

**Digital Service Report 0.1**

# Installation Guide

**Contents**

[1 Introduction 2](file:///D:\Documents\School\Final%20Year%20Project\Final%20Project.docx#_Toc74750817)

[2 Installing Git 2](file:///D:\Documents\School\Final%20Year%20Project\Final%20Project.docx#_Toc74750818)

[3 Setting Up MongoDB 10](file:///D:\Documents\School\Final%20Year%20Project\Final%20Project.docx#_Toc74750819)

[4 Setting Up Heroku and Heroku CLI (Windows) 17](file:///D:\Documents\School\Final%20Year%20Project\Final%20Project.docx#_Toc74750820)

[5 Installing Node.js (Windows) 21](file:///D:\Documents\School\Final%20Year%20Project\Final%20Project.docx#_Toc74750821)

[6 Download Visual Studio Code 26](file:///D:\Documents\School\Final%20Year%20Project\Final%20Project.docx#_Toc74750822)

[7 Setting Up Application API 30](file:///D:\Documents\School\Final%20Year%20Project\Final%20Project.docx#_Toc74750823)

[8 Setting Up Application Frontend 36](file:///D:\Documents\School\Final%20Year%20Project\Final%20Project.docx#_Toc74750824)

# Introduction

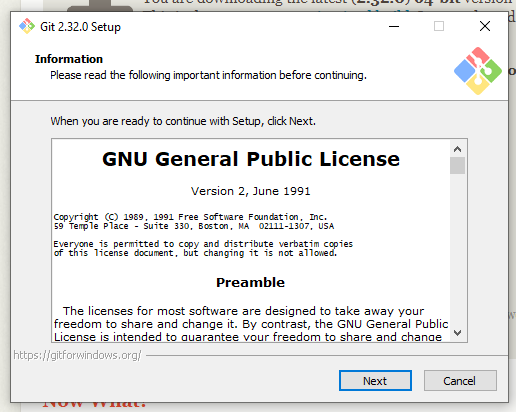
This installation guide is organised in a way that presents the recommended sequence in which Digital Service Report (DSR) can be implemented.

# Installing Git

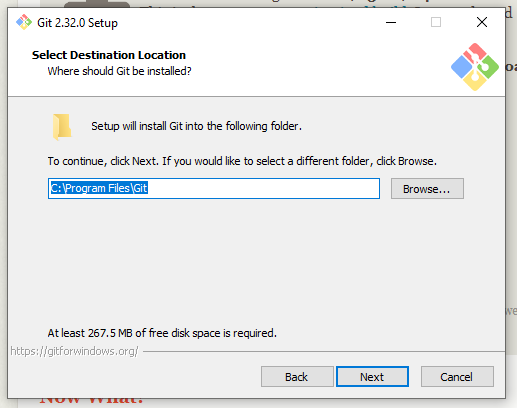
1. Navigate to <https://git-scm.com/downloads>
2. Select the platform and download the installer



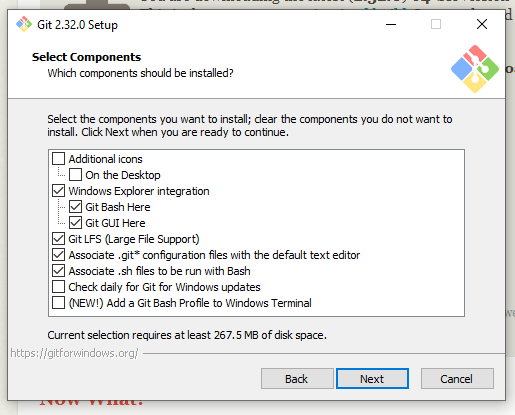
1. Run the installer and click **Next**

c

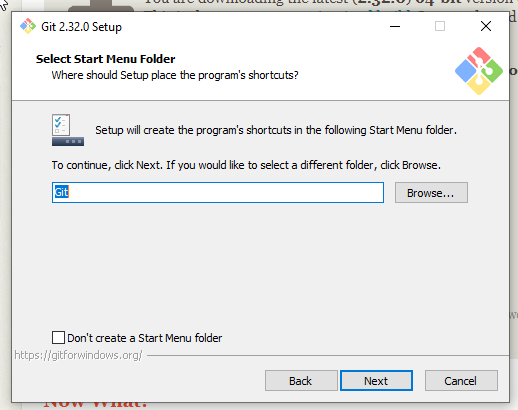
1. Select the installation folder and click **Next**



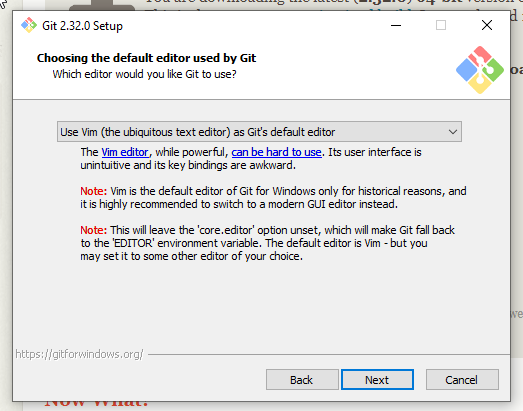
1. Select **Windows Explorer integration, Git LFS, Associate .git\* configuration files with the default text editor** and  **Associate .sh files to be run with bash**. Click **Next**



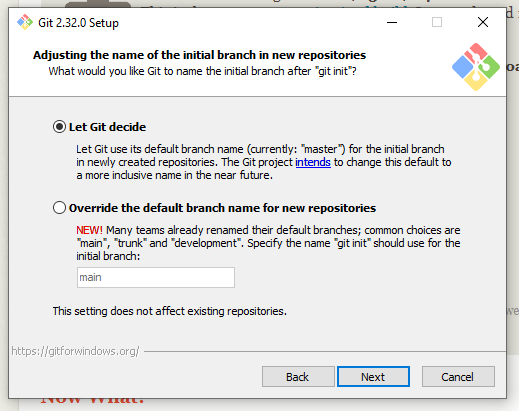
1. Select the shortcut location and click **Next**



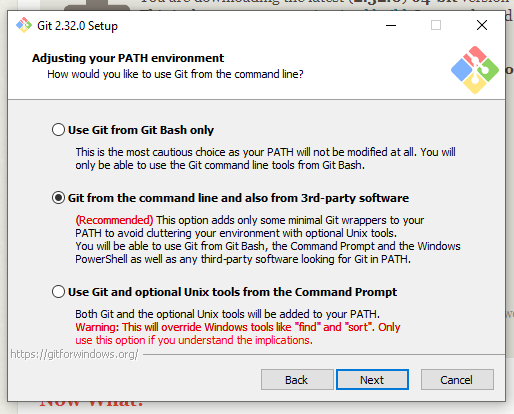
1. Select the default text editor for git and click **Next**



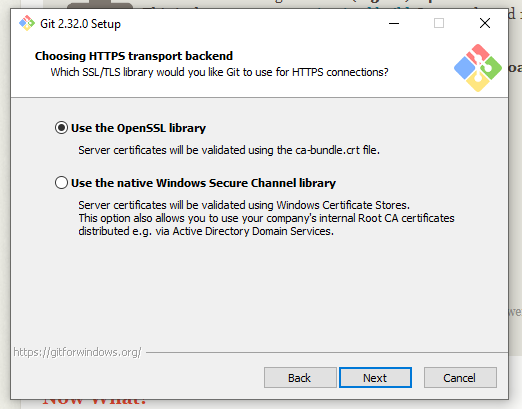
1. Select **Let Git decide** and click **Next**



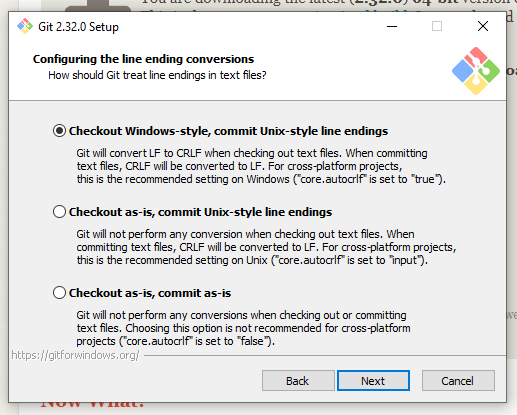
1. Select **Git from the command lines and also from 3rd-party software** and click **Next**



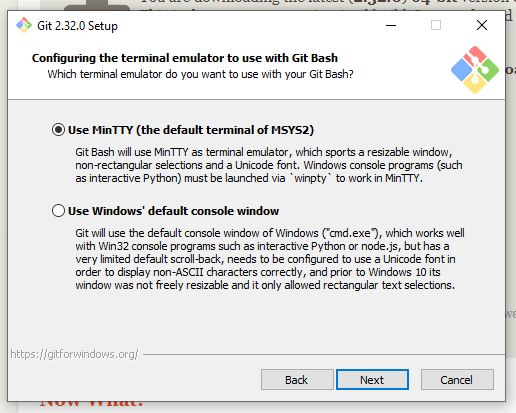
1. Select **Use the OpenSSL library** and click **Next**



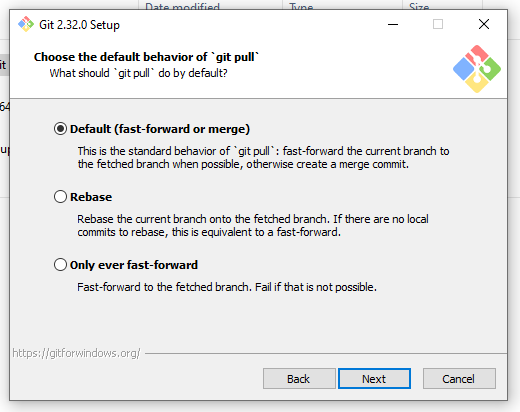
1. Select the line ending conversions and click **Next**



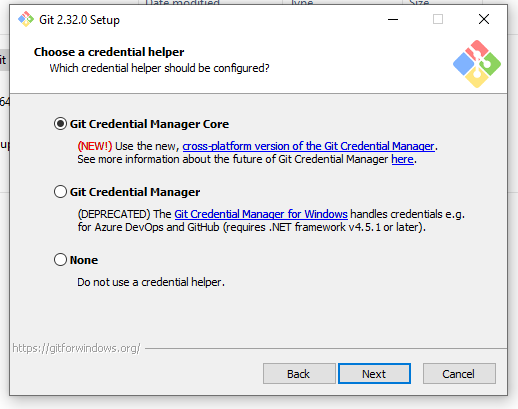
1. Select the emulator to use for Git Bash and select **Next**



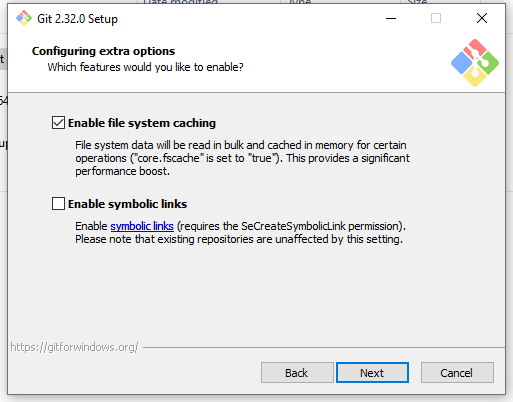
1. Select **Default** for git pulland select **Next**



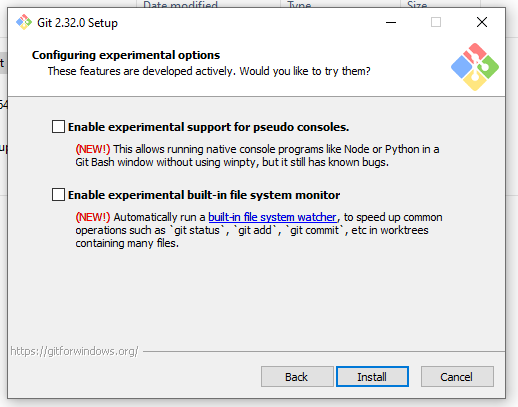
1. Select **Git Credential Manager Core** and select **Next**



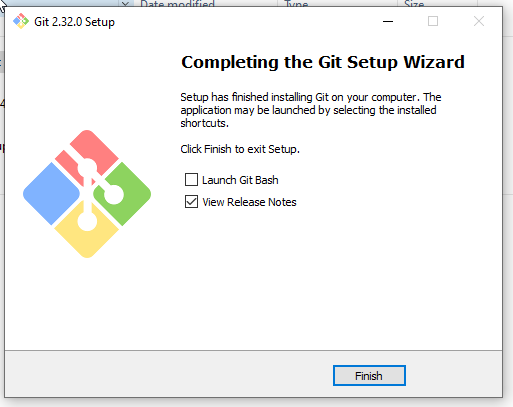
1. Select **Enable file system caching** and select **Next**



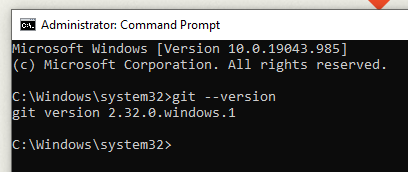
1. Click **Install** to start the installation



1. Click **Finish** to close the installer

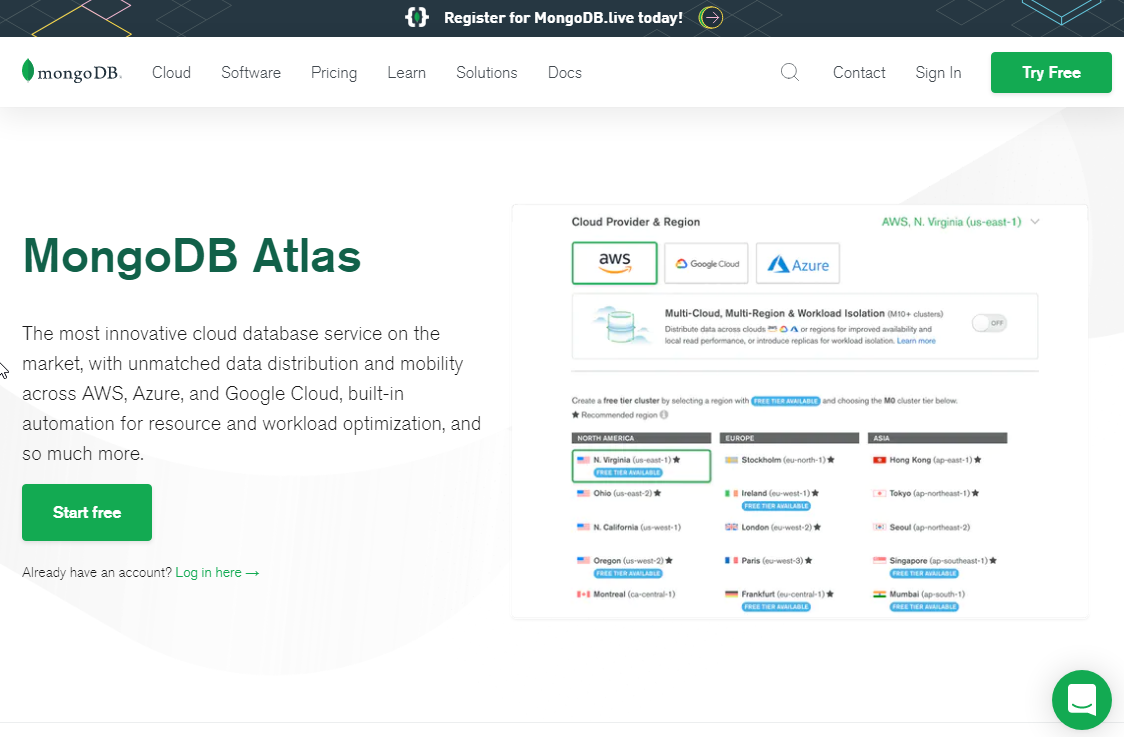


1. Restart the PC. Once the PC has booted, open the command prompt and type **git –version** to show the git version

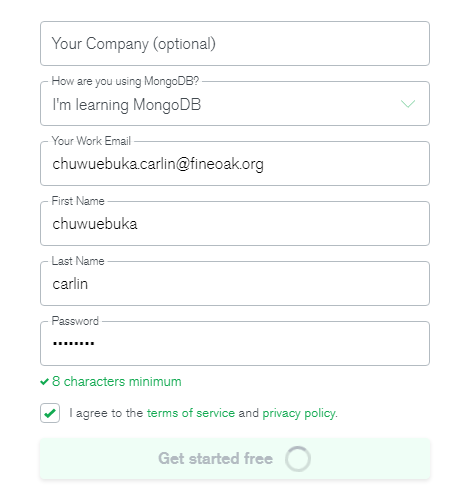


# Setting Up MongoDB

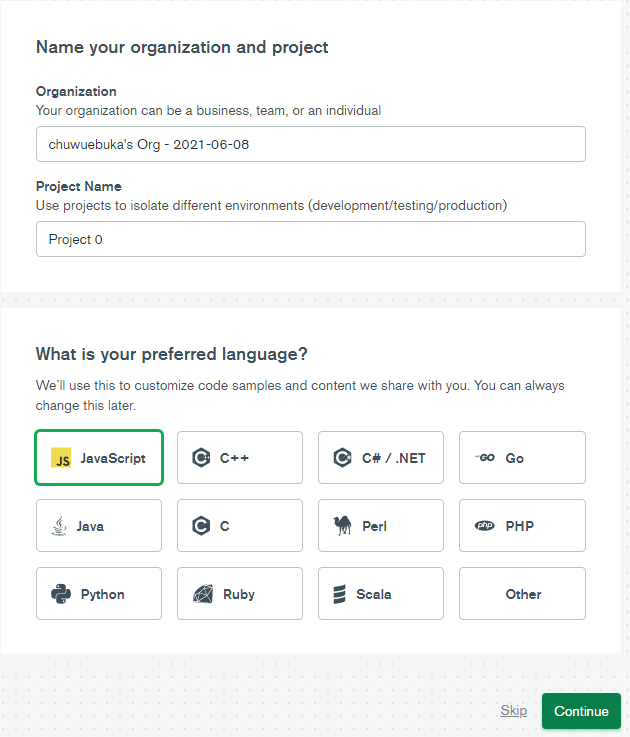
1. Navigate to <https://www.mongodb.com/cloud/atlas>
2. Click on **Start free**



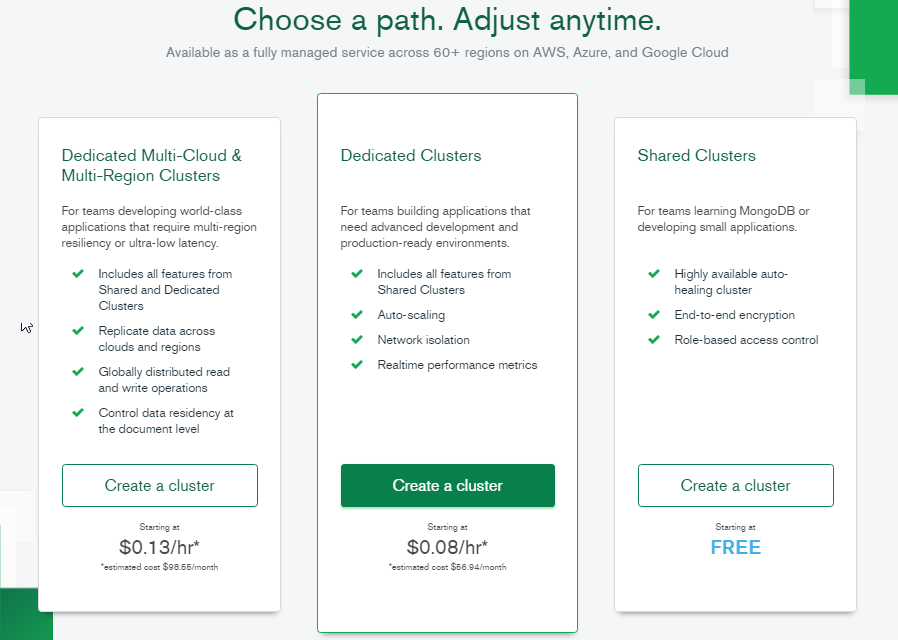
1. Create a mongo account



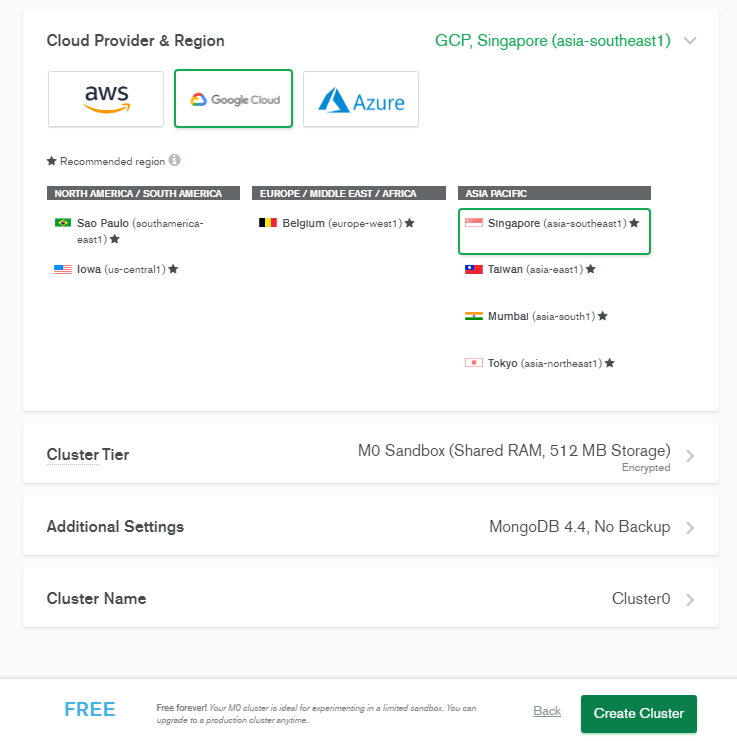
1. Enter your **Organisation** and **Project Name**. Select the **JaveScript** for preferred language and click **continue**.



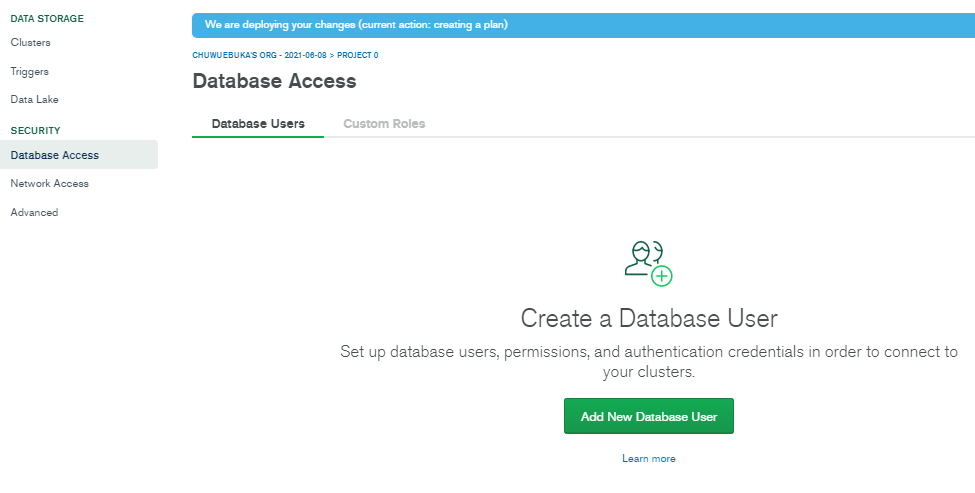
1. Select the cluster type and **Create a cluster**



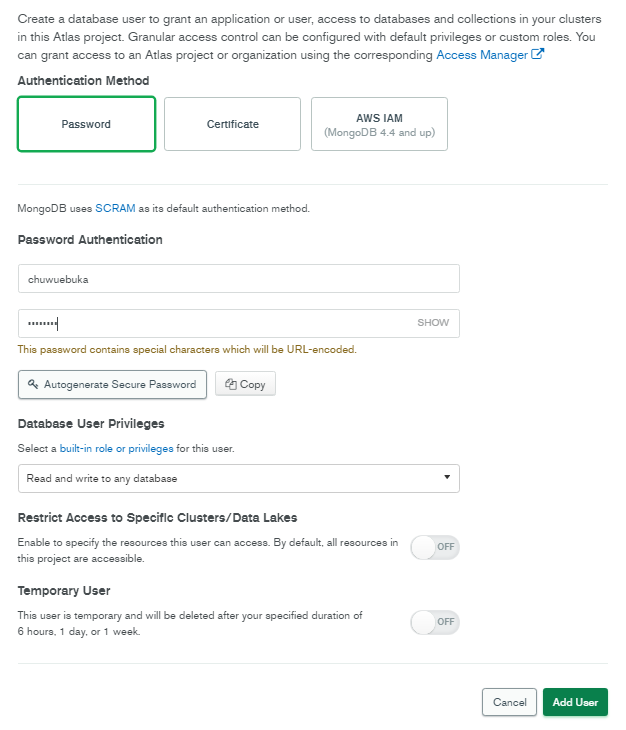
1. Select the **Cloud Provider & Region** and click **Create Cluster**



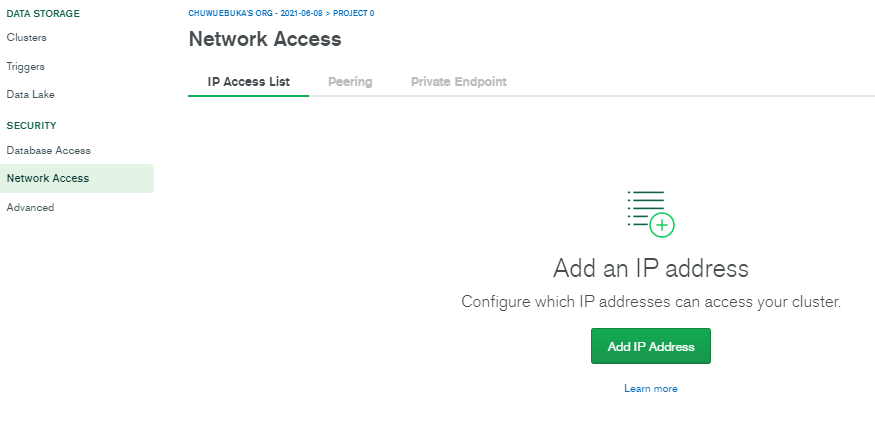
1. Navigate **Database Access** and click **Add New Database User**.



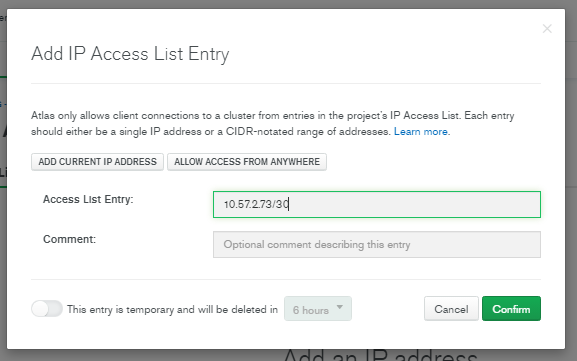
1. Select the **Authentication Method** you like to use but for this guide, **Password** will the prime example. Enter the **username and password** under **Password Authentication** and click **Add User**.



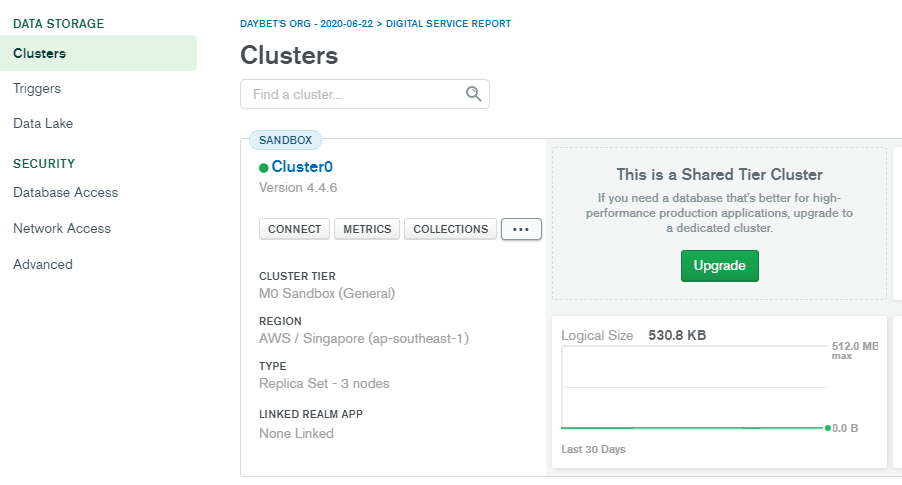
1. Navigate to **Network Access** and click **Add IP Address**



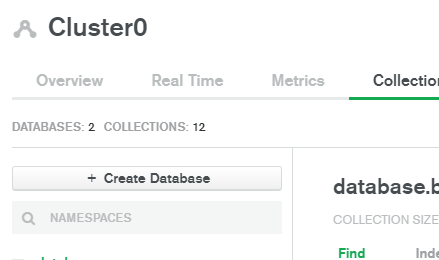
1. Add network address where the application is host from and click **Confirm**.



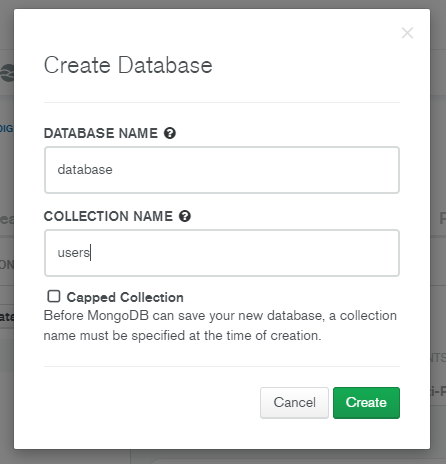
1. Navigate back to **Clusters** and click **COLLECTIONS**



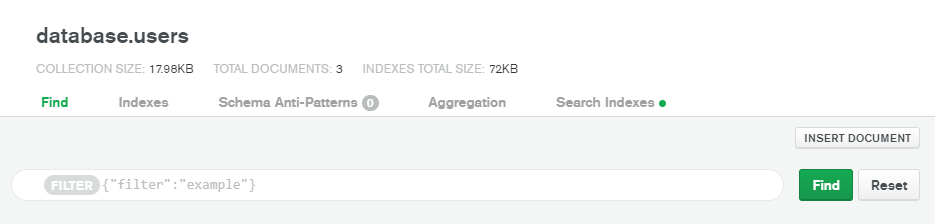
1. Click **+ Create Database**



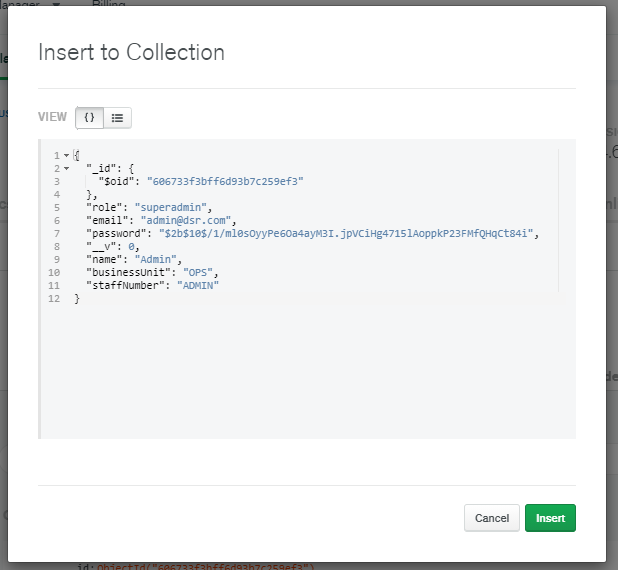
1. Enter your **DATABASE NAME** and type **users** in **COLLECTION NAME.** Click **Create.**



1. In the newly created database, click **INSERT DOCUMENT**.



1. Select the **VIEW** as **{}** and enter the following json text. Click **Insert** next.



{

"\_id": {

"$oid": "606733f3bff6d93b7c259ef3"

},

"role": "superadmin",

"email": "admin@dsr.com",

"password": "$2b$10$/1/ml0sOyyPe6Oa4ayM3I.jpVCiHg4715lAoppkP23FMfQHqCt84i",

"\_\_v": 0,

"name": "Admin",

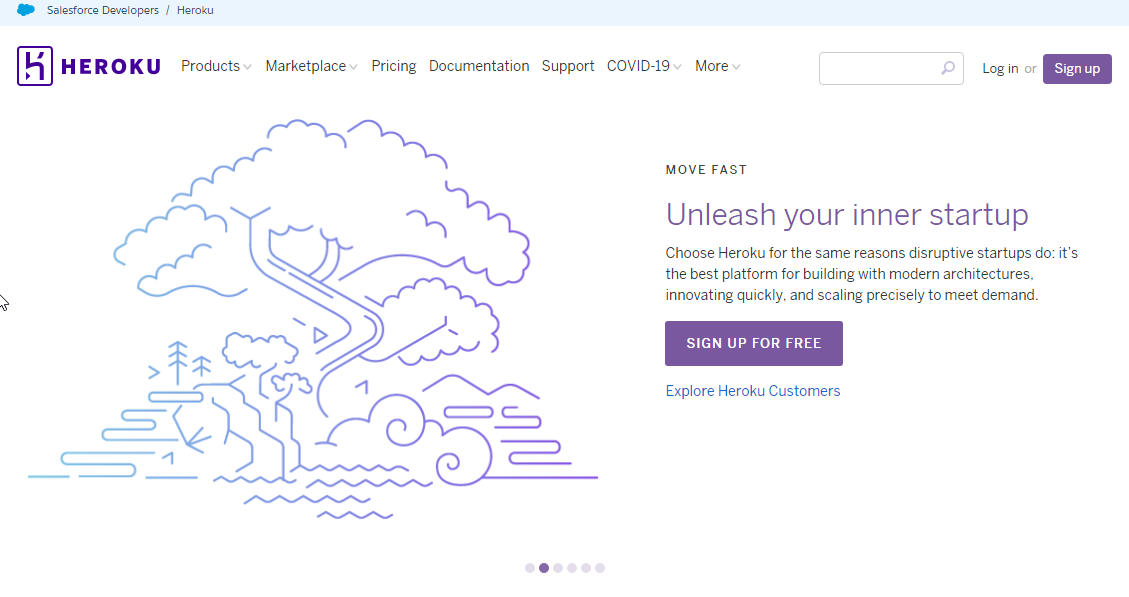
"businessUnit": "OPS",

"staffNumber": "ADMIN"

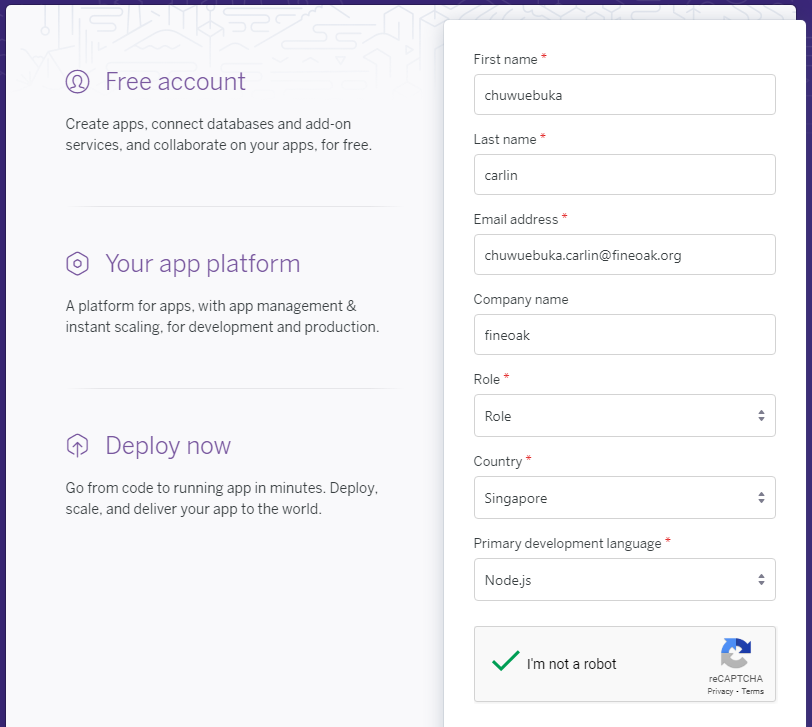
}

# Setting Up Heroku and Heroku CLI (Windows)

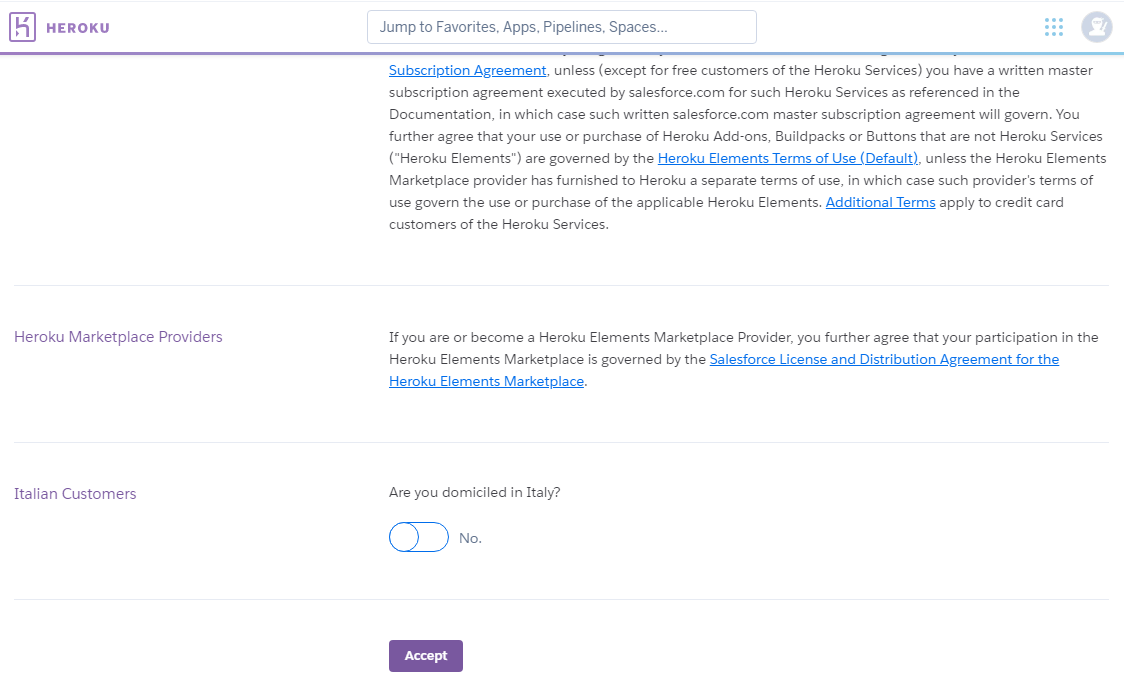
1. Navigate to <https://www.heroku.com> and click **Sign up**



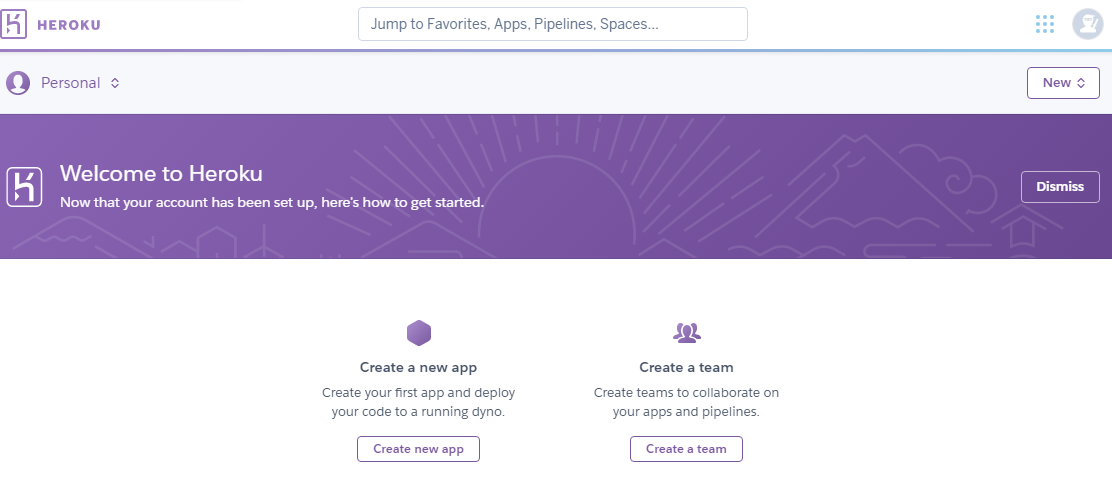
1. Enter the details for account opening. In the **Primary development language**, select **Node.js**. Click **CREATE FREE ACCOUNT**.



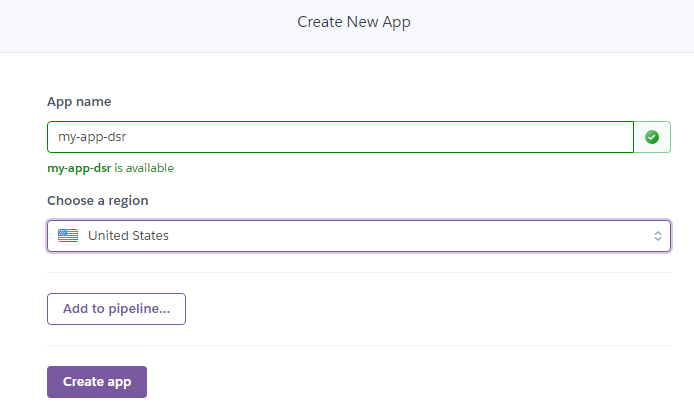
1. Once the account has been created, a verification email will be sent to your email. Verify the email and login to Heroku. Upon log in for the first time, read and click **Accept** to proceed.



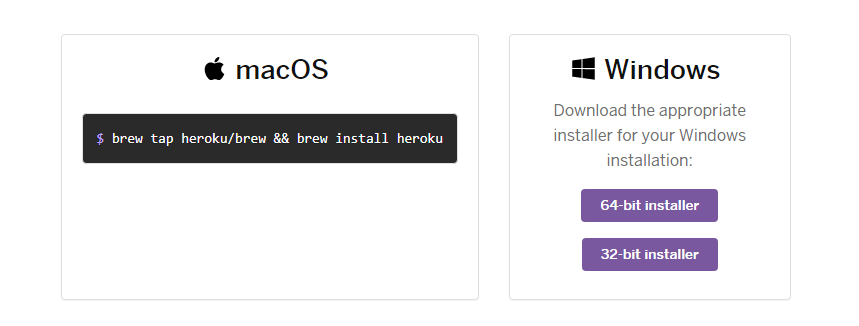
1. Click **Create new app**



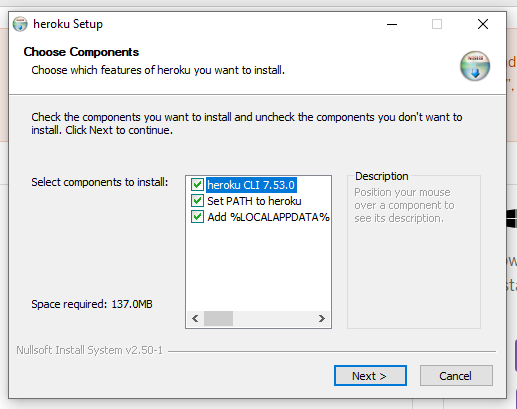
1. Enter a unique **App name** and choose the **region**. Click **Create app.**



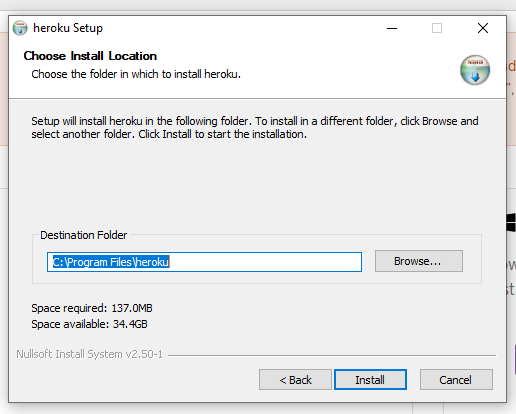
1. Navigate to <https://devcenter.heroku.com/articles/heroku-cli> and select the installation platform.



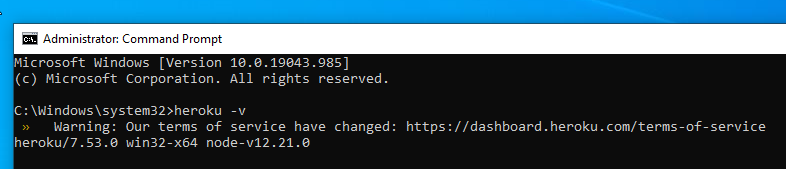
1. Run the installer. Select **Heroku CLI, Set PATH to Heroku** and **Add LOCALAPPDATA** click **Next >**



1. Select the installation folder and click **Install**

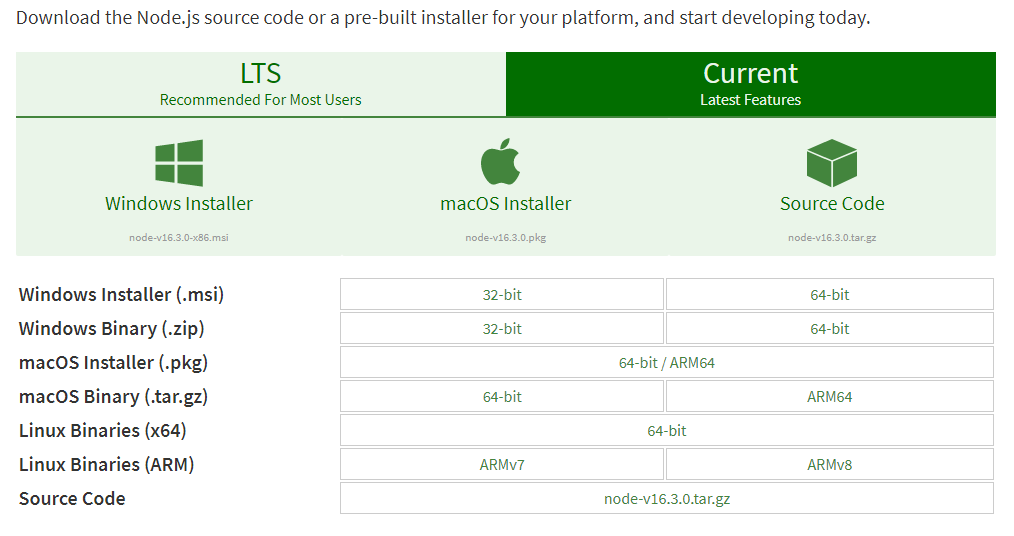


1. Click **Close** to close the installer.
2. Restart the PC. Once the PC is booted up, open the command prompt and type **Heroku -u** to show Heroku version

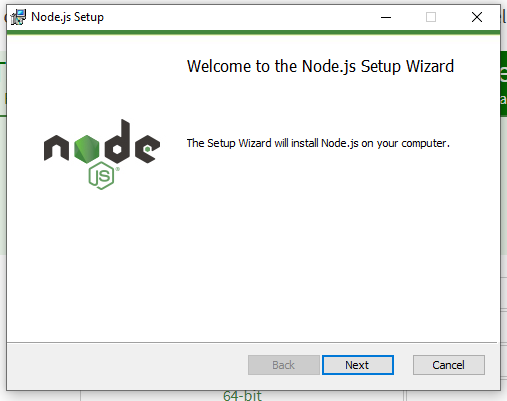


# Installing Node.js (Windows)

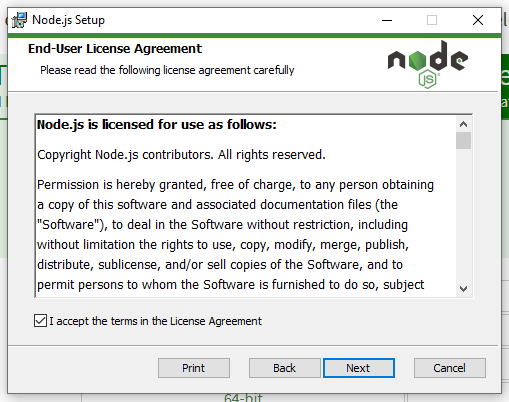
1. Navigate to <https://nodejs.org/en/download/current>
2. Select the platform to download



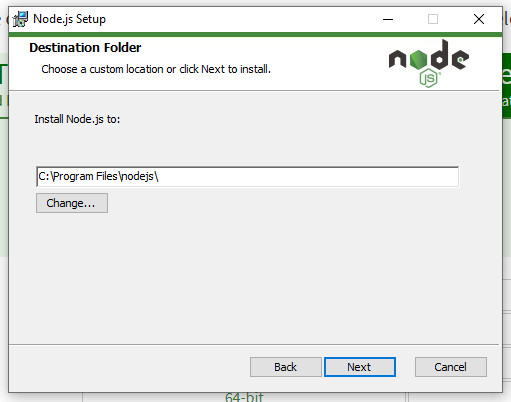
1. Run the installer and click **Next**



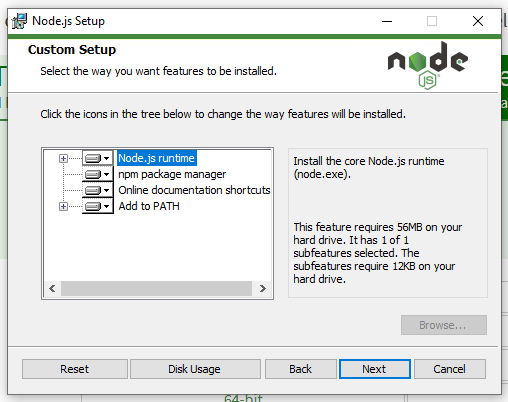
1. Read and check **I accept the terms in the License Agreement** and click **Next**



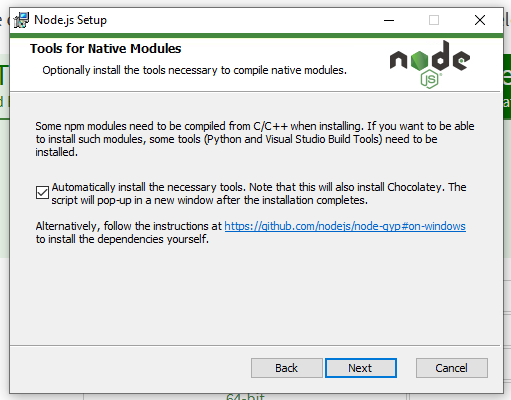
1. Select the installation folder location and click **Next**



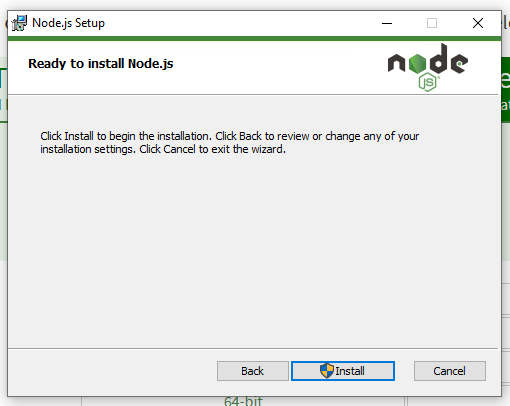
1. Click **Next** to continue



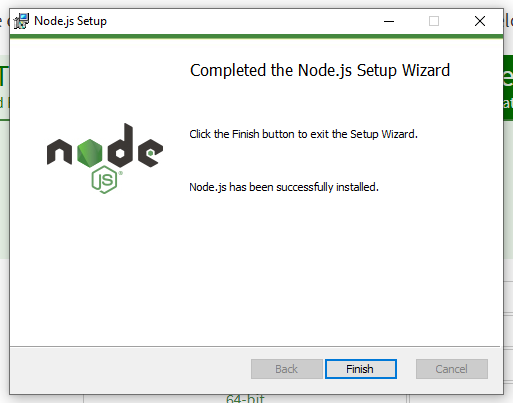
1. Check **Automatically install the necessary tools** and click **Next**



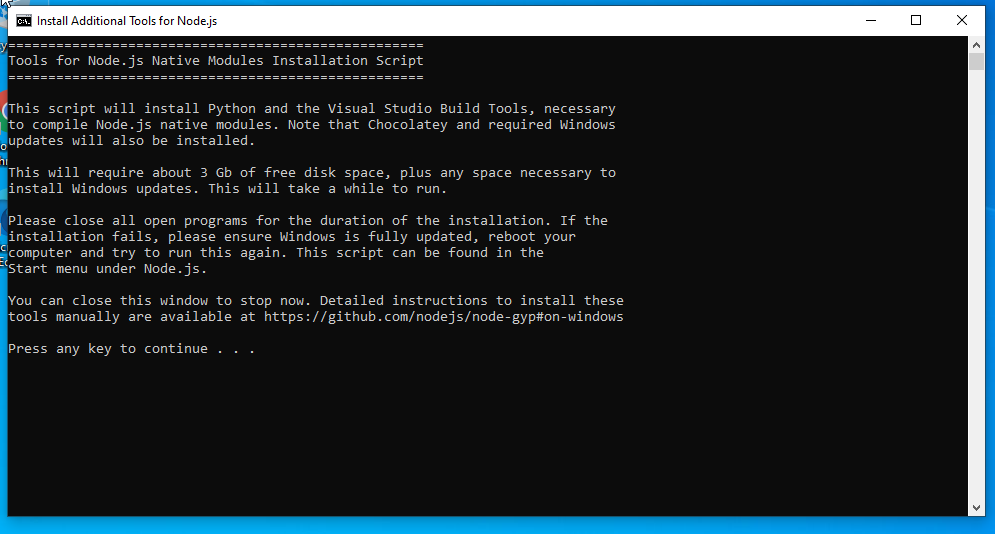
1. Click **Install** to begin the installation



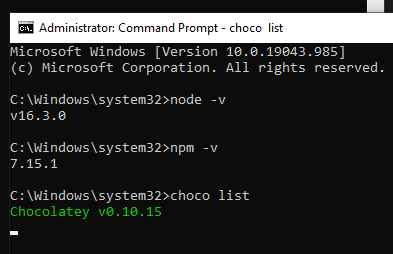
1. Click Finish to proceed



1. A Command prompt window will be prompt to install the necessary tools. Close off all other program in order to process by hitting any key in the prompt.

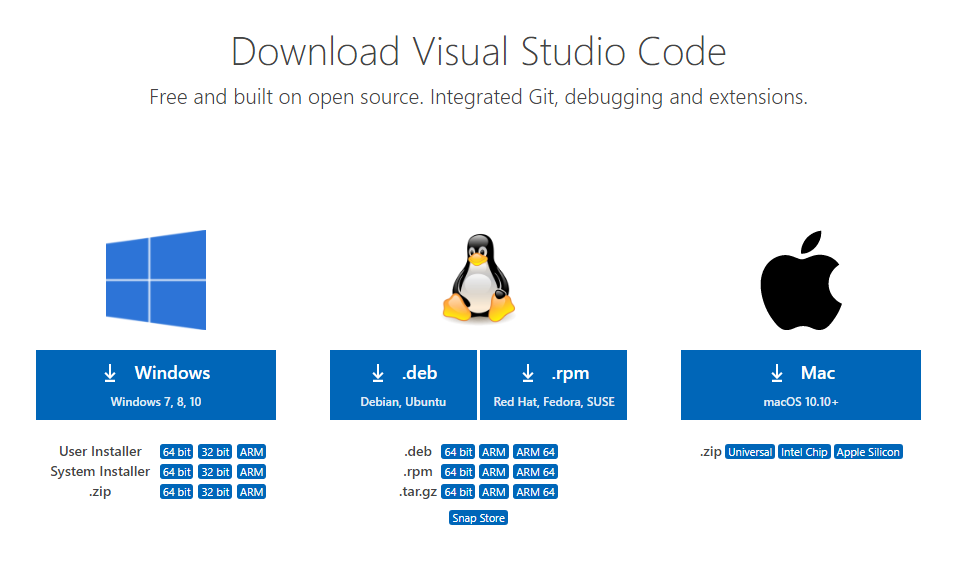


1. Reboot the PC to complete the installation.
2. Open the command prompt, enter **node -v**, **choco list** and **npm -v** to check the tools and node version

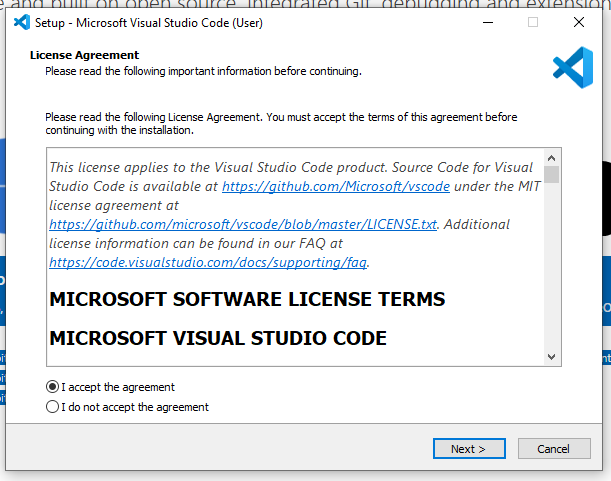


# Download Visual Studio Code

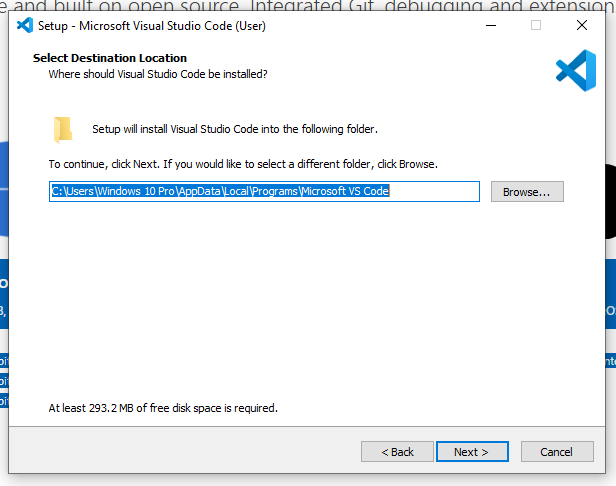
1. Navigate to <https://code.visualstudio.com/download>
2. Select the platform to download



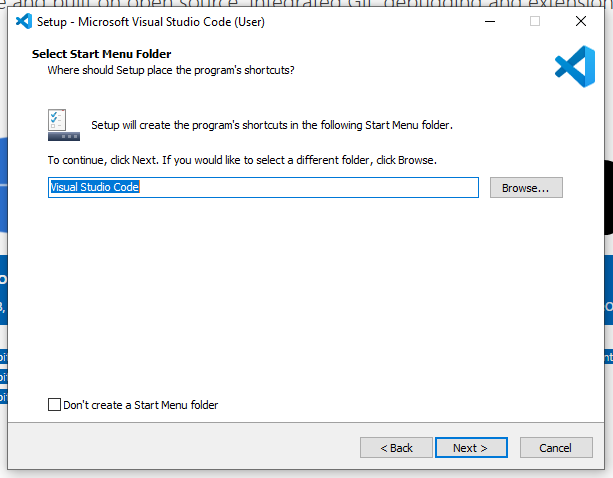
1. Run the installer. Read and select **I accept the agreement** and click **Next >**



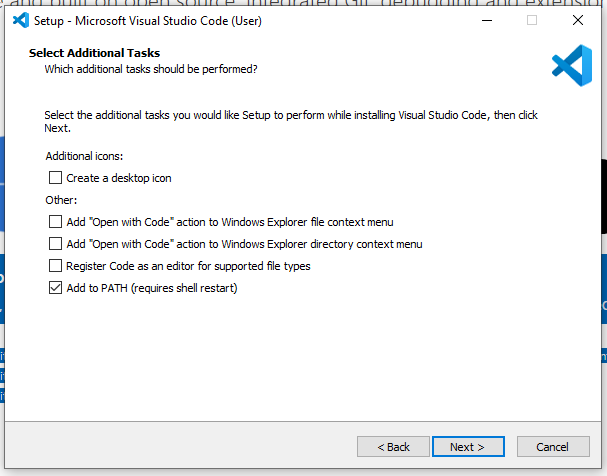
1. Select the location of installation and click **Next >**



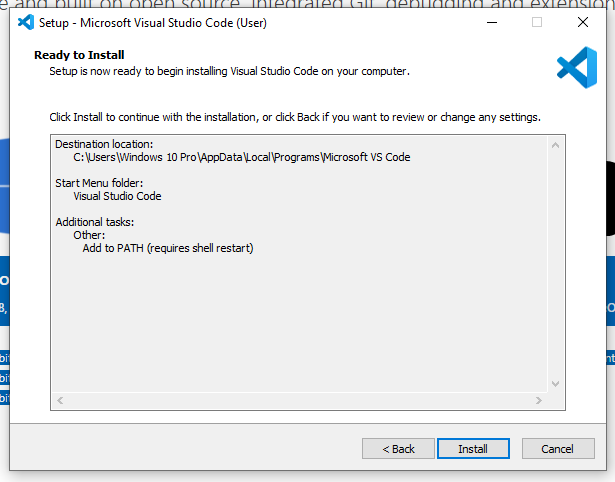
1. Select the shortcut folder and click **Next >**



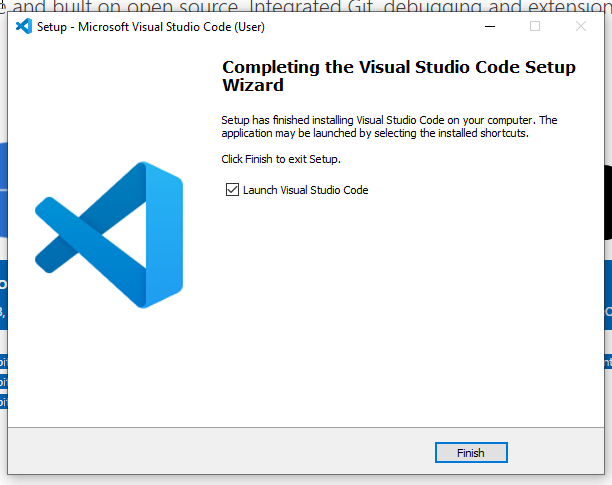
1. Select **Add to PATH** and click **Next >**



1. Click **Install** to begin



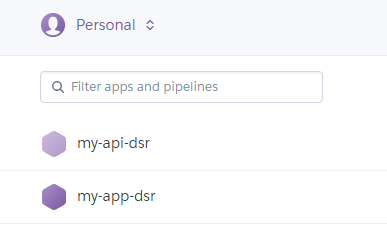
1. Click **Finish** to exit the installer



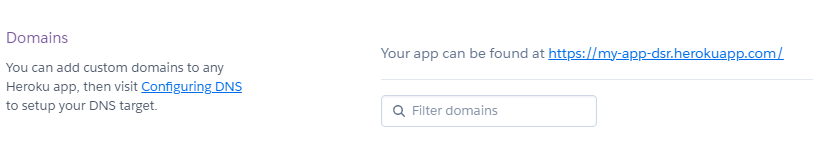
1. Reboot the PC to complete the installation.

# Setting Up Application API

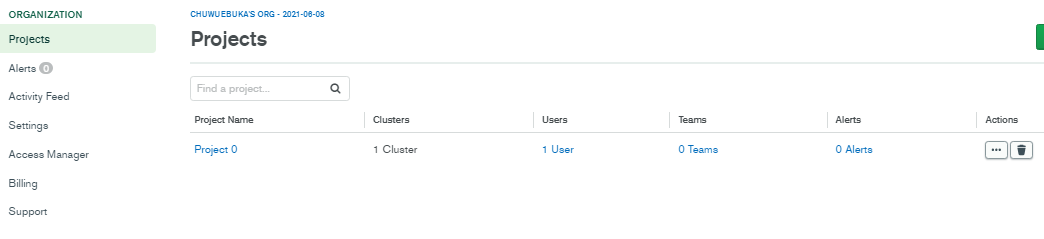
1. Log into **Heroku** and click on your host container. Check on **Setting Up Heroku and Heroku CLI (Windows)** to create the a new host container if it has not been done so.



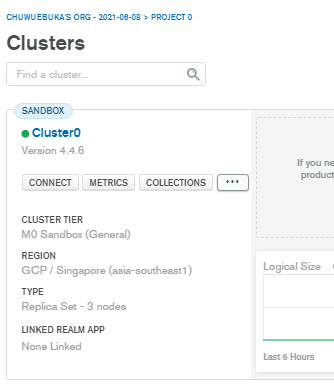
1. Navigate to **Settings**, scroll down to **Domains** and take note of the URL



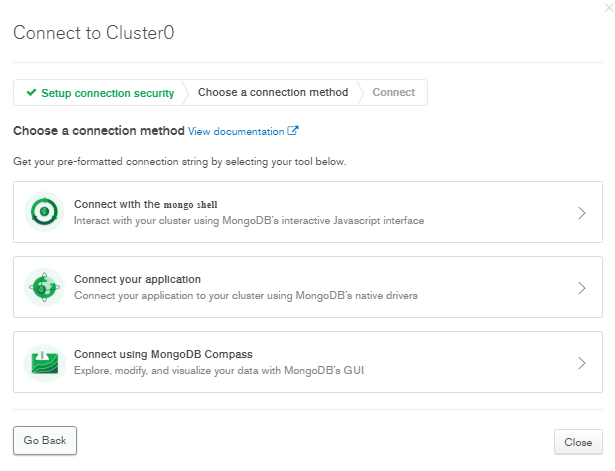
1. Log into **Mongodb** and navigate to **Project** and select the project



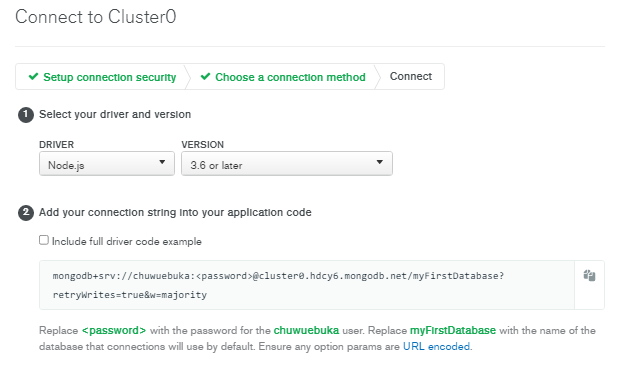
1. Click **CONNECT** under **SANDBOX**



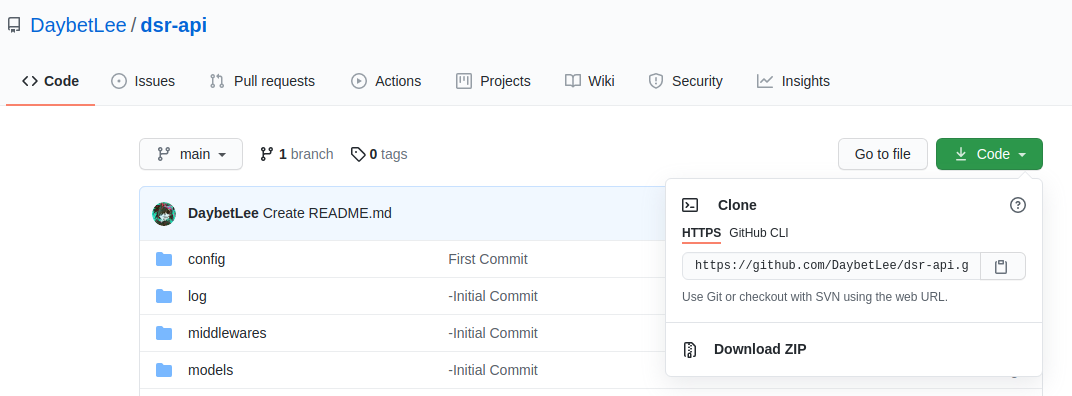
1. Select **Connect your application**



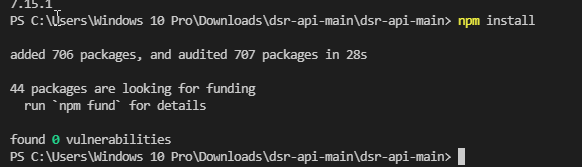
1. Select **Node.js** for **DRIVER** and **3.6 or later** for **VERSION**. Take note of the **driver code**. Replace the **<password>** of the **driver code** with the user password. Ensure the database is the same as the one created in **Setting Up MongoDB** contain the **users** collection. The database name is append in after **mongodb.net/<database name>** in the driver code.



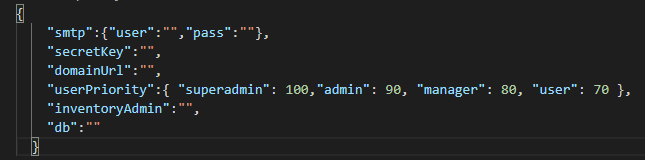
1. Navigate to <https://github.com/DaybetLee/dsr-api/>
2. Click on **Code** and **Download ZIP**



1. Extract the file **dsr-api-main.zip** and open the folder will any integrated development environment (IDE)
2. In the terminal of the IDE, type and enter **npm install** to install the dependencies



1. Open the **config** folder and **production.json** file. Filled in the empty information:

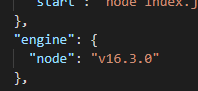


|  |  |
| --- | --- |
| **Key** | **Value** |
| smtp | The smtp server address and the password. |
| secretKey | Self-defined secret key used for jsonwebtoken |
| domainUrl | Heroku domain URL |
| inventoryAdmin | The inventory admin email where the inventory update will be sent to |
| db | MongoDB driver code |

1. Run **node -v** in the terminal and note the node.js version



1. Navigate to **package.json** file and look for **engine**. Update the node version from previous output



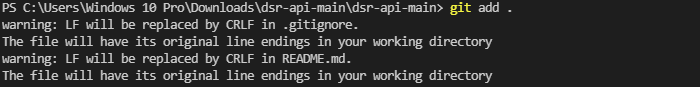
1. In the IDE terminal, enter **git init** to initiate git



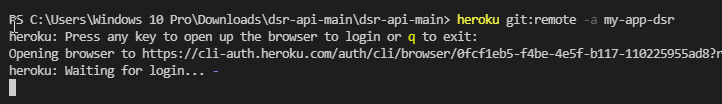
1. Set the global setting email by entering **git config –global user.email <email>**



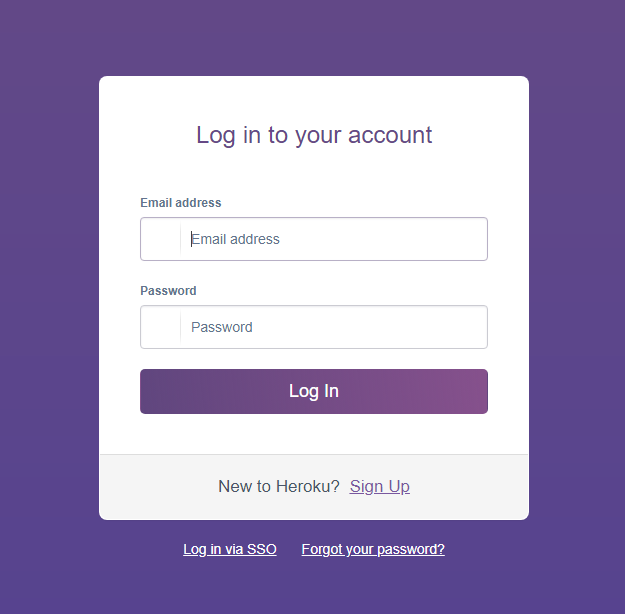
1. Add all file to staging by enter **git add .** in the terminal



1. Initial the first git commit by entering **git commit -m “First commit”** in the terminal
2. In the IDE terminal, enter the **Heroku git:remote -a <app name>**. Ensure Heroku cli and git has been installed prior of this step.



1. The terminal will open the browse to Heroku website, login to complete the adding of Heroku remote git.



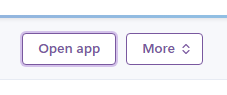
1. Upon successful login, the terminal will prompt the set git remote line



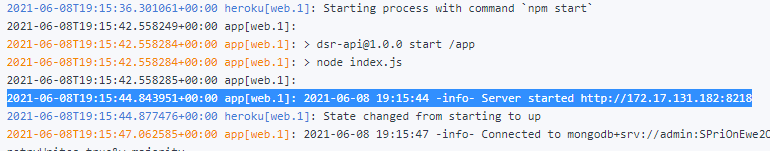
1. Push the local git to Heroku git by entering **git push heroku master.**



1. To verify, access the Heroku. Click on the application and click **More** and **View Logs**

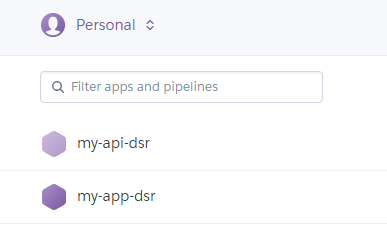


1. The log will show that server is running.

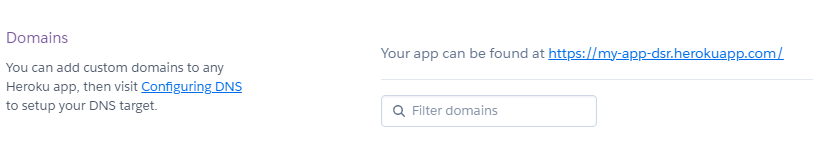


# Setting Up Application Frontend

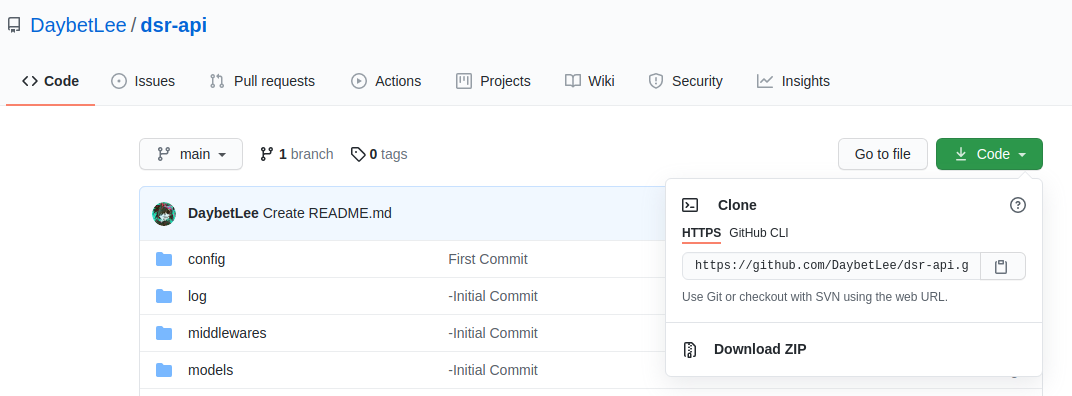
1. Log into **Heroku** and click on your backend application. Check on **Setting Up Heroku and Heroku CLI (Windows)** to create a new host container if it has not been done so. Ensure the backend application is up before proceeding, you can refer to **backend I**



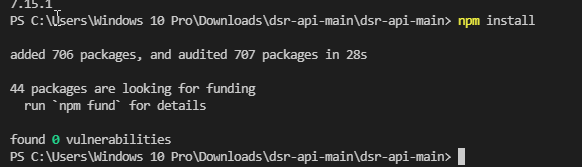
1. Navigate to **Settings**, scroll down to **Domains** and take note of the URL



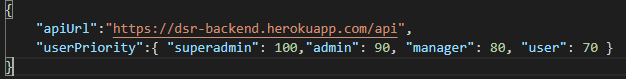
1. Navigate to <https://github.com/DaybetLee/dsr-app/>
2. Click on **Code** and **Download ZIP**



1. Extract the file **dsr-app-main.zip** and open the folder will any integrated development environment (IDE)
2. In the terminal of the IDE, type and enter **npm install** to install the dependencies



1. Open the **config.json** file and add the Heroku backend URL append with **/api**:



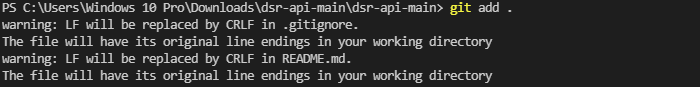
1. In the IDE terminal, enter **git init** to initiate git



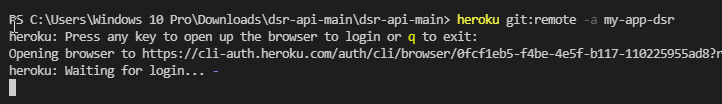
1. Set the global setting email by entering **git config --global user.email <email>**



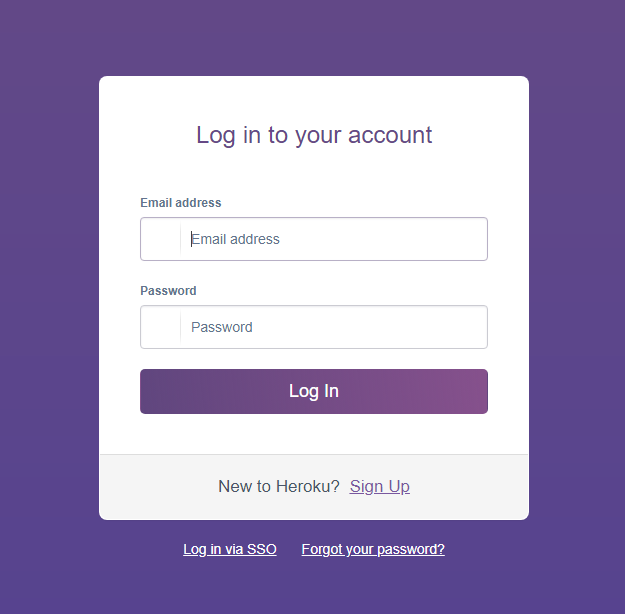
1. Add all file to staging by enter **git add .** in the terminal



1. Initial the first git commit by entering **git commit -m “First commit”** in the terminal
2. In the IDE terminal, enter the **heroku git:remote -a <app name>**. Ensure Heroku cli and git has been installed prior of this step.



1. The terminal will open the browse to Heroku website, login to complete the adding of Heroku remote git.



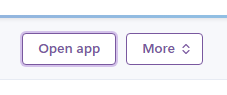
1. Upon successful login, the terminal will prompt the set git remote line



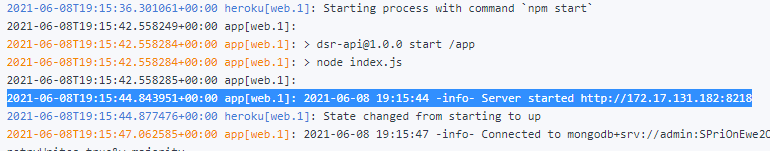
1. Push the local git to Heroku git by entering **git push heroku master.**



1. To verify, access the Heroku. Click on the application and click **More** and **View Logs**



1. The log will show that server is running.



<https://www.mongodb.com/mern-stack> - introduction