EGR111 – Introduction to Computer Science (C Language)

Git and GitHub Introduction and Setup   Rev 1.2

Overview:

The purpose of this activity is to create your online repository to hold not only the code for this course, but likely coding projects that you create or collaborate on for the rest of your career. You'll use Git commands for source code control and commit your work to a GitHub repository.

This will be standard operating procedure for the rest of this course – limited Moodle submissions but regular commits to GitHub. Follow along with the instructor carefully. There are multiple configuration steps. It will pay off at the end when code edits are submitted with a simple click.

Prerequisites:

Prior to beginning the instruction provided in this lesson you must have completed the following:

1. Visual Studio Introduction

Performance Outcomes:

1. Configure local and remote Git repositories.
2. Stage code updates using the VS Code source code interface.
3. Commit code updates using the VS Code source code interface.
4. Publish code to your remote repository.
5. View repositories on your GitHub account.

Resources:

1. GitHub: Where the world builds software · GitHub
2. Setting your username in Git – GitHub Docs
3. Setting your commit email address – GitHub Docs
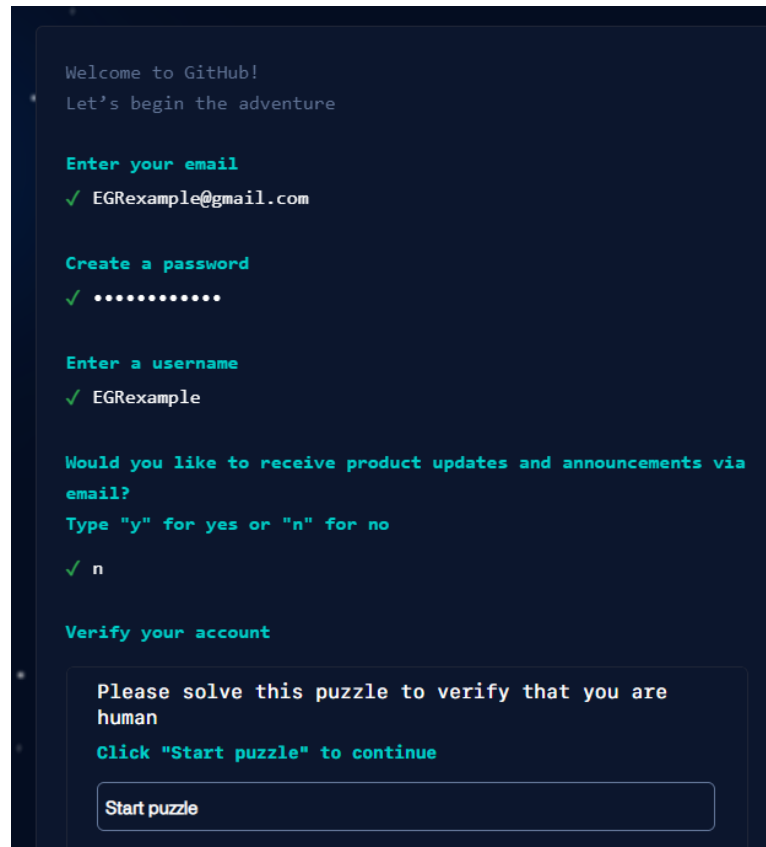4. Working with GitHub in Visual Studio Code

Materials:

1. RPi with VS Code

Directions:

Part 1 – Remote Repository (GitHub) Configuration

1. Follow along with the instructor as they create a new GitHub account at https://github.com. Figure 1 show the current interface for a new GitHub account with the instructor using the EGRexample email address. Use your preferred email address when creating this resource. It is okay and possibly preferred to use an email other than your mail.nmc.edu account. The instructor will discuss.

Figure 1 Creating a New GitHub Account

2. The last step in the new account process is to enter a validation code that is sent to the email address that you provided. Open the new email message from GitHub and copy the code to the form.

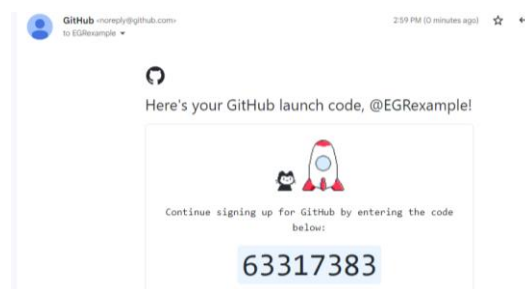*Figure 2 GitHub Prompt for Security Code*

Figure 3 Email with GitHub Launch Code

EGR111 – Introduction to Computer Science (C Language)

3. With the security code entered, you'll be redirected to a main GitHub entry page prompting for you to create your first code repository. The site is a great tool, and there many resources that can help. Be sure to become familiar with it as the course progresses. For now, you'll create a single EGR111 repository that will contain all coding solutions for this course.
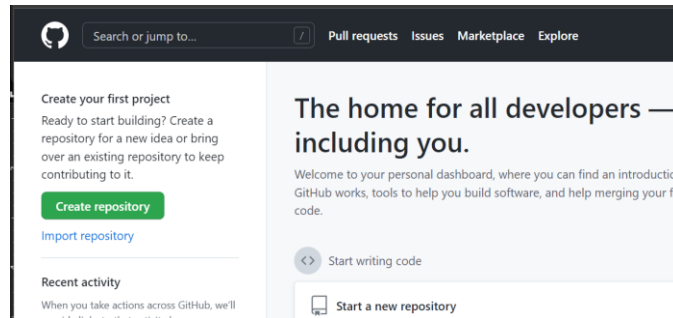

Figure 4 GitHub Create Repository Page

4. Click the Create repository button to start your first repository on GitHub. Your GitHub username is shown as Owner. Enter the **EGR111** as the repository name as shown below. Set the visibility to **Private**. See Figure 5.

Figure 5 Creating the EGR111 Repository as Private

5. Click the check box to add a **README** file and a specific **.gitignore** template file by clicking the drop down and selecting **C** as shown below.
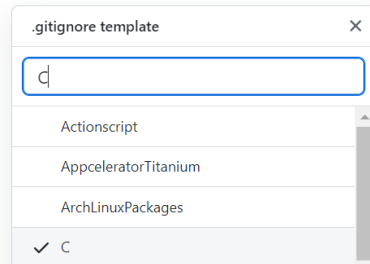
Figure 6 Adding README and .gitignore Files

6. The EGR111 repository is created. Click the edit icon on the README.md file and add **your name** to the title. This is not typical but is required so that the instructor can easily identify your repository. You'll need to click the "Commit Change" button at the bottom of the edit screen to save (commit) your change.
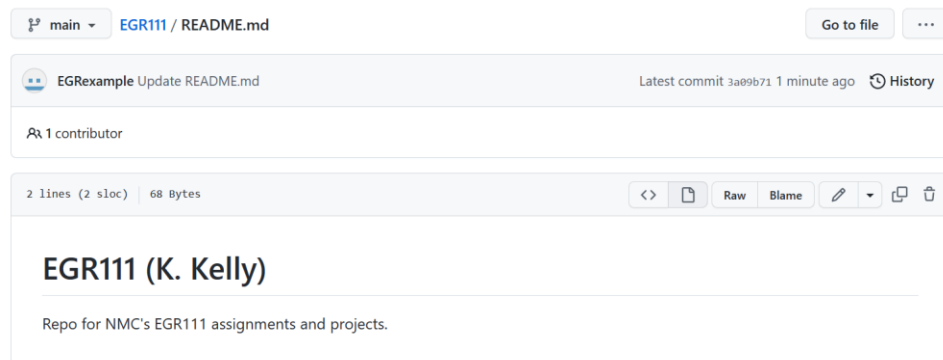

Figure 7 Editing the README File to Include Your Name

7. Next, you'll add the instructor as a collaborator. This is required due to the repository being private. The repository is visible to only you and collaborators. Click the **Settings** menu item and the select **Collaborators**.
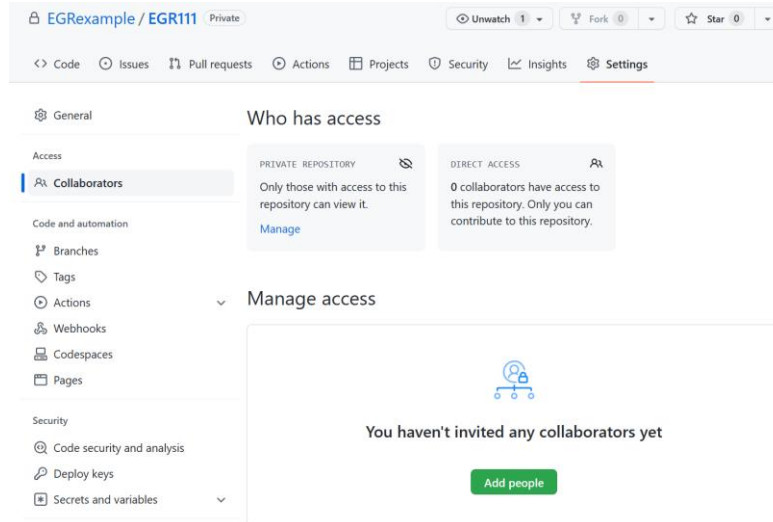
4

Figure 8 Accessing Collaborators in Settings

8. Click the **Add people** button and enter **K2Controls** to add the instructor as a collaborator. Select the K2Controls account shown. The button at the bottom updates to **Add k2controls to this repository**. Click the button to add.
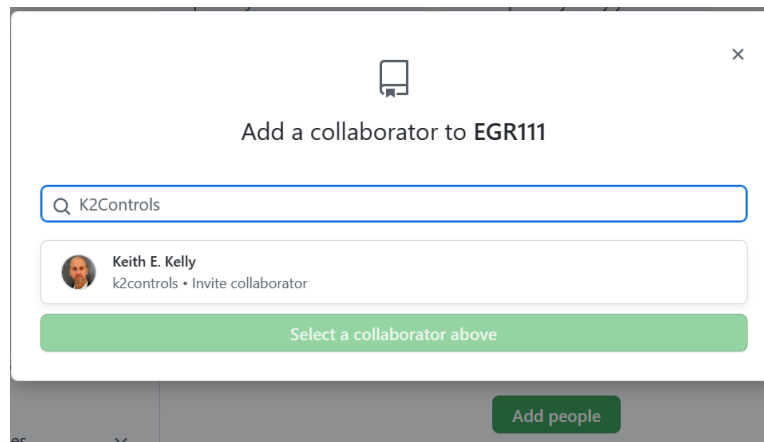

Figure 9 Adding a Collaborator
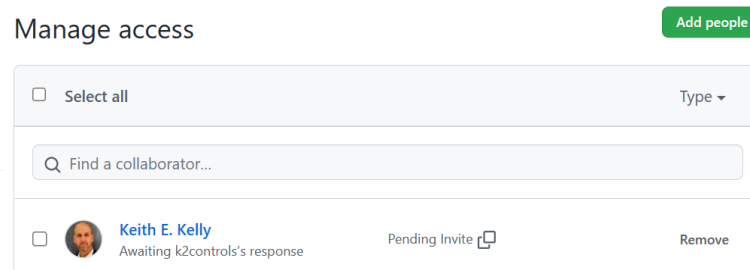
9. The invitation to k2controls is pending.

Figure 10 Pending Access for K2Controls

EGR111 – Introduction to Computer Science (C Language)

Part 2 – Local Repository (VS Code) Configuration

10. The remote GitHub repository is configured. Now configure the local repository.

11. You must set your Git username and email address on the Raspberry Pi. Access your RPi, open a terminal, and enter the git config commands shown below using **your username** and **email**.


Figure 11 Enter Global Values for user.name and user.email

12. Now use the RPi browser to log into your GitHub account. Access your EGR111 repository. Click the <> *Code* button and copy the URL to your repository.

13. Keep the browser window open and launch VS code in your home directory by entering the following command in the terminal.
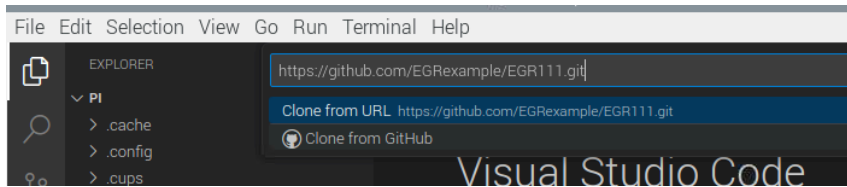
    code .

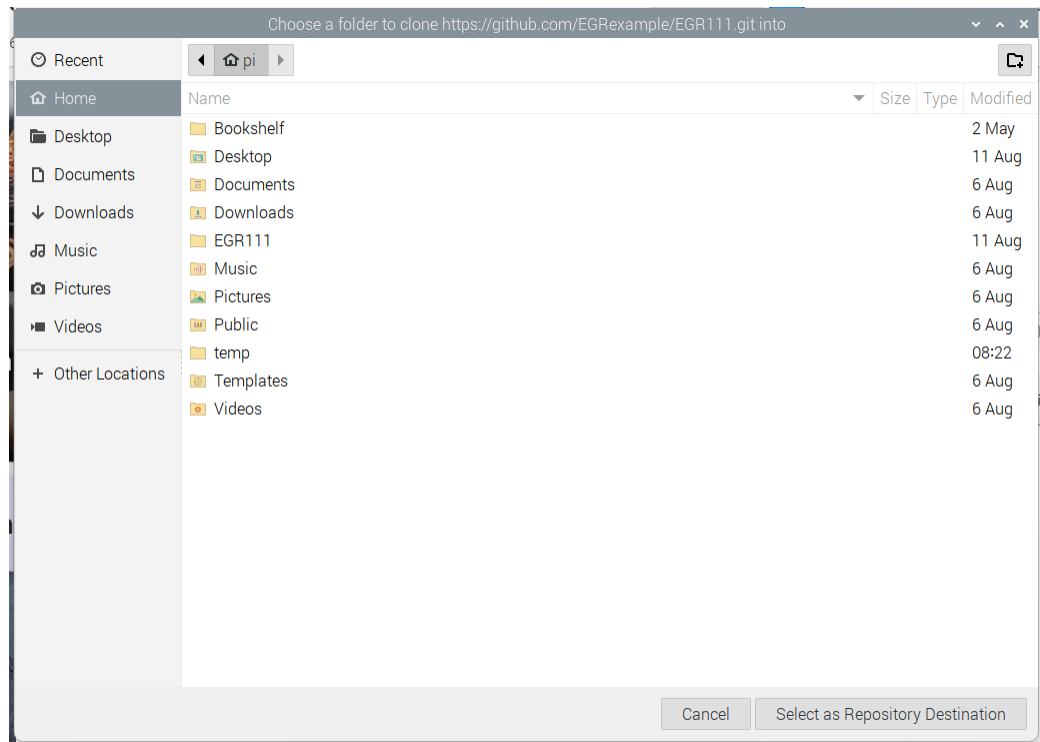14. The Welcome page is displayed. Click the "Clone Git Repository…" option.
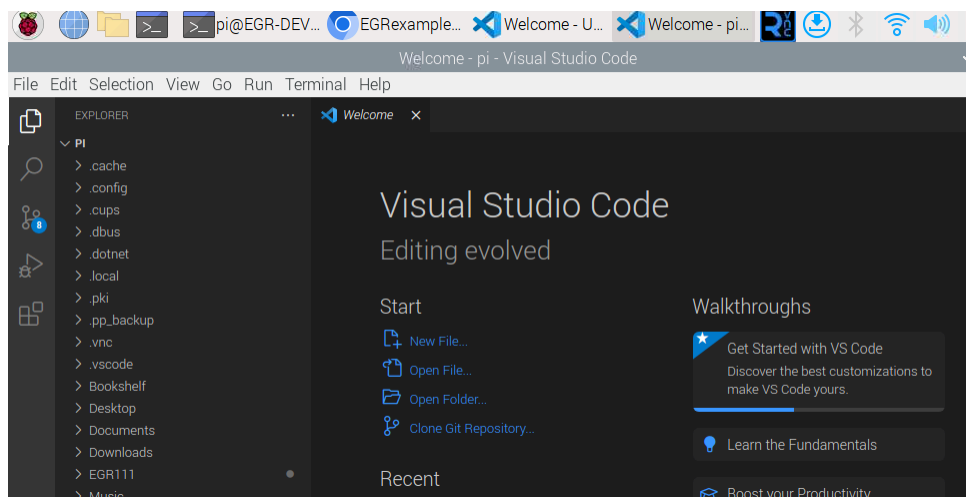
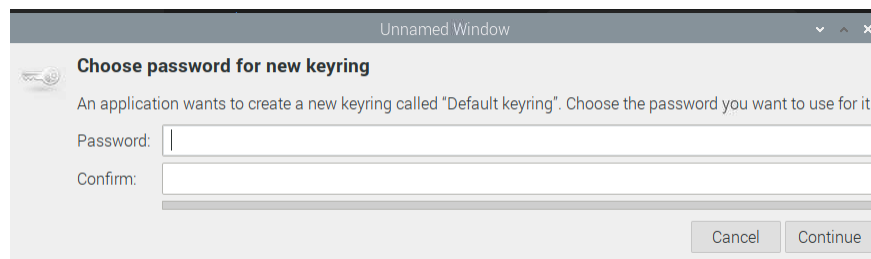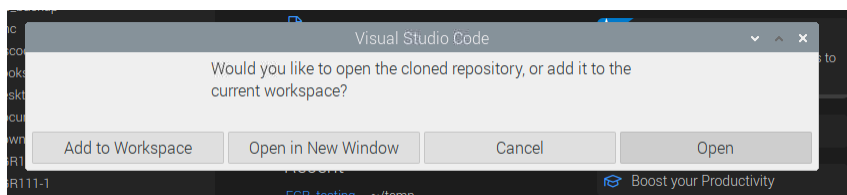15. Paste the URL into the prompt as shown below and press enter.

6

16. Select the pi directory as the Repository Destination.

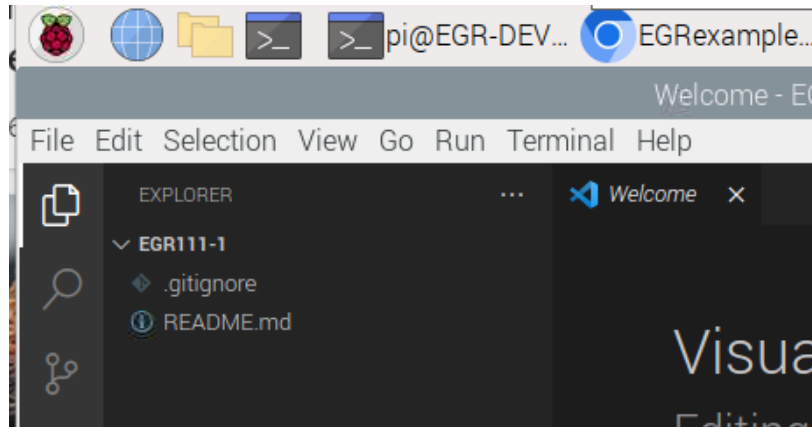17. Allow authentication and click Cancel when the new keyring prompts are displayed.

18. IMPORTANT: Click Open to open the repository. You <u>do not</u> want to Add to Workspace.

19. VS Code is now open in the EGR111 workspace. This workspace will be used for the entire semester.

20. You will access the RPi and launch VS Code in the EGR111 workspace at the start of every class. Let's practice.

    a. Close VS Code.

    b. Close the browser.

    c. Close the terminal window if open.

    d. Open a new terminal window.

    e. Move to the EGR111 directory (**cd EGR111**)

    f. Launch VS Code (**code .**)

21. Make your first commit from VS Code to the remote repo by updating your README.md file. Open README.md in the editor and enter text similar to that shown below.
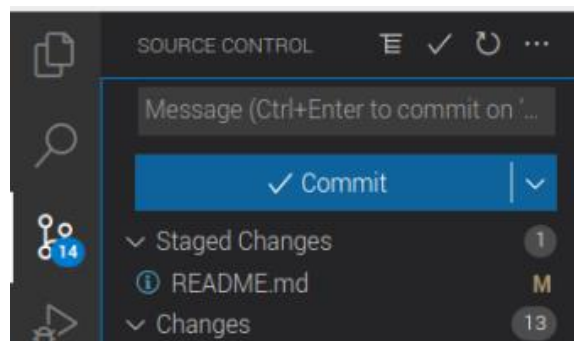

22. Save the update and close the file in the editor.

23. Click on the Source Control tool in the Activity Bar. Note the README.md file has been modified.
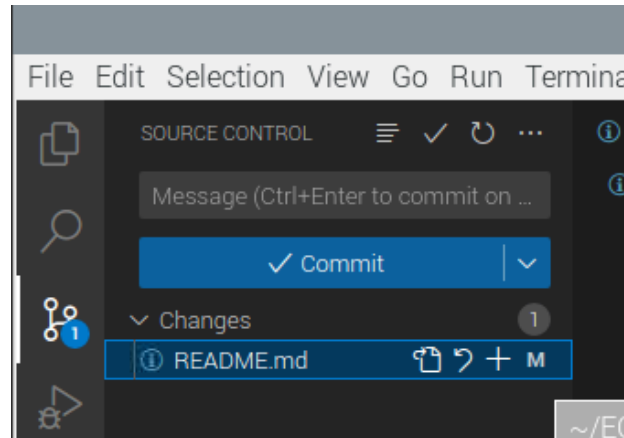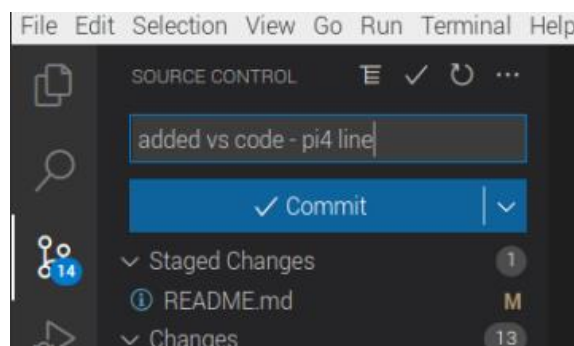
24. Click the plus sign next to README.md to stage this change.

25. The README.md file is now "staged" and is listed in the Staged Changes collection. Only files that have been staged will be committed to the repo.

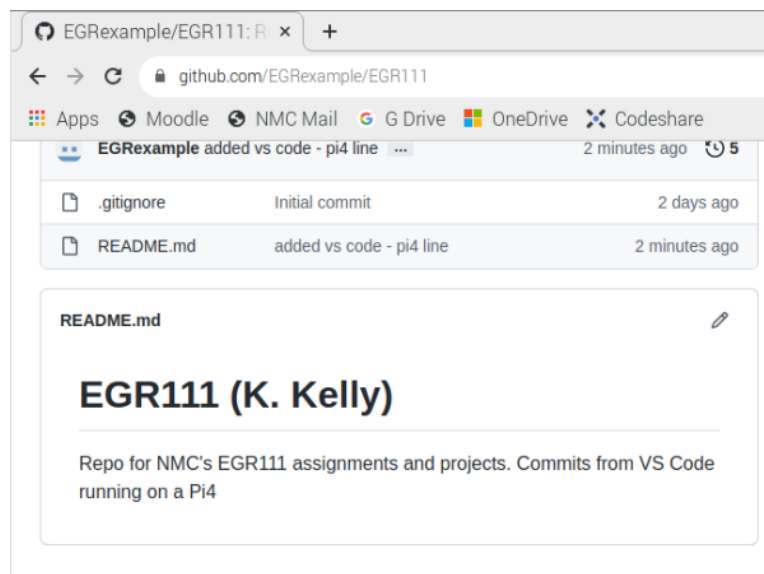26. Enter a message that identifies this commit. See the example below. Click the Commit button.

27. The change has been committed to the local repo. Locate the Publish Branch tool on the status bar. Click this and changes are sync'd with the remote GitHub repository.

28. Open your GitHub EGR111 repo in a browser. The revised README.md is displayed.

29. Recall that you created both a README.md file and a .gitignore file during the setup. Open the VS Code Explorer and verify that both of the remote files are added to your local project.

Assessments:

None but your workspace code must be pushed to your GitHub repository to enable evaluation.