

Setting up connected worlds with your data using GitLab, Bitbucket or GitHub

Contents:

- Notes
- Steps
- Importing Connected Worlds
 - Importing into GitLab
 - Importing into GitHub
 - Importing into Bitbucket
- Deploying
 - Setting up surge and node
 - Setting up your CI service
 - * GitLabCI
 - * CircleCI
- Using your own data

Notes:

1. This guide is designed for someone with a reasonable knowledge of the chosen git services and some technical experience.
2. GitLab will use GitLab CI and both GitHub and Bitbucket will use CircleCI.
3. It is assumed that you will have all relevant accounts and all the needed permissions to follow this guide.
4. It also assumes you are deploying to surge.sh, if you are not then you will need to edit the deploy script to suit your implementation.

Steps:

The only two required steps for running your own connected worlds platform are importing the repository and setting up surge deployment.

If you are planning on deploying to somewhere other than surge or don't plan to deploy in this way you may skip the surge deployment section of the below guide. Due to the sheer number of platforms available to deploy to, we have only made connected worlds support surge by default. This means that if you want to deploy elsewhere you will need to edit the CI scripts your self. This will require more technical knowledge than following this guide as less of the work will have been done for you. Not doing this may require some code to be removed from the CI files if you still use a CI service. If you just want to deploy

locally then you can download the project to your pc and run `yarn start` from a terminal.

Importing Connected Worlds:

Note in all cases the repository may take a few minutes to import.

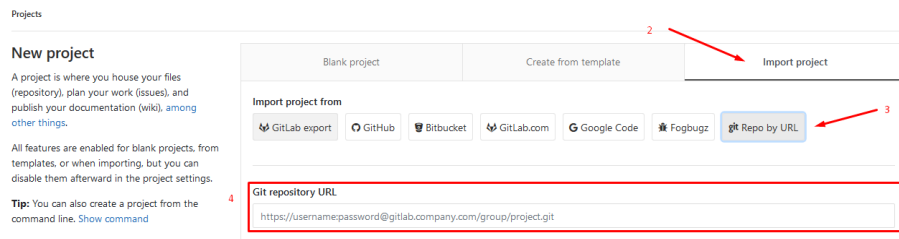
Importing into GitLab:

1. Login to GitLab.
2. Click on "New project" in the top right corner of the screen.
3. Click on "Repo by URL".
4. Click on "Import project" and then paste the following URL:

`https://github.com/eResearchSandpit/VR-Network-Vis`

or here if you have access:

`https://gitlab.ecs.vuw.ac.nz/ENGR301-302-2018/Project-10/VR-Network-Vis`

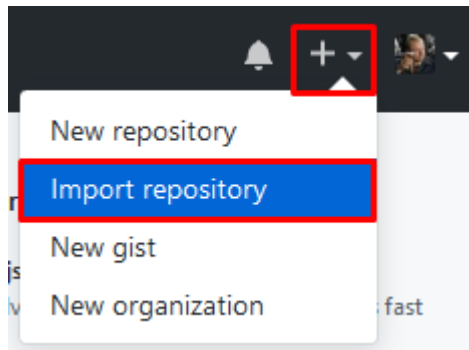


1. Scroll down and rename the project if you would like.
2. Set the visibility level of the project (Private, Internal or Public).
3. Click create project.

In the deployment section follow the GitLabCI guide NOT the CircleCI guide.

Importing into GitHub:

1. Login to GitHub.
2. Press the "+" symbol in the top right corner of GitHub next to you profile picture.
3. Click "Import repository".



1. On the following screen input a name for the repo and then paste the following URL:

<https://github.com/eResearchSandpit/VR-Network-Vis>

or here if you have access:

<https://gitlab.ecs.vuw.ac.nz/ENGR301-302-2018/Project-10/VR-Network-Vis>

1. Press begin import.

Import your project to GitHub

Import all the files, including the revision history, from another version control system.


Your old repository's clone URL

`http://svn.example.org/code/svn`

Learn more about the types of [supported VCS](#).

Your new repository details

Owner

 BoeVonLipwig ▾

Name

Privacy



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

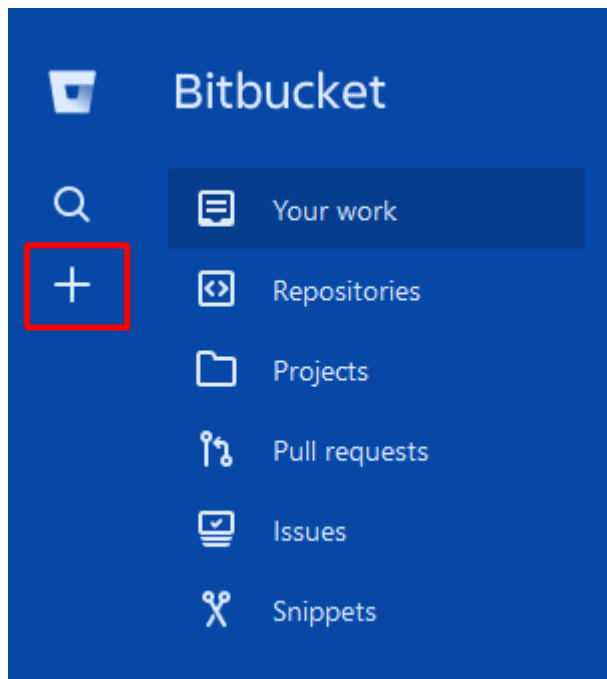
[Cancel](#)

[Begin import](#)

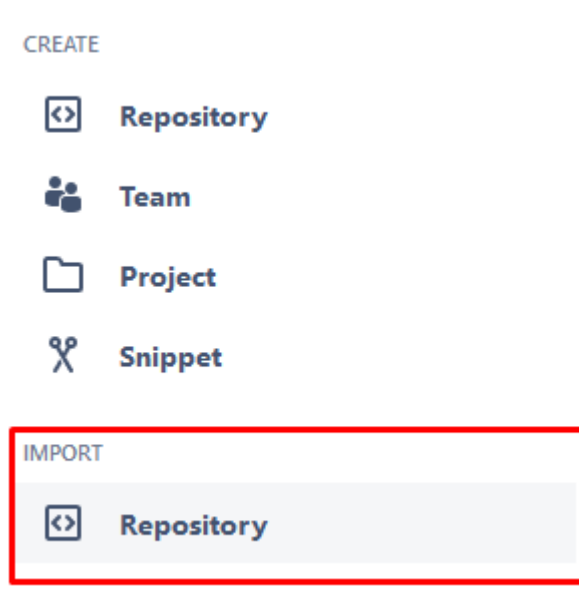
In the deployment, section follow the CircleCI guide NOT the GitLabCI guide.

Importing into Bitbucket:

1. Start by logging in to Bitbucket.
2. Press the "+" symbol in the top left of your screen in the menu bar



1. Then press "Repository" under the import section of the menu.



1. Then name the repository and input the following URL:
<https://github.com/eResearchSandpit/VR-Network-Vis>

or here if you have access:

<https://gitlab.ecs.vuw.ac.nz/ENGR301-302-2018/Project-10/VR-Network-Vis>

Import existing code [Create new repository](#)

Old repository

Source Git

URL *

☐ Requires authorization

New repository

Owner Sindlashau

Repository name *

Access level ☐ This is a private repository

Version control system ☒ Git ☐ Mercurial

[Advanced settings](#)

[Cancel](#)

1. Set the visibility level of the project (Private or Public).
2. Click "Import repository".

In the deployment section follow the CircleCI guide NOT the GitLabCI guide.

Deploying:

Note that if you are using GitLab the CI script is set up to deploy to GitLab pages automatically.

Setting up surge and node:

1. Install NodeJS from the following link: Node installer and follow the on screen instructions.

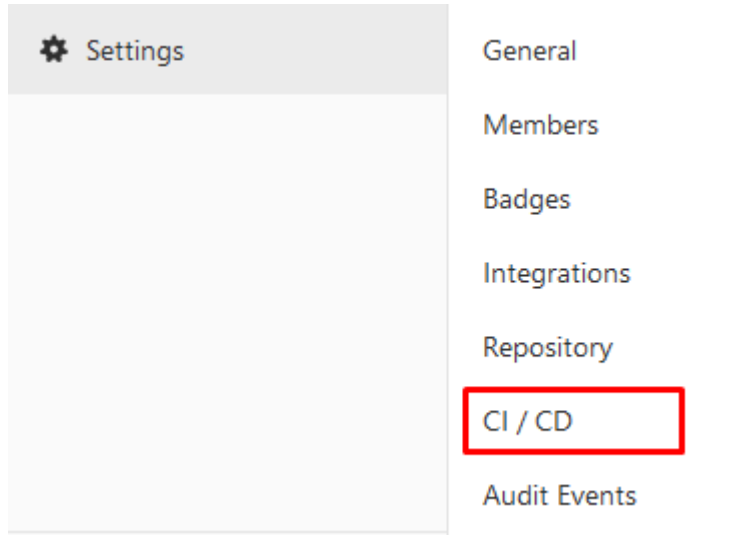
2. Open cmd or some command line equivalent and run the following commands: `npm install --global surge surge token` and follow the instructions on screen(remember the details you input they will be used).
3. Save the token that surge outputs for later.

Setting up your CI service:

Note: follow the GitLabCI guide if you are using GitLab and the CircleCI guide if you are using CircleCI.

GitLabCI:

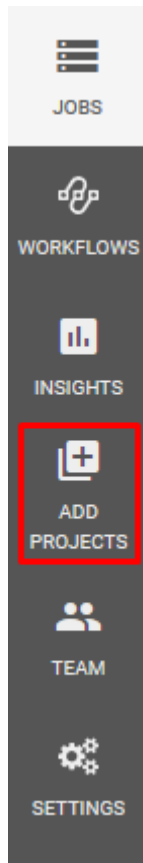
1. Go to the "CI/CD" tab in settings:



1. Expand the "Variables" tab and add three fields called:
 - (a) "SURGE_LOGIN".
 - (b) "SURGE_TOKEN".
 - (c) "URL".
2. Input the following values into each field:
 - (a) In the "SURGE_LOGIN" field input the email you used to generate the token.
 - (b) In the "SURGE_TOKEN" field input the token that you got from the "Setting up surge and node" step.
 - (c) In the "URL" field input whatever URL you want to use for your site so long as it ends in .surge.sh and is not already taken(You can test this by trying to visit it).
3. Click "Save".

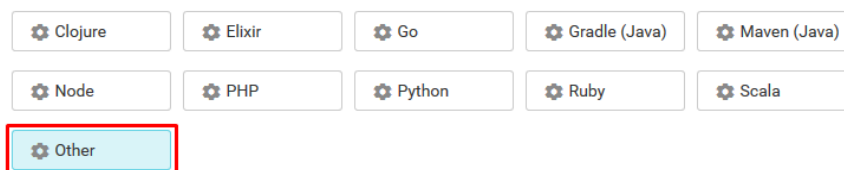
CircleCI:

1. Login to CircleCI using the account related to your project.
2. Click add project button:



1. Locate your project in the list of repositories and click add project.
2. From the change the language from node to other:

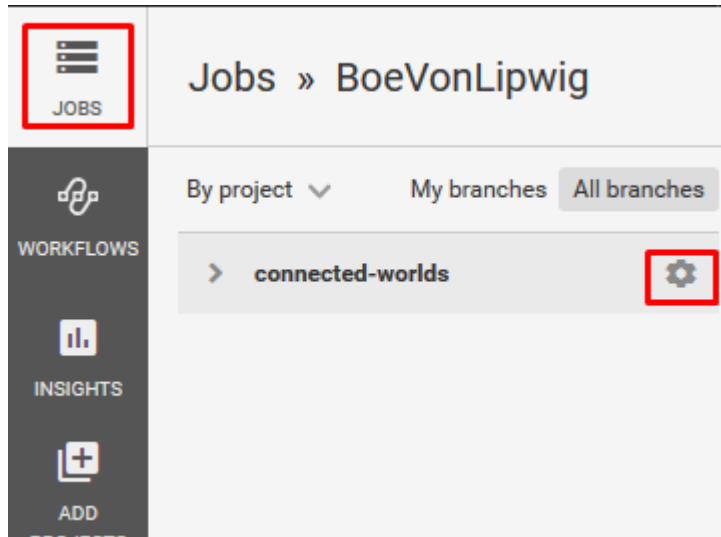
Language



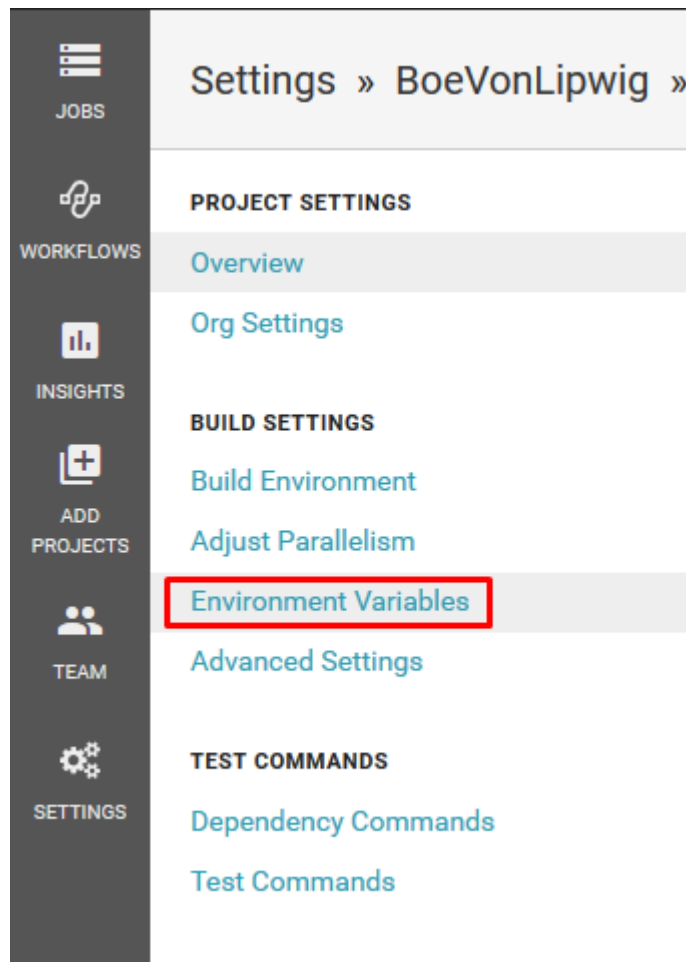
1. Click "Start Building"
2. The build should fail at this point as the environment variables have not

been set

3. Click jobs and then the settings cog next to the correct repository name.



1. Click on "Environment Variables"



1. Click add variable and enter the names and values into the appropriate slot:

Environment Variables for BoeVonLipwig/connected-worlds Import Variables Add Variable

Add environment variables to the job. You can add sensitive data (e.g. API keys) here, rather than placing them in the repository.

Add an Environment Variable

To disable string substitution you need to escape the `$` characters by prefixing them with `\`. For example, a value like `usd$` would be entered as `usd\$`.

Name

Value

Cancel

Add Variable

#	Name	Value
1	"SURGE_LOGIN"	The email you used to generate the token
2	"SURGE_TOKEN"	The token that you got from the "Setting up surge and node" step
3	"URL"	The URL you want to use for your site so long as it ends in .surge.sh and is not already taken(You can test this by trying to visit it)

Using your own data

See the data-uploading-and-formatting document for instructions.