# LABORATORY EXERCISE 1

# INITIALIZING A PROJECT USING CODEIGNITER WITH BOOTSTRAP INTEGRATION AND GITHUB VERSION CONTROL

**Learning Objectives**

By the end of this laboratory exercise, students should be able to:

* Learn how to install and configure CodeIgniter as a PHP framework for web application development.
* Understand the integration of Bootstrap for creating responsive and user-friendly web interfaces.
* Develop skills in setting up a project folder structure and linking CodeIgniter with Bootstrap templates.
* Gain practical experience in using GitHub for version control and project collaboration.
* Apply best practices for project initialization and version tracking in a development environment.

**Prerequisite student experiences and knowledge**

Before starting this exercise, students should have:

* Basic understanding of HTML, CSS, and PHP.
* Familiarity with databases and MySQL basics.
* Experience using a local web server (XAMPP/WAMP/LAMP).
* Knowledge of basic Git commands and GitHub usage.
* Ability to use a text editor/IDE such as Visual Studio Code.

**Background**

This exercise introduces students to initializing a web project using the CodeIgniter framework, a lightweight PHP MVC framework known for speed and simplicity. The integration of Bootstrap ensures that the project interface is responsive and visually appealing. Furthermore, using GitHub provides students with a professional approach to version control, ensuring project tracking, collaboration, and code management.

**Materials/Resources**

* **Personal Computer with Internet Access**
* **XAMPP/WAMP/LAMP server installed**
* **CodeIgniter Framework (latest version)**
* **Bootstrap (via CDN or downloaded files)**
* **Visual Studio Code or any code editor**
* **Git and GitHub Account**
* **Web Browser (Chrome, Firefox, etc.)**

**Laboratory Activity**

**Step 1: Environment Setup**

1. Install and configure XAMPP/WAMP on your machine.
2. Download and extract the latest version of CodeIgniter from your `htdocs` (XAMPP) or `www` (WAMP) directory.
3. Rename the folder as ` **ITE311-LASTNAME**`.
4. Run your local server and test the CodeIgniter installation by accessing:

**http://localhost/ ITE311-LASTNAME**

**Step 2: Initialize Git Repository**

1. Open the project folder in your terminal or Git Bash.
2. Run the following commands:

**git init**

**git add .**

**git commit -m "Initial commit - CodeIgniter setup"**

1. Create a new repository on GitHub named **WebSystem-ITE311**.
2. Link your local project to GitHub:

**git remote add origin https://github.com/yourusername/ WebSystem-ITE311.git**

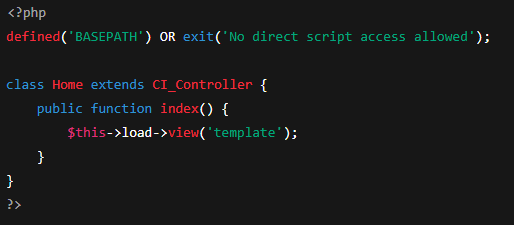
**git push -u origin main**

**Step 3: Integrate Bootstrap**

1. Download Bootstrap from [https://getbootstrap.com](https://getbootstrap.com) or use the CDN links.
2. Navigate to `application/views` and create a new file `template.php`.
3. Add Bootstrap CDN inside your template's `<head>` section.
4. Test by creating a simple navigation bar using Bootstrap classes.

**Step 4: Create a Home Controller and View**

1. Navigate to `application/controllers` and create a file named `Home.php`.
2. Insert the following code:



1. In your browser, visit: **http://localhost/ITE311-LASTNAME /index.php/home**
2. Open your web browser and navigate to http://127.0.0.1:8000 to see your basic homepage.

**Step 5: Push Update Project to GitHub**

1. After Bootstrap integration and controller creation, push changes:

git add .

git commit -m "Added Bootstrap template and Home controller"

git push

**Step 6: Push Update Project to GitHub**

1. Take screenshots of your running home page and GitHub repository.
2. Insert them in your lab report under “Output/Results.”

Output / Results

* Screenshot of successful CodeIgniter setup.
* Screenshot of integrated Bootstrap layout.
* Screenshot of GitHub repository with pushed commits.
* The instructional manual should be printed, and the softcopy uploaded at the time of checking.

**QUESTIONS:**

1. Why is it important to integrate Bootstrap when developing a web project in CodeIgniter?

Integrating Bootstrap in a CodeIgniter web project is important because it helps you quickly design a responsive and professional-looking website without writing a lot of custom CSS. It saves time and ensures your project looks good on all devices.

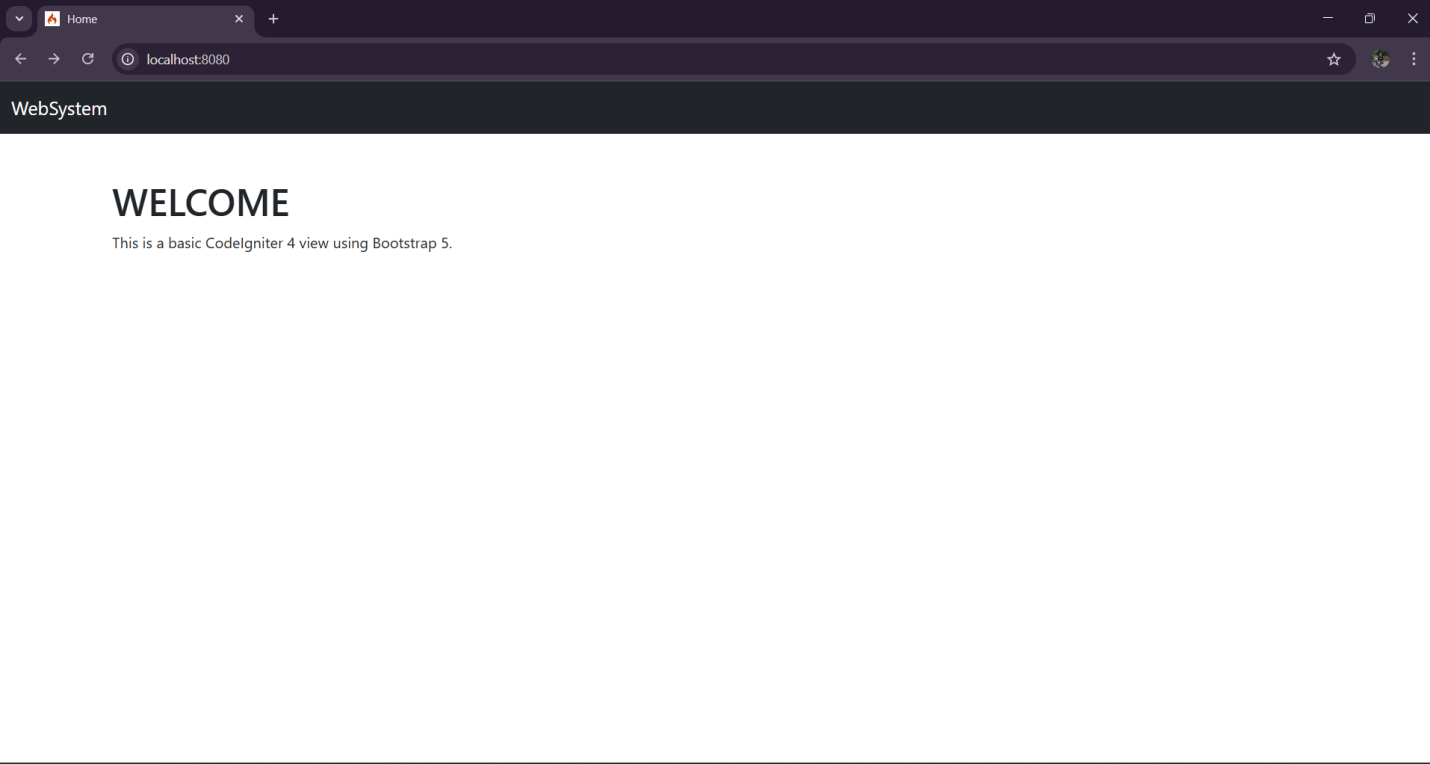
1. How does using GitHub improve the development and management of web applications?

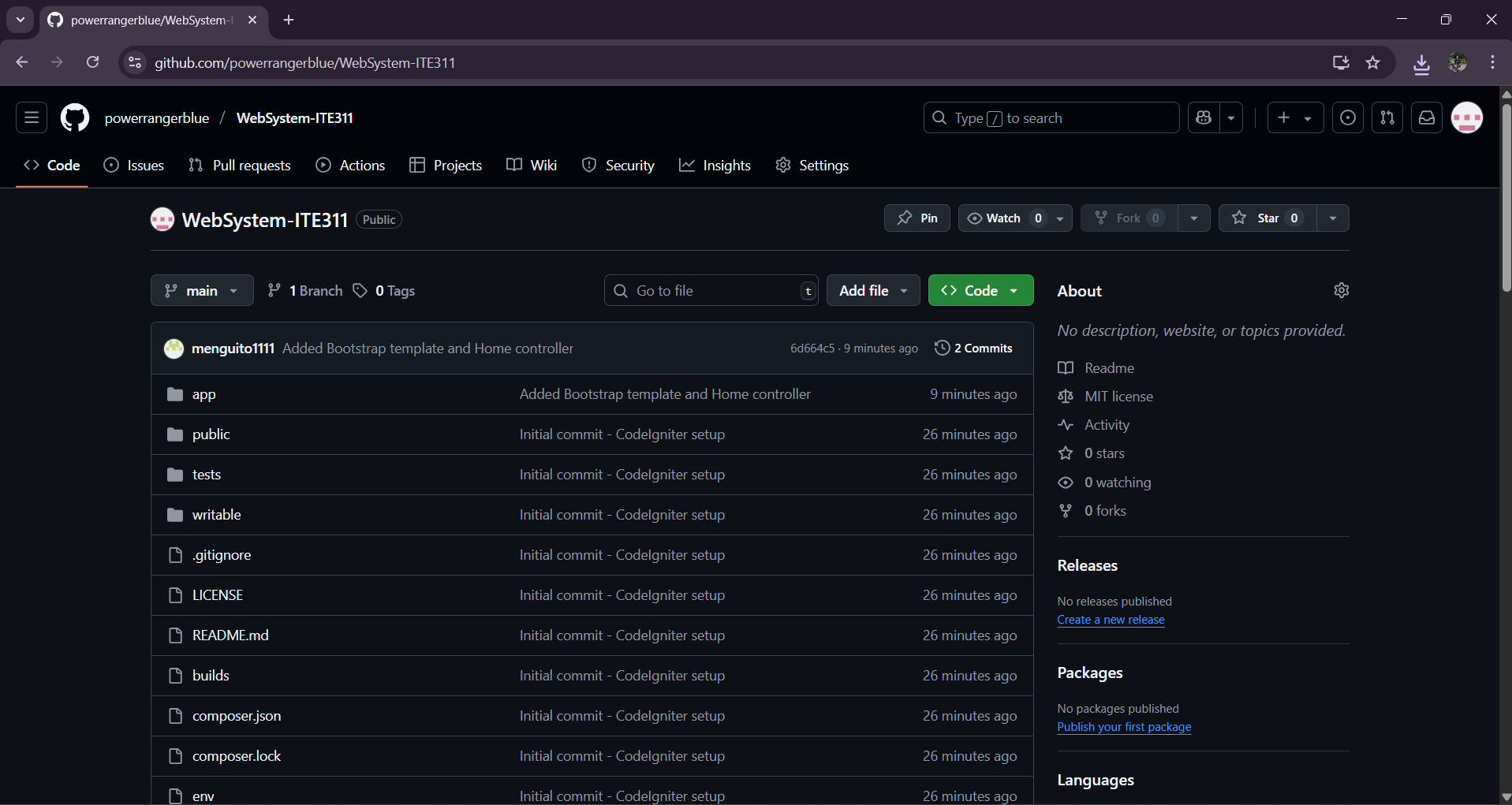
Using GitHub improves the development and management of web applications by allowing you to track changes, collaborate with others, and back up your code online. It makes teamwork easier, helps manage versions, and keeps your project organized and secure.

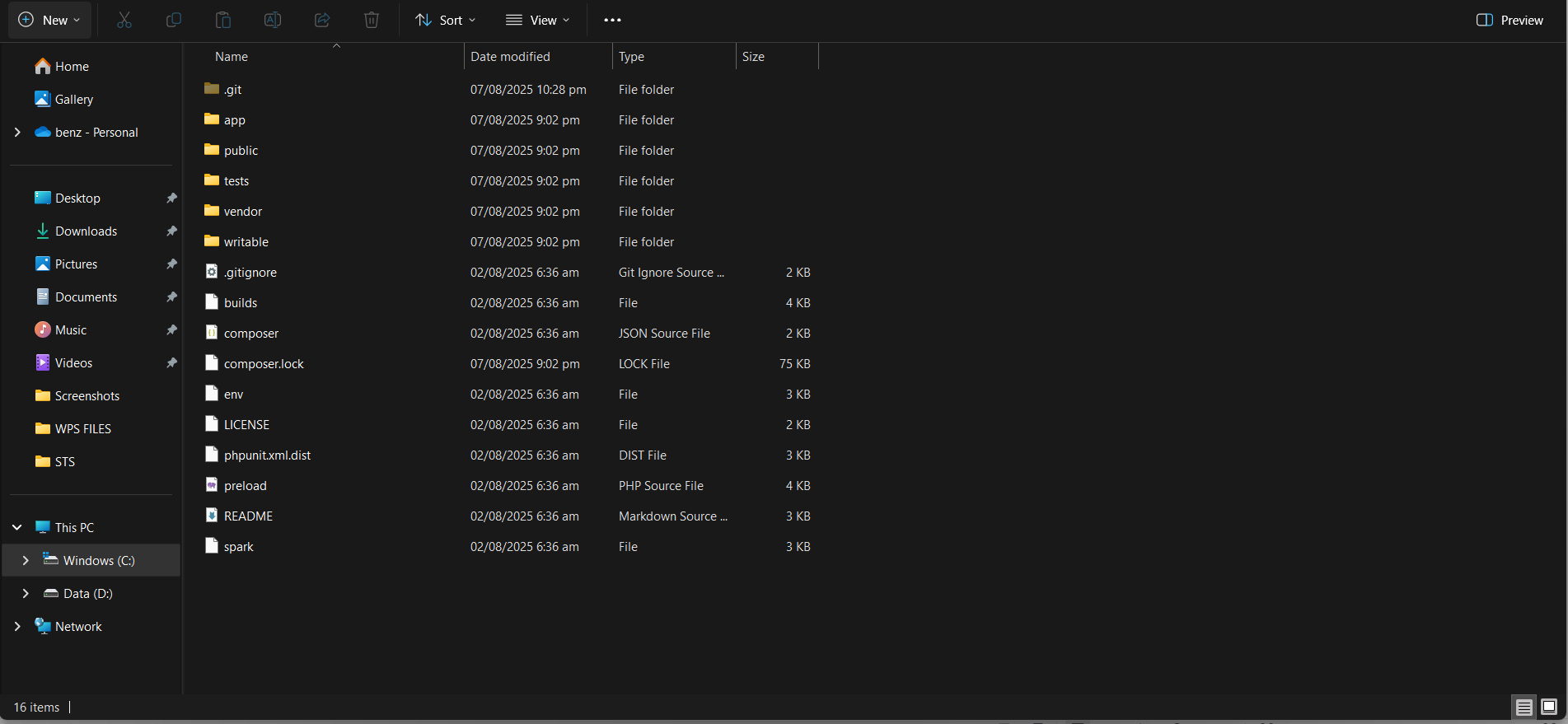
1. What steps did you perform to successfully set up and test the CodeIgniter framework on your local server?

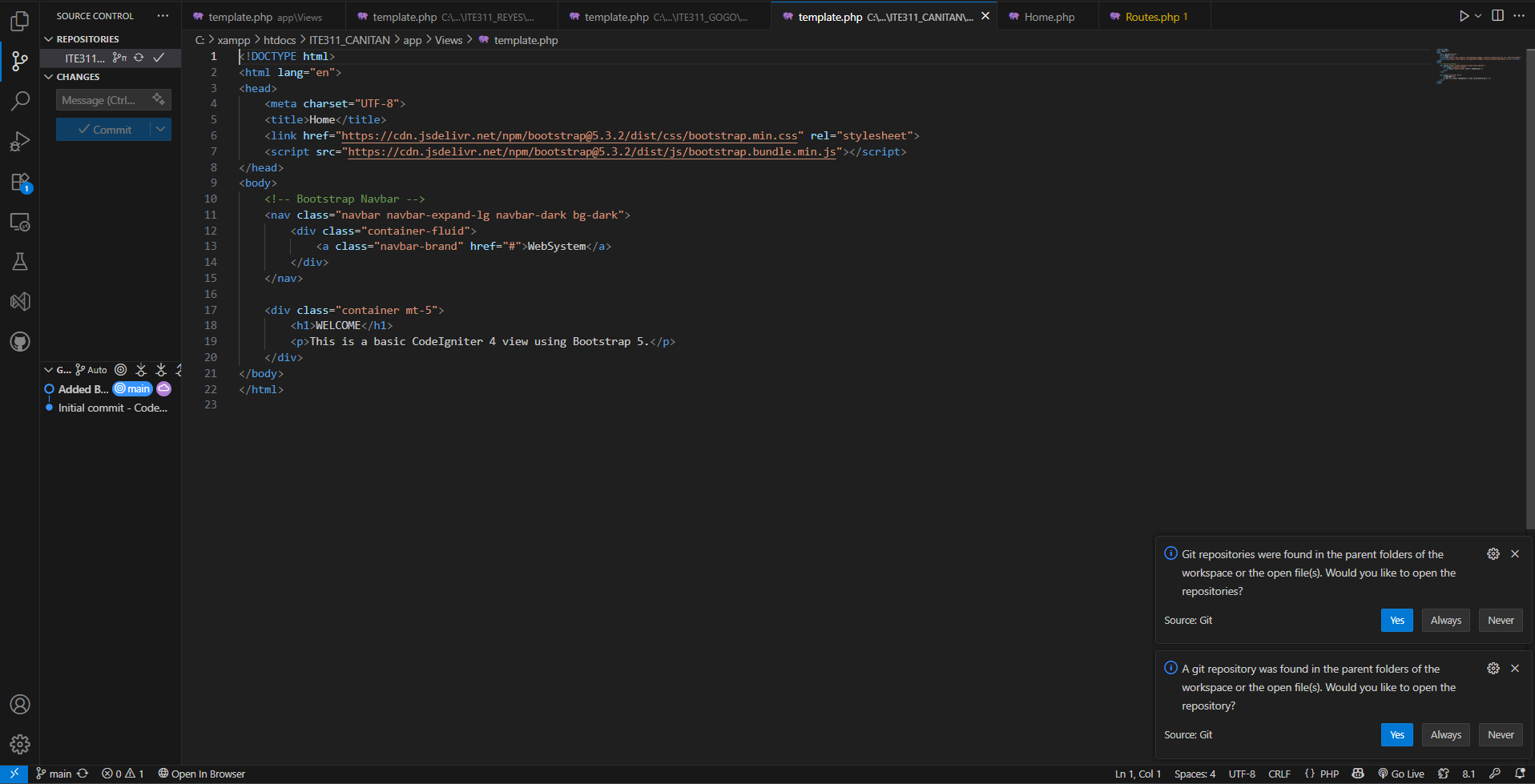
To successfully set up and test the CodeIgniter framework on my local server, I first installed XAMPP and started the Apache server. Then, I downloaded the latest version of CodeIgniter from the official website and extracted it into the `htdocs` directory of XAMPP. After that, I renamed the extracted folder to `ITE311-LASTNAME` to match the required format. Finally, I opened my web browser and accessed `http://localhost/ITE311-LASTNAME`, where I was able to see the CodeIgniter welcome page, confirming that the setup was successful.

**Output / Results**









**USER MANUAL**

**Overview**

**This manual will guide you step-by-step to:**

**Set up CodeIgniter on your local server**

**Initialize a Git repository and connect to GitHub**

**Integrate Bootstrap for styling**

**Create a simple home page using a controller and view**

**Push your project updates to GitHub**

**Step 1: Environment Setup**

**Install XAMPP or WAMP**

**Download XAMPP: https://www.apachefriends.org**

**Install and open XAMPP Control Panel**

**Start Apache and MySQL**

**Download CodeIgniter**

**Visit: https://codeigniter.com**

**Download the latest version of CodeIgniter 4**

**Extract and Rename**

**Move the extracted folder to:**

**C:\xampp\htdocs (for XAMPP)**

**or C:\wamp\www (for WAMP)**

**Rename the folder to:**

**ITE311-LASTNAME (e.g., ITE311-Canitan)**

**Test in Browser**

**Open a browser and visit:**

**http://localhost/ITE311-LASTNAME**

**Step 2: Initialize Git Repository**

**Open your project folder in Git Bash or terminal.**

**Run these commands:**

**git init**

**git add .**

**git commit -m "Initial commit - CodeIgniter setup"**

**Go to GitHub, create a new repository:**

**WebSystem-ITE311**

**Link your project:**

**git remote add origin https://github.com/yourusername/WebSystem-ITE311.git**

**git push -u origin main**

**Step 3: Integrate Bootstrap**

**Download Bootstrap or use the CDN link from the official Bootstrap website.**

**Inside the application/views folder, create a new file named template.php.**

**Add Bootstrap links in the <head> section of the template file.**

**Add a simple navigation bar to test Bootstrap integration.**

**Open the project in your browser to check if the layout displays correctly.**

**Step 4: Create a Home Controller and View**

**Inside the application/controllers folder, create a controller file named Home.php.**

**Load the view file you created (template.php) from this controller.**

**You should see your styled homepage using Bootstrap.**

**Step 5: Push Project Updates to GitHub**

**After making changes (like adding Bootstrap and the controller), open Git Bash or your terminal.**

**Stage the new changes, commit them with a message, and push the updates to GitHub.**

**Confirm that the changes are visible in your GitHub repository.**

**Step 6: Output / Results**

**Take and include the following screenshots in your lab report:**

**The CodeIgniter welcome page (setup confirmation)**

**Your homepage showing the Bootstrap layout**

**Your GitHub repository showing the commits and pushed code**

**Conclusion**

**In conclusion, setting up CodeIgniter with Bootstrap integration and GitHub version control provides a strong foundation for developing modern, responsive web applications. By using XAMPP as the local server, CodeIgniter for the framework, Bootstrap for styling, and GitHub for code management, the development process becomes more organized, efficient, and professional. This setup not only ensures a smooth workflow but also prepares the project for future collaboration and deployment.**