# MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

**SANTOSH, TANGAIL-1902**


## Department of Information and Communication Technology

# Lab Report

**Lab report No:** 06

**Lab report on:** Setup and Practical Implementation of GitHub Copilot and Gemini CLI in my Personal Portfolio Project.

**Course title:** Internet and Web Programming Lab

**Course Code**: ICT-4108

| **Submitted By -** | **Submitted To-** |
|---|---|
| Md. Shahidul Islam<br>ID : IT-21024<br>4th Year 1st Semester<br>Session : 2020-2021<br>Dept. of ICT, MBSTU | Md. Anowar Kabir<br>Lecturer<br>Dept. of ICT,MBSTU |

Date of Performance: Date of Submission: 31.07.2025

**Lab Report No:** 06

**Lab report Title:** Setup and Practical Implementation of GitHub Copilot and Gemini CLI in my Personal Portfolio Project.

## Objective:

To explore, configure, and utilize GitHub Copilot and Gemini CLI in the development of a personal portfolio website, aiming to enhance coding productivity and integrate AI assistance in real-world projects.

## Tools and Technologies Used:

- GitHub Copilot (VS Code extension)
- Gemini CLI (Google AI Studio)
- Visual Studio Code (VS Code)
- Node.js / React / HTML/CSS/JS (for the portfolio)
- Git & GitHub

## Theory:

### GitHub Copilot:

GitHub Copilot is an AI-powered coding assistant developed by GitHub and OpenAI. It suggests whole lines or blocks of code as you type, based on context.

### Gemini CLI:

Gemini CLI allows developers to interact with Google's Gemini AI models from the command line. It is useful for querying documentation, generating code, or getting suggestions directly from the terminal.

**Personal Portfolio Website:**

A personal portfolio website showcases a developer's projects, skills, and experiences. It is often built using HTML, CSS, JS, or frameworks like React.

## Procedure:

### Step 1: Setting Up GitHub Copilot

- Open VS Code.
- Go to Extensions and search for "GitHub Copilot".
- Click Install.
- Sign in with GitHub and activate your Copilot subscription/trial.
- Open a code file, start typing, and Copilot will begin suggesting code completions.

### Step 2: Installing Gemini CLI

- Ensure Node.js and npm are installed.
- Install the CLI using: npm install -g @google/gemini-cli
- Authenticate with Google (you'll need to enable the Gemini API access).
- Test the installation using: gemini whoami

### Step 3: Creating the Portfolio Project

- Initialize the project: npx create-react-app my-portfolio , cd my-portfolio
- Use GitHub Copilot while building:
- Autocomplete layout components
- Generate CSS styles
- Suggest animations
- Use Gemini CLI for:
- Asking for responsive design tips: gemini "How to make a navbar responsive in React?"
- Generating SEO-friendly content

## Implementation: What I Did

1. Created sections: About Me, Projects, Contact Form using Copilot code suggestions.
2. Styled components with Copilot's inline CSS hints.
3. Used Gemini CLI to generate text for project descriptions and SEO titles.
4. Asked Gemini CLI to convert JavaScript to TypeScript for improved type safety.
5. Deployed the project using GitHub Pages.

## Result:

| Features | Helped by Copilot | Helped by Gemini CLI |
|---|---|---|
| Navbar & Layout Code | Yes | No |
| CSS Animations | Yes | No |
| Project Descriptions | No | Yes |
| Debugging Error Messages | Yes | Yes |
| Responsive Design Ideas | No | Yes |

## Conclusion:

The integration of GitHub Copilot and Gemini CLI brought a powerful AI-assisted development environment. GitHub Copilot streamlined the coding process, while Gemini CLI offered contextual assistance and guidance via prompts. Together, they made the development of a personal portfolio both faster and smarter.

**References**:


- https://docs.github.com/en/copilot
- https://ai.google.dev/gemini/gemini-cli
- https://reactjs.org
- https://developer.mozilla.org/en-US/