**Git Quick Guide**

**git checkout -b <branch name>** performed once to create a development branch

**git status** to verify the changes were made to the file.

**git add -A** to stage all changes to be committed.

**git commit -m "<Your commit message here>"** to commit your changes on the develop branch.

**git push origin <branch name>** to push the changes back to the GitHub server

i. After branch is pushed, create a pull request to get it merged into Master

Submit a git pull request from the online repository

**git pull origin master** to pull the latest changes from master onto your branch

**Windows Git: (Configure your git local client)**

GitHub associates commits to GitHub accounts using the email address used to sign the commit. This step is required to correctly link commits to your GitHub account!

Run the following commands to sign your commits with your name & email address:

git config --global user.name "<yourName>"

git config --global user.email "<yourEmailAddress>”

**Git Configurations:**

These settings are valid for both Windows and Linux git installations. They will apply across all repositories accessed by the local client. For Windows installations, these settings will only apply to the workstation git was installed on. For Linux installations, these settings are stored in your home directory and apply to any Linux installation that mounts your home directory.

Linux config paths:

~/.ssh

~/.gitconfig

Windows config paths:

C:/Users/<username>/.ssh

C:/Users/<username>/.gitconfig

**Use SSH keys to authenticate into GitHub.**

Check for existing SSH keys:

Open Git Bash.

Run the following command: ls -al ~/.ssh

Check the directory listing to see if you have a public SSH key. This is typically: id\_rsa.pub

**If no SSH keys exist, generate a new SSH key:**

Open Git Bash.

Run the following command: ssh-keygen -t rsa -b 4096 -C "<yourEmailAddress>"

Press "Enter" to accept the default file location.

Enter a passphrase for the SSH key.

**Add public SSH key to GitHub:**

Open Git Bash.

Run the following command: clip < ~/.ssh/id\_rsa.pub

Login to GitHub using your SSO credentials.

In the upper right-hand corner, click your profile picture and in the drop down menu click "Settings"

In the left sidebar, click "SSH and GPG keys"

Click "New SSH key"

Enter a title for the key, then paste the SSH key you copied to the clipboard into the "Key" field.

Click "Add SSH key"

**Use HTTPS with Personal Access Token (PAT) to authenticate to GitHub.**

Manage your PATs <https://github.com/settings/tokens>

Click "Generate new token"

Name the token & grant "repo" scope.

Access repositories using the following syntax: https://<username>:<PAT>@<GitHubBaseURL>/<RepoOwner>/<Repo>

**Initial Clone of Repo:**

create a ssh key and upload to GitHub.com first

navigate to /.ssh folder in git-bash

git clone https://<username>:<pat>@<github repo>.git

**LINUX Git**

Install Git for Linux

Configure your git local client.

GitHub associates commits to GitHub accounts using the email address used to sign the commit. This step is required to correctly link commits to your GitHub account!

Run the following commands to sign your commits with your name & email address:

git config --global user.name "<yourName>"

git config --global user.email "<yourEmailAddress>"

**Linux config paths:**

~/.ssh

~/.gitconfig

Use SSH keys to authenticate into GitHub.

**Check for existing SSH keys:**

Open Git Bash.

Run the following command: ls -al ~/.ssh

Check the directory listing to see if you have a public SSH key. This is typically: id\_rsa.pub

**If no SSH keys exist, generate a new SSH key:**

Run the following command: ssh-keygen -t rsa -b 4096 -C "<yourEmailAddress>"

Press "Enter" to accept the default file location.

Enter a passphrase for the SSH key.

**Add public SSH key to GitHub:**

Run the following command: clip < ~/.ssh/id\_rsa.pub

Login to GitHub using your SSO credentials.

In the upper right-hand corner, click your profile picture and in the drop down menu click "Settings"

In the left sidebar, click "SSH and GPG keys"

Click "New SSH key"

Enter a title for the key, then paste the SSH key you copied to the clipboard into the "Key" field.

Click "Add SSH key"

**Use HTTPS with Personal Access Token (PAT) to authenticate to GitHub.**

Manage your PATs here.

Click "Generate new token"

Name the token & grant "repo" scope.

Access repositories using the following syntax: https://<username>:<PAT>@<GitHubBaseURL>/<RepoOwner>/<Repo>