

CoreSkills4ai Repository Setup Guide

Mission

Markdown-driven autonomous development system with multi-mode operation for building production-grade, teachable, portfolio-ready projects.

Repository Structure (Hybrid Model)

C:\CoreSkills4ai\

├── agent-system\

MONO-REPO: Core agent system

├── .git\

Single Git repo for agent

├── core\

Agent runtime

├── modes\

Operating mode definitions

├── templates\

Project templates

├── docs\

System documentation

└── docker\

Container definitions

├── projects\

MULTI-REPO: Individual projects

├── portfolio-app-1\

Separate Git repo

├── client-work-a\

Separate Git repo

└── classroom-demos\

Separate Git repo

├── shared\

Shared resources (no Git)

├── templates\

├── configs\

└── utils\

└── README.md

Root overview

Setup Instructions

Step 1: Create Root Structure

```
powershell
```

```
# Create root directory
```

```
mkdir C:\CoreSkills4ai
```

```
cd C:\CoreSkills4ai
```

```
# Create main folders
```

```
mkdir agent-system
```

```
mkdir projects
```

```
mkdir shared
```

```
mkdir shared\templates
```

```
mkdir shared\configs
```

```
mkdir shared\utils
```

Step 2: Initialize Agent System (Mono-Repo)

```
powershell
```

```
cd C:\CoreSkills4ai\agent-system
```

```
# Initialize Git
```

```
git init
```

```
# Create core structure
```

```
mkdir core
```

```
mkdir modes
```

```
mkdir templates
```

```
mkdir docs
```

```
mkdir docker
```

```
# Create initial files
```

```
New-Item -ItemType File -Path "README.md"
```

```
New-Item -ItemType File -Path ".gitignore"
```

```
New-Item -ItemType File -Path "PRD.md"
```

```
New-Item -ItemType File -Path "ARCHITECTURE.md"
```

```
New-Item -ItemType File -Path "TASKS.md"
```

Step 3: Create .gitignore

```
powershell
```

Navigate to agent-system

```
cd C:\CoreSkills4ai\agent-system
```

Create .gitignore with common exclusions

```
@"
```

```
# Python
```

```
__pycache__/
```

```
*.py[cod]
```

```
*$py.class
```

```
*.so
```

```
.Python
```

```
env/
```

```
venv/
```

```
.venv/
```

```
# Node
```

```
node_modules/
```

```
npm-debug.log*
```

```
# Docker
```

```
*.log
```

```
.dockerignore
```

```
# IDE
```

```
.vscode/
```

```
.idea/
```

```
*.swp
```

```
# Environment
```

```
.env
```

```
.env.local
```

```
*.secret
```

```
# OS
```

```
.DS_Store
```

```
Thumbs.db
```

```
# Build outputs
```

```
dist/
```

```
build/
```

```
*.egg-info/
```

```
"@ | Out-File -FilePath ".gitignore" -Encoding UTF8
```

Step 4: First Commit

```
powershell

# Stage all files
git add .

# Create first commit
git commit -m "Initial commit: CoreSkills4ai Agent System foundation"

# Create main branch
git branch -M main
```

Step 5: Link to GitHub (Optional - Do Later)

```
powershell

# When ready to push to GitHub:
# 1. Create repo on GitHub first
# 2. Then run:
# git remote add origin https://github.com/yourusername/coreskills4ai-agent.git
# git push -u origin main
```

Safety Checklist

Before running ANY Git command:

- ☐ Are you in the correct directory? (`(pwd)` to check)
- ☐ Do you have unsaved work? (Save first!)
- ☐ Are you sure about the command? (Google if unsure)

Safe Git Commands (Use These)

```
powershell

git status      # Check current state (ALWAYS SAFE)
git add .       # Stage files (SAFE)
git commit -m "" # Save snapshot (SAFE)
git push        # Upload to GitHub (SAFE)
git log         # View history (SAFE)
```

DANGEROUS Commands (NEVER Use Without Backup)

powershell

```
git reset --hard # ❌ ERASES local changes
git clean -fd # ❌ DELETES untracked files
git push --force # ❌ Can overwrite remote history
```

🏠 Project Setup (For Each New Project)

powershell

Navigate to projects folder

```
cd C:\CoreSkills4ai\projects
```

Create new project

```
mkdir my-new-project
```

```
cd my-new-project
```

Initialize separate Git repo

```
git init
```

Copy agent templates

```
Copy-Item -Recurse "C:\CoreSkills4ai\agent-system\templates\*" -Destination "."
```

First commit

```
git add .
```

```
git commit -m "Initial project setup from agent templates"
```

📋 Next Steps (Tomorrow)

1. ☒ Run this setup script
2. ☒ Initialize agent-system repo
3. ☒ Create first project in projects/
4. Build core agent files (PRD, ARCHITECTURE, TASKS)
5. Create mode-switching system (MD files for each phase)

Troubleshooting

"Git not found"

```
powershell  
  
# Install Git for Windows  
winget install Git.Git
```

"Permission denied"

```
powershell  
  
# Run PowerShell as Administrator  
# Right-click PowerShell → "Run as Administrator"
```

"File already exists"

```
powershell  
  
# Check current directory  
pwd  
  
# List files  
ls  
  
# If in wrong place, navigate to correct location  
cd C:\CoreSkills4ai\agent-system
```

Support

If anything goes wrong:

1. **STOP** - Don't proceed
2. Run `git status` to see current state
3. Take a screenshot of the error
4. Ask me tomorrow before continuing

Your files are SAFE as long as you:

- Stay in `C:\CoreSkills4ai\` directory
- Only use safe Git commands listed above
- Commit often (`git commit` creates save points)