**IST105 - Introduction to Programming**

**Assignment #5**

*Assignment #5 - Version Control and Deployment with Git and GitHub*

**Important Note:**

**Assignments that do not fulfill all the requirements or contain errors that prevent successful execution on the EC2 instance will receive a grade of zero. Ensure your repository is complete, functional, and publicly accessible before submitting. Test thoroughly to catch and fix any issues before final submission.**

**Prerequisite:**

### Completion of Assignment #4, including the setup of an EC2 instance and creation of a Python and PHP web application.

### Basic knowledge of Git and GitHub, including commands for creating branches and deploying repositories.

### **Objective:**

In this assignment, you will continue working on the data management web application from Assignment #4 by incorporating version control practices and branching strategies using Git and GitHub. You will also deploy the GitHub repository to an AWS EC2 instance and confirm its accessibility through a public IP address.

**Assignment Tasks:**

**1. Setup and Version Control:**

* **Initialize a Git Repository:**

Create a local Git repository in the project directory that contains the files from Assignment #4 (i.e., form.php, data\_management.py, and process.php).

* **Commit Initial Code:**

Commit the code as it was in Assignment #4 to your local Git repository.

**2. GitHub Repository and Branching:**

* **Create a GitHub Repository:**

Create a new public GitHub repository and push your local repository to GitHub.

* **Create Branches:**

Set up the following branches in your GitHub repository:

* + main - This branch should contain the final, tested version of the application.
  + development - Use this branch for testing and debugging changes before merging them into main.
  + feature1 - Use this branch for implementing new features or modifications specifically related to Assignment #5.

**3. Code Modifications on Branches:**

* **Update Code on feature1 Branch:**

Add comments or minor modifications in feature1 to explain your use of Git and branching.

* **Merge Changes to development:**

After testing modifications in feature1, merge the branch into development and confirm that all functionality works as expected.

* **Merge Changes to main:**

After thorough testing in development, merge the code to the main branch to mark it as the final version.

**4. Deploy to EC2 Instance:**

* **Clone the GitHub Repository to EC2:**
  + Connect to your EC2 instance via SSH:

**ssh -i <your-key-pair.pem> ec2-user@<your-public-ip>**

* + Use git clone to retrieve your GitHub repository on the EC2 instance:

**git clone <your-github-repository-url>**

* + Change directory into your project folder (replace <repository-name> with the actual name of your GitHub repository):

**cd <repository-name>**

* **File Permissions:**

Ensure that the data\_management.py script has the correct permissions to be executed:

**sudo chmod +x /var/www/html/data\_management.py**

* **Start Apache Server:**

Start or restart the Apache server to ensure it’s running your updated code:

**sudo service httpd start**

**# or**

**sudo service httpd restart**

**5. Test and Access the Web Application:**

* Public IP Access:
* Verify that the deployed application is accessible through the public IP address of your EC2 instance.
* Test Form Submission:
* Open the form in your web browser using http://<your-public-ip>/form.php, enter values, and verify the correct output on the results page.

**Submission Requirements:**

* **GitHub Repository Link:**

Submit the public link to your GitHub repository, which should include the branches main, development, and feature1.

**[Insert your URL here]**

* **Screenshot of Web Application:**

Include a screenshot of the deployed web application showing the public IP address in the URL.

**[Insert your screenshot here]**

**Important Notes:**

* **Repository Completeness:** Ensure all files from Assignment #4 are included in your GitHub repository.
* **Functionality:** Make sure your application is fully functional on the EC2 instance and accessible via the public IP address.
* **Proper Permissions:** Double-check file permissions and configurations on your EC2 instance.

**Tips:**

* **Git Commands Recap:** Use git branch, git checkout, git merge, and git push commands effectively to manage branches.
* **Testing Before Merging:** Always test changes on the feature1 and development branches before merging to main.