

Solutions that Fit

A quick setup and beginner’s usage guide

**GITHUB User Guide**

Confidential © D+H, 2016

Contents

[Note 2](#_Toc472604339)

[History 3](#_Toc472604340)

[Install git GUI Client 4](#_Toc472604341)

[Setup steps 4](#_Toc472604342)

[Basic git commands 7](#_Toc472604343)

[Config Command 7](#_Toc472604344)

[Clone Repository command 7](#_Toc472604345)

[Branching commands 7](#_Toc472604346)

[Get latest from the git server 7](#_Toc472604347)

[Publish changes to git server 8](#_Toc472604348)

[Merging git branches 8](#_Toc472604349)

# Note

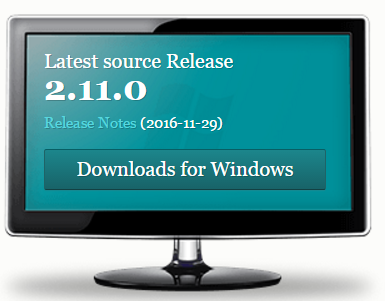
This is living document, will be updated very frequently.

# History

|  |  |  |  |
| --- | --- | --- | --- |
| Change Description | Author | Reviewer | Date |
| Initial Draft | Allwin | Dan/Lou | Jan/19/2017 |

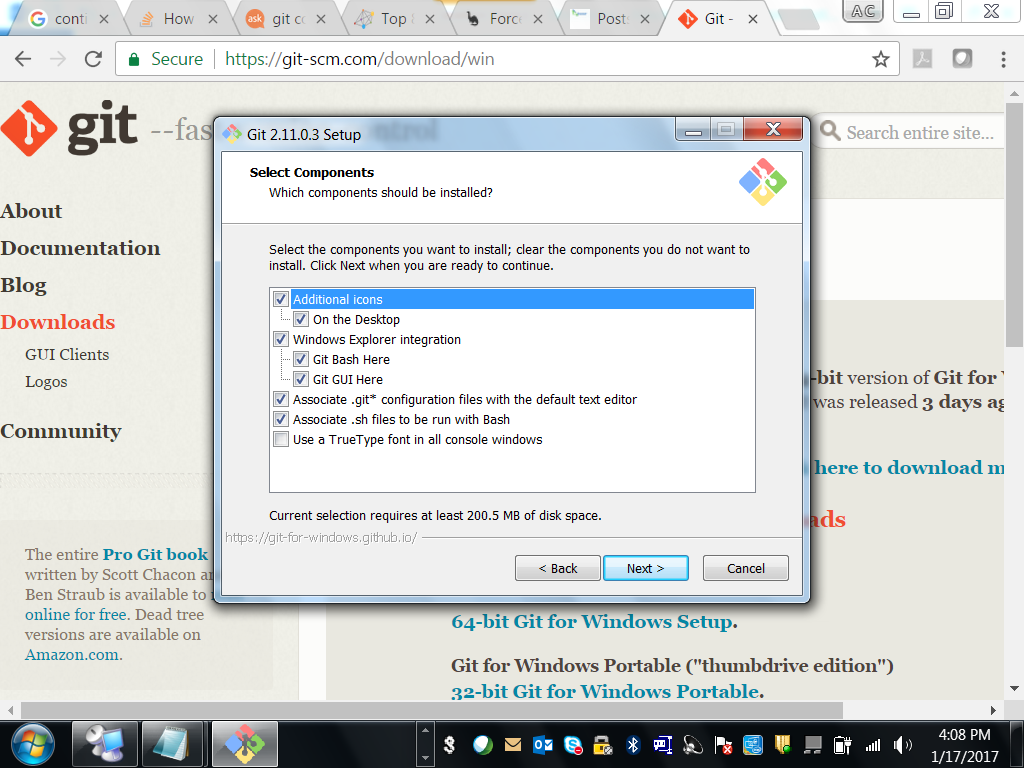
# Install git GUI Client

git client tool consist of UI and command line usage as well. This can be downloaded from the <https://git-scm.com/downloads>. Download the latest for windows from the website.

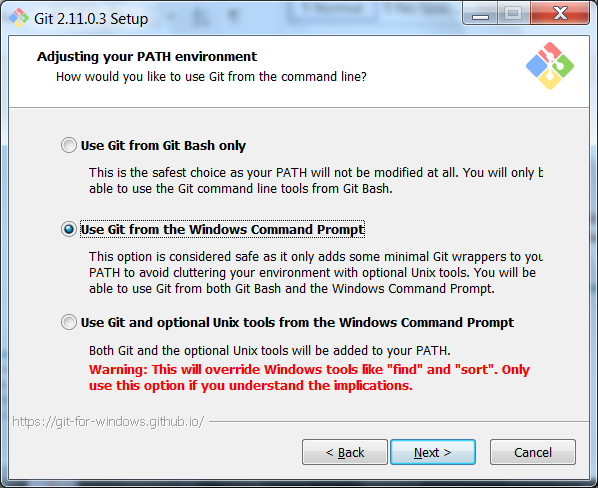


## Setup steps

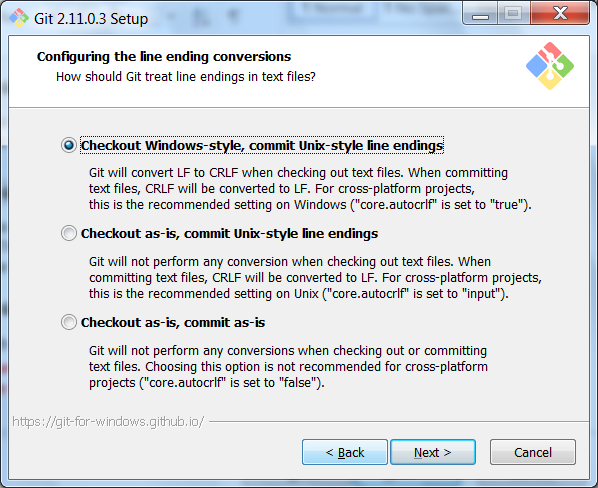
1. Select components



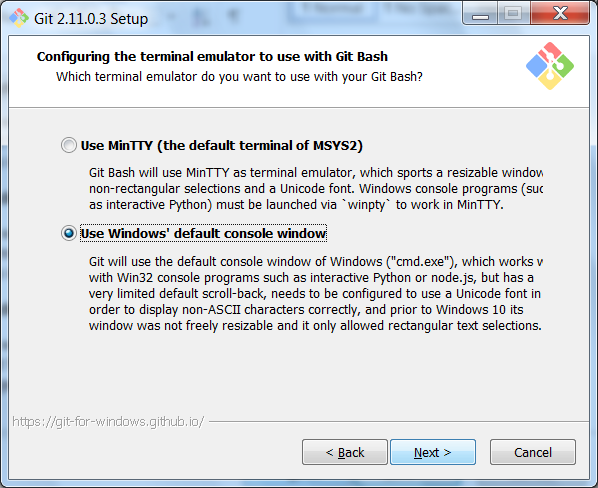
1. Setting up the Path environment variable for windows. By choosing git from command prompt also.



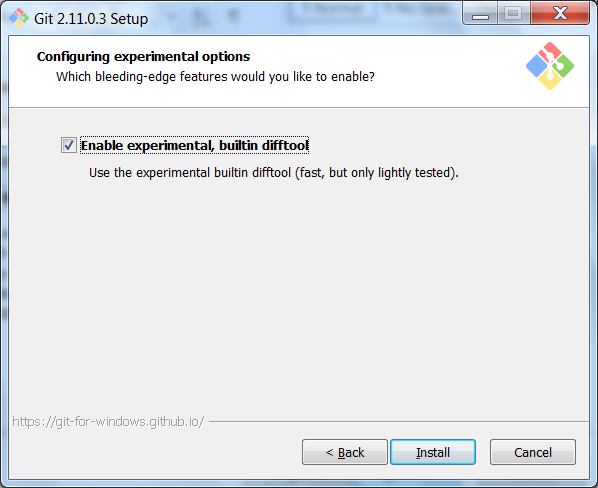
1. Line ending conversion options, as most our team environment is Windows, opting out for windows style is best



1. Terminal emulator as per team environment, good to choose windows options



1. Diff tool for conflict management



To confirm installation was successful open command prompt and type the git and press enter. You will see the git help be displayed.

# Basic git commands

This is not an exhaustive list of commands. And not all the options avalible in the command is described. This list for beginners to quick start using git repositories

### Config Command

Configuration of git environment is done using this command. Mostly for updating user name and email id.

Ex.

|  |  |
| --- | --- |
| **git config --list** | To list all the configuration of git global and local repository configuration |
| **git config --global http.sslverify false** | To ignore the ssl verification |

### Clone Repository command

Change directory accordingly to the folder where the repository to be newly imported from the server.

**cmd:**

**git clone <repositoryURL>**

### Branching commands

To create and publish local branch, following is the set of command

|  |  |
| --- | --- |
| git branch | To get the list of branch that are available. And to know which branch that we are currently in. The current branch will have \* symbol infront of the name |
| git branch <newbranchname> | To create new branch |
| git checkout <branchname> | To transfer between branch |
| git push origin <newbranchname> | To publish the local branch. Always checkout to the branch before you publish it. |

### Get latest from the git server

To get the latest from git server.

Note: if you have any uncommitted changes, git will not allow you to get the latest. Either you have to commit or ignore the changes before you pull

CMD

**git pull**

### Publish changes to git server

To publish the changes using command line, you have to follow the follow steps. These steps are not common for all environment. But these are the best or good to follow in case of any code publish

1. git Add
   * To add the new files to the changes
2. git Commit
   * Confirming that these changes need to be published to server. Or atleast these changes need to be persevered.
   * Always good to give a valid and complete meaningful messages in commit message
   * Smart commit is possible and enabled by most of server, please check for this and recommend to use smart commit whenever possible
3. git pull
   * Getting the latest from the server
   * This is a mandatory step, after commit always get the latest from the server and do a conflict merge if necessary before publishing the code
4. git push
   * Update or publish the code to the git central repository

### Merging git branches

To merge code from one branch to another.

Steps,

1. git checkout <tobranch>
   * Open the branch where the code need to be imported
2. git merge <fromBranch>
   * give the branch name from where the code needed
3. git Pull
   * get latest from the server(always)
4. git push
   * Publish the changes to server