

Report  
Software Version Control  
Use Git as SVC for your project



Department of Systemics  
School of Computer Science  
UNIVERSITY OF PETROLEUM & ENERGY STUDIES,  
DEHRADUN- 248007. Uttarakhand

Submitted By

**Shivam Kumar**

SAP ID: 500083476

Semester VII

B. Tech (Computer Sc. and Engineering), Hons

Submitted to : **Dr. Achala Shakya**

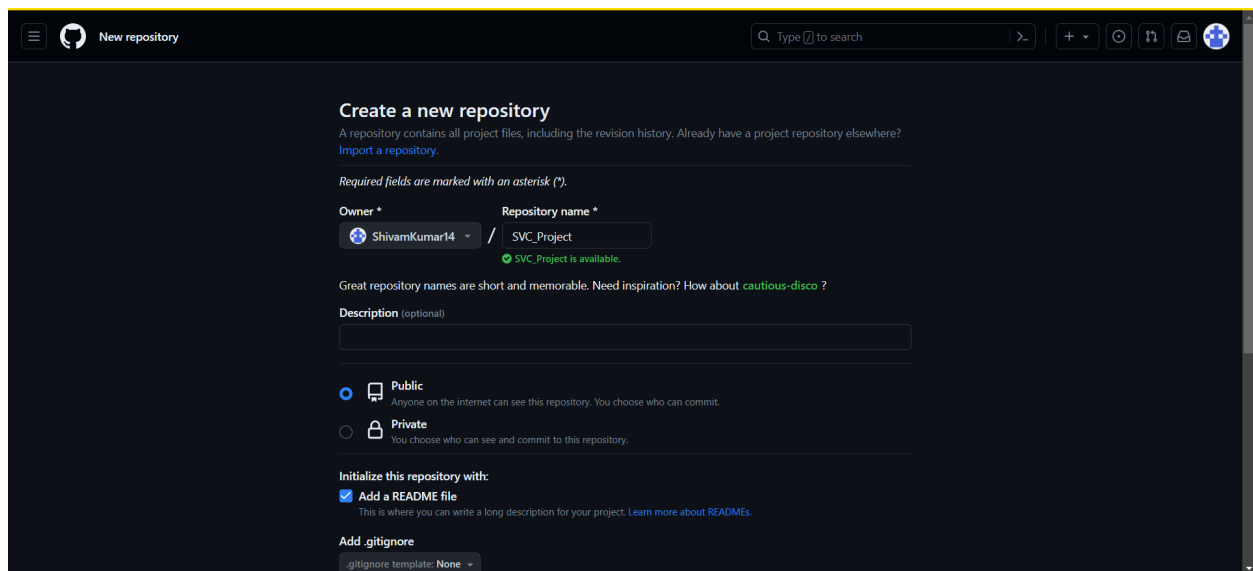
## 1. Introduction

Version control is an essential aspect of modern software development, allowing teams to collaborate efficiently, track changes, and manage codebase history. This report outlines the steps taken to implement version control using Git for a project, including repository setup, branching, and handling changes.

I have worked on a simple static website and pushed its code to a remote repository which is GitHub. This will highlight the process of using version control in tracking the changes I made during the working of this project.

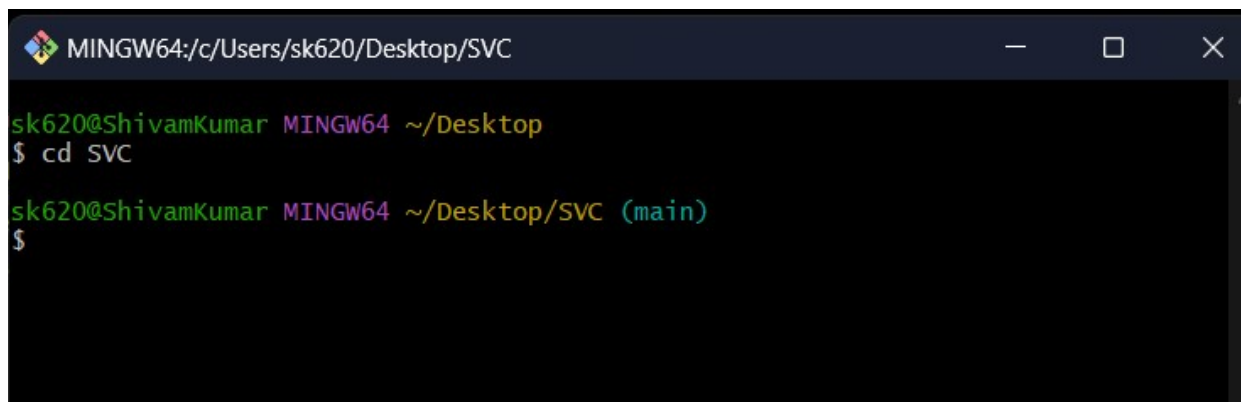
## 2. Repository Initialization

The first step involved creating a repository on GitHub. This central repository serves as the main codebase for the project, allowing collaborative development.

A screenshot of the GitHub 'Create a new repository' page. The page has a dark theme. At the top, there's a search bar and navigation icons. The main heading is 'Create a new repository'. Below it, a subtext says 'A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.' There's a note: 'Required fields are marked with an asterisk (\*).' The 'Owner' field is set to 'ShivamKumar14' and the 'Repository name' field is 'SVC\_Project', with a green checkmark indicating 'SVC\_Project is available.' Below this, a message says 'Great repository names are short and memorable. Need inspiration? How about cautious-disco?'. The 'Description' field is optional and empty. There are two radio buttons for visibility: 'Public' (selected) and 'Private'. Under 'Public', it says 'Anyone on the internet can see this repository. You choose who can commit.' Under 'Private', it says 'You choose who can see and commit to this repository.' There's a section 'Initialize this repository with:' with a checked box for 'Add a README file' and a link to 'Learn more about READMEs'. At the bottom, there's a section 'Add .gitignore' with a dropdown menu set to 'None'.

## 4. Initial Code Addition

Path to project directory:

A screenshot of a terminal window with a dark background. The title bar shows 'MINGW64:/c/Users/sk620/Desktop/SVC'. The terminal text shows the user 'sk620@ShivamKumar' in a 'MINGW64' environment at '~/Desktop'. The user enters '\$ cd SVC' and the prompt changes to 'sk620@ShivamKumar MINGW64 ~/Desktop/SVC (main)'. The user then enters '\$'.

Initialize Git: If your project is not yet a Git repository, you need to initialize it:

*git init*

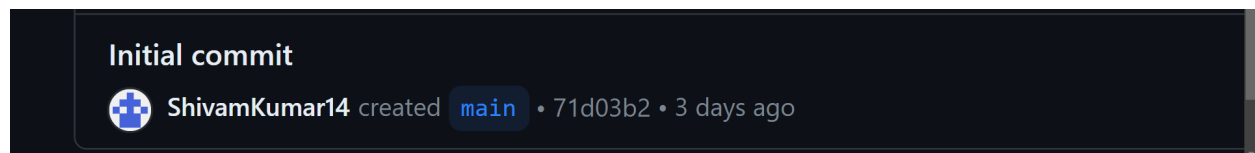
Add Remote Repository: Add the URL of your GitHub repository as the remote origin:

*git remote add origin https://github.com/ShivamKumar14/SVC-Project.git*

```
sk620@ShivamKumar MINGW64 ~/Desktop/SVC (main)
$ git init
sk620@ShivamKumar MINGW64 ~/Desktop/SVC (main)
$ git remote add origin https://github.com/ShivamKumar14/SVC-Project.git
```

The changes were staged, committed, and pushed to the main branch of the remote repository using the following commands:

*git add ., git commit -m "commit message", git push origin main*



After Initial Commit, made some changes in some files of the code, it will maintain the history of 'when and what' changes i made.

```
sk620@ShivamKumar MINGW64 ~/Desktop/SVC (main)
$ git add .

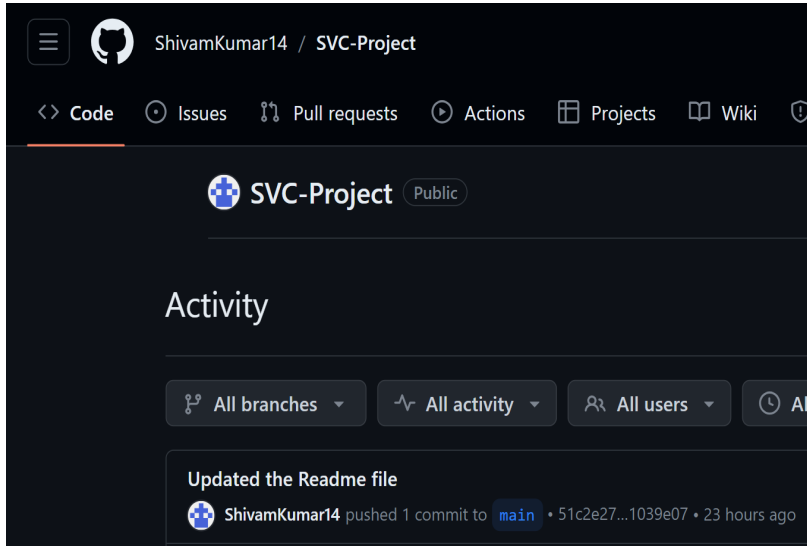
sk620@ShivamKumar MINGW64 ~/Desktop/SVC (main)
$ git commit -m "changed the calucaltion part"
[main c9147cf] changed the calucaltion part
2 files changed, 5 insertions(+), 5 deletions(-)

sk620@ShivamKumar MINGW64 ~/Desktop/SVC (main)
$ git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 456 bytes | 456.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/ShivamKumar14/SVC-Project.git
   3825b06..c9147cf  main -> main

sk620@ShivamKumar MINGW64 ~/Desktop/SVC (main)
$
```

## 5. Updating the Readme

Changes were made in the local text editor in the readme file and were pushed in GitHub to reflect updates about the project.



```
MINGW64:/c/Users/sk620/Desktop/SVC
sk620@ShivamKumar MINGW64 ~/Desktop
$ cd SVC

sk620@ShivamKumar MINGW64 ~/Desktop/SVC (main)
$ git add .

sk620@ShivamKumar MINGW64 ~/Desktop/SVC (main)
$ git commit -m "Updated the Readme file"
[main 63b247d] Updated the Readme file
1 file changed, 1 insertion(+), 1 deletion(-)

sk620@ShivamKumar MINGW64 ~/Desktop/SVC (main)
$ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 420 bytes | 420.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/ShivamKumar14/SVC-Project.git
1039e07..63b247d main -> main

sk620@ShivamKumar MINGW64 ~/Desktop/SVC (main)
$
```