AI ASSISTED CODING LAB

ASSIGNMENT 14.4

ENROLLMENT NO :2503A51L25

BATCH NO: 19

NAME: SRIRAMOJU AJAY

TASK1

TASK1 DESCRIPTION:- **AI-Assisted Portfolio Website**
Scenario:
A student wants to showcase their projects, skills, and contact details in a portfolio website. Instead of writing all code manually, they want to speed up the process using GitHub Copilot.

PROMPT:- Create a simple and responsive student portfolio website with HTML and CSS. Include a navigation menu (Home, About, Projects, Contact), a projects grid with hover effects, and a contact form.

CODE:-

HTML:-

style.html > html > body > section#contact > form

```html
<!DOCTYPE html>
<html lang="en">
<head>···
</head>
<body>
    <header>
        <nav>
            <ul>
                <li><a href="#home">Home</a></li>
                <li><a href="#about">About</a></li>
                <li><a href="#projects">Projects</a></li>
                <li><a href="#contact">Contact</a></li>
            </ul>
        </nav>
    </header>
    <section id="home">
        <h1>Welcome to My Portfolio</h1>
        <p>Hello! I'm a student passionate about web development.</p>
    </section>
    <section id="about">
        <h2>About Me</h2>
        <p>I am learning to build modern, responsive websites and love to solve problems with code.</p>
    </section>
    <section id="projects">
        <h2>Projects</h2>
        <div class="projects-grid">
            <div class="project-card">···
            </div>
            <div class="project-card">···
            </div>
            <div class="project-card">···
            </div>
        </div>
    </section>
    <section id="contact">
        <h2>Contact</h2>
        <form>0
            <input type="text" placeholder="Your Name" required>
            <input type="email" placeholder="Your Email" required>
            <textarea placeholder="Your Message" required></textarea>
            <button type="submit">Send</button>
        </form>
    </section>
</body>
</html>
```

CSS:-

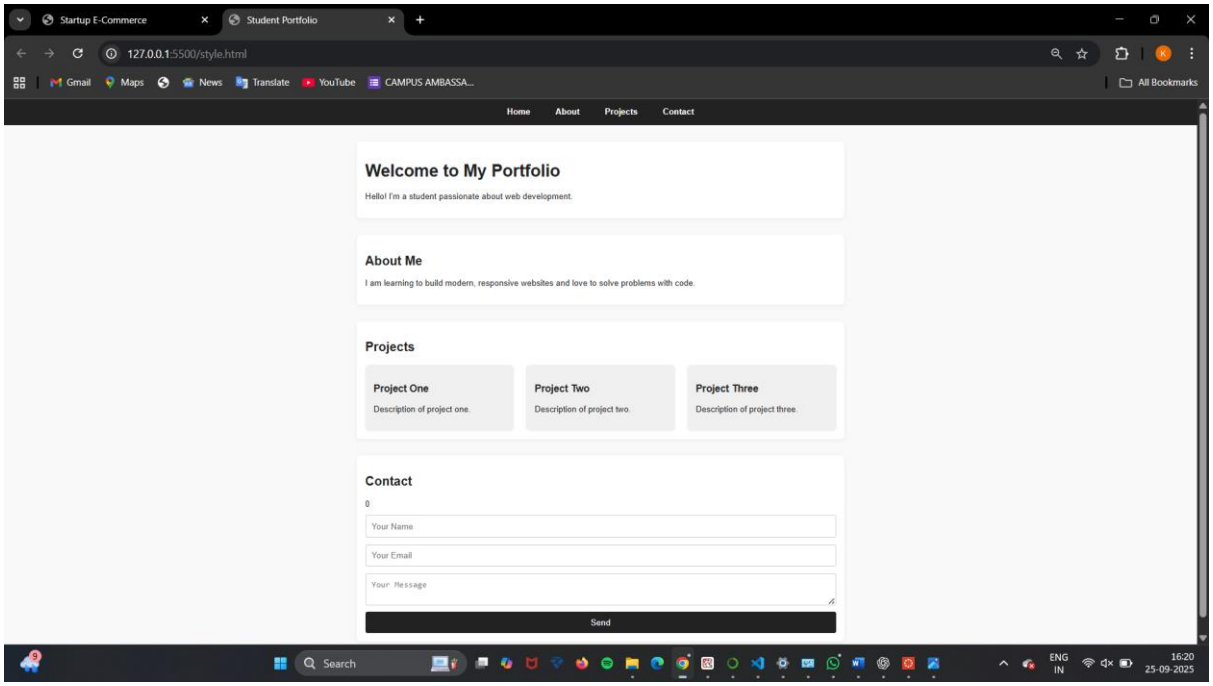
style.css > section

```css
body {
    font-family: Arial, sans-serif;
    margin: 0;
    padding: 0;
    background: #f9f9f9;
    color: #222;
}

header {
    background: #222;
    color: #fff;
    padding: 1rem 0;
}

nav ul {
    display: flex;
    justify-content: center;
    list-style: none;
    margin: 0;
    padding: 0;
}

nav ul li {
    margin: 0 1.5rem;
}

nav ul li a {
    color: #fff;
    text-decoration: none;
    font-weight: bold;
}

section {
    max-width: 900px;
    margin: 2rem auto;
    padding: 1rem;
    background: #fff;
    border-radius: 8px;
    box-shadow: 0 2px 8px rgba(0,0,0,0.05);
}

.projects-grid {
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(220px, 1fr));
    gap: 1.5rem;
```

style.css > section

```
48
49  .project-card {
50      background: #f0f0f0;
51      padding: 1rem;
52      border-radius: 8px;
53      box-shadow: 0 1px 4px rgba(0,0,0,0.07);
54      transition: transform 0.2s, box-shadow 0.2s;
55  }
56
57  .project-card:hover {
58      transform: translateY(-8px) scale(1.03);
59      box-shadow: 0 6px 18px rgba(0,0,0,0.15);
60      background: #e0e7ff;
61  }
62
63  form {
64      display: flex;
65      flex-direction: column;
66      gap: 0.8rem;
67  }
68
69  input, textarea {
70      padding: 0.7rem;
71      border: 1px solid #ccc;
72      border-radius: 4px;
73      font-size: 1rem;
74  }
75
76  button {···
85  }
86
87  button:hover {
88      background: #444;
89  }
90
91  @media (max-width: 600px) {
92      nav ul {
93          flex-direction: column;
94          align-items: center;
95      }
96      section {
97          margin: 1rem;
98          padding: 0.5rem;
99      }
100 }
```

OUTPUT:-



OBSERVATION:- I observed that using AI tools like GitHub Copilot significantly reduces the time required to create a portfolio website. The AI suggested responsive HTML and CSS code for navigation, project grids, and contact forms. This task demonstrated how AI can speed up the development process while still requiring human review for customization and styling.

TASK2

TASK2 DESCRIPTION:- AI-Generated Restaurant Landing Page

Scenario:

A local restaurant needs a simple landing page with a navigation bar, menu highlights, and an image gallery. The developer wants to quickly generate it using AI assistance.

PROMPT:- Generate a responsive restaurant website with navigation, menu cards, image slider, contact info, and hover effects using HTML, CSS, and JavaScript.

CODE:-

HTML:-

style.html > html > body > section#contact > form

```
<!DOCTYPE html>
<html lang="en">
<head>···
</head>
<body>
    <header>
        <nav>
            <ul>
                <li><a href="#home">Home</a></li>
                <li><a href="#about">About</a></li>
                <li><a href="#projects">Projects</a></li>
                <li><a href="#contact">Contact</a></li>
            </ul>
        </nav>
    </header>
    <section id="home">
        <h1>Welcome to My Portfolio</h1>
        <p>Hello! I'm a student passionate about web development.</p>
    </section>
    <section id="about">
        <h2>About Me</h2>
        <p>I am learning to build modern, responsive websites and love to solve problems with code.</p>
    </section>
    <section id="projects">
        <h2>Projects</h2>
        <div class="projects-grid">
            <div class="project-card">···
            </div>
            <div class="project-card">···
            </div>
            <div class="project-card">···
            </div>
        </div>
    </section>
    <section id="contact">
        <h2>Contact</h2>
        <form>0
            <input type="text" placeholder="Your Name" required>
            <input type="email" placeholder="Your Email" required>
            <textarea placeholder="Your Message" required></textarea>
            <button type="submit">Send</button>
        </form>
    </section>
</body>
</html>
```

CSS:-

restaurant.css > #prev

```css
body {
    font-family: 'Segoe UI', Arial, sans-serif;
    margin: 0;
    background: #f8f8f8;
    color: #222;
}

header {
    background: #b71c1c;
    color: #fff;
    padding: 1rem 0;
}

nav ul {
    display: flex;
    justify-content: center;
    list-style: none;
    margin: 0;
    padding: 0;
}

nav ul li {
    margin: 0 1.5rem;
}

nav ul li a {
    color: #fff;
    text-decoration: none;
    font-weight: bold;
}

section {
    max-width: 900px;
    margin: 2rem auto;
    padding: 1rem;
    background: #fff;
    border-radius: 8px;
    box-shadow: 0 2px 8px rgba(0,0,0,0.05);
}

.menu-cards {
    display: flex;
    gap: 1.5rem;
    flex-wrap: wrap;
    justify-content: center;
```

restaurant.css > #prev

```
73    .slider {
77        text-align: center;
78    }
79
80    .slider img {
81        width: 100%;
82        border-radius: 8px;
83        box-shadow: 0 2px 8px rgba(0,0,0,0.08);
84    }
85
86    .slider-btn {
87        position: absolute;
88        top: 50%;
89        transform: translateY(-50%);
90        background: #b71c1c;
91        color: #fff;
92        border: none;
93        font-size: 2rem;
94        padding: 0.3rem 0.8rem;
95        border-radius: 50%;
96        cursor: pointer;
97        opacity: 0.8;
98        transition: background 0.2s;
99    }
100
101   #prev { left: -40px; }
102   #next { right: -40px; }
103
104   .slider-btn:hover {
105       background: #d32f2f;
106   }
107
108   @media (max-width: 700px) {
109       .menu-cards {
110           flex-direction: column;
111           align-items: center;
112       }
113       .slider {
114           max-width: 95vw;
115       }
116       #prev, #next {
117           left: 10px;
118           right: 10px;
119       }
120   }
```

OBSERVATION:- I learned how AI can assist in generating a restaurant landing page with a navigation bar, menu highlights, and an image gallery. The code produced by AI was responsive and included hover effects. However, I observed that while AI provided the basic structure, additional manual refinements were needed to match the design expectations.

TASK3

TASK3 DESCRIPTION:- **AI-Powered Event Registration Form**

Scenario:

SR University is hosting a tech fest. They need a web-based registration form for students. The form must validate user input in real-time.

PROMPT:- Create a restaurant landing page with a navigation bar, menu highlights styled as cards, and a JavaScript image slider for the gallery section.

CODE:-

HTML:-

registration.html > html > body

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>SRU Tech Fest Registration</title>
    <link rel="stylesheet" href="registration.css">
</head>
<body>
    <div class="form-container">
        <h2>Tech Fest Registration</h2>
        <form id="regForm" autocomplete="off">···
        </form>
        <div id="successMsg"></div>
    </div>
    <script>
        // Real-time validation
        const name = document.getElementById('name');
        const email = document.getElementById('email');
        const phone = document.getElementById('phone');
        const dept = document.getElementById('dept');
        const eventSel = document.getElementById('event');
        const form = document.getElementById('regForm');
        const successMsg = document.getElementById('successMsg');

        function validateName() {···

        function validateEmail() {···

        function validatePhone() {···

        function validateDept() {···

        function validateEvent() {···

        name.addEventListener('input', validateName);
        email.addEventListener('input', validateEmail);
        phone.addEventListener('input', validatePhone);
        dept.addEventListener('change', validateDept);
        eventSel.addEventListener('change', validateEvent);

        form.addEventListener('submit', function(e) {···
    </script>
</body>
</html>
```

CSS:-

registration.css > #successMsg

```css
body {
    background: #f0f4f8;
    font-family: 'Segoe UI', Arial, sans-serif;
    margin: 0;
}

.form-container {
    background: #fff;
    max-width: 400px;
    margin: 3rem auto;
    padding: 2rem 2.5rem;
    border-radius: 10px;
    box-shadow: 0 4px 24px rgba(0,0,0,0.08);
}

h2 {…
}

form {…
}

label {
    font-weight: 500;
    margin-bottom: 0.2rem;
}

input, select {…
}

input:focus, select:focus {
    border: 1.5px solid #1976d2;
}

button {…
}

button:hover {…
}

.error {…
}

#successMsg {…
}
```

OBSERVATION:- This task highlighted the use of AI in generating a registration form with real-time input validation. I observed that AI-generated JavaScript handled input checks effectively, reducing coding effort. The task also showed that AI assistance can improve form usability by suggesting features like error messages and validation prompts, which enhance user experience.

TASK4

TASK4 DESCRIPTION:- **AI-Assisted E-Commerce Product Page**

Scenario:

A startup wants a basic e-commerce product page to display products with prices and an “Add to Cart” button.

PROMPT:- Generate a restaurant landing page with a navigation bar, menu highlights as styled cards, and a JavaScript image slider for the gallery section.

CODE:-

HTML:-

product.html > html > body > main > div.product-grid

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Startup E-Commerce</title>
    <link rel="stylesheet" href="product.css">
</head>
<body>
    <header>
        <h1>Shop Products</h1>
        <div class="cart-counter">
            Cart: <span id="cartCount">0</span>
        </div>
    </header>
    <main>
        <div class="product-grid">···
        </div>
    </main>
    <script>
        // Add to Cart functionality with cart counter
        let cartCount = 0;
        const cartCountSpan = document.getElementById('cartCount');
        const addCartBtns = document.querySelectorAll('.add-cart-btn');

        addCartBtns.forEach(btn => {
            btn.addEventListener('click', () => {
                cartCount++;
                cartCountSpan.textContent = cartCount;
            });
        });
    </script>
</body>
</html>
```

CSS:-

```
# product.css > {} @media (max-width: 700px)
  1 > body {···
  6   }
  7
  8 > header {···
 18   }
 19
 20 > .cart-counter {···
 28   }
 29
 30 > main {···
 34   }
 35
 36 > .product-grid {···
 40   }
 41
 42 > .product-card {···
 49   }
 50
 51 > .product-card:hover {···
 54   }
 55
 56 > .product-card img {···
 61   }
 62
 63 > .product-card h3 {···
 67   }
 68
 69 > .product-card p {···
 74   }
 75
 76 > .add-cart-btn {···
 86   }
 87
 88   .add-cart-btn:hover {
 89       background: #1565c0;
 90   }
 91
 92   @media (max-width: 700px) {
 93 >     header {···
 97       }
 98 >     main {···
100       }
101 >     .product-grid {···
103       }
104   }
```

OBSERVATION:- I observed that AI-generated code provided a quick way to design a product display page with prices and “Add to Cart” functionality. The structure was simple yet functional. However, I noticed that while AI generated the HTML and CSS effectively, integrating dynamic cart functionality would still require additional logic and backend support.