Table of contents – Aidin – GitLab guide

Getting started

1. Download Visual Studio Code.
2. Download Git.
3. Download the GitLab extension in Visual Studio Code.
4. Add access key to connect GitLab with your computer.

- Visual Studio Code

Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft with the Electron Framework, for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add functionality.

- **Download Visual Studio Code at https://code.visualstudio.com/**

- Git installation

Git (/ɡɪt/) is a distributed version control system that tracks changes in any set of computer files, usually used for coordinating work among programmers who are collaboratively developing source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows (thousands of parallel branches running on different computers).

- **Download Git at https://git-scm.com/book/en/v2/Getting-Started-Installing-Git**

- GitLab extension installation

F
Sdf
Sd
Fsd
Fsdf
Sdf
Sd

GitLab terminal commands

- Confirm Git is installed

You can determine if Git is already installed on your computer by opening a terminal and running this command:

- **git --version**

If Git is installed, the output is:

- **git version X.Y.Z**

- Configure Git

To start using Git from your computer, you must enter your credentials to identify yourself as the author of your work. The u sername and email address should match the ones you use in GitLab.

In your shell, add your username:
- **git config --global user.name "your_username"**

Add your email address:
- **git config --global user.email "your_email_address@example.com"**

To check the configuration, run:
- **git config --global --list**

- Choose a repository

Before you begin, choose the repository you want to work in. You can use any project you have permission to access on GitLab.com or any other GitLab instance.

1. Go to "https://gitlab.liu.se/da-proj/microcomputer-project-laboratory-d/2023/g08/docs".
2. In the upper-right corner, select **Fork**.
3. Choose a namespace for your fork.

The project becomes available at "https://gitlab.com/<your-namespace>/sample-project/".

- Clone a repository

When you clone a repository, the files from the remote repository are downloaded to your computer, and a connection is created. This connection requires you to add credentials. You can either use **SSH** or **HTTPS**. **SSH** is recommended.

-- Clone with SSH

Clone with SSH when you want to authenticate only one time.

1. Authenticate with GitLab by following the instructions in the SSH documentation.

2. Go to your project's landing page and select **Clone**. Copy the URL for **Clone with SSH**.

3. Open a terminal and go to the directory where you want to clone the files. Git automatically creates a folder with the repository name and downloads the files there.

4. Run this command:
- **git clone git@gitlab.liu.se:da-proj/microcomputer-project-laboratory-d/2023/g08/docs.git**

5. To view the files, go to the new directory:
- **cd "sample-project"**

-- Clone with HTTPS

Clone with HTTPS when you want to authenticate each time you perform an operation between your computer and GitLab.

1. Go to your project's landing page and select **Clone**. Copy the URL for **Clone with HTTPS**.

2. Open a terminal and go to the directory where you want to clone the files.

3. Run the following command. Git automatically creates a folder with the repository name and downloads the files there.
- **git clone https://gitlab.liu.se/da-proj/microcomputer-project-laboratory-d/2023/g08/docs.git**

Gfhgfhfg
Hgf
Hgf
H
Gfh
Gf
Gfh
Fg
Hgf
H
Hf
Ghf

h