


# Jenkins Pipeline


Autor: Michał Sarota

1. Aby zbudować pipeline, potrzebujemy
  - poprawnie zainstalowanego Jenkinsa
2. Tworzenie pipeline
  - Aby rozpocząć pracę, należy dodać nowego joba i wybrać pipeline”


**Enter an item name**  
  
» Required field




**Freestyle project**  
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used to build other Jenkins projects.




**Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and other complex build systems.




**Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, or different build parameters.



**Folder**  
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates items as long as they are in different folders.




**Multibranch Pipeline**  
Creates a set of Pipeline projects according to detected branches in one SCM repository.



**Organization Folder**  
Creates a set of multibranch project subfolders by scanning for repositories.


If you want to create a new item from other existing, you can use this option:



Copy from

**OK**

- Następnie zaznaczamy odpowiednie opcje

 **Jenkins**

Dashboard › sprawozdanie ›

GeneralBuild TriggersAdvanced Project OptionsPipeline

Description

[Plain text] [Preview](#)

☐ Discard old builds ?

☐ Do not allow concurrent builds

☐ Do not allow the pipeline to resume if the controller restarts

☒ GitHub project

Project url ?

☐ Pipeline speed/durability override ?

☐ Preserve stashes from completed builds ?

☐ This project is parameterized ?

☐ Throttle builds ?

## Build Triggers

- ☐ Build after other projects are built ?
- ☐ Build periodically ?
- ☐ GitHub hook trigger for GITScm polling ?
- ☐ Poll SCM ?
- ☐ Disable this project ?
- ☐ Quiet period ?
- ☐ Trigger builds remotely (e.g., from scripts) ?

# Pipeline

## Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

[https://github.com/InzynieriaOprogramowaniaAGH/MDO2022\\_S.git](https://github.com/InzynieriaOprogramowaniaAGH/MDO2022_S.git)

Credentials ?

- none -

 Add

Add Repository

**Add Repository**

**Branches to build** ?

**Branch Specifier (blank for 'any')** ?

\*/MS400537\_lab5

**Add Branch**

**Repository browser** ?

(Auto)

**Additional Behaviours**

**Add** ▾

**Script Path** ?

ITE/GCL07/MS400537/Lab05/Jenkinsfile

☒ **Lightweight checkout** ?

**Pipeline Syntax**


**Save** **Apply**

- Linki
  - a. Project URL
    - i. [https://github.com/InzynieriaOprogramowaniaAGH/MDO2022\\_S/tree/MS400537\\_lab5/ITE/GCL07/MS400537/Lab05](https://github.com/InzynieriaOprogramowaniaAGH/MDO2022_S/tree/MS400537_lab5/ITE/GCL07/MS400537/Lab05)
  - b. Repository URL
    - i. [https://github.com/InzynieriaOprogramowaniaAGH/MDO2022\\_S.git](https://github.com/InzynieriaOprogramowaniaAGH/MDO2022_S.git)


- c. Branch specifier
  - i. \*/MS400537\_lab5
- d. Script path
  - i. ITE/GCL07/MS400537/Lab05/Jenkinsfile


### 3. Build


Aby utworzyć build, należy w dashboardie pipeline wybrać "Build now"


 **Jenkins**


Dashboard ▸ sprawozdanie ▸


 Back to Dashboard


 Status


 Changes


 Build Now


 Configure


 Delete Pipeline


 Full Stage View

 GitHub


 Open Blue Ocean

 Rename


 Pipeline Syntax


 Build History

trend ^


 Filter builds...

No builds

 Atom feed for all

 Atom feed for failures

## Pipeline sprawozdanie

 Recent Changes

## Stage View

No data available. This Pipeline has not yet run.

## Permalinks

Pipeline zawiera 5 kroków: checkout scm, build, test, publish i cleanup

Link do pliku:

[https://github.com/InzynieriaOprogramowaniaAGH/MDO2022\\_S/blob/MS400537\\_lab5/ITE/GCL07/MS400537/Lab05/Jenkinsfile](https://github.com/InzynieriaOprogramowaniaAGH/MDO2022_S/blob/MS400537_lab5/ITE/GCL07/MS400537/Lab05/Jenkinsfile)

checkout scm - pobranie informacji dotyczących repozytorium, branchy, przejście do odpowiednich katalogów

build - zbudowanie projektu

- odbywa się to poprzez zbudowanie następującego dockerfile

```
FROM node:16
RUN git clone https://github.com/iamkun/dayjs.git
WORKDIR dayjs
RUN npm install
RUN npm run build
```

- Za pomocą pliku budowany jest kontener, który klonuje repozytorium, pobiera dependencje oraz buduje projekt

test - włączenie automatycznych testów zawartych w projekcie

- odbywa się to poprzez zbudowanie następującego dockerfile

```
FROM builder
WORKDIR dayjs
RUN npm run test
```

- Za pomocą pliku budowany jest kontener na bazie poprzedniego (buildera) i uruchamiane są w nim testy automatyczne

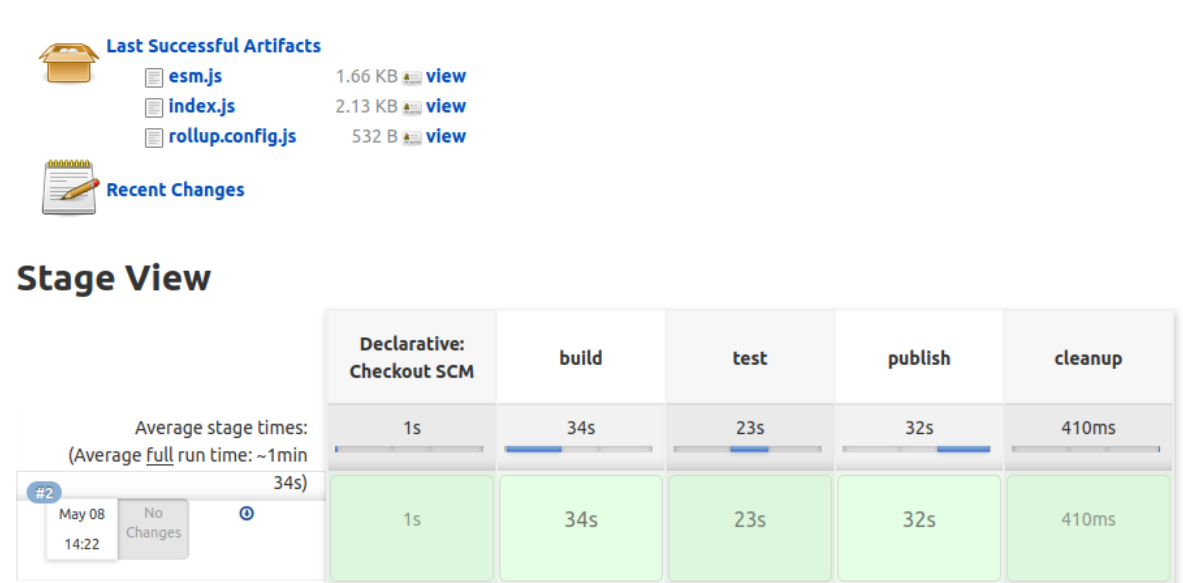
publish - opublikowanie katalogu build i umożliwienie jego ściągnięcia jako gotowy program. Ściągnąć wynik można poprzez wejście w szczegóły builda.

- Z kontenera "builder" przekopiywana jest zawartość folderu build, w którym znajdują się pliki będące rezultatem wykonania komendy odpowiadającej za budowanie

cleanup - usunięcie niepotrzebnych już martwych kontenerów.

deploy - krok niezawarty w pipeline. W rzeczywistości w tym kroku nasz projekt umieszczali byśmy gdzieś na naszych serwerach/bucketach po to, żeby został on dostarczony do innych (przykładowo - jeśli tworzymy aplikację internetową umieszczoną na serwerach AWS, to deploy jest przeniesieniem folderu wynikowego build na odpowiedni bucket S3, do którego łączą się klienci korzystający z naszego rozwiązania).

## Wynik builda



Widzimy, że wszystkie stage przeszły poprawnie.  
Dodatkowo do pobrania mamy artefakty.