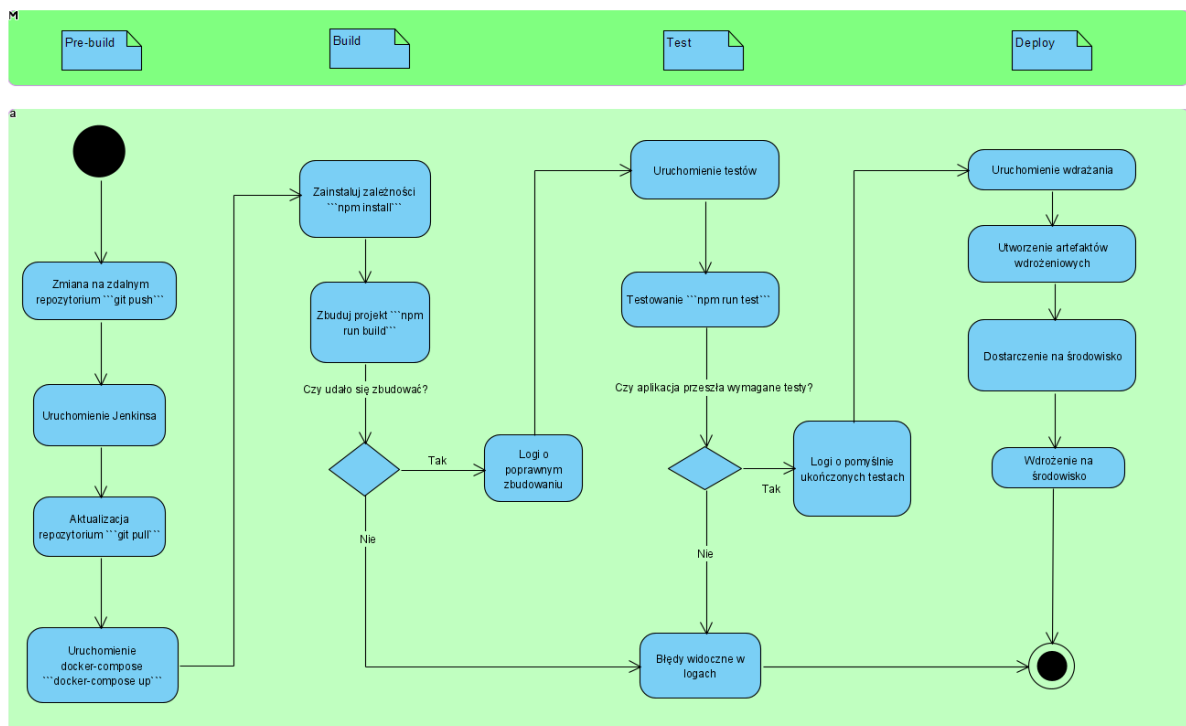


<b>Imię i nazwisko:</b> Tomasz Artecki	<b>Wydział:</b> WIMiP	<b>Grupa laboratoryjna:</b> 1
<b>Sprawozdanie 6</b> <b>Temat:</b> Pipeline CI/CD Plan		<b>Data oddania:</b> 22.04.2022

## 1. Wykorzystane technologie

- Git, Github
- Typescript, node.js
- Docker, docker-compose
- Docker hub
- Jenkins
- Komunikator deltachat ( <https://github.com/deltachat/deltachat-desktop> )

## 2. Diagram aktywności



## 3. Omówienie pipeline'u

- Początek pipeline – kiedy wykryte są zmiany na zdalnym repozytorium **git push** uruchamia się Jenkins i wykonuje instrukcje zawarte w Jenkinsfile.
- Repozytorium jest aktualizowane do najnowszej wersji **git pull**
- W odpowiednim folderze z plikiem docker-compose uruchamiany jest proces budowania za pomocą **docker-compose up**
- Instalacja wymaganych zależności oraz budowanie projektu, kolejno **npm install** oraz **npm run build**

- e. W przypadku gdy budowanie się nie powiedzie Jenkins kończy swoją pracę pozostawiając logi z błędami. W przeciwnym wypadku możemy zobaczyć logi o poprawnym zbudowaniu i przechodzimy dalej.
- f. Kolejną fazą jest uruchomienie testów oraz testowanie **npm run test**
- g. Jeśli testowanie się nie powiedzie to podobnie jak powyżej Jenkins kończy pracę zostawiając logi, a w przeciwnym wypadku kontynuuje przepływ.
- h. Kolejną fazą jest uruchomienie wdrażania i utworzenie artefaktów wdrożeniowych.
- i. Jenkins kończy pracę dostarczając i wdrażając program na środowisko produkcyjne.

#### 4. Diagram wdrożeniowy

