

Technical Setup Checklist for MLA60970

Overview

This checklist ensures you have all the technical tools configured before beginning Week 1. Complete all items in order - the setup builds on itself.

Phase 1: Video Tutorials (Required Viewing)

Watch VS Code Tutorial

“Visual Studio Code Crash Course” by Traversy Media (32 min) - **Link:** https://www.youtube.com/watch?v=WPqXP_kLzpo - **Focus:** Installation, interface, file management, extensions - **Take notes on:** Creating files, using extensions, basic shortcuts

Watch GitHub Tutorial

“Git and GitHub for Beginners” by freeCodeCamp (first 40 min) - **Link:** <https://www.youtube.com/watch?v=RGOj5yH7evk> - **Focus:** Account creation, repositories, basic workflow, GitHub Pages - **Take notes on:** Creating repos, commits, publishing to web

Phase 2: Software Installation

Install VS Code

- ☐ Download from code.visualstudio.com
- ☐ Install and launch successfully
- ☐ Complete welcome setup
- ☐ Can create and save a new file

Create GitHub Account

- ☐ Sign up at github.com
- ☐ Use professional username
- ☐ Use school email address
- ☐ Verify email address
- ☐ Add profile picture and basic info

Phase 3: Essential Configuration

VS Code Extensions (Install all 5)

Install through Extensions panel (Ctrl/Cmd+Shift+X): 1. ☐ **Live Server** - Local development server 2. ☐ **Prettier - Code Formatter** - Auto-formatting 3. ☐ **Auto Rename Tag** - HTML tag management 4. ☐ **HTML CSS Support** - Enhanced syntax support 5. ☐ **Color Highlight** - Visual color preview

VS Code Settings

Configure through Settings (Ctrl/Cmd+,): - [] Enable “Format On Save” - [] Set Prettier as default formatter - [] Enable “Auto Save” - [] Verify settings are saved

Phase 4: Project Setup

Create Course Workspace

- ☐ Create folder on computer: `MLA60970-WebDev`
- ☐ Open folder in VS Code (File → Open Folder)
- ☐ Create subfolder: `Week1`
- ☐ Verify you can navigate files in VS Code

Test HTML Development

Create test file `test.html` with this content:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Setup Test</title>
</head>
<body>
  <h1>Hello, Web Development!</h1>
  <p>VS Code and Live Server are working correctly.</p>
</body>
</html>
```

Test checklist: - [] File saves automatically - [] Code formats when saved (Prettier working) - [] Right-click → “Open with Live Server” available - [] Page opens in browser successfully - [] Changes in VS Code appear instantly in browser

Phase 5: GitHub Integration

Test GitHub Workflow

- ☐ Create new repository on GitHub: `web-dev-test`
- ☐ Make repository public
- ☐ Add README file using GitHub’s web interface
- ☐ Can edit files directly on GitHub (no command line needed)
- ☐ Can view repository at `github.com/yourusername/web-dev-test`

GitHub Pages Setup (Web Interface Only)

- ☐ Go to repository Settings
- ☐ Find “Pages” section
- ☐ Set source to “Deploy from branch: main”
- ☐ Note your GitHub Pages URL
- ☐ Verify simple webpage publishes successfully

Note: This course uses GitHub’s web interface exclusively - no command line or terminal commands required. Works the same on Mac and PC.

Phase 6: Final Verification

Complete Workflow Test

Test the complete development workflow: 1. [] Create HTML file in VS Code 2. [] Use Live Server to preview locally 3. [] Upload file to GitHub repository 4. [] View published page on GitHub Pages 5. [] Make changes and verify they appear online

Professional Setup Check

- ☐ VS Code workspace is organized with clear folder structure
- ☐ GitHub profile looks professional
- ☐ Can efficiently create, edit, and publish web content
- ☐ Comfortable with basic VS Code shortcuts and features

Troubleshooting Resources

VS Code Issues

- **Check:** VS Code Setup Guide
- **Common:** Extension not working → restart VS Code
- **File access:** Ensure folder permissions are correct

GitHub Issues

- **Check:** GitHub Setup Guide
- **Common:** Email not verified → check inbox and spam
- **Pages not working:** Repository must be public

Get Help

- **Instructor support** for course-specific issues
- **Peer collaboration** through course discussions
- **Online documentation** for technical problems

Next Steps After Setup

Ready for Course Content

- ☐ Review Preparatory Materials checklist
- ☐ Read Digital Communication Literacy
- ☐ Complete Design Thinking Preparation
- ☐ Begin Week 1 when course starts

Professional Development Context

This technical setup provides: - **Industry-standard tools** used by professional web developers - **Version control skills** essential for any technical career - **Web publishing capabilities** for portfolio and project sharing - **Foundation for advanced learning** in web development

Complete technical setup enables you to focus on learning design and development concepts rather than troubleshooting tools. This professional workflow will serve you throughout the course and beyond.