

SEFSC GitHub

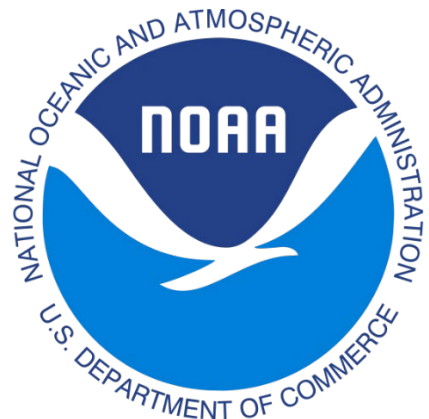
Standard Operating Procedure (SOP)

December 2024

National Oceanic and Atmospheric Administration (NOAA)

Southeast Fisheries Science Center (SEFSC)

Operations and Management Division (OMI)





Document History/Revisions

Release No.	Date	Author	Revision Description
1.0	10/20/2020	David Chevrier	Document Creation
1.1	1/2/2021	David Chevrier	Added clarification to Git-Secrets requirement and updated GitHub Owners' responsibilities
1.2	2/2/2021	David Chevrier	Added information about automatic URL redirects for repositories and GitHub Pages as well as information about the Pricing Plan for the SEFSC organization
1.3	5/11/2021	David Chevrier	Added SEFSC GitHub User Agreement Requirement
1.4	5/15/2021	James Primrose	SE Adopted; links modified
1.5	6/15/2021	James Primrose	GitKraken Configuration
1.6	6/22/2021	Molly Stevens	Git / R-Studio / RMarkdown Configuration
1.7	12/10/2024	James Houston	Updated contact information. Updated links. Updated screenshots. Included GCP backup information.



Approval Signatures

Name	Role	Signature
Luis Noguero	ISSO	
Richard Rasch	SO	
Braydon Mikesell	AO	



Table of Contents

Document History/Revisions	1
Approval Signatures	2
Introduction	4
Roles and Responsibilities	5
General SOP	6
First Steps	6
Local Git Installation	6
GitHub Account Creation	7
SEFSC Repository Naming Convention	8
Previously Hosted GitHub Repositories	8
Creating a New GitHub Repository	8
Update Repository's Remote URL	11
Collaboration	12
Security	12
Overall Policy	12
Acceptable Content	12
Git-Secrets Requirement (gitleaks)	13
NOAA Requirements	14
README File	14
License File	14
Repository Deletion	14
Auto-redirect GitHub Repositories and Pages	14
SEFSC GitHub Plan Information	16
Additional Information	16
NOAA Policies	16
SVN Migration and Reference Sheet	16
Git Reference	16
Gold Copy	16
GitKraken	16
Git / R-Studio / RMarkdown	16
Questions or Concerns	16



Please read this entire document and contact the [SEFSC GitHub Admins](#) with any questions.

Introduction

[Git](#) is a distributed version-control system for tracking changes in any set of files, originally designed for coordinating work among programmers cooperating on source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows.

As with most other distributed version-control systems, every Git directory on every computer is a full-fledged repository with complete history and full version-tracking abilities, independent of network access or a central server.

[GitHub](#) provides hosting for software development and version control using Git. It offers the distributed version control and source code management (SCM) functionality of Git, plus its own features. It provides access control and several collaboration features such as bug tracking, feature requests, task management, continuous integration and wikis for every project. It is the largest host of source code in the world.

The SEFSC greatly benefits from having the ability to collaborate, share, backup, and utilize version control on any project.



Roles and Responsibilities

Role	Responsibilities	Produced Resource
GitHub Admins	Administrates all repositories and users on GitHub under the SEFSC organization. Handles all sharing and permissions. Receives security scan reports and works with users to correct any issues. Considers users feedback and modifies SOP if necessary, while adhering to NOAA policies.	Created SEFSC GitHub Admins email group.
GitHub Users	Users of the SEFSC Repositories are agreeing to abide by and follow these procedures	Created GitHub Users email group



General SOP

Git is an officially supported version control software for the SEFSC. GitHub is the online service to store repositories created with Git which allows for collaboration and sharing of files. All government work performed at/for the SEFSC and stored on GitHub must follow this SOP. No private work is to be stored as part of your government account or linked to the SEFSC organization on GitHub. No government repositories may be stored in a personal GitHub account. NOAA's official GitHub policy may be found [here](#).

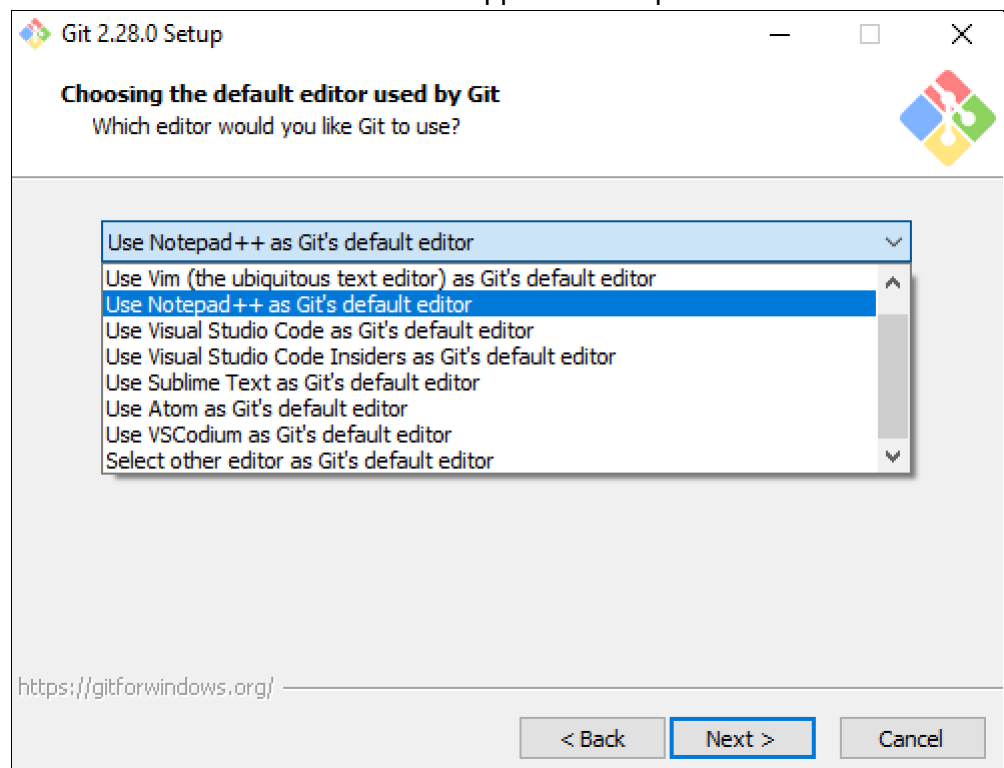
All projects that are to be launched in a SEFSC production environment must be using a software versioning (version control) system. Git together with GitHub is an approved SEFSC option for that version control requirement. Users of GitHub agree to follow all the procedures outlined in this document.

First Steps

Local Git Installation

Submit a ticket to the [SEFSC OMI Service Portal](#) with the following information:

- Request Git installation on your PC.
- Indicate the preferred default text editor for Git.
 - The current available options are shown in the screenshot below.
 - Note that this software must be approved and pre-installed.



- Request Git Desktop installation, if needed.
 - The Git installation includes Git Bash.



GitHub Account Creation

Sign up for a [GitHub](#) account using your NOAA.gov email address. **Note:** **Personal accounts are not allowed to store government work.**

Your username should include **-NOAA** at the end of it to clearly delineate that you are developing software on behalf of NOAA.

Example:

JaneSmith-NOAA
JSmith-NOAA

[Two factor authentication](#) must be enabled before being added to the SEFSC organization.

Fill out the [SEFSC GitHub User Agreement](#). Once you and your supervisor have signed it, submit it to the [SEFSC OMI Service Portal](#) for the SEFSC ISSO to review/approve. Once the ISSO has approved the request and you have set up two factor authentication, your account will be added to the [GitHub Users](#) email list and the SEFSC organization.

SEFSC Repository Naming Convention

In order to maintain consistency and clarity, we are recommending that all SEFSC repositories follow the same naming convention. The name of your repository should be:

Division Name (abbreviation) - Branch Name (abbreviation) - ProjectName. The division and branch names are from the point of view of the program that uses the product (Product Owner). "For example, our Priority Based Resourcing application would be SEFSC-OMI-PBR". The Branch name may be omitted in the case of a product being used across an entire division.

If using a GitHub repository for research paper publications, include your username (minus any numbers). A paper written by David Chevrier would be named OMI-DDB-DCHEVRIER-MySecretPaper. You may omit the branch abbreviation if you are a Division Chief or report directly to one.

Previously Hosted GitHub Repositories

If you have previously created repositories on GitHub which contain government work, you will need to migrate them into the SEFSC organizational account.

- Open your repository.
- Go to Settings, then Options.



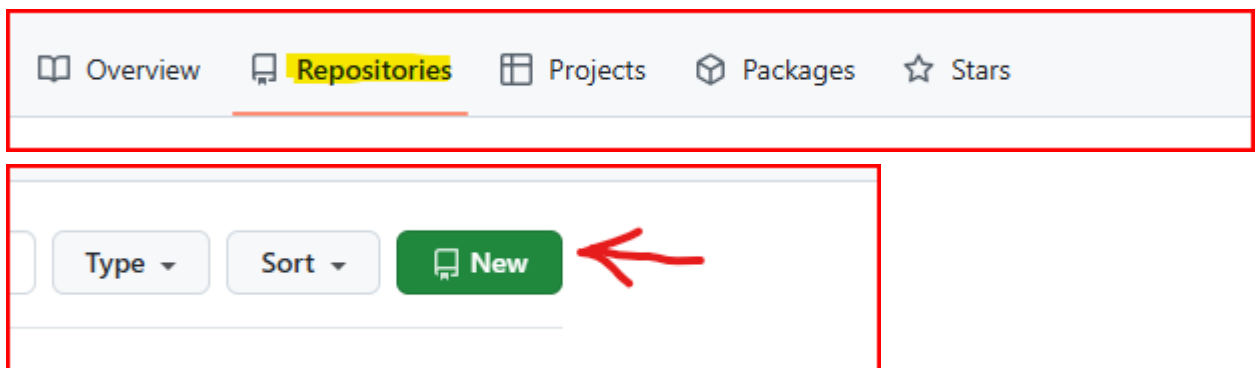
- Click Transfer at the bottom of the page.
- Type SEFSC-SVC-GitHub.
- All of your settings will remain, including outside collaborators.

[SEFSC GitHub Admins](#) receive notification and will examine the repository to ensure it meets NOAA security and SEFSC configuration requirements. If you are experiencing issues transferring the GitHub repository to the SEFSC organization. Please submit a help ticket to the [SEFSC OMI Service Portal](#).

There may be reasons that can prevent the repository move to the SEFSC organization and GitHub Admins can discuss solutions for your specific circumstance.

Creating a New GitHub Repository

1. Create a new **private** repository on GitHub in your government account



2. Select the **SEFSC/SEFSC-Template** under [Repository template](#). This will automatically setup your Readme file, License file, and security scan code as defined in this document. Leave "Include all branches" unchecked.




Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk ().*


Repository template

 SEFSC/SEFSC-Template ▾

Start your repository with a template repository's contents.

☐ Include all branches
Copy all branches from SEFSC/SEFSC-Template and not just the default branch.

Owner * **Repository name ***

 SEFSC ▾

 /


RepoName-Here


✔ RepoName-Here is available.


Great repository names are short and memorable. Need inspiration? How about [psychic-broccoli](#) ?


Description (optional)

Optional description here

☐  **Public**
Anyone on the internet can see this repository. You choose who can commit.

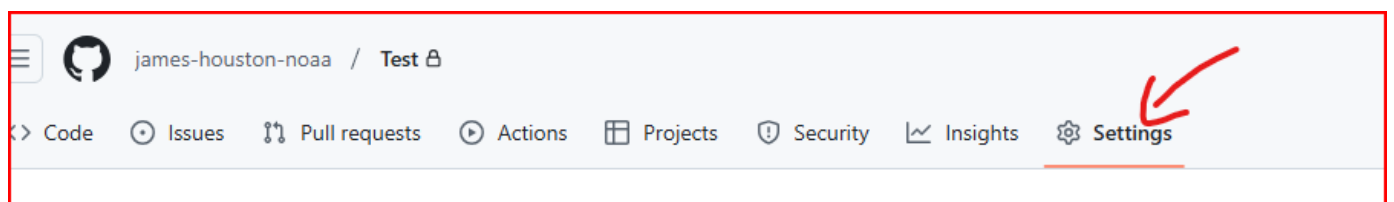
☐  **Internal**
NOAA-NMFS [enterprise members](#) can see this repository. You can choose who can commit.

☒  **Private**
You choose who can see and commit to this repository.

 You are creating a private repository in the SEFSC organization (NOAA-NMFS).

Create repository

3. Open the new repository and go to Settings, then Options





4. Scroll to the bottom and click “Transfer ownership”
5. For the new owner, select or type **SEFSC**

Transfer repository: james-houston-noaa/Test

Transfer this repository to another user or to an organization where you have the ability to create repositories.


Required fields are marked with an asterisk ().*

⚠ To understand admin access, teams, issue assignments, and redirects after a repository is transferred, see [Transferring a repository](#) in GitHub Help.

Transferring may be delayed until the new owner approves the transfer.

New owner *

☒ Select one of my organizations

 SEFSC ▾

☐ Specify an organization or username

Repository name *

Test

✔ Test is available.

Type james-houston-noaa/Test to confirm. *

james-houston-noaa/Test

I understand, transfer this repository.

6. Submit a help ticket to the [SEFSC OMI Service Portal](#). We will review the repository to make sure it follows our SOP and add it to the SEFSC organization account.

Update Repository's Remote URL

After a repository is moved into the SEFSC organization, make sure to update your remote url with the following command:

```
git remote set-url origin https://github.com/SEFSC/REPOSITORY.git
```



If the repository was configured as ssh instead of https, use this command:

```
git remote set-url origin git@github.com:SEFSC/REPOSITORY.git
```

Collaboration

To add any internal or external members as a collaborator on your repository, please submit a help ticket to the [SEFSC OMI Service Portal](#). A “team” (group of GitHub users) can be set up if you are working with the same members often. Include the role you want them to have. Available options are: Read, Triage, or Write. (Maintain and Admin are reserved for SEFSC owners only.) NOAA rules state that only NOAA Federal and Affiliated developers can commit directly to a SEFSC repository (they must have an NOAA.gov email address). Non-NOAA collaborators must use pull-request for collaboration and can only be assigned a “Read” role.

Security

Overall Policy

No usernames, passwords, login information, port numbers, IP addresses, server names, Application Programming Interface (API) keys, Personally Identifiable Information (PII), Business Identifiable Information (BII), or confidential data may be stored in any file hosted on GitHub. If writing software, add all of those values to a single file and use the “[gitignore](#)” command to prevent Git from storing it. That file will have to be copied over to production if needed. ***This applies even if that code will only ever be in a private repository.*** Make sure to use the “gitignore” command before your first commit or Git will not ignore the file and you will have to untrack it. More information can be found [here](#).

Using [Encrypted Secrets](#) inside of GitHub is also acceptable.

Acceptable Content

[NOAA GitHub Guidelines](#):

“The following are the requirements that must be met before code, data, or documentation produced by NOAA staff can be released to GitHub.com:

- **Scientific Products:** The code, data, or documentation must be a scientific product, defined by the NOAA Scientific Integrity Policy (NAO 202-735D) as “Presentation of the results of scientific activities including the analysis, synthesis, compilation, or translation of scientific information and data into formats for the use of NOAA, the Department of Commerce, or the Nation.”
- **Associated Product Risk:** The scientific product must be reasonably classifiable as FISMA Low, as outlined by the Federal Information Security Management Act of 2002. FISMA Low classification includes only information for which the unauthorized disclosure, unauthorized modification, unauthorized destruction, or disruption of access can be expected to have a limited adverse



effect on organizational operations, organizational assets, or individuals. If the effect of such events would be serious, severe, or catastrophic, the information cannot be released under the authority of this memo.

• **No Controlled Unclassified Information (CUI):** The information being released must not contain any sensitive data or information. NOAA staff must confirm that there are no account passwords, user logins, API keys, or other restricted information contained in the scientific product prior to releasing it to GitHub.com, through the use of the git-secrets scanning tool. Additionally, the information owner must confirm that there is no Personally Identifiable Information (PII) or Business Identifiable Information (BII) included with the content, nor any information that would allow access to PII or BII, they must also confirm that the data is not a CUI subcategory as listed at the following link: <https://www.archives.gov/cui> ”

Git-Secrets Requirement (gitleaks)

The NOAA requirement to use git-secrets will be replaced with an improved program called gitleaks. This provides an easy means of automating a security scan by scanning for API keys and other credentials accidentally left in the code. The scan will be automatically run using a “GitHub Action” every time a “push” or “pull request” is completed. Gitleaks is **required** using the following method:

- Create a folder named “.github” in the main directory of your local repository, at the same level as your “.git” folder. Note the period at the start of the folder name.
- Inside of the “.github” folder, create another new folder named “workflows”.
- In the “workflows” folder, add a file named “secretScan.yml”
- Copy and paste the following code to that file, making sure to keep the spaces exactly as typed below:

```
name: gitleaks

on: [push,pull_request]

jobs:
  gitleaks:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v2
        with:
          fetch-depth: '0'
      - name: gitleaks-action
        uses: zricethezav/gitleaks-action@master
```

Note: If the SEFSC official template was used to create a repository, this file was already created for you. Adding this code is only needed for existing repositories that will be added to our organization.



NOAA Requirements

NOAA requires the following two files to be added to every repository hosted on GitHub:

README File

A ReadMe file must be included with every repository with the following text:

"This repository is a scientific product and is not official communication of the National Oceanic and Atmospheric Administration, or the United States Department of Commerce. All NOAA GitHub project code is provided on an 'as is' basis and the user assumes responsibility for its use. Any claims against the Department of Commerce or Department of Commerce bureaus stemming from the use of this GitHub project will be governed by all applicable Federal law. Any reference to specific commercial products, processes, or services by service mark, trademark, manufacturer, or otherwise, does not constitute or imply their endorsement, recommendation or favoring by the Department of Commerce. The Department of Commerce seal and logo, or the seal and logo of a DOC bureau, shall not be used in any manner to imply endorsement of any commercial product or activity by DOC or the United States Government."

You should also include additional information relevant to the work by answering these questions: who worked on this project, when this project was created, what the project does, why the project is useful, how users can get started with the project, where users can get help with your project, and who maintains and contributes to the project.

License File

A license file is required by NOAA and must be included in the repository with the following text:

"Software code created by U.S. Government employees is not subject to copyright in the United States (17 U.S.C. §105). The United States/Department of Commerce reserve all rights to seek and obtain copyright protection in countries other than the United States for Software authored in its entirety by the Department of Commerce. To this end, the Department of Commerce hereby grants to Recipient a royalty-free, nonexclusive license to use, copy, and create derivative works of the Software outside of the United States."

Repository Deletion

Please submit a help ticket to the [SEFSC OMI Service Portal](#) with the name of the repository you want to have deleted, along with the reasons why, and the SEFSC admins will approve and perform this task.

Auto-redirect GitHub Repositories and Pages

All GitHub repositories' URLs that were previously created will automatically redirect to the URL of the repository's current location once moved into the SEFSC organization. This includes changes to the names of the repositories. This allows for previously published URLs to remain the same.



The one caveat is **you can not create** a new repository on your account with the same name as the original repository. That will break the re-direct. All of this has been confirmed with internal tests and a 1/26/2021 meeting with GitHub engineers.

If the repository transferred used GitHub Pages, the Pages will also transfer. However, the URLs do not auto-redirect. This can be accomplished manually however following these steps:

1. Transfer the existing repository to the SEFSC organization, following this document.
2. Replacing “username” with your GitHub username, create a new public repository named “username.github.io”, if you don't have one already. (If anyone has a private GitHub Page, contact the [SEFSC GitHub Admins](#) before completing these steps. Private GitHub Pages are only available on paid accounts. They are included in our organization.)
 - a. Be sure to add the required Readme, License, and secret scanning files as described in this document.
 - b. For ease of use, I have created a template you may use. Once you are a member of the SEFSC organization, use the template named “SEFSC/SEFSC-SVC-GitHub.github.io”. Change the folder name and the context of the index.html to match the steps below.
3. Create a folder in the new repository with the exact name of the original repository.
4. In that folder, create a file named “index.html”
5. Copy and paste the following code, replacing the three URLs to the new links of the GitHub Page under the SEFSC organization:

```
<!DOCTYPE html>
<meta charset="utf-8">
<title>Redirecting to https://SEFSC.github.io/originalNameOfRepo</title>
<meta http-equiv="refresh" content="0; URL=https://SEFSC.github.io/originalNameOfRepo/">
<link rel="canonical" href="https://SEFSC.github.io/originalNameOfRepo/">
```

6. *(Optional)* If more than one GitHub Page was moved into the SEFSC organization, repeat steps 3 to 5 for each of them, within this same new repository using the original repository name.
7. Within that new repository, go to Settings-->Options and make sure GitHub Pages is turned on.

Once you complete this, if someone goes to your old GitHub Page URL, it will automatically transfer them to the new URL.



SEFSC GitHub Plan Information

The SEFSC is currently using the Enterprise Cloud Plan. To see the details of what is included with this plan, [visit this page](#).

Additional Information

NOAA Policies

NOAA's official GitHub policy may be found [here](#). This contains all the official rules that also apply to SEFSC.

SVN Migration and Reference Sheet

[https://training.github.com/downloads/subversion-migration/
Subversion versus Git/GitHub](https://training.github.com/downloads/subversion-migration/Subversion%20versus%20Git/GitHub)

Git Reference

[Official Git Documentation](#)

[Git Cheat Sheet](#)

[NOAA Fisheries Integrated Toolbox Best Practices](#)

[GitHub Best Practices](#)

Gold Copy

NMFS will perform daily backups for all repositories under the SEFSC organization as part of the GitHub Enterprise Cloud. Backups are stored within a dedicated [storage bucket](#) within the Google Cloud Platform.

GitKraken

[GitKraken Install Document](#)

Git / R-Studio / RMarkdown

[Git / RMarkdown / R-Studio Configuration](#)

Questions or Concerns

Feel free to reach out to [SEFSC GitHub Admins](#) with any questions or concerns about GitHub or this SOP.