>  </div>  </div>  <div class="footer-bottom">  <p>&copy; 2025 Sheilla Rizki Fadillah. All rights reserved. | Designed with love</p>  </div>  </div>  </footer>  <script src="script.js"></script>  </body>  </html> |

* Styles.css

|  |
| --- |
| /\* Reset and Base Styles \*/  \* {  margin: 0;  padding: 0;  box-sizing: border-box;  }  body {  font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;  line-height: 1.6;  color: #333;  background-color: #f8f9fa;  }  .container {  max-width: 1200px;  margin: 0 auto;  padding: 0 20px;  }  /\* Design Principle 1: Hierarchy - Clear visual hierarchy through typography \*/  h1,  h2,  h3,  h4,  h5,  h6 {  font-weight: 600;  margin-bottom: 1rem;  }  h1 {  font-size: 2.5rem;  }  h2 {  font-size: 2rem;  }  h3 {  font-size: 1.5rem;  }  h4 {  font-size: 1.25rem;  }  /\* Design Principle 2: Contrast - High contrast for readability \*/  .btn {  padding: 12px 24px;  border: none;  border-radius: 6px;  cursor: pointer;  font-size: 1rem;  font-weight: 500;  text-decoration: none;  display: inline-block;  transition: all 0.3s ease;  margin-right: 10px;  }  .btn-primary {  background-color: #007bff;  color: white;  }  .btn-primary:hover {  background-color: #0056b3;  transform: translateY(-2px);  }  .btn-secondary {  background-color: transparent;  color: #007bff;  border: 2px solid #007bff;  }  .btn-secondary:hover {  background-color: #007bff;  color: white;  }  /\* Header Styles \*/  .header {  background-color: #fff;  box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);  position: fixed;  top: 0;  left: 0;  right: 0;  z-index: 1000;  }  .header .container {  display: flex;  justify-content: space-between;  align-items: center;  padding: 1rem 20px;  }  .logo h1 {  color: #007bff;  margin: 0;  font-size: 1.8rem;  }  .navigation ul {  display: flex;  list-style: none;  gap: 2rem;  }  .nav-link {  text-decoration: none;  color: #333;  font-weight: 500;  transition: color 0.3s ease;  position: relative;  }  .nav-link:hover,  .nav-link.active {  color: #007bff;  }  .nav-link.active::after {  content: "";  position: absolute;  bottom: -5px;  left: 0;  width: 100%;  height: 2px;  background-color: #007bff;  }  .mobile-menu-toggle {  display: none;  flex-direction: column;  cursor: pointer;  }  .mobile-menu-toggle span {  width: 25px;  height: 3px;  background-color: #333;  margin: 3px 0;  transition: 0.3s;  }  /\* Main Content Layout \*/  .main-content {  margin-top: 80px;  padding: 2rem 0;  }  .main-content .container {  display: grid;  grid-template-columns: 300px 1fr;  gap: 2rem;  }  /\* Design Principle 3: Alignment - Consistent alignment throughout \*/  .sidebar {  position: sticky;  top: 100px;  height: fit-content;  }  .profile-card {  background: white;  border-radius: 12px;  padding: 2rem;  box-shadow: 0 4px 20px rgba(0, 0, 0, 0.1);  margin-bottom: 2rem;  text-align: center;  }  .profile-image {  margin-bottom: 1.5rem;  }  .profile-image img {  width: 150px;  height: 150px;  border-radius: 50%;  object-fit: cover;  border: 4px solid #007bff;  }  .profile-info h2 {  color: #333;  margin-bottom: 0.5rem;  }  .title {  color: #007bff;  font-weight: 500;  margin-bottom: 1.5rem;  }  .contact-info {  text-align: left;  margin-bottom: 1.5rem;  }  .contact-info p {  margin-bottom: 0.5rem;  font-size: 0.9rem;  }  .social-links {  display: flex;  gap: 1rem;  justify-content: center;  }  .social-link {  padding: 8px 16px;  background-color: #f8f9fa;  color: #007bff;  text-decoration: none;  border-radius: 6px;  font-size: 0.9rem;  transition: all 0.3s ease;  }  .social-link:hover {  background-color: #007bff;  color: white;  }  .stats-card {  background: white;  border-radius: 12px;  padding: 2rem;  box-shadow: 0 4px 20px rgba(0, 0, 0, 0.1);  }  .stats-card h3 {  margin-bottom: 1.5rem;  text-align: center;  color: #333;  }  .stat-item {  display: flex;  justify-content: space-between;  align-items: center;  padding: 1rem 0;  border-bottom: 1px solid #eee;  }  .stat-item:last-child {  border-bottom: none;  }  .stat-number {  font-size: 1.5rem;  font-weight: bold;  color: #007bff;  }  .stat-label {  font-size: 0.9rem;  color: #666;  }  /\* Content Styles \*/  .content {  background: white;  border-radius: 12px;  box-shadow: 0 4px 20px rgba(0, 0, 0, 0.1);  overflow: hidden;  }  .section {  padding: 3rem;  display: none;  }  .section.active {  display: block;  }  .section-title {  color: #333;  margin-bottom: 2rem;  padding-bottom: 1rem;  border-bottom: 3px solid #007bff;  display: inline-block;  }  /\* Hero Section \*/  .hero {  text-align: center;  padding: 2rem 0;  }  .hero-title {  font-size: 3rem;  color: #333;  margin-bottom: 1rem;  }  .hero-subtitle {  font-size: 1.2rem;  color: #666;  margin-bottom: 2rem;  max-width: 600px;  margin-left: auto;  margin-right: auto;  }  .hero-buttons {  margin-top: 2rem;  }  /\* About Section \*/  .about-content p {  font-size: 1.1rem;  margin-bottom: 2rem;  color: #555;  line-height: 1.8;  }  /\* Design Principle 4: Repetition - Consistent grid patterns \*/  .skills-grid {  display: grid;  grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));  gap: 2rem;  margin-bottom: 3rem;  }  .skill-category {  background: #f8f9fa;  padding: 2rem;  border-radius: 8px;  border-left: 4px solid #007bff;  }  .skill-category h3 {  color: #007bff;  margin-bottom: 1rem;  }  .skill-category ul {  list-style: none;  }  .skill-category li {  padding: 0.5rem 0;  color: #555;  position: relative;  padding-left: 1.5rem;  }  .skill-category li::before {  content: " ";  position: absolute;  left: 0;  color: #007bff;  }  .education {  background: #f8f9fa;  padding: 2rem;  border-radius: 8px;  }  .education-item h4 {  color: #007bff;  margin-bottom: 0.5rem;  }  .education-item p {  color: #666;  }  /\* Portfolio Section \*/  .portfolio-grid {  display: grid;  grid-template-columns: repeat(auto-fit, minmax(300px, 1fr));  gap: 2rem;  }  .portfolio-item {  background: #f8f9fa;  border-radius: 12px;  overflow: hidden;  transition: transform 0.3s ease, box-shadow 0.3s ease;  }  .portfolio-item:hover {  transform: translateY(-5px);  box-shadow: 0 8px 30px rgba(0, 0, 0, 0.15);  }  .portfolio-image img {  width: 100%;  height: 200px;  object-fit: cover;  }  .portfolio-content {  padding: 1.5rem;  }  .portfolio-content h3 {  color: #333;  margin-bottom: 1rem;  }  .portfolio-content p {  color: #666;  margin-bottom: 1rem;  }  .portfolio-tags {  display: flex;  gap: 0.5rem;  flex-wrap: wrap;  }  .tag {  background: #007bff;  color: white;  padding: 0.25rem 0.75rem;  border-radius: 15px;  font-size: 0.8rem;  }  /\* Contact Section \*/  .contact-content {  display: grid;  grid-template-columns: 1fr 1fr;  gap: 3rem;  }  .contact-info-section h3 {  color: #007bff;  margin-bottom: 1rem;  }  .contact-info-section p {  color: #666;  margin-bottom: 2rem;  line-height: 1.8;  }  .contact-methods {  background: #f8f9fa;  padding: 2rem;  border-radius: 8px;  }  .contact-method {  margin-bottom: 1rem;  color: #555;  }  .contact-method strong {  color: #007bff;  }  /\* Form Styles \*/  .contact-form {  background: #f8f9fa;  padding: 2rem;  border-radius: 12px;  }  .form-group {  margin-bottom: 1.5rem;  }  .form-group label {  display: block;  margin-bottom: 0.5rem;  color: #333;  font-weight: 500;  }  .form-group input,  .form-group textarea {  width: 100%;  padding: 12px;  border: 2px solid #ddd;  border-radius: 6px;  font-size: 1rem;  transition: border-color 0.3s ease;  }  .form-group input:focus,  .form-group textarea:focus {  outline: none;  border-color: #007bff;  }  .form-group input.error,  .form-group textarea.error {  border-color: #dc3545;  }  .error-message {  color: #dc3545;  font-size: 0.9rem;  margin-top: 0.5rem;  display: block;  }  .success-message {  background: #d4edda;  color: #155724;  padding: 1rem;  border-radius: 6px;  margin-bottom: 1rem;  border: 1px solid #c3e6cb;  }  /\* Footer Styles \*/  .footer {  background-color: #333;  color: white;  padding: 3rem 0 1rem;  margin-top: 4rem;  }  .footer-content {  display: grid;  grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));  gap: 2rem;  margin-bottom: 2rem;  }  .footer-section h3 {  color: #007bff;  margin-bottom: 1rem;  }  .footer-section p {  color: #ccc;  line-height: 1.6;  }  .footer-section ul {  list-style: none;  }  .footer-section ul li {  margin-bottom: 0.5rem;  }  .footer-section ul li a {  color: #ccc;  text-decoration: none;  transition: color 0.3s ease;  }  .footer-section ul li a:hover {  color: #007bff;  }  .footer-bottom {  text-align: center;  padding-top: 2rem;  border-top: 1px solid #555;  color: #ccc;  }  /\* Responsive Design \*/  @media (max-width: 768px) {  .main-content .container {  grid-template-columns: 1fr;  }  .sidebar {  position: static;  order: 2;  }  .content {  order: 1;  }  .navigation ul {  display: none;  }  .mobile-menu-toggle {  display: flex;  }  .hero-title {  font-size: 2rem;  }  .contact-content {  grid-template-columns: 1fr;  }  .skills-grid {  grid-template-columns: 1fr;  }  .portfolio-grid {  grid-template-columns: 1fr;  }  .section {  padding: 2rem 1rem;  }  }  @media (max-width: 480px) {  .container {  padding: 0 15px;  }  .hero-title {  font-size: 1.8rem;  }  .hero-subtitle {  font-size: 1rem;  }  .profile-card {  padding: 1.5rem;  }  .stats-card {  padding: 1.5rem;  }  }  /\* Animation Classes \*/  .fade-in {  animation: fadeIn 0.6s ease-in;  }  @keyframes fadeIn {  from {  opacity: 0;  transform: translateY(20px);  }  to {  opacity: 1;  transform: translateY(0);  }  }  .slide-in-left {  animation: slideInLeft 0.6s ease-out;  }  @keyframes slideInLeft {  from {  opacity: 0;  transform: translateX(-50px);  }  to {  opacity: 1;  transform: translateX(0);  }  }  /\* Loading States \*/  .loading {  opacity: 0.6;  pointer-events: none;  }  .spinner {  border: 2px solid #f3f3f3;  border-top: 2px solid #007bff;  border-radius: 50%;  width: 20px;  height: 20px;  animation: spin 1s linear infinite;  display: inline-block;  margin-right: 10px;  }  @keyframes spin {  0% {  transform: rotate(0deg);  }  100% {  transform: rotate(360deg);  }  } |

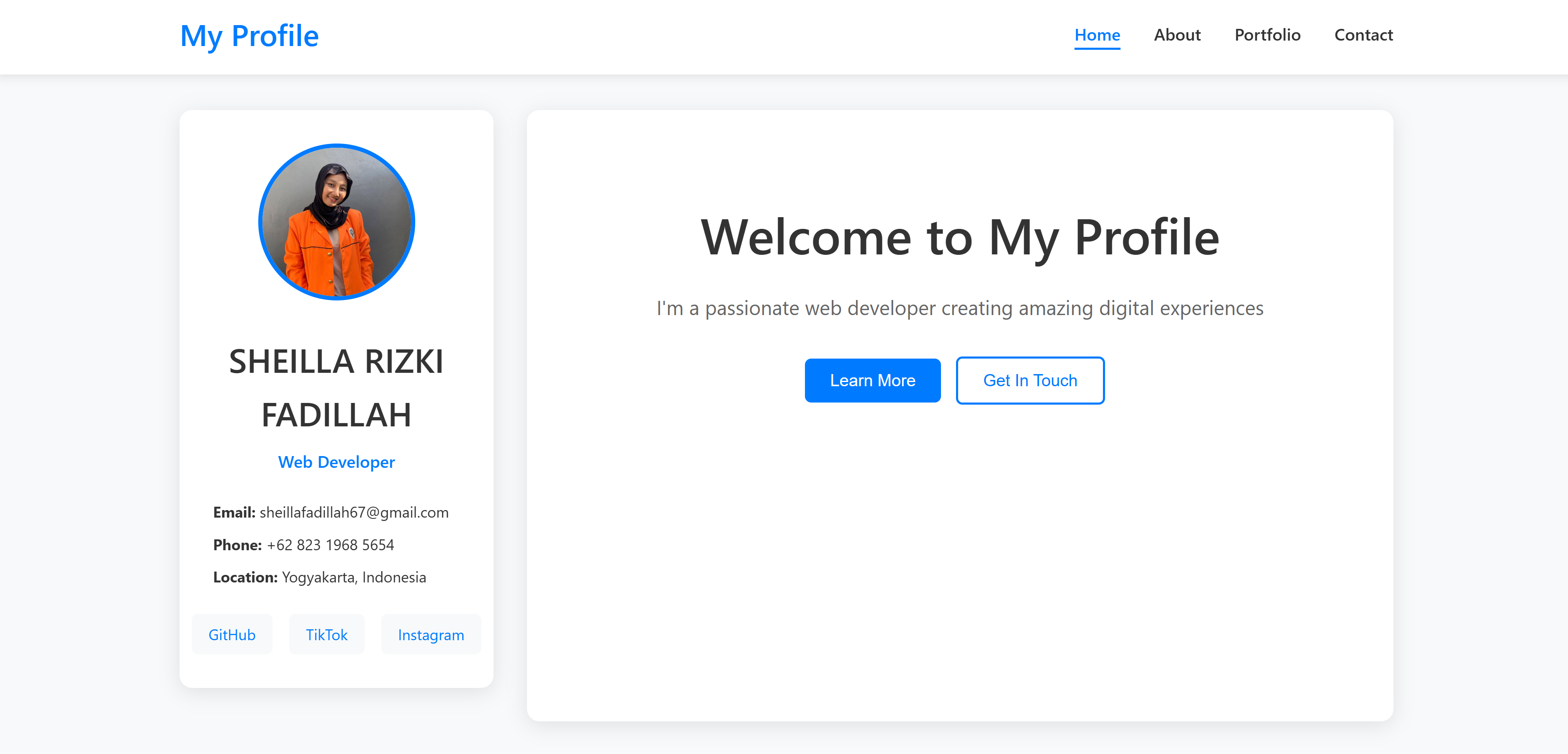
* Script.js

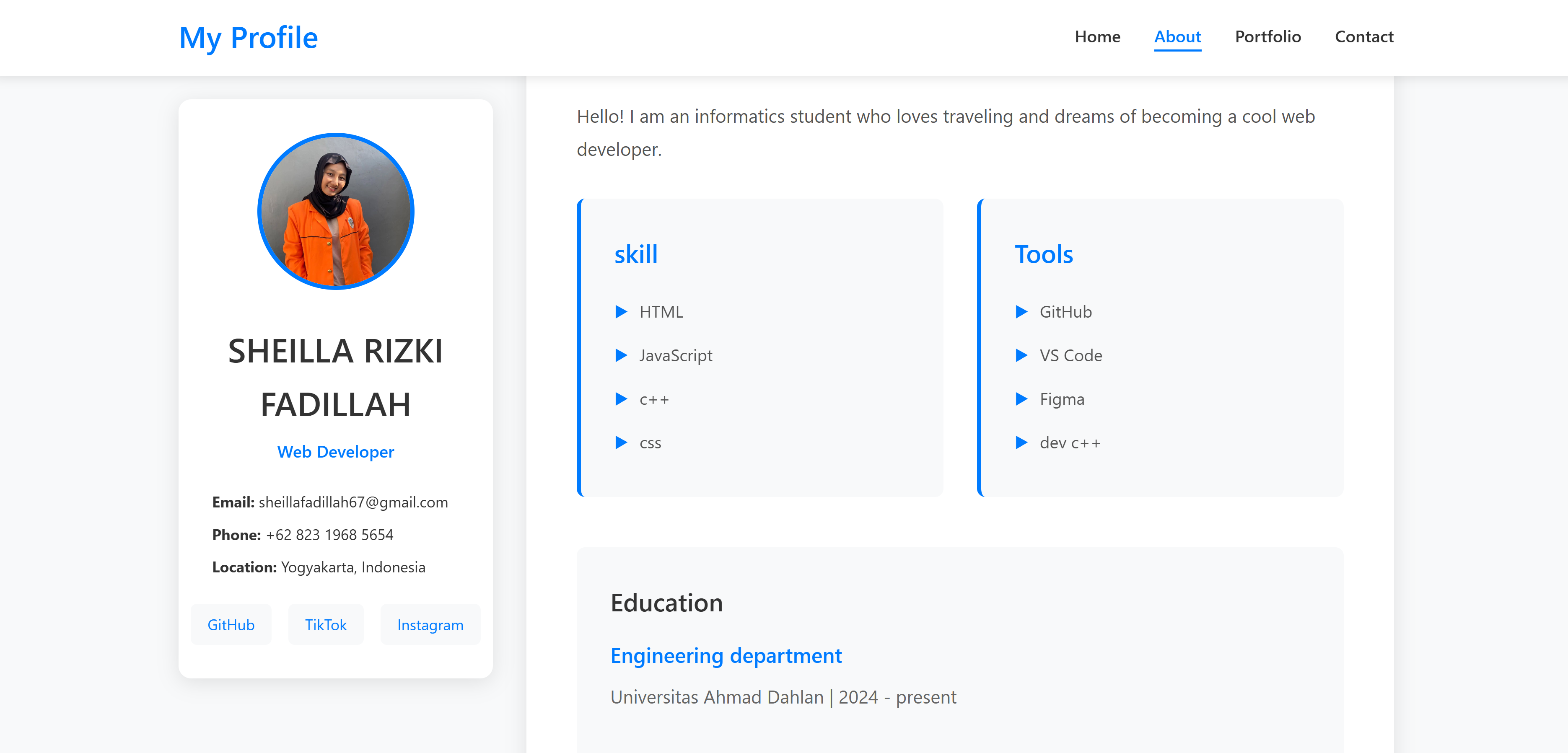
|  |
| --- |
| // DOM Content Loaded Event  document.addEventListener("DOMContentLoaded", () => {  initializeNavigation()  initializeFormValidation()  initializeMobileMenu()  initializeAnimations()  })  // Navigation Functions  function initializeNavigation() {  const navLinks = document.querySelectorAll(".nav-link")  const sections = document.querySelectorAll(".section")  navLinks.forEach((link) => {  link.addEventListener("click", function (e) {  e.preventDefault()  const targetId = this.getAttribute("href").substring(1)  showSection(targetId)  setActiveNavLink(this)  })  })  // Show home section by default  showSection("home")  }  function showSection(sectionId) {  const sections = document.querySelectorAll(".section")  sections.forEach((section) => {  section.classList.remove("active")  })  const targetSection = document.getElementById(sectionId)  if (targetSection) {  targetSection.classList.add("active")  targetSection.classList.add("fade-in")  }  }  function setActiveNavLink(activeLink) {  const navLinks = document.querySelectorAll(".nav-link")  navLinks.forEach((link) => {  link.classList.remove("active")  })  activeLink.classList.add("active")  }  function scrollToSection(sectionId) {  showSection(sectionId)  const navLink = document.querySelector(`a[href="#${sectionId}"]`)  if (navLink) {  setActiveNavLink(navLink)  }  }  // Mobile Menu Functions  function initializeMobileMenu() {  const mobileToggle = document.querySelector(".mobile-menu-toggle")  const navigation = document.querySelector(".navigation")  if (mobileToggle) {  mobileToggle.addEventListener("click", function () {  navigation.classList.toggle("mobile-active")  this.classList.toggle("active")  })  }  }  // Form Validation Functions  function initializeFormValidation() {  const contactForm = document.getElementById("contactForm")  if (contactForm) {  contactForm.addEventListener("submit", handleFormSubmit)  // Real-time validation  const inputs = contactForm.querySelectorAll("input, textarea")  inputs.forEach((input) => {  input.addEventListener("blur", function () {  validateField(this)  })  input.addEventListener("input", function () {  clearError(this)  })  })  }  }  function handleFormSubmit(e) {  e.preventDefault()  const form = e.target  const formData = new FormData(form)  // Client-side validation  if (!validateForm(form)) {  return false  }  // Show loading state  const submitButton = form.querySelector('button[type="submit"]')  const originalText = submitButton.textContent  submitButton.innerHTML = '<span class="spinner"></span>Sending...'  submitButton.disabled = true  // Submit form via AJAX  fetch("process\_contact.php", {  method: "POST",  body: formData,  })  .then((response) => response.json())  .then((data) => {  if (data.success) {  showSuccessMessage("Message sent successfully! Thank you for contacting me.")  form.reset()  } else {  showErrorMessage(data.message || "An error occurred. Please try again.")  }  })  .catch((error) => {  console.error("Error:", error)  showErrorMessage("An error occurred. Please try again.")  })  .finally(() => {  // Reset button state  submitButton.textContent = originalText  submitButton.disabled = false  })  }  function validateForm(form) {  let isValid = true  const requiredFields = form.querySelectorAll("[required]")  requiredFields.forEach((field) => {  if (!validateField(field)) {  isValid = false  }  })  return isValid  }  function validateField(field) {  const value = field.value.trim()  const fieldName = field.name  let isValid = true  let errorMessage = ""  // Clear previous errors  clearError(field)  // Required field validation  if (field.hasAttribute("required") && !value) {  errorMessage = `${getFieldLabel(fieldName)} is required.`  isValid = false  }  // Specific field validations  if (value && isValid) {  switch (fieldName) {  case "name":  if (value.length < 2) {  errorMessage = "Name must be at least 2 characters long."  isValid = false  } else if (!/^[a-zA-Z\s]+$/.test(value)) {  errorMessage = "Name can only contain letters and spaces."  isValid = false  }  break  case "email":  const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/  if (!emailRegex.test(value)) {  errorMessage = "Please enter a valid email address."  isValid = false  }  break  case "phone":  if (value && !/^[+]?[0-9\s\-$$$$]+$/.test(value)) {  errorMessage = "Please enter a valid phone number."  isValid = false  }  break  case "subject":  if (value.length < 5) {  errorMessage = "Subject must be at least 5 characters long."  isValid = false  }  break  case "message":  if (value.length < 10) {  errorMessage = "Message must be at least 10 characters long."  isValid = false  }  break  }  }  if (!isValid) {  showFieldError(field, errorMessage)  }  return isValid  }  function showFieldError(field, message) {  field.classList.add("error")  const errorElement = document.getElementById(`${field.name}-error`)  if (errorElement) {  errorElement.textContent = message  }  }  function clearError(field) {  field.classList.remove("error")  const errorElement = document.getElementById(`${field.name}-error`)  if (errorElement) {  errorElement.textContent = ""  }  }  function getFieldLabel(fieldName) {  const labels = {  name: "Name",  email: "Email",  phone: "Phone",  subject: "Subject",  message: "Message",  }  return labels[fieldName] || fieldName  }  function showSuccessMessage(message) {  const form = document.getElementById("contactForm")  const existingMessage = form.querySelector(".success-message")  if (existingMessage) {  existingMessage.remove()  }  const successDiv = document.createElement("div")  successDiv.className = "success-message"  successDiv.textContent = message  form.insertBefore(successDiv, form.firstChild)  // Remove message after 5 seconds  setTimeout(() => {  successDiv.remove()  }, 5000)  }  function showErrorMessage(message) {  const form = document.getElementById("contactForm")  const existingMessage = form.querySelector(".error-message")  if (existingMessage) {  existingMessage.remove()  }  const errorDiv = document.createElement("div")  errorDiv.className = "error-message"  errorDiv.style.background = "#f8d7da"  errorDiv.style.color = "#721c24"  errorDiv.style.padding = "1rem"  errorDiv.style.borderRadius = "6px"  errorDiv.style.marginBottom = "1rem"  errorDiv.style.border = "1px solid #f5c6cb"  errorDiv.textContent = message  form.insertBefore(errorDiv, form.firstChild)  // Remove message after 5 seconds  setTimeout(() => {  errorDiv.remove()  }, 5000)  }  // Animation Functions  function initializeAnimations() {  // Animate elements when they come into view  const observerOptions = {  threshold: 0.1,  rootMargin: "0px 0px -50px 0px",  }  const observer = new IntersectionObserver((entries) => {  entries.forEach((entry) => {  if (entry.isIntersecting) {  entry.target.classList.add("fade-in")  }  })  }, observerOptions)  // Observe portfolio items  const portfolioItems = document.querySelectorAll(".portfolio-item")  portfolioItems.forEach((item) => {  observer.observe(item)  })  // Observe skill categories  const skillCategories = document.querySelectorAll(".skill-category")  skillCategories.forEach((category) => {  observer.observe(category)  })  }  // Utility Functions  function debounce(func, wait) {  let timeout  return function executedFunction(...args) {  const later = () => {  clearTimeout(timeout)  func(...args)  }  clearTimeout(timeout)  timeout = setTimeout(later, wait)  }  }  // Smooth scrolling for anchor links  function smoothScroll(target) {  const element = document.querySelector(target)  if (element) {  element.scrollIntoView({  behavior: "smooth",  block: "start",  })  }  }  // Profile image upload functionality (optional)  function initializeImageUpload() {  const profileImg = document.getElementById("profile-img")  if (profileImg) {  profileImg.addEventListener("click", () => {  const input = document.createElement("input")  input.type = "file"  input.accept = "image/\*"  input.onchange = (e) => {  const file = e.target.files[0]  if (file) {  const reader = new FileReader()  reader.onload = (e) => {  profileImg.src = e.target.result  }  reader.readAsDataURL(file)  }  }  input.click()  })  }  }  // Initialize additional features  document.addEventListener("DOMContentLoaded", () => {  initializeImageUpload()  })  // Export functions for global access  window.scrollToSection = scrollToSection  window.showSection = showSection |

* process\_contact.php

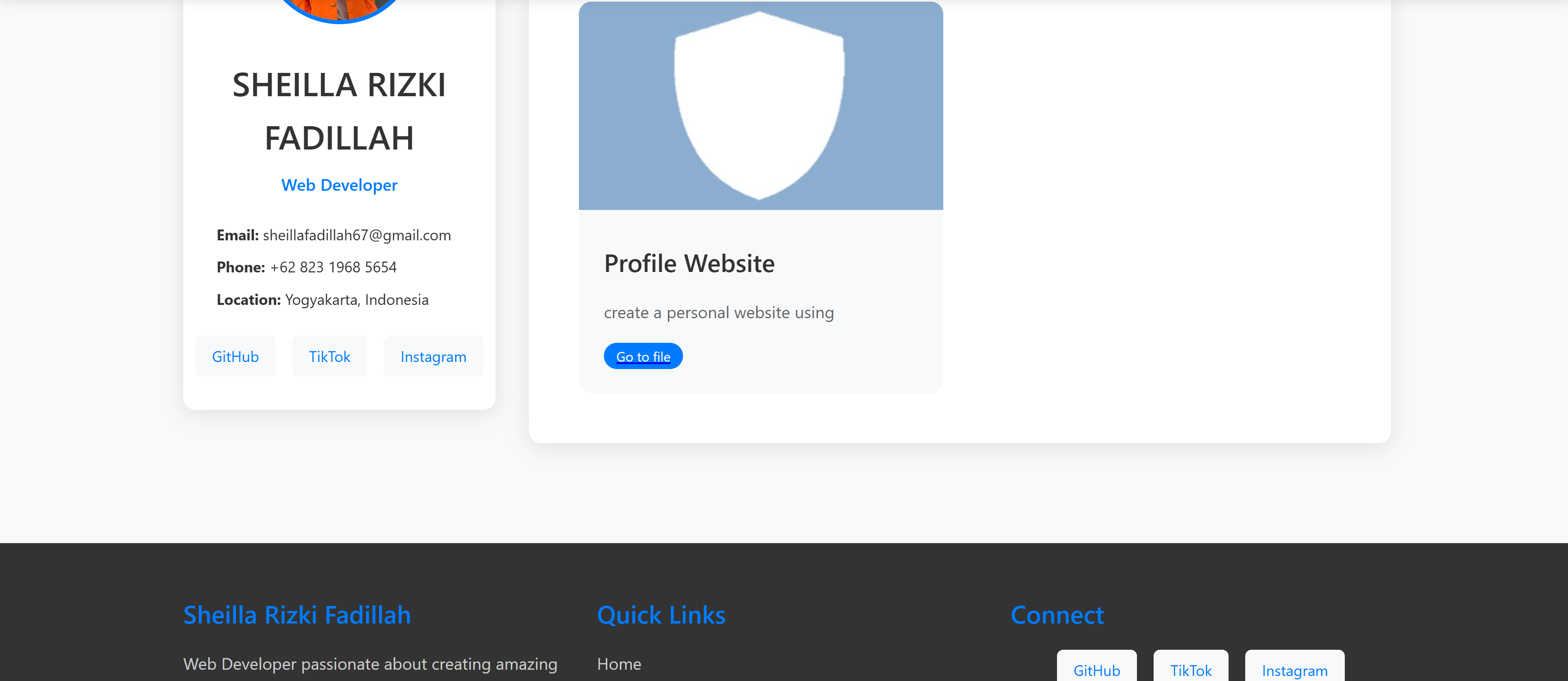
|  |
| --- |
| <?php  // Set content type to JSON  header('Content-Type: application/json');  // Enable error reporting for debugging (remove in production)  error\_reporting(E\_ALL);  ini\_set('display\_errors', 1);  // Function to sanitize input data  function sanitizeInput($data) {  $data = trim($data);  $data = stripslashes($data);  $data = htmlspecialchars($data);  return $data;  }  // Function to validate email  function validateEmail($email) {  return filter\_var($email, FILTER\_VALIDATE\_EMAIL);  }  // Function to validate phone number  function validatePhone($phone) {  return preg\_match('/^[\+]?[0-9\s\-$$$$]+$/', $phone);  }  // Response array  $response = array(  'success' => false,  'message' => '',  'errors' => array()  );  // Check if form was submitted via POST  if ($\_SERVER['REQUEST\_METHOD'] !== 'POST') {  $response['message'] = 'Invalid request method.';  echo json\_encode($response);  exit;  }  // Get and sanitize form data  $name = isset($\_POST['name']) ? sanitizeInput($\_POST['name']) : '';  $email = isset($\_POST['email']) ? sanitizeInput($\_POST['email']) : '';  $phone = isset($\_POST['phone']) ? sanitizeInput($\_POST['phone']) : '';  $subject = isset($\_POST['subject']) ? sanitizeInput($\_POST['subject']) : '';  $message = isset($\_POST['message']) ? sanitizeInput($\_POST['message']) : '';  // Server-side validation  $errors = array();  // Validate name  if (empty($name)) {  $errors['name'] = 'Name is required.';  } elseif (strlen($name) < 2) {  $errors['name'] = 'Name must be at least 2 characters long.';  } elseif (!preg\_match('/^[a-zA-Z\s]+$/', $name)) {  $errors['name'] = 'Name can only contain letters and spaces.';  }  // Validate email  if (empty($email)) {  $errors['email'] = 'Email is required.';  } elseif (!validateEmail($email)) {  $errors['email'] = 'Please enter a valid email address.';  }  // Validate phone (optional field)  if (!empty($phone) && !validatePhone($phone)) {  $errors['phone'] = 'Please enter a valid phone number.';  }  // Validate subject  if (empty($subject)) {  $errors['subject'] = 'Subject is required.';  } elseif (strlen($subject) < 5) {  $errors['subject'] = 'Subject must be at least 5 characters long.';  }  // Validate message  if (empty($message)) {  $errors['message'] = 'Message is required.';  } elseif (strlen($message) < 10) {  $errors['message'] = 'Message must be at least 10 characters long.';  }  // If there are validation errors, return them  if (!empty($errors)) {  $response['message'] = 'Please correct the errors below.';  $response['errors'] = $errors;  echo json\_encode($response);  exit;  }  // Prepare data for saving  $contactData = array(  'timestamp' => date('Y-m-d H:i:s'),  'name' => $name,  'email' => $email,  'phone' => $phone,  'subject' => $subject,  'message' => $message,  'ip\_address' => $\_SERVER['REMOTE\_ADDR'] ?? 'Unknown',  'user\_agent' => $\_SERVER['HTTP\_USER\_AGENT'] ?? 'Unknown'  );  // Format data for text file  $dataString = "=== NEW CONTACT MESSAGE ===" . PHP\_EOL;  $dataString .= "Timestamp: " . $contactData['timestamp'] . PHP\_EOL;  $dataString .= "Name: " . $contactData['name'] . PHP\_EOL;  $dataString .= "Email: " . $contactData['email'] . PHP\_EOL;  $dataString .= "Phone: " . ($contactData['phone'] ?: 'Not provided') . PHP\_EOL;  $dataString .= "Subject: " . $contactData['subject'] . PHP\_EOL;  $dataString .= "Message: " . PHP\_EOL . $contactData['message'] . PHP\_EOL;  $dataString .= "IP Address: " . $contactData['ip\_address'] . PHP\_EOL;  $dataString .= "User Agent: " . $contactData['user\_agent'] . PHP\_EOL;  $dataString .= "=========================" . PHP\_EOL . PHP\_EOL;  // Save to text file  $filename = 'contact\_messages.txt';  $result = file\_put\_contents($filename, $dataString, FILE\_APPEND | LOCK\_EX);  if ($result === false) {  $response['message'] = 'Failed to save message. Please try again.';  echo json\_encode($response);  exit;  }  // Success response  $response['success'] = true;  $response['message'] = 'Thank you for your message! I will get back to you soon.';  echo json\_encode($response);  ?> |

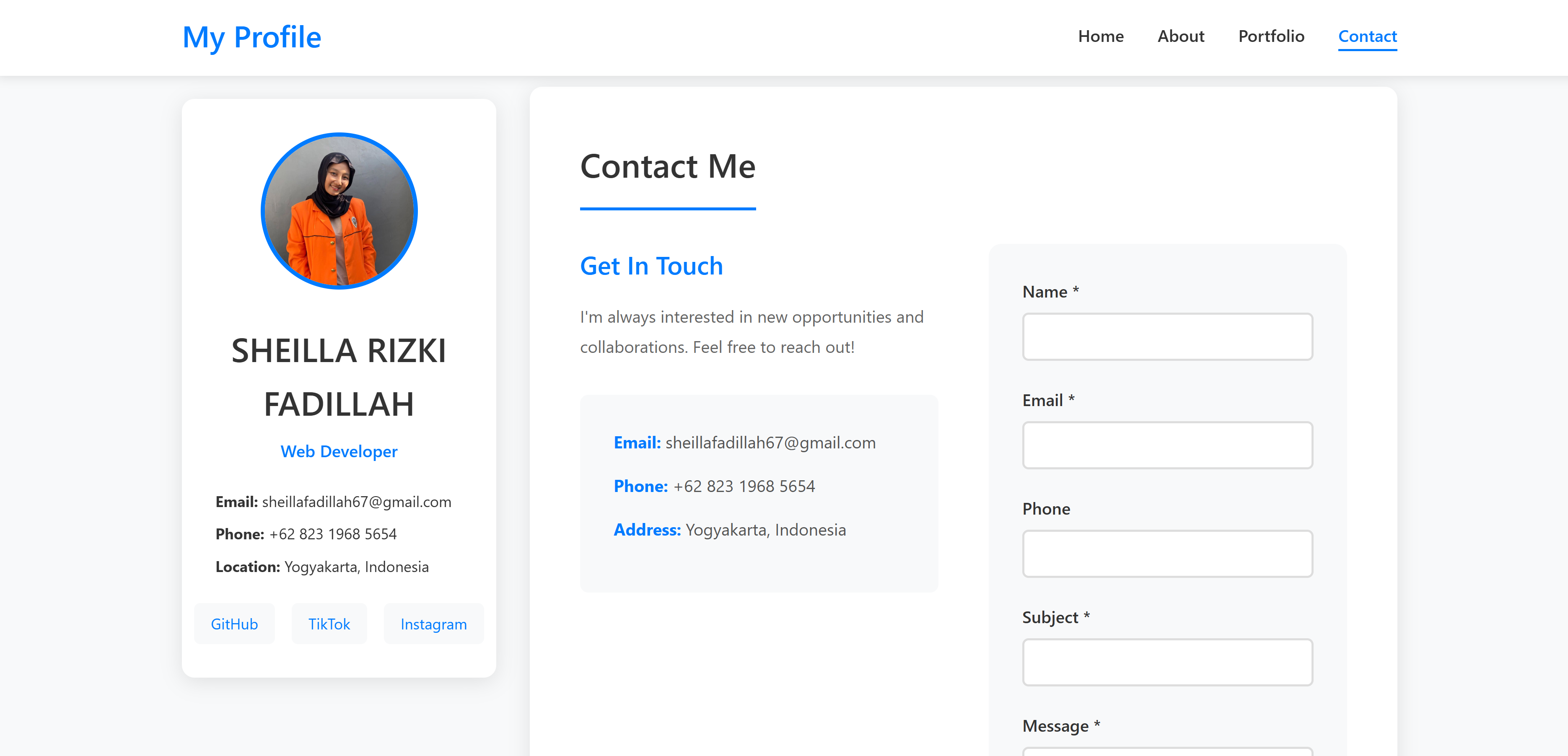
#### H. DESKRIPSI TAMPILAN

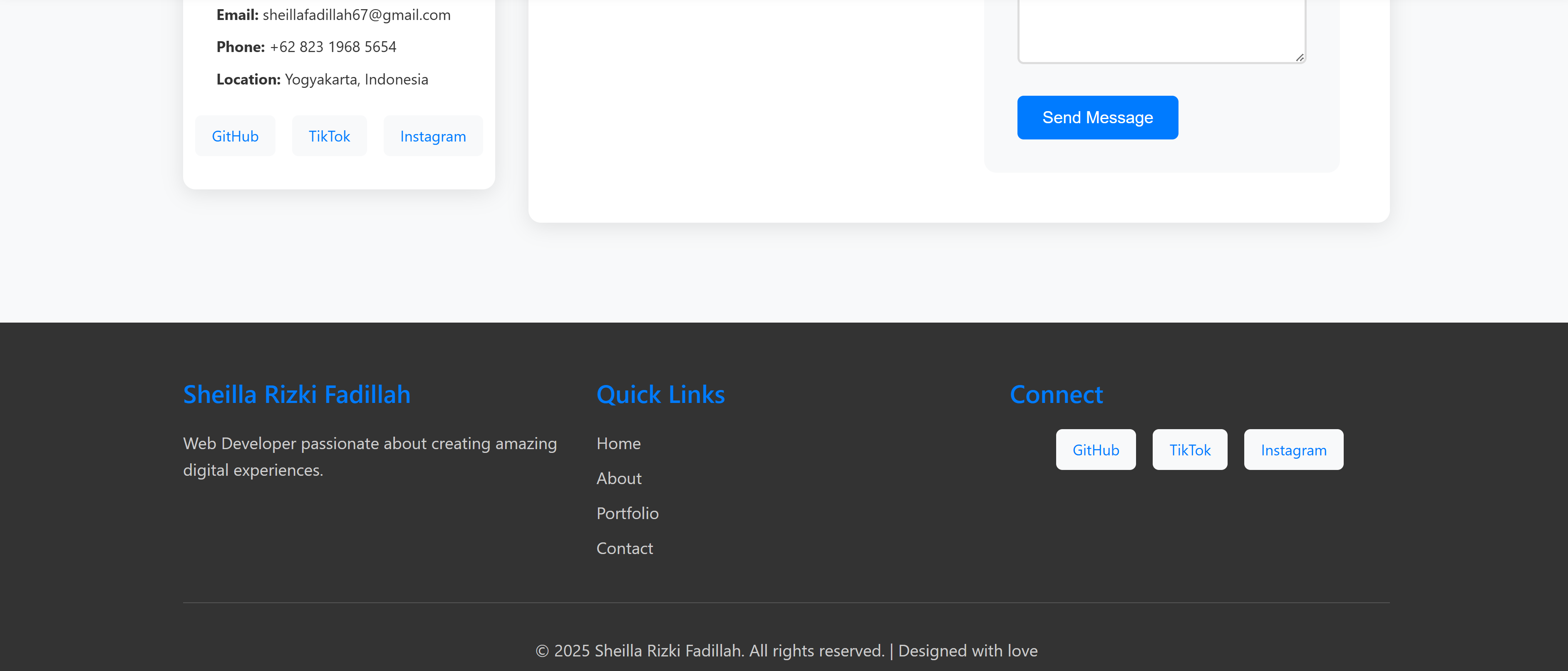














* Kontras (Contras)

Kontras berfungsi untuk menyoroti elemen-elemen penting agar lebih mudah dikenali. Pada desain ini, misalnya, nama "My profile" pada header diberi ukuran font lebih besar dan warna yang mencolok untuk menarik perhatian pengunjung. Selain itu, perbedaan warna latar pada header, main, dan footer membantu memperjelas struktur halaman dan memandu mata pengguna.

* Repetition (Pengulangan)

Pengulangan diterapkan untuk menjaga konsistensi desain, seperti penggunaan jenis font yang seragam untuk judul dan paragraf, serta warna yang sama untuk tautan navigasi. Ini membuat halaman terasa lebih menyatu dan mudah diingat, sekaligus memperkuat identitas visual.

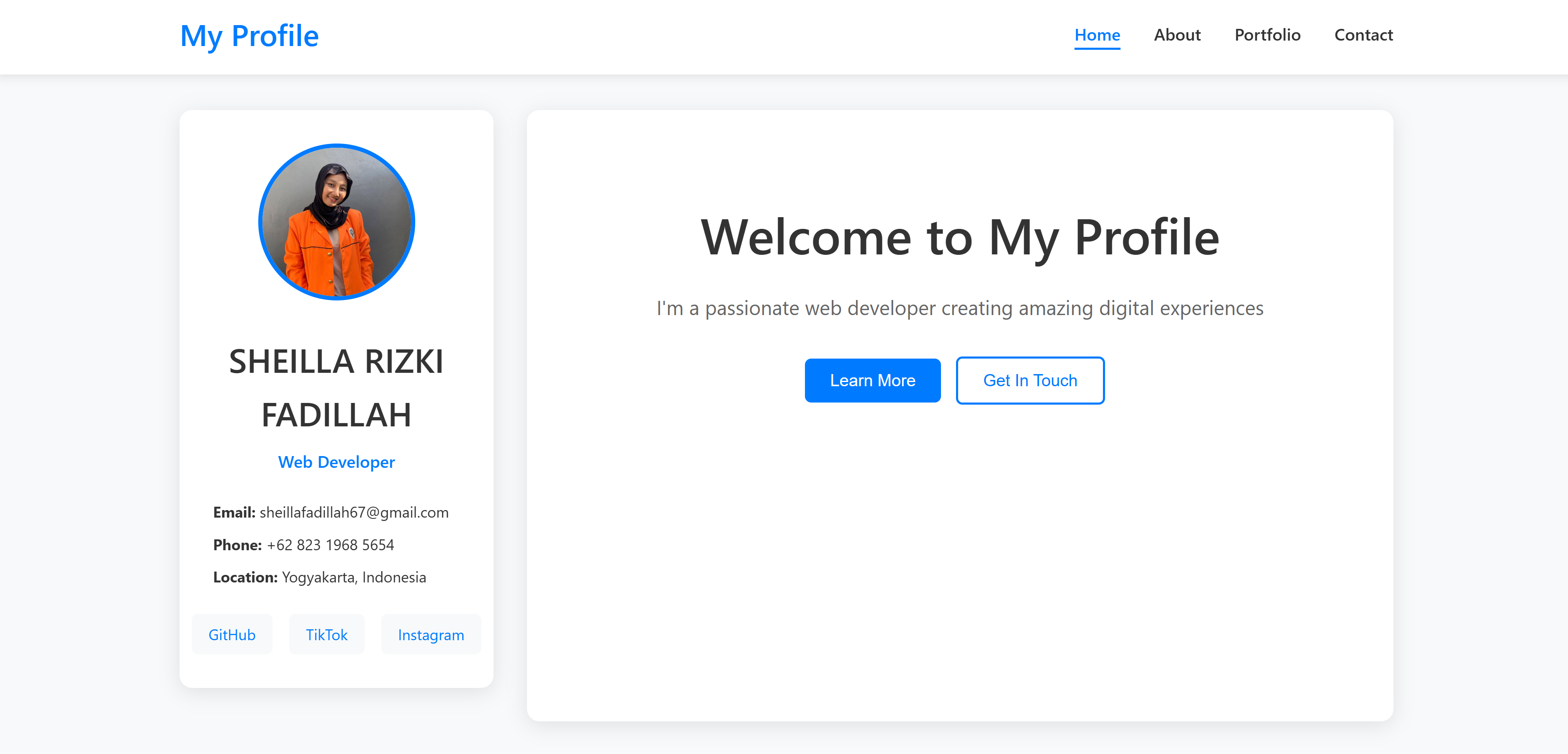
* Alignment (Penjajaran)

Setiap elemen pada halaman ini dirancang untuk sejajar dengan rapi, memastikan teks, gambar, dan navigasi berada dalam garis yang jelas. Dengan menggunakan teknik seperti flexbox atau grid, desain ini menciptakan kesan teratur dan profesional, memudahkan pengunjung dalam membaca dan menavigasi konten.

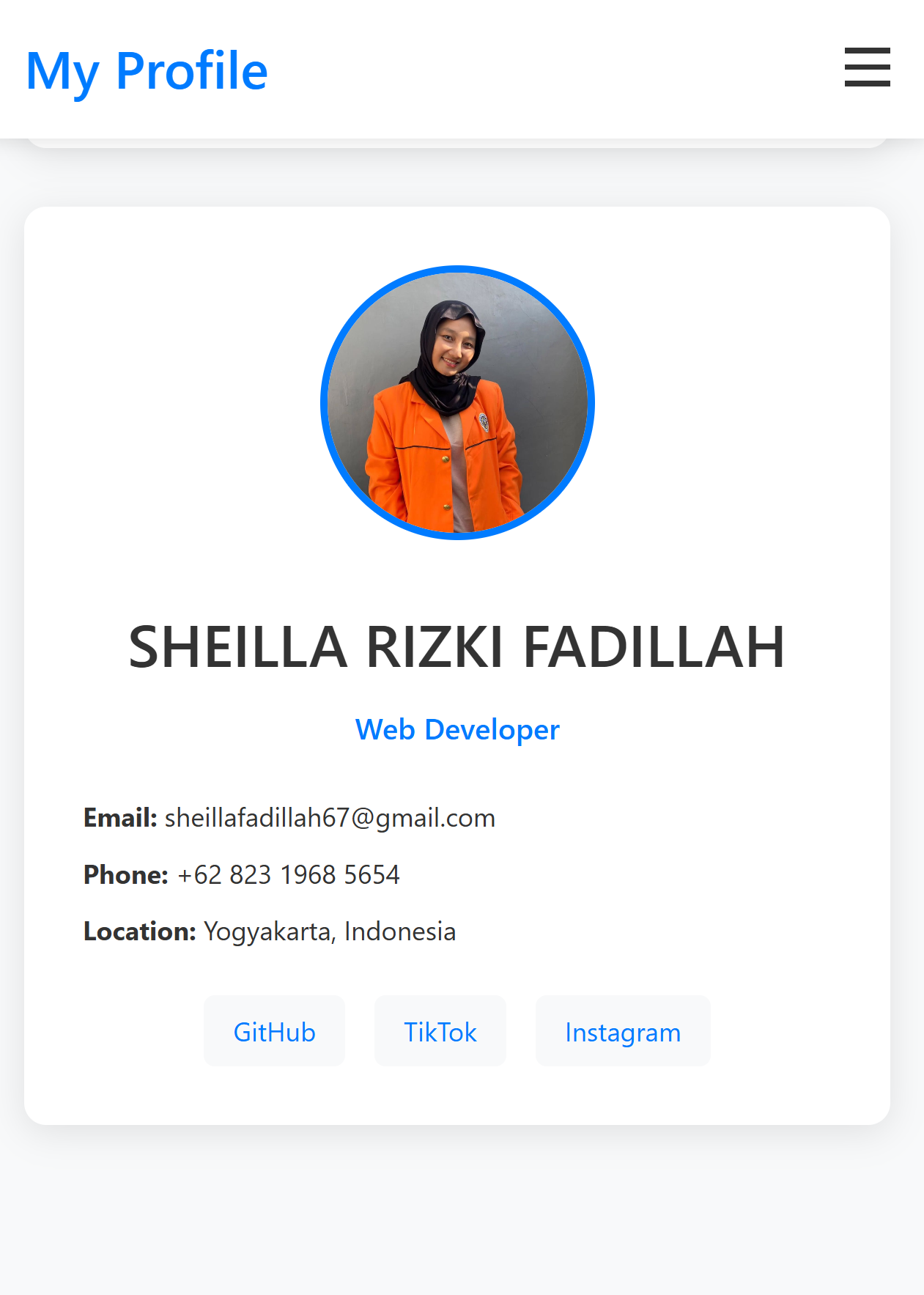
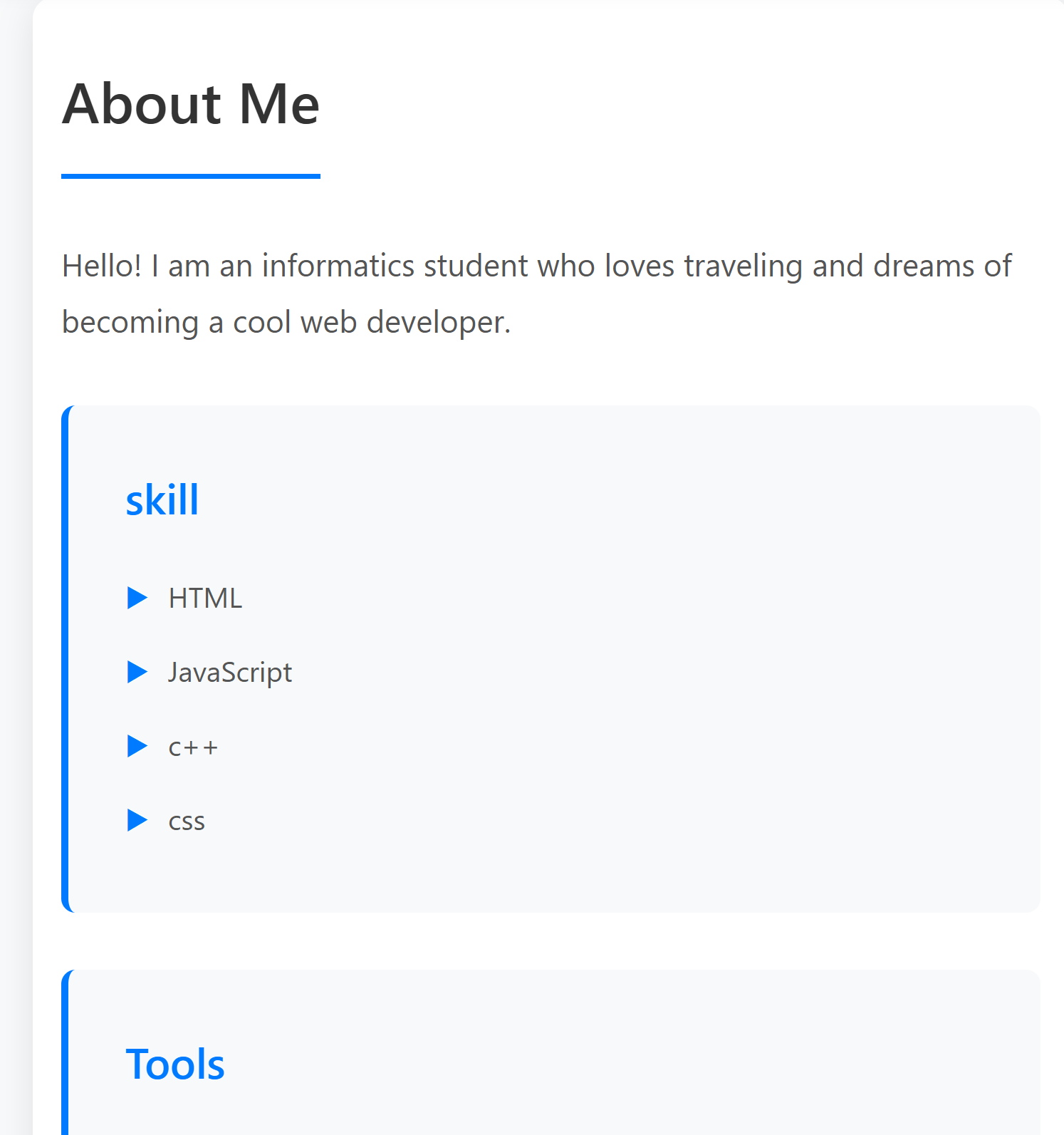
* Proximity (Kedekatan)

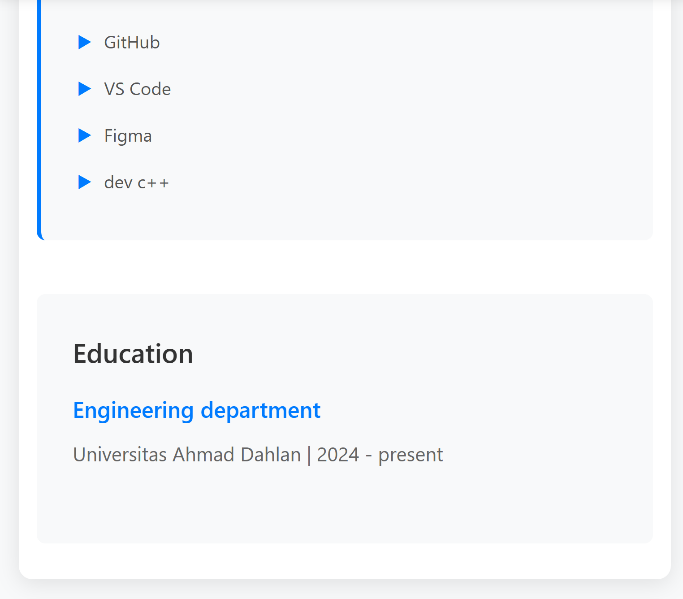
Elemen yang saling berkaitan dikelompokkan untuk memperjelas hubungan antar konten. Misalnya, menu navigasi dikumpulkan dalam header, informasi pribadi ditempatkan dalam satu blok di main, dan hak cipta ditempatkan secara terpisah di footer. Ini membuat struktur halaman lebih intuitif dan mudah dipahami.

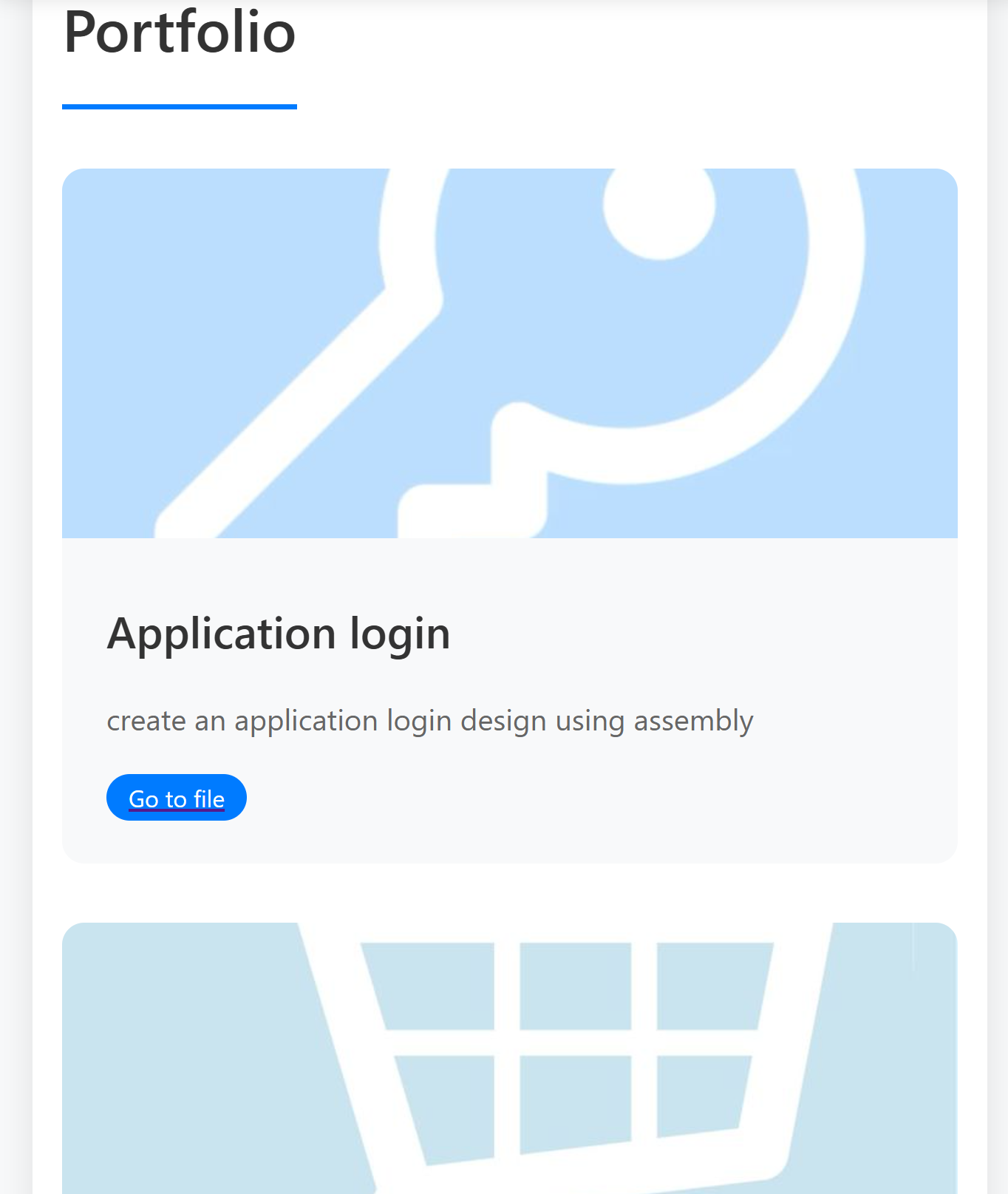
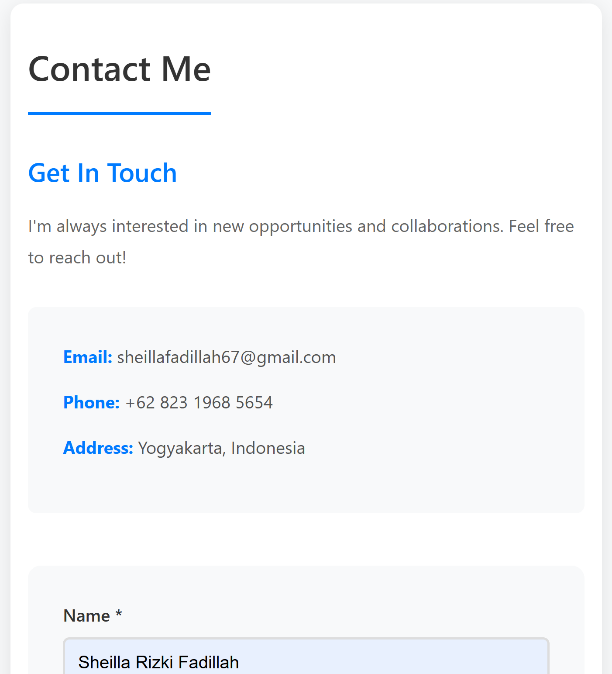
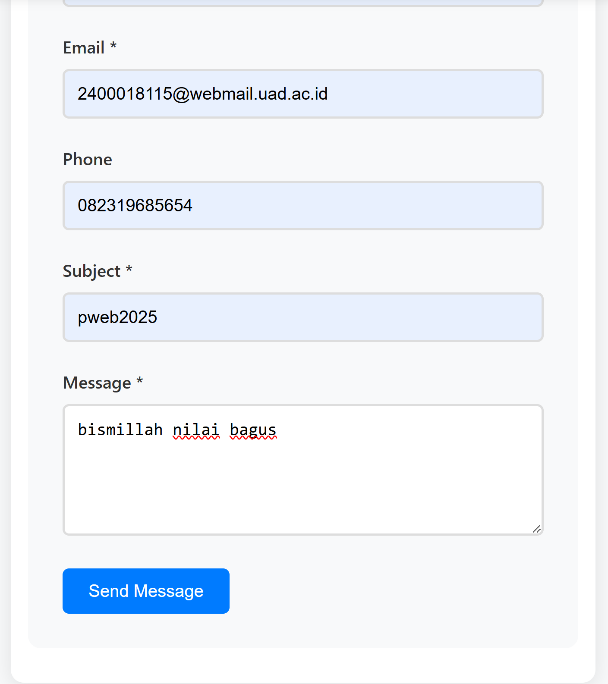
#### I. TAMPILAN WEB

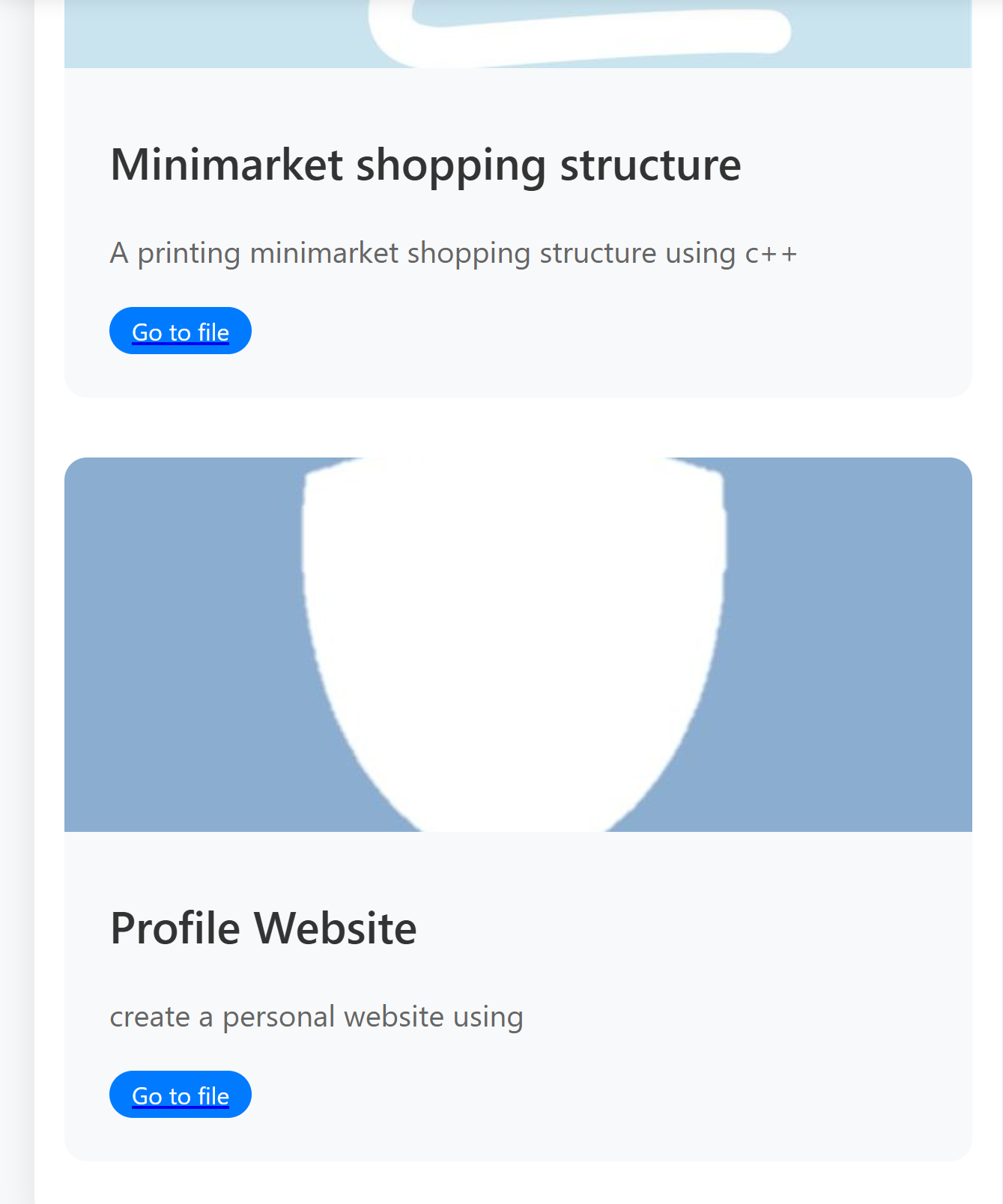


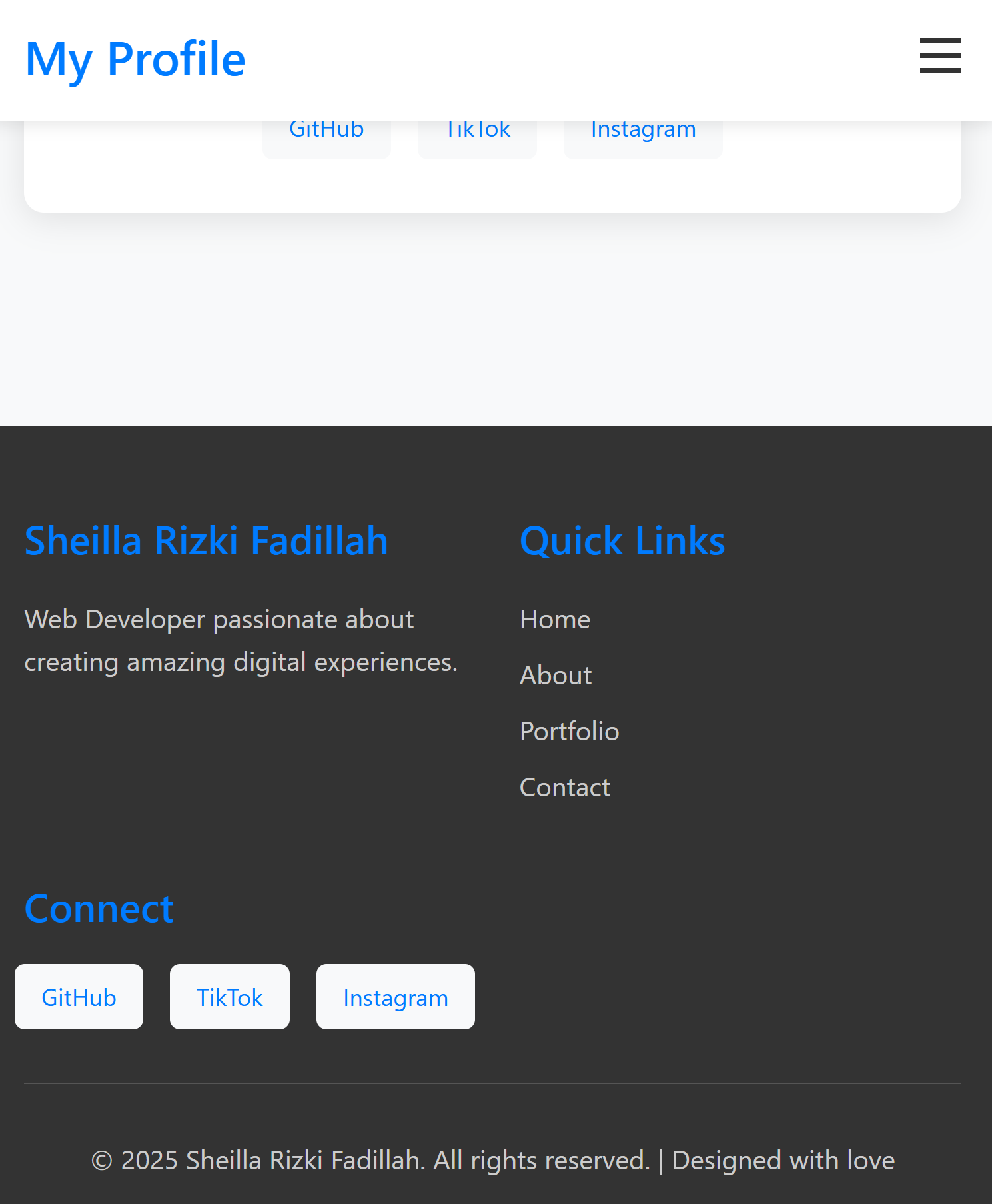
#### J. TAMPILAN RESPONSIVE LAYOUT

** **

****

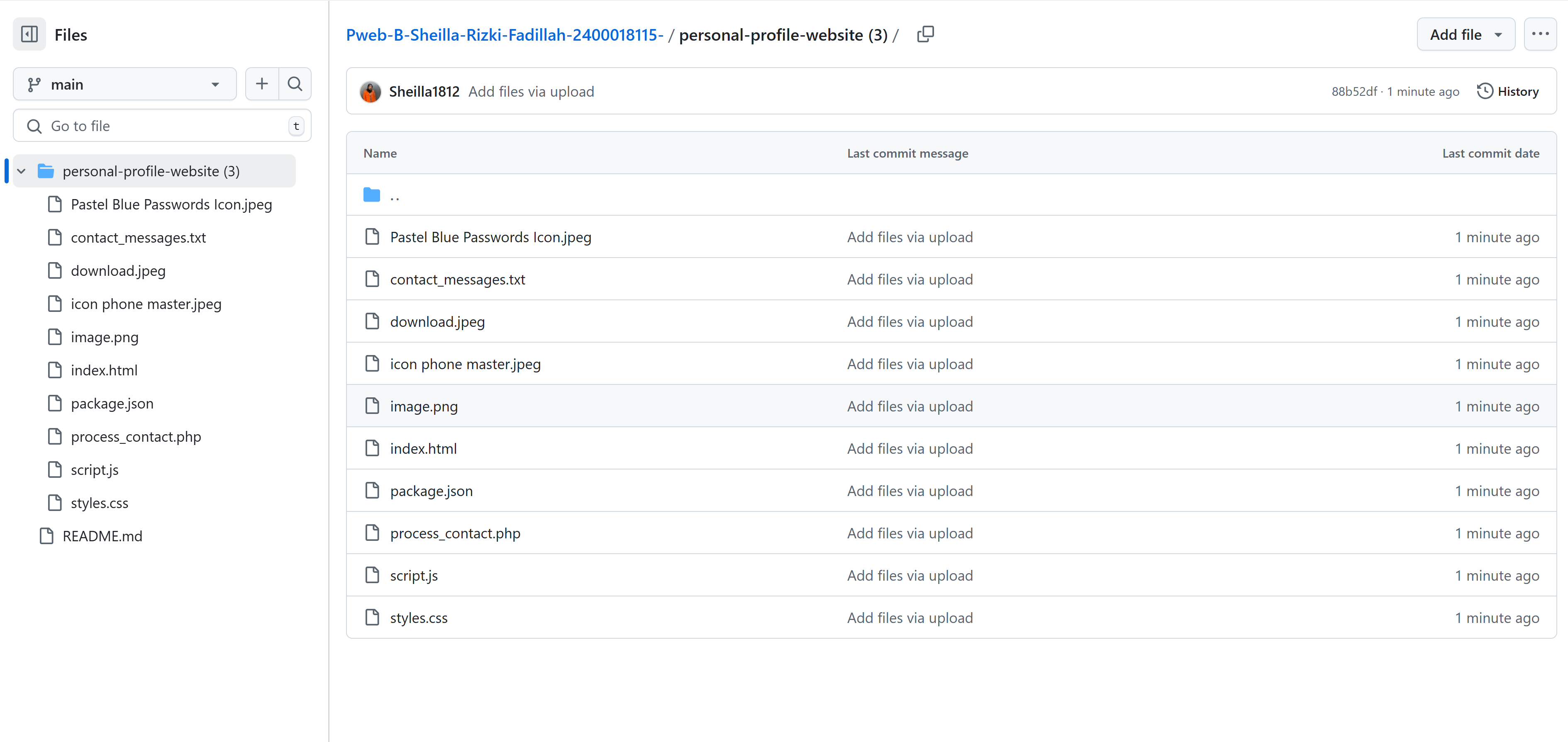
****

****

****

Website ini dibuat dengan pendekatan **responsive layout**, memungkinkan tampilan menyesuaikan secara otomatis dengan berbagai ukuran layar, seperti desktop, tablet, dan smartphone. Dengan memanfaatkan **media query** pada CSS, elemen-elemen seperti header, navigasi, dan konten utama dirancang agar tetap rapi dan mudah diakses pada semua perangkat, memastikan pengalaman pengguna yang konsisten.

**K. ESKRPSI SCHRENSHOOT GITHUB**

****

**L. ANALISIS PENGERJAAN TUGAS PROJEK**

Proyek ini saya selesaikan dalam waktu sekitar satu minggu. Selama proses pembuatannya, saya banyak belajar tentang dasar-dasar pengembangan web, termasuk penggunaan HTML, CSS, JavaScript dan PHP. Untuk panduan dan referensi, saya sering merujuk ke sumber-sumber online seperti W3Schools. Proyek ini tidak membutuhkan biaya tambahan, karena saya menggunakan perangkat dan software gratis yang sudah tersedia, seperti visual studio code untuk pengujian. Namun, saya menghadapi beberapa tantangan, terutama dalam memahami struktur dan sintaks dari HTML, CSS, dan JavaScript dan PHP, yang masih cukup baru bagi saya. Meski begitu, saya merasa bangga dengan hasil akhirnya, karena proyek ini memberi saya pengalaman berharga dalam merancang halaman web dari awal, termasuk pengelolaan tata letak, penambahan elemen visual, dan interaktivitas dasar.