**Using the command line**

1. In your command line, navigate to your project directory. Type git init to initialize the directory as a Git repository.
2. Type git remote add origin https://github.com/SunlongNgouv/github-upload.git
3. Type git add .
4. Type git commit -m "initializing repository"
5. Type git push -u origin main to push the files you have locally to the remote on GitHub. (You may be asked to log in.)

**Note:** You can also use a password protected SSH key to connect to GitHub. See [Connecting to GitHub with SSH](https://help.github.com/en/github/authenticating-to-github/connecting-to-github-with-ssh) in our documentation to learn more.

Using GitHub Desktop

**Using GitHub Desktop**

GitHub Desktop doesn't allow you to add a new remote for an existing directory, so instead we'll copy the contents of your existing folder to our repo. If you'd like to keep your existing folder, you may want to use the command line or one of the other tools.

1. In GitHub Desktop, click on **File** and **Clone a repository**.
2. Click on the **URL** tab.
3. Paste the URL from this repository.
4. Move the contents of your local repository to this directory.
5. Create a commit by entering a commit message and then clicking on **Commit to main**
6. Click **Publish branch** in the top right corner to push your repository to GitHub.

Using Visual Studio Code

**Using Visual Studio Code**

1. In Visual Studio Code, open the folder for your project.
2. Click the icon on the left for **Source Control**.
3. On the top of the Source Control panel, click the **Git icon**.
4. If the files you see match the repository you want to create, click **Initialize Repository**.
5. Next to the word **CHANGES**, click the symbol of the plus sign to stage all of the changes.
   * This is part of the two stage commit. You can use this staging function to create meaningful commits throughout the development process.
6. In the box in the Source Control panel, type a commit message. Something like "initial commit - moving project" could work.
7. Click the checkmark at the top of the Source Control panel.
8. Open the integrated terminal found under View > Integrated Terminal.
9. In your command line, type git remote add origin https://github.com/SunlongNgouv/github-upload
10. In the Source Control Panel, click the expandable three dots that open a menu of options.
11. When asked if you'd like to publish the branch, click **Okay**.

Using Atom

**Using Atom**

1. In Atom, open the folder for your project
2. At the top of your screen, click **Packages**. Select **GitHub**, and then toggle the **Git Tab** from the drop-down menu.
3. Select **Create Repository** within the Git tab on the right-hand size of your screen.
4. Select **Init** to accept the default prompt of the pop up window
5. In the Git tab, you can see that your files are ready for staging. It *should* be accounted for, but double check to make sure that none of your binaries or files that you listed in the .gitignore are listed in this dialog menu.  
   - If they are, double check your .gitignore file to make sure they're included or remove them from your directory.
6. Select **Stage All**  
   - This is part of the two stage commit. You can use this staging function to create meaningful commits throughout the development process.
7. In the box at the bottom of the Git panel, type a commit message. Something like "initial commit - moving project" could work.
8. Select **Commit**
9. Close Atom
10. In your command line, navigate to your project directory.
11. Type git remote add origin https://github.com/SunlongNgouv/github-upload
12. Return to Atom, and select the [Up/Down arrow icon](https://user-images.githubusercontent.com/13326548/36766999-34ff2bb2-1bed-11e8-90c6-3c97d0837244.png) at the bottom of your Git Tab
13. Click [Push](https://user-images.githubusercontent.com/13326548/36767211-5fd34ce6-1bee-11e8-964a-f49bed227c02.png), above the noted dialog.
14. Return to your repository, and note a successful push by finding your files on GitHub's code tab.

Using Eclipse

**Using Eclipse**

1. In Eclipse, from the Eclipse Marketplace, install the [eGit](http://www.eclipse.org/egit/) GitHub plugin.
2. Open your existing project.
3. Display the **Git Repositories** window by selecting Window > Show View > Other > Git > Git Repositories.
4. Click the **Create a Git Repository** button on the Git Repositories pane.
5. Make changes to your project and create a commit.
6. Push the changes to your default branch.
7. When asked for a remote, paste the URL you copied earlier.
8. Click next, and enter the branch name.