Front End:

Applications used:

Node.js

express

Eslint

Jest

Supertest

Node.js and express were used to build a small and simple front-end application. Previously an apache server was used but to enable linting and testing a switch to node was made.

Eslint is a pluggable static code analyzer for javascript, it is configurable via yml file and works with node through npm.

Jest is a testing framework for javascript enabling unit tests. Struggles with choosing unit tests since the front end barely even has functions there’s nothing really to unit test.

Front-End:

Simple express server application, returning the index.html when addressed. The index.html site has a button which upon being pressed request the data from the backend and displays it on the page.

Server index.js

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Main Page that gets sent as the response

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Linting:

Eslint settings:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

The env: section is to let the linter know which global variables it can expect.

Extends: “eslint:recommended” activates a subset of recommended rules

run by using “npm run lint”

Testing:

Test to check if the app.get() function returns something or is empty, using jest

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Making the server and checking if it responds.

Run by using “npm test”

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

The forceExit flag forces jest to exit after all tests have been run through.

Pipeline:

Azure devops Pipeline

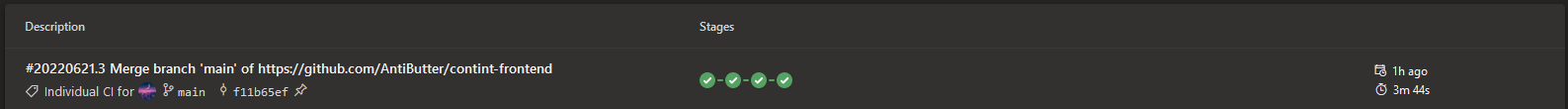
4 Stages:

Linting

Testing

Building

Pushing/Deployment



Pipeline run through triggered by a push to the main branch or manually.

Azure devops yamls File:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Image gets tagged by the build id which is the number assigned to the pipeline when it’s ran. Number of times the pipeline has been run.

Linting stage, sets the environment and executes the npm run lint command to start eslint.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Testing stage runs the test command.

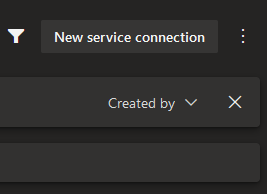
Ein Bild, das Text enthält.

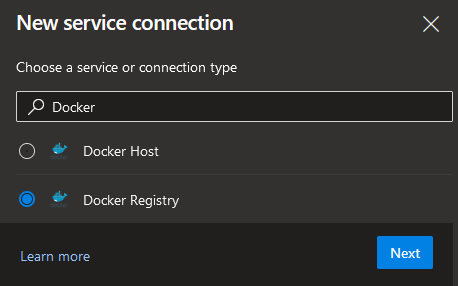
Automatisch generierte Beschreibung

Build and Publish stage, using the azure devops docker task with a service connection to dockerhub to build and then push the image to dockerhub.

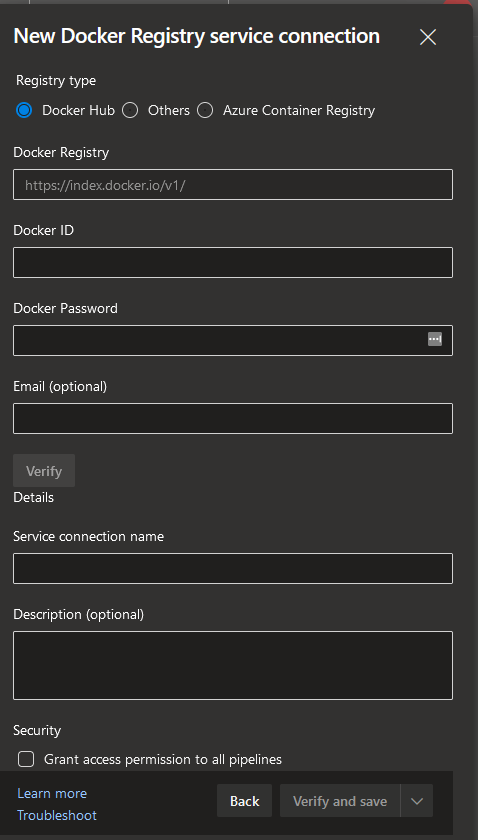
Service Connection:

Project Settings -> Service Connections -> new Service Connection





Docker Registry



To store the dockerhub login information safely

Other service connection is the Github Repository so the pipeline can react to pushes to the main branch.

Successfully Run Pipeline:

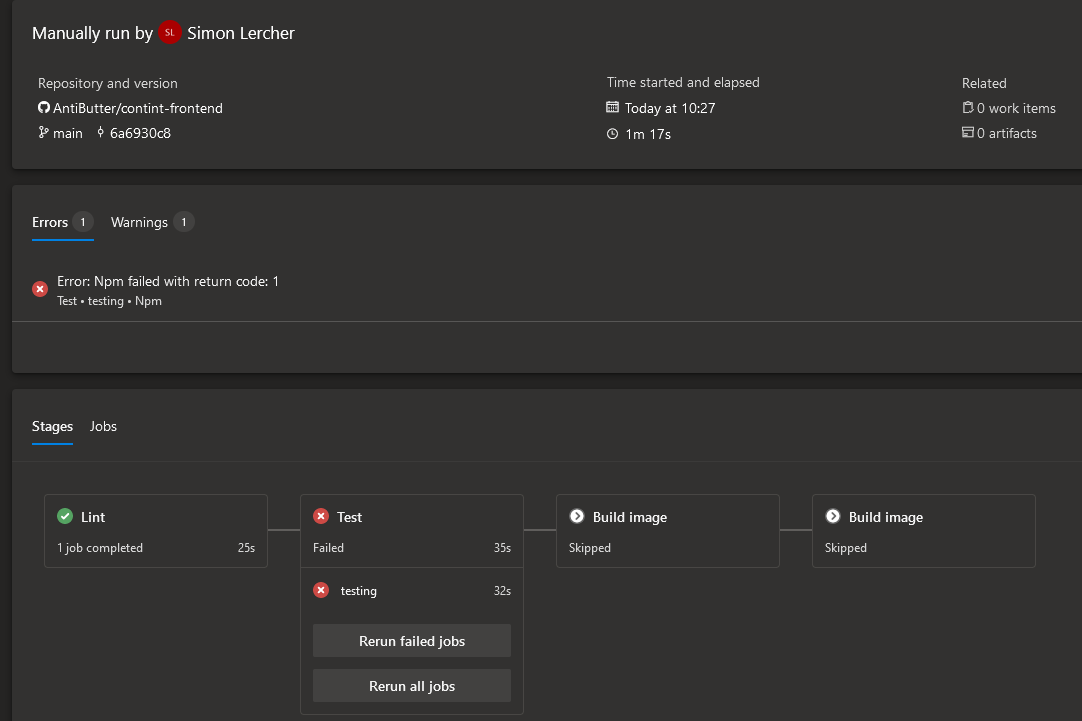
Triggered by Push to the Github repository.

Ein Bild, das Text, Screenshot, Monitor, Bildschirm enthält.

Automatisch generierte Beschreibung

Failed Pipeline:

Manually Run



Running the program:

Pulling from dockerhub -> docker pull antibutter/frontend:$(desiredVersion)

docker run -p 5000:5000 antibutter/frontend:$(pulledVersion)

Links:

<https://github.com/AntiButter/contint-frontend>

<https://hub.docker.com/repository/docker/antibutter/frontend>

<https://dev.azure.com/if20b117/Conint-Frontend/_build>