

Lab10 Omówienie implementacji

Autor:

Kamila Nowak, 306481

Przygotowano dla:

Aplikacji będącej prostym komunikatorem wysyłającym i odbierającym ciąg znaków.

Kiedy:

Projekt przygotowano w ramach zajęć z przedmiotu Metodyki Devops podczas kilku laboratoriów.

Implementacja następujących technologii w końcowym projekcie:

- Git (github, git hooks, webhooks)
- Docker (compose, registry)
- Jenkins (Jenkinsfile)
- JavaScript.
- Język powłoki bash

UML Diagram:

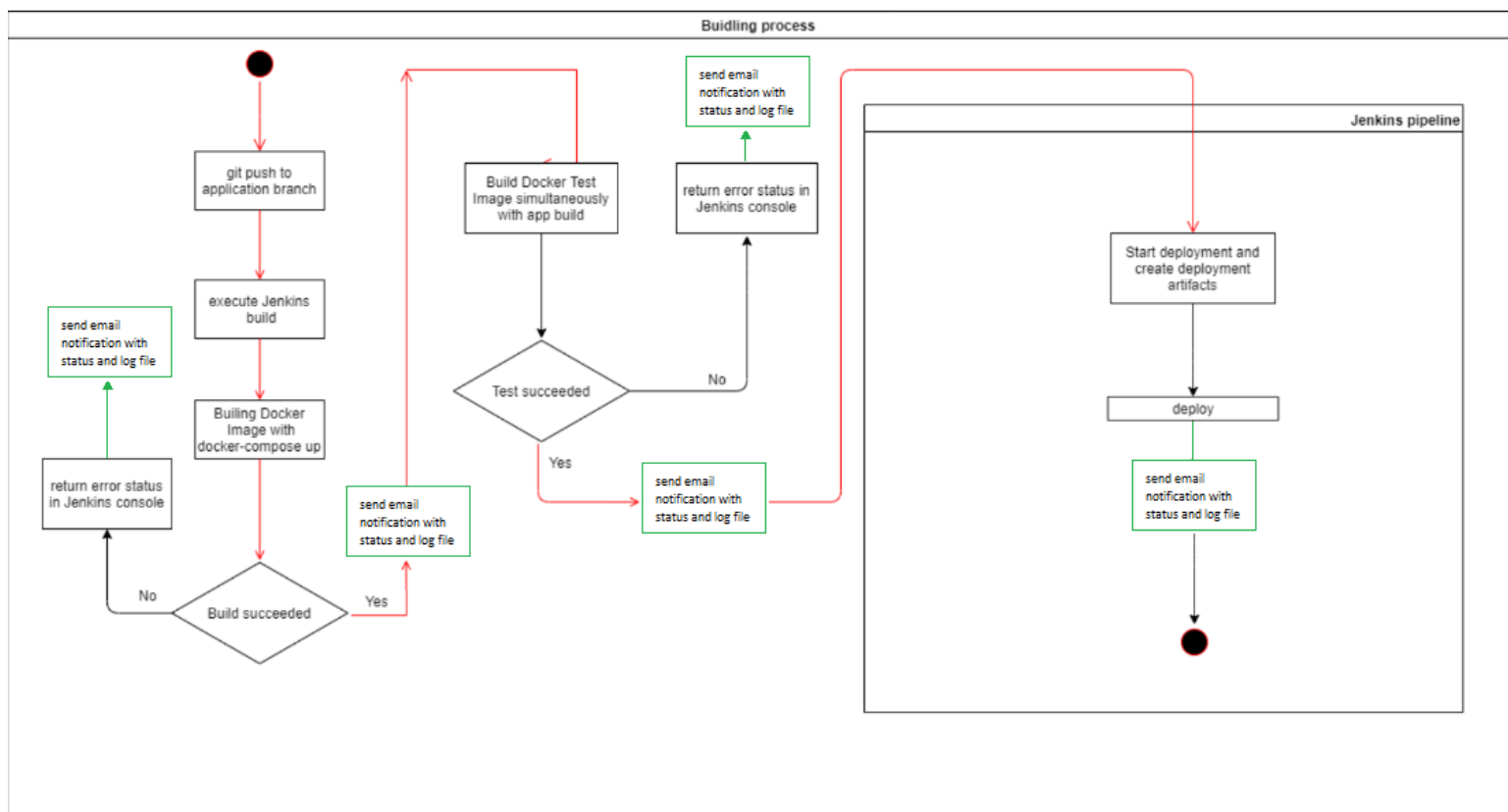
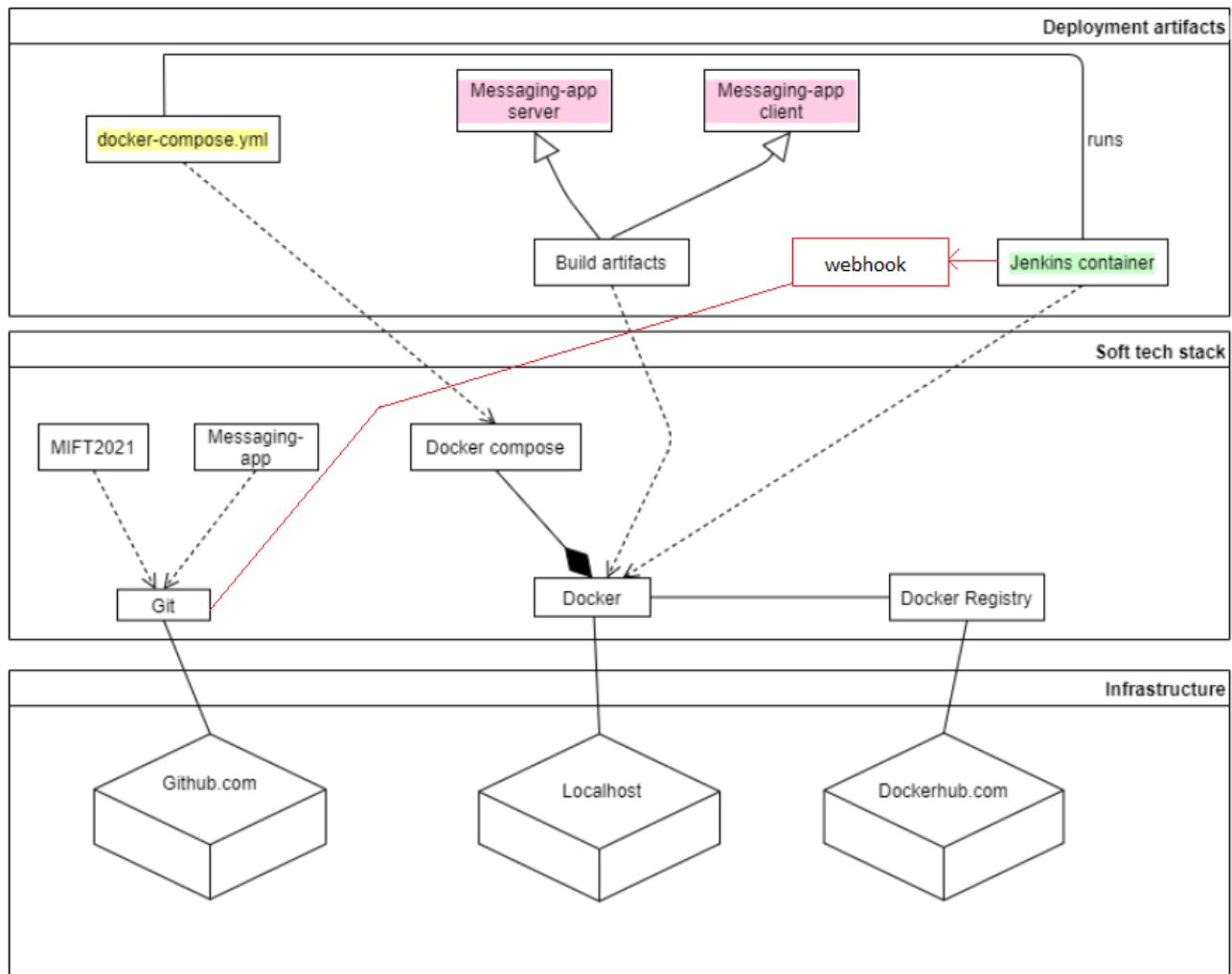


Diagram wdrożeniowy UML:



Tabelaryczne zestawienie do diagramu UML:

Link do pliku: <https://github.com/KamilaNowak/messaging-app-for-docker/blob/master/Jenkinsfile>

Krok: wykonanie build Jenkinsa	Technologia: Jenkins	Link do pliku powyżej	Linia: 6	Zgodność z planem: Tak, stage build poprawnie się wykonuje
budowa i uruchomienie obrazu dockera	Docker	powyżej	12	Tak, czasem błędy z powodu już działającego obrazu. Rozwiązanie: odkomentowanie 2 linijek i usunięcie obrazów
budowa stage'u testowego	Jenkins	powyżej	23	Tak, wykonuje się poprawnie

deployment artefaktów	Jenkins	powyżej	39	Tak, status wysyłany na adres e-mail (tak jak w poprzednich stage;ach)
wysyłanie notyfikacji mailowych	Jenkins + SMTP	powyżej	57	status oraz plik z logami wysyłany na email po każdym stage'u

Różnice w diagramie wdrożeniowym:

dodanie webhooka	Github + Jenkins	konfiguracja Github i Jenkins	-	Tak, po zrobieniu push na repozytorium build Jenkinsowy uruchamia się ponownie
------------------	------------------	-------------------------------	---	--