

Installation instructions

Acquiring the project

The software is available at [Wpbwww/SRE-International-Team \(github.com\)](https://github.com/Wpbwww/SRE-International-Team). The project can either be cloned through GitHub, or a ZIP file can be downloaded as well.

Once the project has been downloaded, navigate to the folder containing it (if downloading a ZIP, unzip it in a place of your choosing).

Setting up MongoDB

In order to be able to open and analyze repositories, a MongoDB database needs to be created first. Navigate into your MongoDB compass, and create a database with the name 'double-c'. Otherwise, this program will not be able to run.

Setting up the project

Should you encounter an error, please refer to our Error Troubleshooting section.

After opening the project folder, navigate to it in the command line (on windows, you can right click the project folder to get its address as text, which you can then use in the terminal):

```
C:\Users\holub>cd C:\Users\holub\Desktop\SRE-International-Team-main
C:\Users\holub\Desktop\SRE-International-Team-main>
```

Next, use the 'npm run install-dependencies' command to install all the tools required by the software:

```
C:\Users\holub\Desktop\SRE-International-Team-main>npm run install-dependencies
> double-c@1.0.0 install-dependencies
> npm run server-install && npm run client-install

> double-c@1.0.0 server-install
> cd backend && npm install

changed 12 packages, and audited 268 packages in 2s
30 packages are looking for funding
  run `npm fund` for details

11 vulnerabilities (5 moderate, 5 high, 1 critical)
To address all issues, run:
  npm audit fix
Run `npm audit` for details.
```

After installation finishes, your output should look something like:

```
added 3 packages, changed 89 packages, and audited 1794 packages in 57s

149 packages are looking for funding
  run `npm fund` for details

58 vulnerabilities (1 low, 9 moderate, 33 high, 15 critical)

To address issues that do not require attention, run:
  npm audit fix

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
```

After installing the dependencies, execute 'npm run server' to set up a server for the project, assuming all dependencies were installed successfully, the terminal output should look like this:

```
C:\Users\holub\Desktop\SRE-International-Team-main>npm run server

> double-c@1.0.0 server
> cd backend && nodemon server.js --ignore client

[nodemon] 2.0.20
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting node server.js
server is listening on port 4538...
```

Next, open a new terminal window (do not close the first one), and navigate to the project folder. Then, use the 'npm run client' command to set up the frontend side of the software. After running the command, the second terminal should look something this:

```
C:\Users\holub>cd C:\Users\holub\Desktop\SRE-International-Team-main
C:\Users\holub\Desktop\SRE-International-Team-main>npm run client

> double-c@1.0.0 client
> cd client && npm start

> client@0.1.0 start
> react-scripts start

(node:8104) [DEP0148] DeprecationWarning: Use of deprecated folder mapping "." in the "exports" field module resolution
of the package at C:\Users\holub\Desktop\SRE-International-Team-main\client\node_modules\postcss-safe-parser\node_modules\postcss\package.json.
Update this package.json to use a subpath pattern like "./*".
(Use `node --trace-deprecation ...` to show where the warning was created)
[fws]: Project is running at http://0.0.0.0:3000/
[fws]: webpack output is served from
[fws]: Content not from webpack is served from C:\Users\holub\Desktop\SRE-International-Team-main\client\public
[fws]: 404s will fallback to /
Starting the development server...
```

In addition, the default browser on your computer should automatically open your localhost connection, and enter the website.

Navigating the Site

To test out the site's functionality, try importing a repository by pressing the import button:



Try to import the 'Pytorch' github repository:

A form titled "Import Repo" in a dark gray font. Below the title, there are two input fields. The first field is labeled "Owner" and contains the text "pytorch". The second field is labeled "RepoName" and also contains the text "pytorch". Below these fields, there are two blue buttons: "CANCEL" on the left and "IMPORT" on the right, both in white, uppercase, sans-serif font.

For any repository, you can get their import owner and name by looking at the link: github.com/owner/reponame.

After pressing OK, the system will start importing the relevant data, this process may take a while, as long as the load wheel is spinning there is no problem. After a few minutes, you can see your new project, and entering it will allow you to look at the repository analysis:

2 Repos Found In All Repo



Pytorch

Pytorch



Likoli



61004



Dec 16th, 2022

View

Report

TimeLine

- create-at
Aug 13th, 2016
- latest-release
Dec 16th, 2022
- latest-update
Dec 16th, 2022
- latest-push
Dec 16th, 2022

2018-7-27

2018-10-2

SUBMIT

SHOW RELEASE



Once inside the analysis window, you can select the time slice you want to analyse, the repository version you would like to analyse, and so on!

Compatibility Description

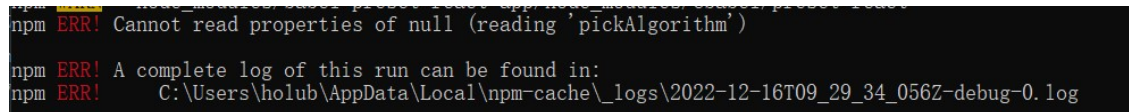
At present, the installation of automatic decompression and compilation of installation packages is not supported. It only supports downloading and decompressing compressed packages on github, and running the project in IDE environment that supports Node.JS.

Release Notes

The detailed update log of each version can be found on [GitHub](#).

Error Troubleshooting

Error 1: Cannot read properties of null (reading 'pickAlgorithm'), or similar disclaimers during execution of 'npm run install-dependencies'



```
npm ERR! Cannot read properties of null (reading 'pickAlgorithm')
npm ERR! A complete log of this run can be found in:
npm ERR! C:\Users\holub\AppData\Local\npm-cache\_logs\2022-12-16T09_29_34_056Z-debug-0.log
```

If during installation of dependencies you run into this error, follow these steps:

1. Using terminal, navigate to the project folder (the same folder where you executed the install-dependencies)
2. Execute the following command: 'npm cache clear -force'
3. Using terminal, navigate to the backend side of the project (use 'cd backend' if already in the project folder)
4. Execute the following command: 'npm cache clear -force'
5. Using terminal, navigate to the client side of the project (use 'cd client' if already in the project folder)
6. Execute the following command: 'npm cache clear -force'

These steps should clear the node cache for the whole project, executing 'npm run install-dependencies' now should lead to no errors. Remember to execute 'npm run install-dependencies' again, otherwise the dependencies will not be installed.


Error 2: an error occurs when importing a repository on the website

If during importing a project, the system immediately stops loading, and shows a red error message, it may be the case that the github authenticator key has switched. authenticator key allows for faster loading of data, however sometimes the key may expire/not work on your machine.

To remedy this, you will need to create your own GitHub authentication token. For steps on how to create one, please refer to [this website](#). Create a classic token, for scope access you do not need to give the token access to anything. In the end, you will be left with a long string which you should copy:

Tokens you have generated that can be used to access the [GitHub API](#).

Make sure to copy your personal access token now. You won't be able to see it again!

✓ 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 

Delete

Then, navigate to `projectfolder/backend/controllers/dash.js`. There, find the 'const octokit' variable, and switch the old auth token with your new one (the variable is located somewhere around line 10):

```
const octokit = new Octokit({  
  auth: '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',  
});
```

The nature of why some auth tokens expire early is unknown to us, the default auth key in our program should function for 90 days since the writing of this guide (until 16th of march), however it may be deleted prematurely.