

Imię i nazwisko: Kinga Baryczka

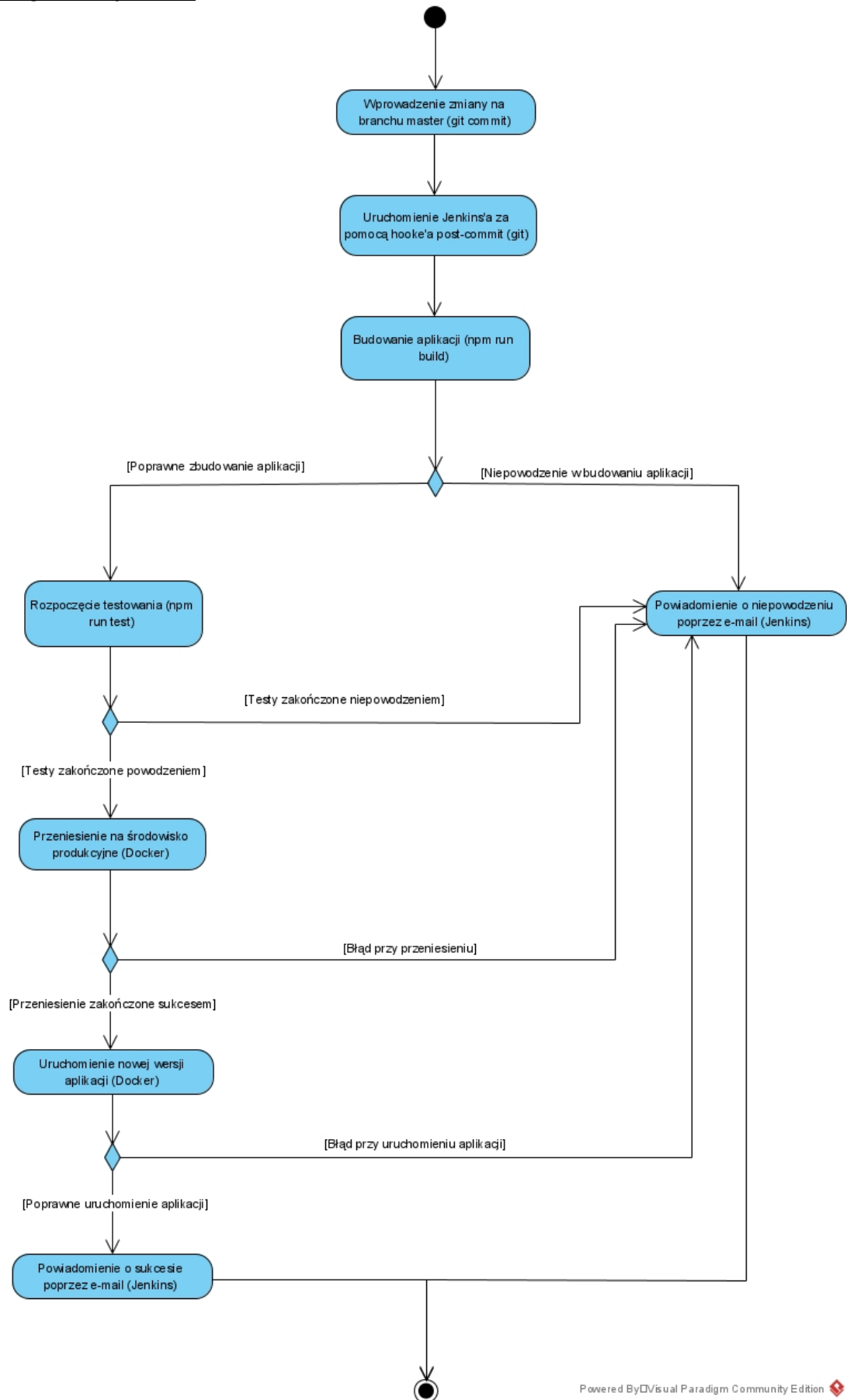
Data wykonania: 23.04.2021 r.

PLAN WDROŻENIOWY APLIKACJI „DELTACHAT”

ZASTOSOWANE TECHNOLOGIE:

- GIT (W TYM GIT HOOKS),
- DOCKER (W TYM DOCKER-COMPOSE),
- DOCKERREGISTRY (DOCKERHUB),
- BASH,
- JENKINS

1. Diagram aktywności:

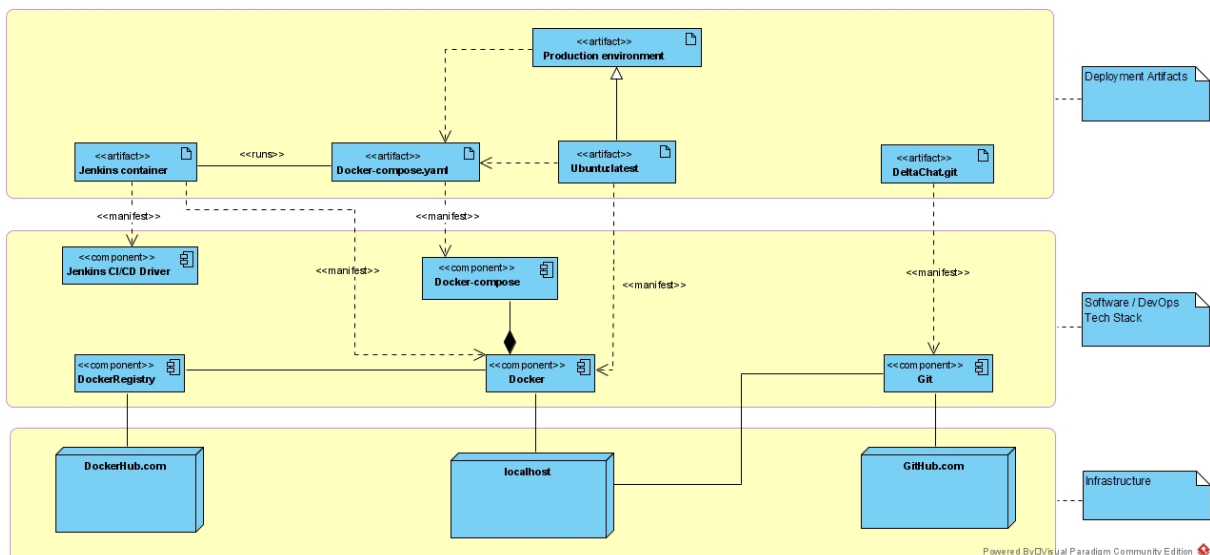


- 1) Po wprowadzeniu zmian na repozytorium na GitHubie uruchamiany jest Jenkins za pomocą hooke'a post-commit.
- 2) Następnie Jenkins uruchamia budowanie aplikacji za pomocą komendy *npm run build*.
W przypadku niepoprawnego zbudowania, Jenkins wysyła powiadomienie na e-mail o niepowodzeniu w budowaniu aplikacji.
- 3) W przypadku gdy aplikacja zbuduje się poprawnie, Jenkins uruchamia testowanie za pomocą polecenia *npm run test && npm run test-e2e*. Testy dzielą się na:
 - jednostkowe,
 - integracyjne,
 - E2E.

W przypadku, gdy testy zakończą się niepowodzeniem Jenkins wysyła o tym powiadomienie na e-mail.

- 4) Przeniesienie na środowisko produkcyjne odbywa się za pomocą Dockera. Dockerfile utworzy nowe środowisko produkcyjne na podstawie obrazu ubuntu:latest.
W przypadku niepowodzenia w przeniesieniu na środowisko produkcyjne Jenkins wyśle powiadomienie na e-mail.
- 5) Uruchomienie dockerowego kontenera zawierającego najnowszą wersję aplikacji.
W przypadku błędu podczas uruchomienia aplikacji zostanie wysłane powiadomienie na e-mail.
- 6) Powiadomienie o sukcesie poprzez e-mail.

2. Diagram wdrożeniowy



Użyta infrastruktura:

- DockerHub.com,
- localhost,
- GitHub.com.

Użyty tech-stack:

- DockerRegistry,
- Docker,
- Docker-compose,
- Git,
- Jenkins CI/CD Driver.

Użyte artefakty:

- Kontener dockerowy zawierający Jenkinsa,
- Docker-compose.yaml,
- Obraz ubuntu:latest,
- Repozytorium z aplikacją DeltaChat,
- Kontener dockerowy zawierający środowisko produkcyjne.