

 UserGuide.md

Visual Git - User Guide

VisualGit is an easy-to-use, visually-oriented desktop client for Git aimed at helping students learn the standard Git workflow. The following details steps for installation, and a rundown of the basic features of VisualGit. This guide assumes you have a basic understanding of Git and its workflow. More information can be found [here](#).

Origins

VisualGit was brought to life by Elliot Whiley and Harvey Rendell. For a summary of their project, refer to their [cool poster](#). You can find their original repo [here](#).

Installation

Prerequisites

npm (Node Package Manager) is used to manage VisualGit's dependencies, therefore it is required to install and run VisualGit. Follow the installation instructions below:

Centos-based systems

```
sudo yum install npm
```

Debian-based systems

```
sudo apt-get install npm
```

Mac

If you have homebrew installed:

```
brew install npm
```

Otherwise download and install version 10.15.3 of [Node.js](#). It comes with npm version 6.4.1. Upgrade this to npm 6.9.0 and install important global packages by running the commands below:

```
sudo npm install -g npm@6.9.0
sudo npm install -g electron node-pre-gyp
```

Windows

Download and install the latest version 10.15.3 of [Node.js](#). It comes with npm version 6.4.1. Upgrade this to npm 6.9.0 and install important global packages by running the commands below:

```
npm install -g npm@6.9.0
npm install -g electron node-pre-gyp
```

Type "\$PSHome" in a PowerShell window to get your PowerShell path. Add the PowerShell path to your PATH system environment variable. then...

```
npm install windows-build-tools -g
```

Ubuntu

Install npm and node

```
sudo apt install npm
sudo apt install nodejs-legacy
```

Upgrade node to 10.15.3 and npm to 6.9.0

```
sudo npm install -g n
sudo n 10.15.3
sudo npm install npm@6.9.0 -g
```

Install important global packages

```
sudo npm install -g electron node-pre-gyp --unsafe-perm=true --allow-root
sudo apt install libcurl4-gnutls-dev
```

change the permission and verify cache

```
sudo chmod +777 -R $HOME/
sudo npm cache verify
```

Setup

Please ensure that the path to your repository folder has no spaces in it before you clone to avoid errors.

Clone with either HTTPS or SSH:

HTTPS

```
git clone https://github.com/kblincoe/VisualGit_SE701_2019_2.git
```

SSH

```
git clone git@github.com:kblincoe/VisualGit_SE701_2019_2.git
```

then...

```
cd VisualGit_SE701_2019_2
npm install
npm start
```

You can log in to VisualGit using your GitHub credentials.

Features

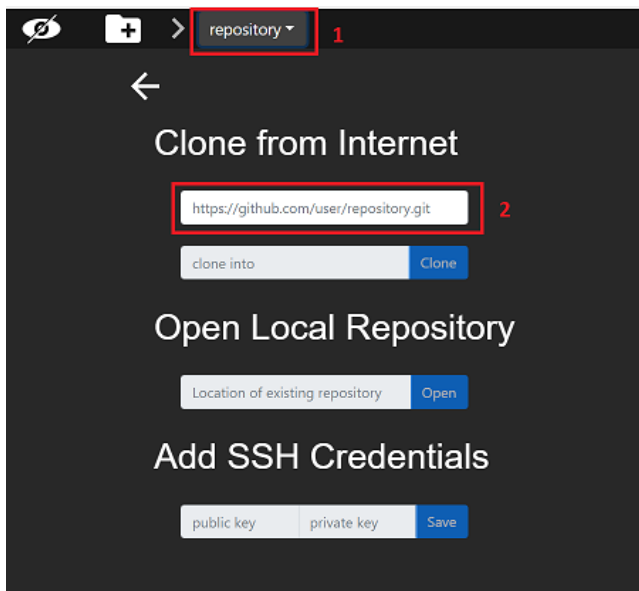
Opening / Cloning repositories

Repositories can be added by two methods; either by cloning the remotely hosted repository or opening it from the local file system.

Clone

To clone a remote repository, you will need the git repository address. You can enter this in two ways, as shown in the image below:

1. You can select your repository from the drop-down menu at the top of the screen. This will automatically fill in the git repository address.
2. You can manually type the git repository address.



Once the git repository address has been entered, select the location you wish to clone your repository to on your machine, and click clone.

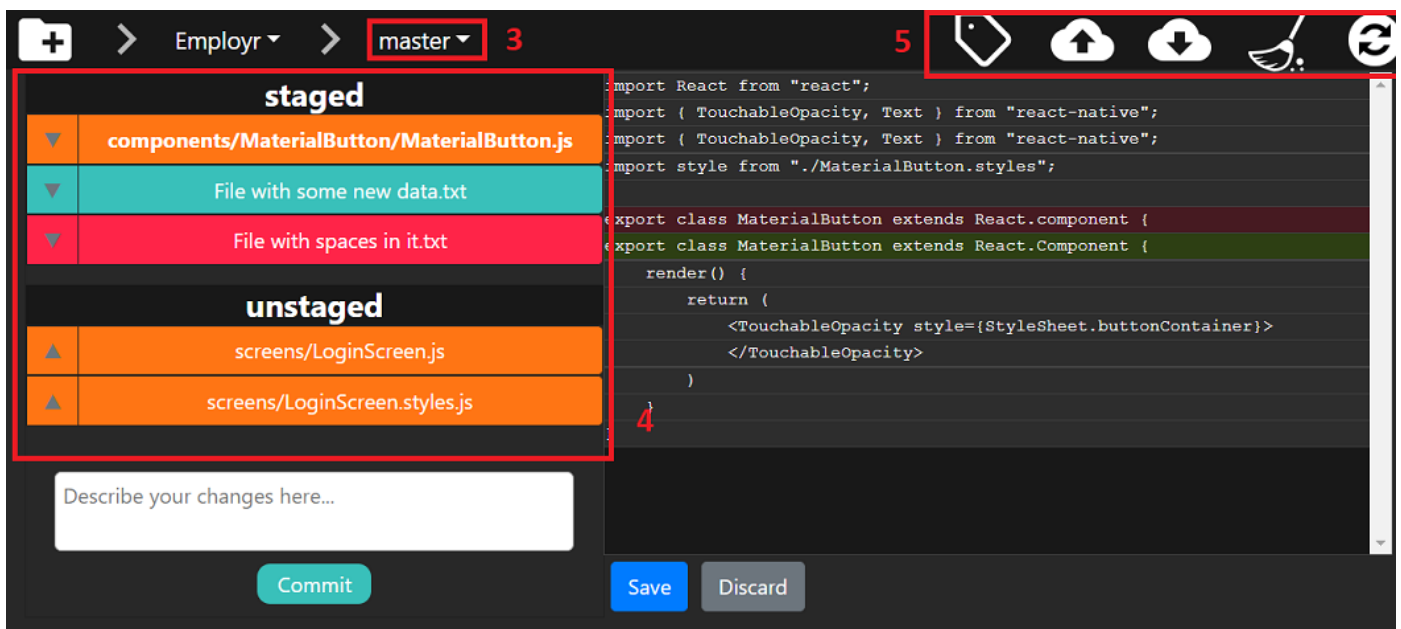
Open local repository

You can open a locally saved repository by clicking on the field 'Location of existing repository'. Navigate to the directory you wish to open, then click 'open'.

Adding SSH credentials

SSH credentials can also be added into VisualGit. To add your keys to VisualGit, click on the field noting the public key, and navigate to the directory in which your public key is located. The program will automatically detect the public/private key pairing. Once that is done, click 'Save'.

Adding & Committing



When changes are made in a repository which is open, the interface will update by showing each file which has uncommitted changes (4). These changes will be displayed with the following colour scheme:

- Teal: Added file
- Orange: Modified file
- Red: Deleted file

This is used to allow users to see the different types of changes easily and once they click on the files, the file differences will be shown. The file differences are shown line by line with green lines representing added lines and red representing deleted lines. Any other parts of the file are unchanged.

It is also possible to switch to another branch using the branch selector, located at the top of the application (3).

Pulling, Pushing, Tagging, and Other features

There are several useful Git functions which are available in the toolbar at the top right of the application (5). These buttons correspond to pulling, pushing, tagging, cleaning, and sync. More information about some of these functions can be found [here](#).