[SCHOOL NAME] | Department of Career & Technical Education

Program: Computer Technology / Programming & Software Development

Course Code: [Insert Code, e.g., TE81]

DOCUMENT: ENVIRONMENT SETUP GUIDE	**VERSION: 2025.1**
PROJECT: Web Application MVP	**LAB TYPE:** Development Studio

---

OBJECTIVE

To establish a standardized local development environment to ensure code consistency, portability, and alignment with industry-standard backend workflows.

STEP 1: CODE EDITOR CONFIGURATION (IDE)

Industry Standard: Visual Studio Code (VS Code)

1. **Installation:** Ensure VS Code is installed from the official Software Center.
2. **Required Extensions:** Open the Extensions Marketplace (Ctrl+Shift+X) and install:
    - **Prettier:** For standardized code formatting.
    - **PHP Intelephense:** For logic and syntax highlighting.
    - **Live Server:** For real-time frontend testing.
    - **Docker:** (If using containerized environments).

STEP 2: VERSION CONTROL SETUP (GIT)

Requirement: Every student must have an active GitHub account and local Git configuration.

1. **Identity Configuration:** Open the terminal and execute:
    - git config --global user.name "Your Full Name"
    - git config --global user.email "your.email@example.com"
2. **SSH/Token Authentication:** Follow the instructor’s guide to link your local machine to your GitHub account to enable secure push/pull operations.

STEP 3: BACKEND RUNTIME (LOCAL SERVER)

Depending on the assigned team track, configure one of the following:

Option A: Standard Stack (XAMPP/MAMP)

- Launch the control panel and start **Apache** and **MySQL**.
- Root Directory: Move your project files to the htdocs or www folder.

Option B: Containerized Environment (Docker - Advanced Track)

- *Note: This mirrors professional enterprise deployments.*
- Navigate to your project folder in the terminal and run:
    - docker-compose up -d
- Verify that your PHP container and MySQL instance are running via the Docker Dashboard.

STEP 4: DIRECTORY ARCHITECTURE

All projects must follow this standardized file structure for successful MVP verification:

Plaintext

```
/your-project-name
│
├── /assets          # Images, CSS, and Client-side JS
├── /config          # Database connection and environment variables
├── /sql             # Database schema and seed files
├── /src             # Core PHP logic and backend scripts
├── index.php        # Application entry point
└── README.md        # Technical documentation (Required)
```

STEP 5: VERIFICATION CHECKLIST

Before beginning Sprint 1, you must confirm the following:

- [ ] I can access localhost in my browser.
- [ ] I can create a new branch in Git (git checkout -b setup).
- [ ] My database connection file is correctly configured (but not pushed with plain-text passwords).

INSTRUCTOR SIGN-OFF

Final Setup Verified: _____ (Initials) | **Date:** _____