Complete MERN Stack Development Environment Setup Guide for Windows

Contents

Prerequisites Checklist

Before starting, ensure you have:

- Windows 10 or Windows 11
- Administrator access to your computer
- Stable internet connection
- At least 4GB free disk space

Part 1: Node.js Installation and Testing

Step 1: Download Node.js

1. Visit the official Node.js website: **https://nodejs.org/**
2. Download the **LTS (Long Term Support)** version - this is recommended for most users
3. Choose the "Windows Installer (.msi)" for 64-bit systems

Step 2: Install Node.js

1. Run the downloaded .msi file as administrator
2. Follow the installation wizard:
    - Accept the license agreement
    - Choose installation location (default: C:\Program Files\nodejs\)
    - **IMPORTANT**: Make sure "Add to PATH" is checked ✅
    - Select "Automatically install necessary tools" ✅

Step 3: Verify Node.js Installation

1. Press Windows Key + R, type cmd, press Enter
2. In Command Prompt, type:
3. node --version

Expected output: v18.x.x or v20.x.x (current LTS version)

4. Test npm (Node Package Manager):
5. npm --version

Expected output: 9.x.x or higher

Step 4: Test Node.js Functionality

1. Create a test file:
2. echo console.log("Hello from Node.js!"); > test-node.js
3. Run the test:
4. node test-node.js

Expected output: Hello from Node.js!

Troubleshooting Node.js PATH Issues

If commands are not recognized:

1. **Manual PATH Setup:**
    - Press Windows Key + X → System
    - Click "Advanced system settings"
    - Click "Environment Variables"
    - Under "System Variables", find and select "Path"
    - Click "Edit" → "New"
    - Add: C:\Program Files\nodejs\
    - Add: C:\Users\[YourUsername]\AppData\Roaming\npm
    - Click OK and restart Command Prompt

Part 2: Git Installation and Configuration

Step 1: Download Git

1. Visit: **https://git-scm.com/download/windows**
2. Download "64-bit Git for Windows Setup"

Step 2: Install Git

1. Run the installer as administrator
2. **Important Installation Options:**
    - Use Git from the command line and 3rd-party software ✅
    - Use bundled OpenSSH ✅
    - Use the OpenSSL library ✅

- Checkout Windows-style, commit Unix-style line endings ✅
- Use Windows' default console window ✅
- Enable file system caching ✅

Step 3: Verify Git Installation

1. Open Command Prompt or PowerShell
2. Check Git version:
3. git --version

Expected output: git version 2.x.x.windows.x

Step 4: Configure Git (First-time setup)

1. Set your username:
2. git config --global user.name "Your Name"
3. Set your email:
4. git config --global user.email "your.email@example.com"
5. Verify configuration:
6. git config --list

Step 5: Test Git Functionality

1. Create a test repository:
2. mkdir git-testcd git-testgit init
3. Create and commit a test file:
4. echo "# Test Repository" > README.mdgit add README.mdgit commit -m "Initial commit"

Expected output: Successful commit message

Part 3: Visual Studio Code Installation and Setup

Step 1: Download VS Code

1. Visit: **https://code.visualstudio.com/**
2. Click "Download for Windows"
3. Download the "User Installer" (recommended)

Step 2: Install VS Code

1. Run the installer
2. **Important Installation Options:**
    - Add "Open with Code" action to Windows Explorer file context menu ✅
    - Add "Open with Code" action to Windows Explorer directory context menu ✅
    - Register Code as an editor for supported file types ✅
    - Add to PATH ✅

Step 3: Verify VS Code Installation

1. Open Command Prompt
2. Test VS Code command:
3. code --version

Expected output: Version information

4. Test opening VS Code from command line:
5. code .

This should open VS Code in the current directory

Step 4: Essential VS Code Extensions for MERN Stack

Install these extensions through VS Code:

1. **JavaScript (ES6) code snippets** - by charalampos karypidis
2. **ES7+ React/Redux/React-Native snippets** - by dsznajder
3. **Auto Rename Tag** - by Jun Han
4. **Bracket Pair Colorizer 2** - by CoenraadS
5. **GitLens** - by Eric Amodio
6. **Prettier - Code formatter** - by Prettier
7. **ESLint** - by Microsoft
8. **Thunder Client** - by RangaVadhineni (REST API testing)

Installation Method:

- Open VS Code

- Press Ctrl + Shift + X (Extensions panel)
- Search for each extension name
- Click "Install"

Part 4: Complete System Test for MERN Stack

Step 1: Create a Test MERN Project Structure

1. Open Command Prompt as Administrator
2. Create project directory:
3. mkdir mern-test-projectcd mern-test-project

Step 2: Initialize Node.js Project

1. Initialize npm:
2. npm init -y
3. Install basic MERN dependencies:
4. npm install express mongoose react react-domnpm install -D nodemon concurrently

Step 3: Create Basic Server File

1. Create server.js:
2. echo const express = require('express'); const app = express(); const PORT = 5000; app.get('/', (req, res) => res.send('MERN Stack Server Running!')); app.listen(PORT, () => console.log(` Server running on port ${PORT}`)); > server.js

Step 4: Test the Setup

1. Open VS Code in project directory:
2. code .
3. Open terminal in VS Code (Ctrl + ~)
4. Run the server:
5. node server.js
6. Open browser and go to: http://localhost:5000
7. Expected result: "MERN Stack Server Running!" message

Step 5: Initialize Git Repository

1. In VS Code terminal:
2. git initgit add .git commit -m "Initial MERN setup"

Part 4: Environment Variables and PATH Configuration

Understanding Windows PATH

The PATH environment variable tells Windows where to find executable files.

Current PATH Verification

1. Open Command Prompt
2. Check current PATH:
3. echo %PATH%

Adding Custom Paths (if needed)

1. Press Windows Key + X → System
2. Advanced system settings → Environment Variables
3. Under "User variables" or "System variables", find "Path"
4. Click "Edit" → "New"
5. Add your custom path (e.g., C:\MyTools\bin)

Common MERN Stack Paths to Verify

- Node.js: C:\Program Files\nodejs\
- NPM Global: C:\Users\[Username]\AppData\Roaming\npm
- Git: C:\Program Files\Git\cmd
- VS Code: C:\Users\[Username]\AppData\Local\Programs\Microsoft VS Code\bin

Part 5: Troubleshooting Common Issues

Issue 1: "Command not recognized"

Solution:

- Restart Command Prompt after installation
- Check PATH environment variable
- Reinstall with "Add to PATH" option enabled

Issue 2: npm permission errors

Solution:

- Run Command Prompt as Administrator
- Or configure npm to use a different directory:
- npm config set prefix %APPDATA%\npm

Issue 3: Git authentication issues

Solution:

- Use Git Credential Manager (comes with Git for Windows)
- For GitHub, use Personal Access Tokens instead of passwords

Issue 4: VS Code not opening from command line

Solution:

- Reinstall VS Code with "Add to PATH" option
- Or add manually: C:\Users\[Username]\AppData\Local\Programs\Microsoft VS Code\bin

---

Additional Resources

Official Documentation

- **Node.js:** https://nodejs.org/en/docs/
- **Git:** https://git-scm.com/doc
- **VS Code:** https://code.visualstudio.com/docs

MERN Stack Learning Resources

- **MongoDB University:** https://university.mongodb.com/
- **React Documentation:** https://reactjs.org/docs/
- **Express.js Guide:** https://expressjs.com/en/guide/
- **YouTube:** www.youtube.com/@WebDevSimplified
- The University of Helsinki-**Deep Dive Into Modern Web Development**

  : https://fullstackopen.com/en/