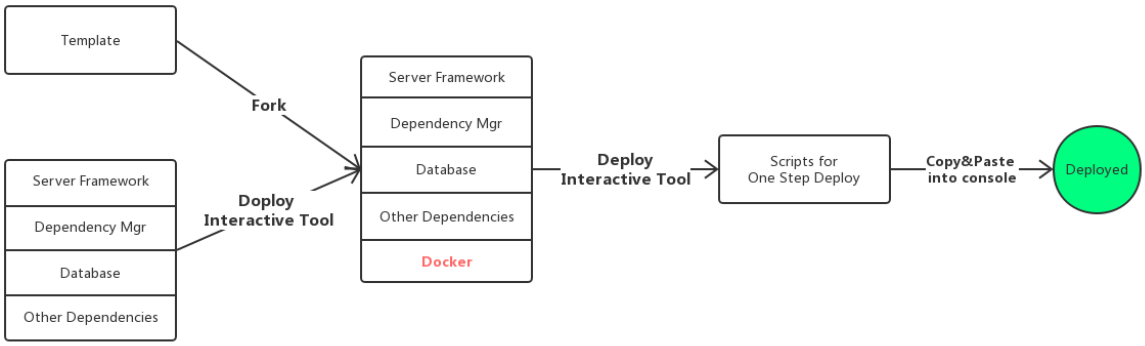
Doploy Build Document

Zhufeng Xu(zx2245), Mengyu Han(mh3881)

Implementation of Doploy

Doploy is a Docker-based configuration tool that helps deploy backend server for a web application in a faster and simpler way.

The whole project generate Docker scripts according to different configurations of servers. Typically, it takes in JSON configurations and outputs several script files that activate an one-step deploy feature.



All functionalities including:

- (Build deployable feature in customized projects) Automatic Docker scripts generator
- (Build deployable feature using our template projects) Template branches redirector
- Interactive configuration website

All implementations and documents in the process graph can be accessed in the Doploy Github repository.

Interactive configuration website can be accessed with the link here. And the front end codes for Interactive website can be accessed here.

Note that the interactive website, along with its backend codes, resides on my private AWS server with the domain name *www.do-ploy.com* which was bought 1 month ago, and with a valid HTTPS certificate. You can also re-build the project according to the build details below.

Build Details

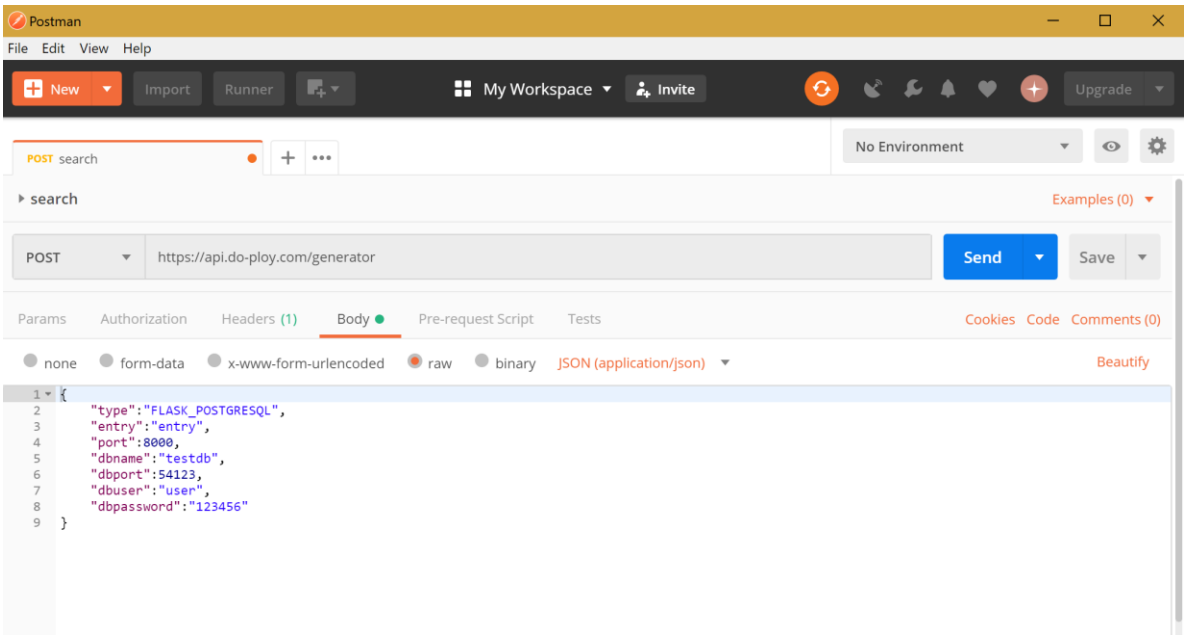
The whole Doploy project backend is based on:

- Java 8+
- Maven 3.5.2+
- Springboot 2.1.1+

- MySQL 8+

The functionality part is in the master branch of its Github project. The rebuild steps are:

1. Download the Github project and the dependencies above
2. Create the default database 'alansite'(it's defined in the src/main/resources/application.yaml) with both username and password 'doploy'.
3. Locate the bash terminal to the root directory of the project
4. Run '*mvn package && java -jar target/Doploy-LATEST.jar*' to build and run the project
5. The local server will be in the port of 8090(can be accessed with localhost:8090/generator, but the server exposes only REST API with a POST method, you may need extra tool like Postman to simulate the API calls. A sample call screenshot with Postman is attached below)



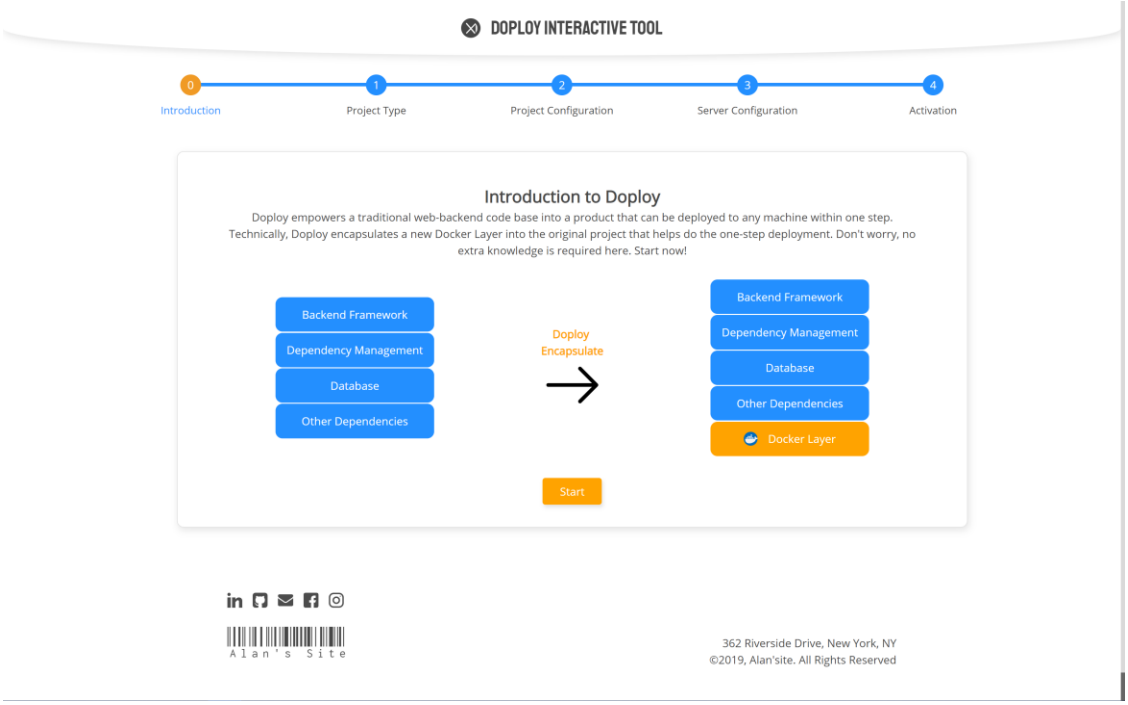
For interactive website, the frontend part, there're other dependencies:

- Parcel.js
- Node.js

The rebuild steps are:

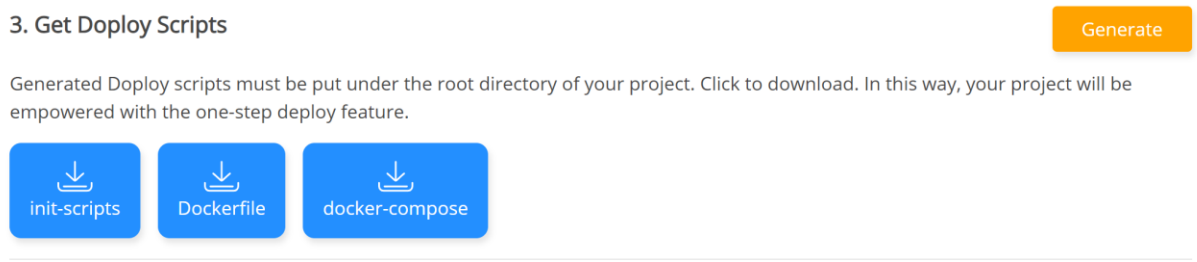
1. Download the Github project and dependencies above
2. Locate the bash terminal to the root directory of the project
3. Run '*npm run start*' to deploy the website locally on port 1234

After the build, you can see the frontend interactive website like this:



This is also consistent with the online version on https://www.do-ploy.com/doploy.

Another thing to mention is, since Doploy generates files for customized projects (see the screenshot below), the storage of these files is supported by [AWS S3](). This is not free, and if you want to generate customized script files, you will also need to register your own S3 and change the bucket name and the corresponding bucket region in 'src/main/java/cn/alandelip/logic/impl/S3Upload.java'.



Other Related Documents

Different preset Doploy templates: You can access them in [different branches of the Github project]()

How to define Docker build files: [Docker Document]()

Where to find sample Docker files: [Docker Hub]()

How to generate customized project configurations: [Apache FreeMarker Document]()