Plugin Based Microbiome Analysis Graphical User Interface (PluMA-GUI) Version 0 - Installation Guide

**Team Members: Rishabh Vaidya, Cesia Bulnes, Bhavyta Chauhan**

**Product Owner(s)**: Trevor Cickovski

**Mentor(s)**: Trevor Cickovski

**Instructor**: Masoud Sadjadi

***Abstract***

*This document presents the information necessary to gain a good understanding of PluMA: Plugin-based Microbiome Analysis. Our goal with PluMA is to facilitate the construction of flexible and lightweight analysis pipelines through which a developer can implement a new algorithm in their programming language of choice, and easily test and debug within a larger pipeline alongside stages in different languages that potentially use different file formats.*

*PluMA accomplishes this through plugins, and has a large collection available in its plugin pool, implemented in various programming languages for both the CPU and GPU. Plugins can be run sequentially to form a pipeline, and can be easily added, removed or substituted through our user interface. Since plugins are dynamically loaded, you can assemble a pipeline by downloading PluMA and only the plugins that you need from the pool.*

*Currently, PluMA can be intricate for those who don’t have coding experience or have never used the terminal. With the GUI, users can create the same pipeline formation and install plugins with a very easy user experience that does not require any coding whatsoever.*

**Installing Electron:**

**How to install npm and Node.js:**

To download Node.js which includes npm, the user must navigate to:

<https://nodejs.org/en/download/>

If the user wants the latest version of npm, then he/she must run this command:

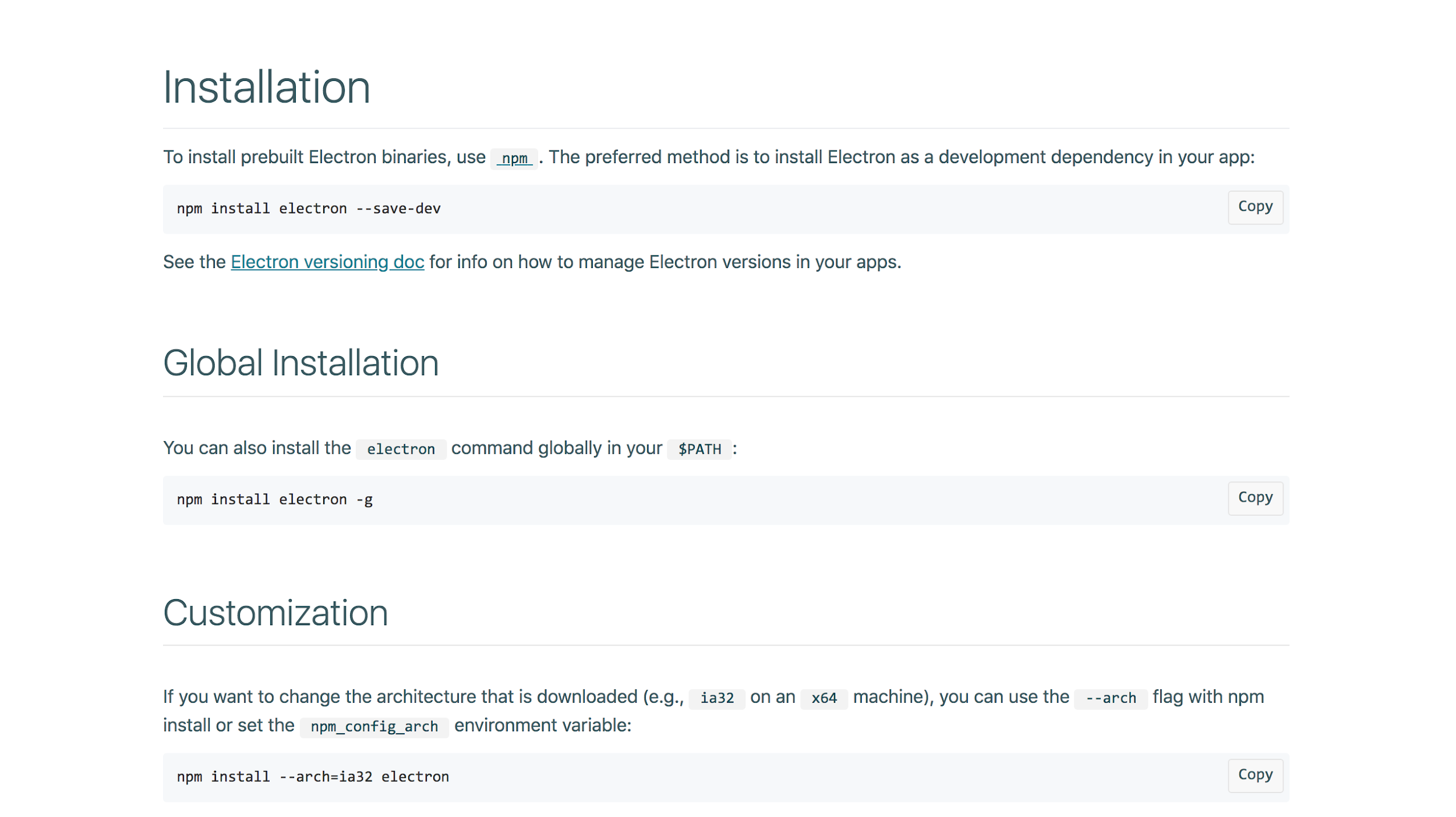
npm install npm@latest -g

**How to install Electron:**

In order to install Electron, the user must navigate to the Electron installation page:

<https://electronjs.org/docs/tutorial/installation>

The page has instructions on how to install the application in three configurations:



Once, this step is complete, the user can move on to downloading PluMA.

**Downloading PluMA:**

**How to download PluMA:**

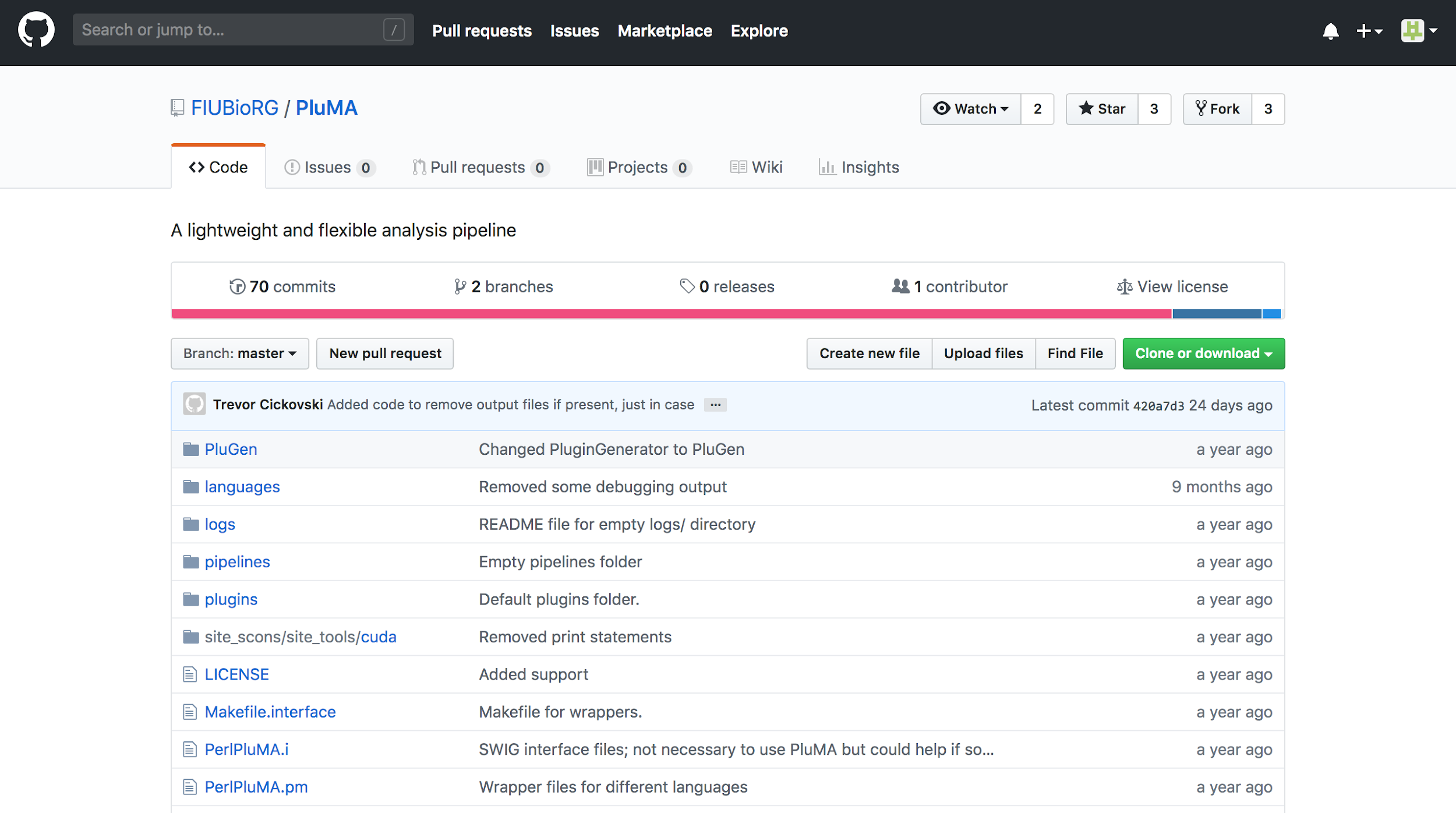
PLUMA is hosted by the Bioinformatics Research Group (BioRG) at Florida International University at <http://biorg.cis.fiu.edu/pluma>, with source code now available through GitHub:

<https://github.com/FIUBioRG/PluMA>

PLUMA users must agree to MIT software license regulations, part of the Open Source Initiative (OSI). We provide a copy of this license at the end of this User Guide. Finally, we make available a plugin pool of dynamically loadable plugin libraries implemented in various languages at:

<http://biorg.cs.fiu.edu/pluma/plugins>

which contains links to repositories of various PLUMA plugins developed by external users. Pipelines can be assembled using combinations of these plugin extensions.



In order to compile PluMA, the user can refer to the BioRG PluMA guide:

<http://biorg.cis.fiu.edu/pluma/userguide.pdf>

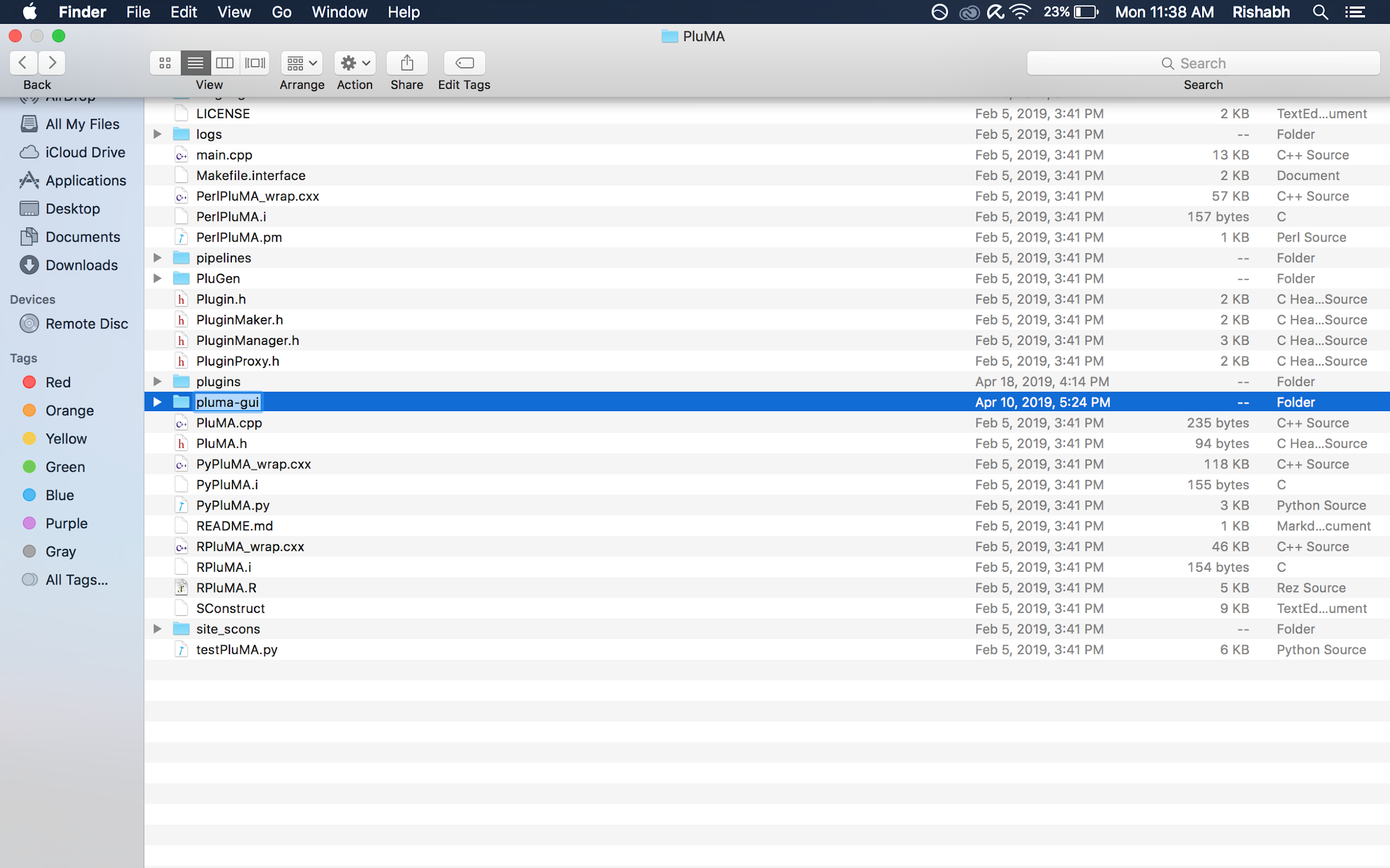
**Downloading PluMA-GUI:**

**How to download PluMA-GUI:**

The PluMA-GUI is a desktop application that provides an enhanced user friendly experience for using the PluMA software. The source code for the PluMA-GUI desktop application is available through BitBucket:

<https://sp-jira.cis.fiu.edu:7443/projects/PLMA/repos/pluma-gui/browse>

**Note: This project folder must be placed inside the main PluMA directory, so that the application can access the installed plugins.**



Once the gui is downloaded, the application can be run by navigating to the pluma-gui directory and typing in the command “electron . ”

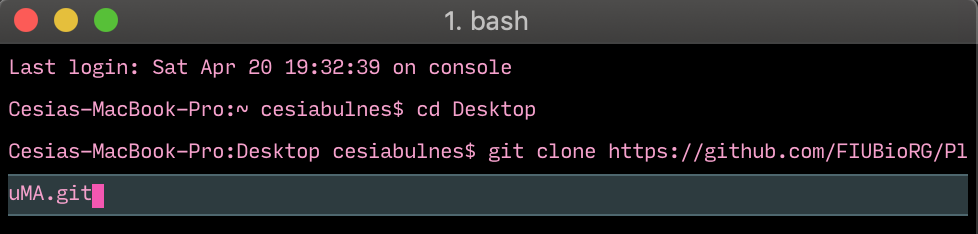
**Installing Dependencies:**

Before we begin, I want to make sure you have git installed. Please refer to this [link](https://git-scm.com/book/en/v2/Getting-Started-Installing-Git) to install git on your terminal.

In the command line, we will be installing the PluMA project under the Desktop folder.

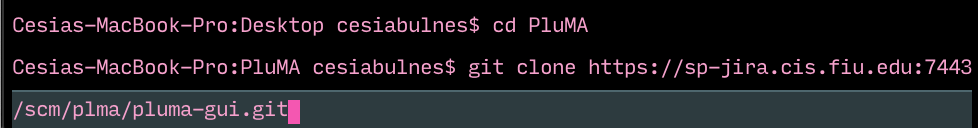
The following command will be used to install

git clone https://github.com/FIUBioRG/PluMA.git



Inside the PluMA folder we will be installing our project pluma-gui

git clone https://sp-jira.cis.fiu.edu:7443/scm/plma/pluma-gui.git

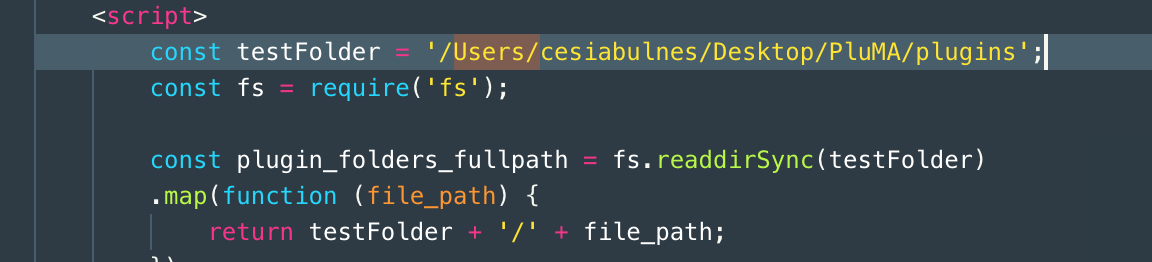


From there, we will be installing any dependencies you may not currently have, such as [electron](https://electronjs.org/docs/tutorial/installation), [cheerio](https://cheerio.js.org), [node](https://docs.npmjs.com/downloading-and-installing-node-js-and-npm), etc. If there are any dependencies you may not have, you can also do npm install -g.

Before launching the application this step MUST BE DONE IN ORDER FOR IT TO WORK. The team had trouble in storing information with our different routes to where PluMA and pluma-gui was stored.

In readmeparser.js, please edit the first line, and replace with your user name after /Users/<your user name>/Desktop/PluMA/plugins

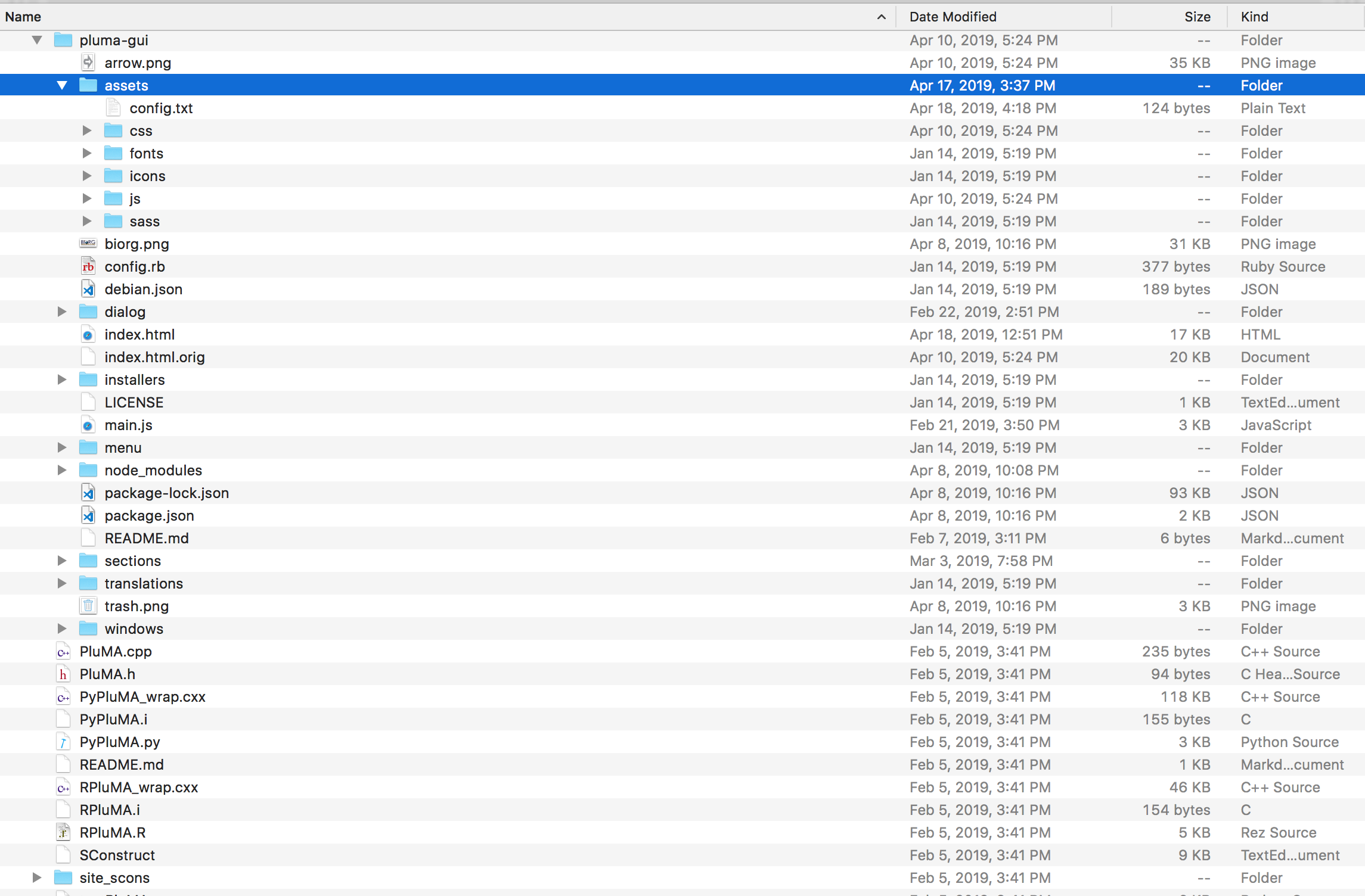


Please do the same steps in index.html, in lines 60,211 & 255



**Files To Be Modified:**

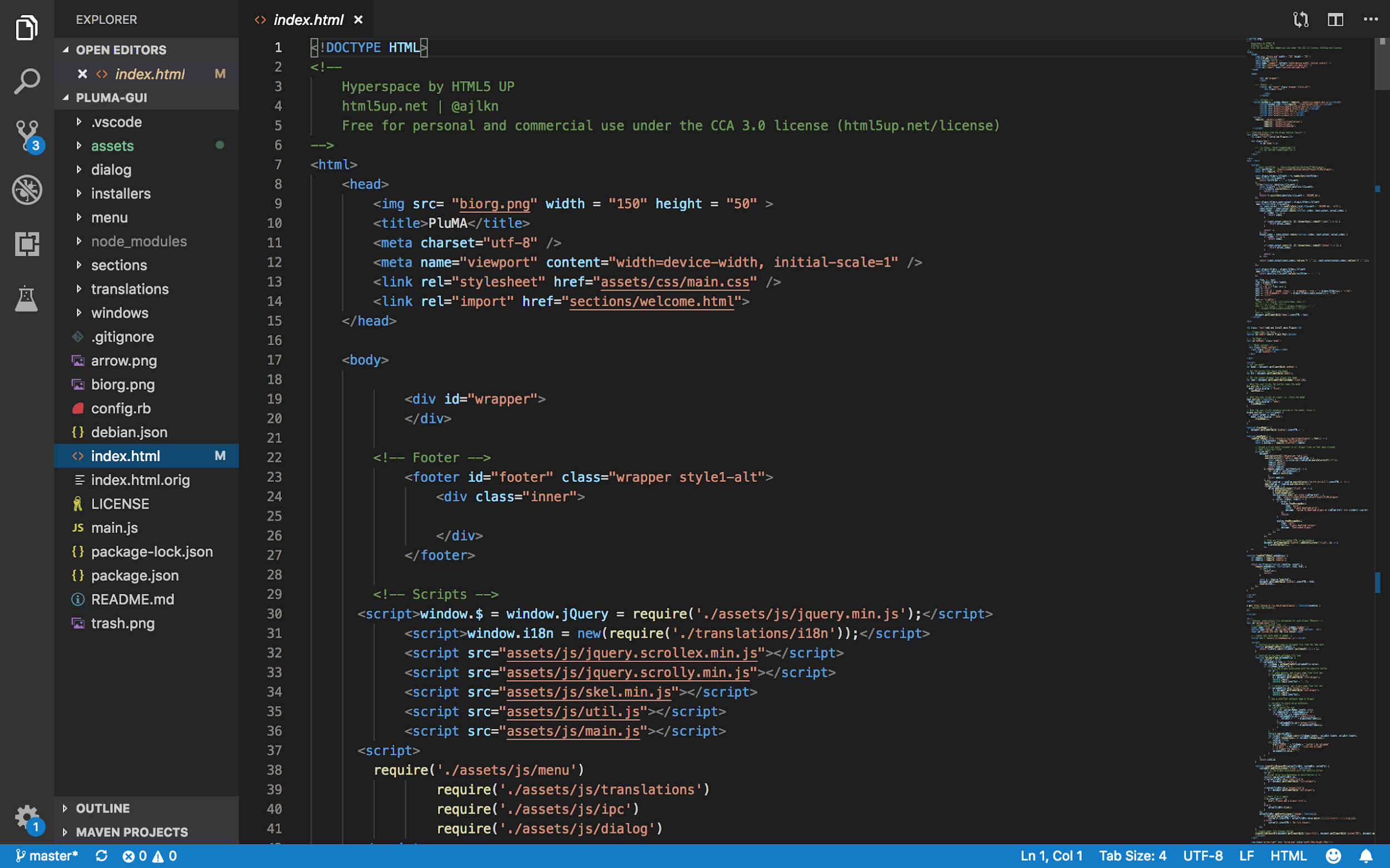
In the pluma-gui directory, the main source code file is index.html. If the developer wishes to make any changes to the code, he/she can add/remove code from the index.html file. Along with this file, there are css and js files that can also be modified in the assets directory within pluma-gui.



The necessary files along with index.html are located in the assets folder. The assets folder includes the css files, the js files, the config.txt file and more.

Other than the assets folder, the various other folders are for building the application and running it. Some of the folders contain code for creating more windows but these are blank as of now. The developer can wish to add/modify those windows.

**Source Code(index.html):**

****