

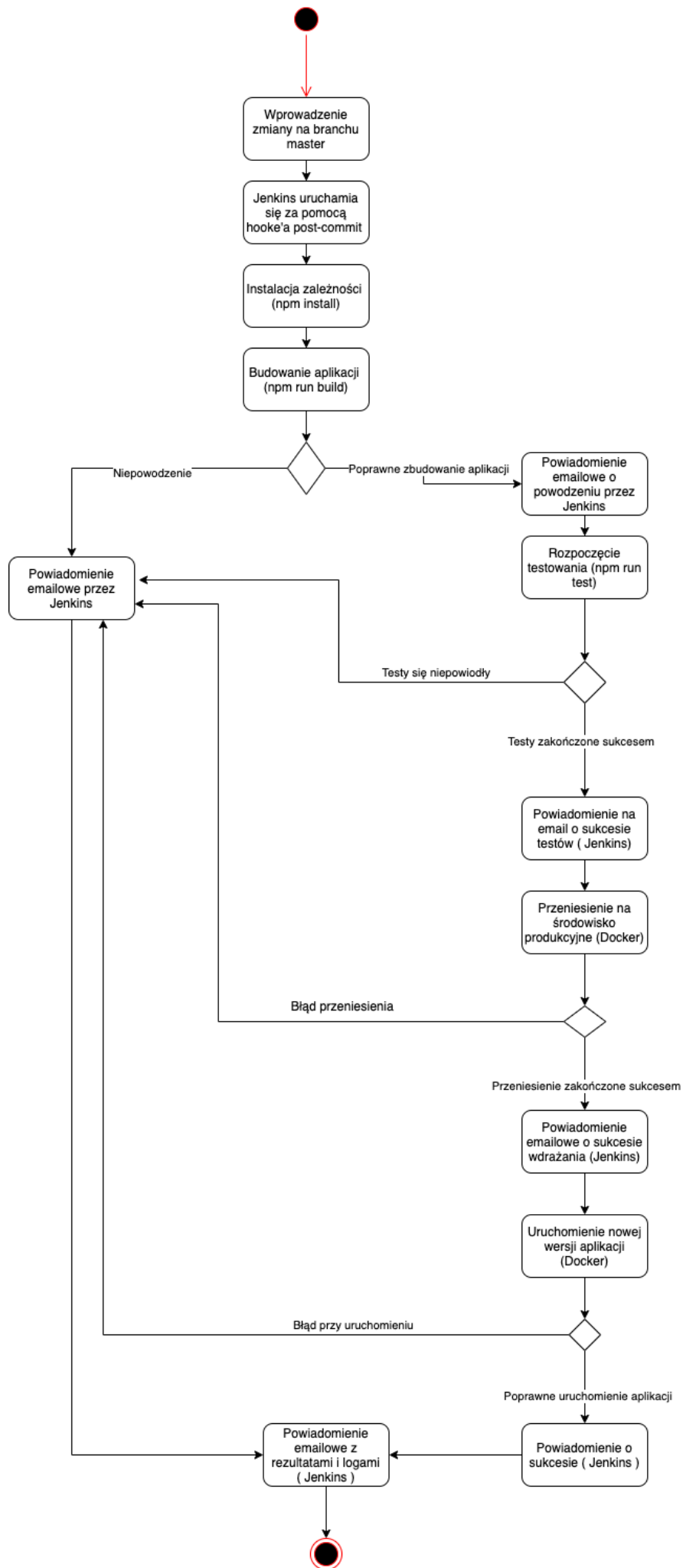
Imię i nazwisko: Marcin Zachura

Czas wykonania: 23.04.2021 - 14.06.2021

Rzeczywisty plan wdrożeniowy aplikacji “Deltachat”

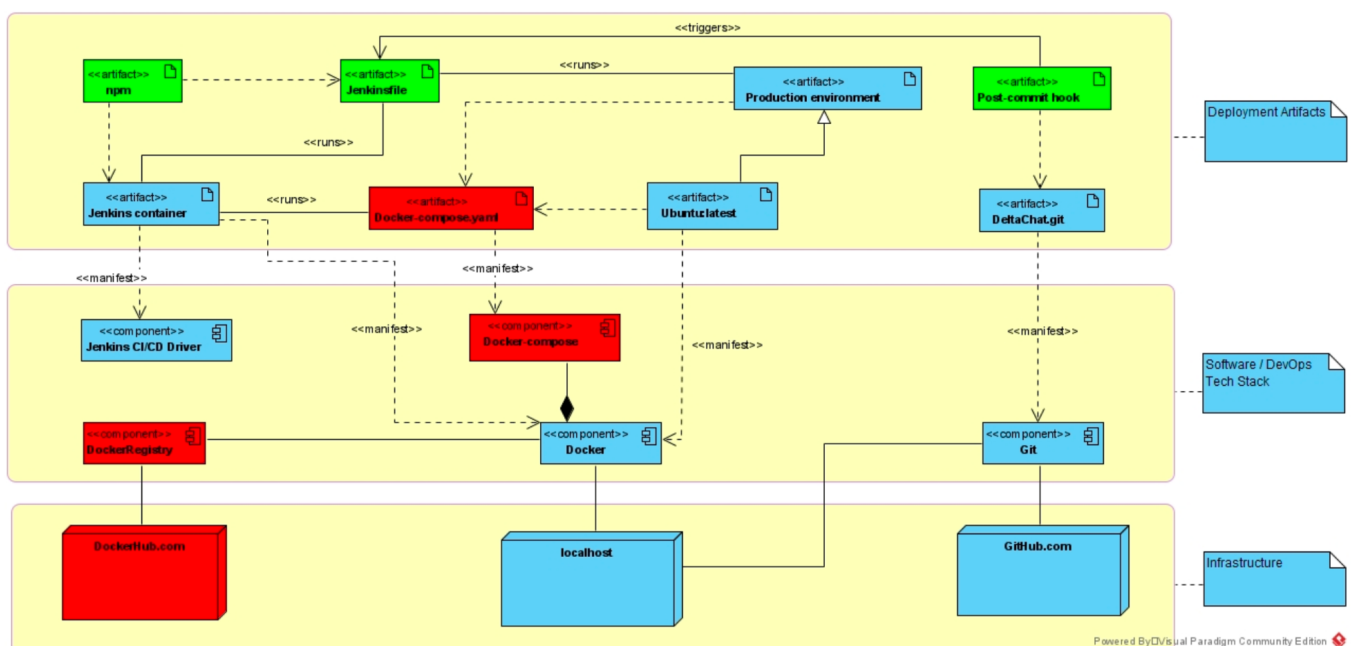
Zastosowane technologie:

- git
- docker
- bash
- jenkins
- NPM



Nazwa Kroku	Technologia	Link do pliku	Numer linii	Zgodność z planem, komentarz
Zmiana na branchu master	Git	-	-	Zgodnie z planem - dodanie zmiany poprzez commit na repo Deltachatu
Uruchomienie Jenkinsa za pomocą post-commit	Git	Post-commit	4	Zgodnie z planem - post-commit uruchamia build na Jenkinsie za pomocą curl
Instalowanie zależności	Npm	Jenkinsfile	7	Dodano do planu – Należało zainstalować zależności potrzebne do zbudowania aplikacji
Budowanie aplikacji	Npm	Jenkinsfile	8	Zgodnie z planem – Zbudowanie aplikacji poleceniem npm run build
Powiadomienie na email o niepowodzeniu budowania	Jenkins	Jenkinsfile	11-16	Zgodnie z planem - W razie niepoprawnego zbudowania aplikacji pipeline jest przerywany i wysyłane jest powiadomienie na email
Powiadomienie na email o sukcesie budowania	Jenkins	Jenkinsfile	17-21	Dodano do planu - Nie przewidywano w planie powiadomień na email
Rozpoczęcie testowania	Npm	Jenkinsfile	27	Zgodne z planem - Po poprawnym uruchomieniu aplikacji w stage test uruchomiano testowanie jednostkowe, integracyjne i e2e za pomocą npm run test
Powiadomienie na email o niepowodzeniu testowania	Jenkins	Jenkinsfile	30-35	Zgodnie z planem - W razie, gdy testy zakończą się niepowodzeniem pipeline jest przerywany i wysyłane jest powiadomienie na email

Powiadomienie na email o sukcesie testowania	Jenkins	Jenkinsfile	36-40	Dodano do planu - Nie przewidywano w planie powiadomień na email
Przeniesienie na środowisko produkcyjne	Docker	Jenkinsfile Dockerfile.deploy	46	Zgodnie z planem - przeniesienie aplikacji na obraz dockerowy
Powiadomienie na email o niepowodzeniu wdrażania	Jenkins	Jenkinsfile	49-54	Zgodnie z planem - W razie gdy wdrożenie zakończy się niepowodzeniem, pipeline jest przerywany i wysyłane jest powiadomienie na email
Powiadomienie na email o sukcesie wdrażania	Jenkins	Jenkinsfile	55-59	Dodano do planu - Nie przewidywano w planie powiadomień na email
Uruchomienie nowej wersji aplikacji	Docker	-	-	Nie zaimplementowano ponieważ usunięto uruchamianie aplikacji z planu
Powiadomienie o sukcesie uruchamiania	Jenkins	-	-	Nie zaimplementowano ponieważ nie uruchamiano aplikacji
Powiadomienie o niepowodzeniu uruchamiania	Jenkins	Jenkinsfile	63-76	Dodano do planu - Nie przewidywano w planie powiadomień na email



Etap	Nazwa kroku	Technologia	Link do pliku	Zgodność z planem, komentarz
Infrastructure	Github	Git	-	Zgodnie z planem - wykorzystano go, aby przechowywać repozytorium
	Localhost	-	-	Zgodnie z planem
	DockerHub.com	DockerHub	-	Usunięto z planu - nie zaimplementowano publikowania obrazów DockerHub
Software/ DevOps Techstack	Git	Git	-	Zgodnie z planem
	Docker	Docker	-	Zgodnie z planem
	Docker-compose	Docker-compose	-	Usunięto z planu - zamiast implementacji docker-compose wykorzystano npm install
	Jenkins CI/CD driver	Jenkins	-	Zgodnie z planem - wykorzystano go jako narzędzie CI/CD
	DockerRegistry	Docker	-	Usunięto z planu - nie publikowano obrazów
Deployment artifacts	Deltachat.git	Git	Repo	Zgodnie z planem - wykorzystano sforkowane repozytorium z kodem źródłowym aplikacji
	Post-commit hook	Git	Post-commit	Dodano do planu - Post-commit hook za pomocą curl uruchamiał pipeline po commicie
	Production enviroment	Docker	Dockerfile.deploy	Zgodnie z planem - przeniesiono aplikację na środowisko produkcyjne, którym był obraz

				dockerowy zawierający kod źródłowy aplikacji
	Ubuntu:latest	Docker	-	Zgodnie z planem - wykorzystano obraz ubuntu w środowisku produkcyjnym
	Docker- compose.yaml	Docker	-	Usunięto z planu - nie był potrzebny z powodu posiadania tylko jednego obrazu
	Jenkinsfile	Jenkins	Jenkinsfile	Dodano do planu - aby uruchamiać Jenkins pipeline
	Npm	Npm	-	Dodano do planu - aby testować i budować aplikację
	Jenkins container	Jenkins, Docker	-	Zgodnie z planem - wykorzystano go jako kontener dockerowy, w którym uruchamiany był jenkins