

## DevOps Practice Assignment

This assignment is designed to help you practice Git, Linux, branching, deployment, and scripting using a simple three-tier application.

**Suggested Application Type:** Create a simple Task Manager application with the following structure: - Frontend: HTML, CSS, JavaScript - Backend: Node.js (Express) or Python (Flask) - Database: SQLite or MongoDB

**Hint:** Ask ChatGPT or Gemini to generate a minimal three-tier folder structure for the Task Manager app.

**Part 1:** Generate the Application 1. Use ChatGPT or Gemini to generate a three-tier Task Manager application. 2. Copy the generated code into a local project folder with a clean structure.

**Hint:** Keep the folder names consistent and readable.

**Part 2: Initialize Git** 1. Initialize a Git repository in your local project folder. 2. Add all files and create the first commit. 3. Create a new GitLab repository and push your local repository to it.

**Hint:** Make sure your remote URL is correct before pushing.

**Part 3: Deploy in WSL** 1. Clone the repository inside your WSL environment. 2. Install the required dependencies for backend and frontend. 3. Start the application components from WSL.

**Hint:** Ensure you install the correct dependencies for your chosen stack.

**Part 4: Create a Service Monitoring Shell Script** 1. Write a shell script that checks if the frontend, backend, and database services are running. 2. The script should output a simple status report. 3. Add the script to the repository and commit it to the main branch.

**Hint:** Use process names or port checks for service validation.

**Part 5: Create a Feature Branch** 1. Create a new branch from main. 2. Make a small change in the frontend, such as updating a header or text. 3. Commit and push this change to the new branch.

**Hint:** Name your branch based on the feature being added.

**Part 6: Deploy UI Change in WSL** 1. Switch to the feature branch inside WSL. 2. Run the updated application with the UI modification applied.

**Hint:** Pull the latest changes before deploying.

**Part 7: Merge the Feature Branch** 1. Open a merge request in GitLab. 2. Provide a meaningful title and description. 3. Complete the merge. 4. Pull the updated main branch into your WSL environment and redeploy.

**Hint:** Ensure there are no conflicts before merging.

Part 8: Document the Assignment 1. Create a file named `docs/assignment_report.md` within the repository. 2. Document the steps followed, issues faced, and observations. 3. Push the documentation to GitLab.

Hint: Use clear headings for better readability.