|  |
| --- |
| Subject Name: **Source Code Management**  Subject Code: **24CSE0106**  Cluster: **Alpha**  Department: **DCSE**    **Submitted By: Submitted To:**  Name: Drishti Dr Renu Popli  G02 Department of Computer Science   & Engineering  2410990142 Chitkara University Institute of   Engineering and Technology, Rajpura. |
| **# Index Task 1.1#**   |  |  |  | | --- | --- | --- | | **S. No** | **Program Title** | **Page No.** | | **1.** | **To install and configure Git Client on your local system** | 01 | | **2.** | **Setting up GitHub Account and Adding Collaborators on GitHub Repository** | 21 | | **3.** | **To merge two branches within a Git repository.** | 24 | | **4.** | **To demonstrate push and pull operations in Git.** | 27 | |
| **Practical No. : 1**  **Aim:** To install and configure Git Client on local system.  **Theory:** Git is a distributed version control system that tracks changes in code, allowing multiple developers to collaborate efficiently. It manages project history through commits, branches, and merges, ensuring code integrity. Git enables reverting to previous versions, resolving conflicts, and syncing with remote repositories like GitHub for seamless teamwork.  **Procedure:**   1. Download Git from <https://git-scm.com/downloads> by clicking ‘Download for Windows’ option in the computer screen graphics.   Git installation wizard steps:  A screenshot of a computer  Description automatically generated   1. Click on Next followed by many other next’ s. |
| 1. Click on install      1. Click on finish |
| 1. Verify Git Bash Installation using the command: git –version 2. Configure User Details and User Commands |

|  |
| --- |
| **Practical No. : 2** |
| **Aim:** Setting up GitHub Account and Adding Collaborators on GitHub Repository  **Theory:**  Whenever you make a repository in GitHub, not everyone has the permission to change or push codes into your repository. The users have a read-only access. In order to allow other individuals to make changes to your repository, you need to invite them to collaborate to the project.  GitHub also restricts the number of collaborators we can invite within a period of 24 hours. If we exceed the limit, then either we have to wait for 24-hours or we can also create an organization to collaborate with more people.  Being a collaborator, the user can create, merge and close pull requests in the repository. They can also remove them as the collaborator.  **Procedure:**   1. Login to your GitHub account and you will land on thehomepage as shown below. Click on Repositories option in the menu bar. |

|  |
| --- |
| 1. Click on the ‘New’ button in the top right corner.     3. Enter the Repository name and add the description of the repository.  4. Select if you want the repository to be public or private. |

|  |
| --- |
| 5. If you want to import code from an existing  repository select the import code option.    6. Now, you have created your repository successfully.  7. To add collaborators to your repository, open your repository and select settings option in the  navigation bar. |

|  |
| --- |
| 8. Click on Collaborators option under the access tab.    9. After clicking on collaborators, GitHub asks you to enter your password to confirm the access to the  repository. |

|  |
| --- |
| 10.After entering the password, you can manage access andadd/remove team members to your project.  11.To add members, click on the add people option  and search the id of your respective team member.    12.To remove any member, click on remove option  available in the last column of member’s respective row. |
| **Practical 3: Merging Two Branches**   * **Aim:** To merge two branches within a Git repository. * **Theory:** Merging branches in Git allows you to combine changes from one branch into another. It is a fundamental process in collaborative workflows, ensuring all contributions are integrated into a single codebase. * **Procedure:**  1. **Create a new Branch and Switch to it:**      * 1. Here, the green star represents the current working branch  1. **Make Changes to a file in the new branch and commit them:** |
| * 1. echo command allows the user to create and add the content of the file at same place.   2. Adding and committing the change  1. **Switch back to main branch:**      * 1. Command ‘git checkout branch\_name' changes the current working branch to branch\_name named branch.  1. **Modify another file in the main branch and commit the changes:** |
| The above commands show steps to create a new file in the Master Branch, adding, committing, and checking the status of the changes.   1. **Merge the new branch into the main branch**     **Practical 4: Push/Pull Using Git**   * **Aim:** To demonstrate push and pull operations in Git. * **Theory:** Push transfers committed changes from the local repository to the remote repository, while pull retrieves updates from the remote repository. * **Procedure:**   + **Push the changes to the remote repository using git push.**      * Make changes directly on the remote repository (e.g., via GitHub interface). |
|  |
| Git log command after committing the changes. |