

**MINISTERUL EDUCAȚIEI ȘI CERCETĂRII AL REPUBLICII MOLDOVA**  
**Universitatea Tehnică a Moldovei**  
**Facultatea Calculatoare Informatică și Microelectronică**  
**Departamentul Ingineria Software și Automatică**

**Admis la susținere**  
**Șef de departament:**  
**Fiodorov I. dr., conf. univ.**

-----  
„\_\_\_” \_\_\_\_\_ 2022

# **Analiza și compararea instrumentelor pentru integrarea și livrarea continuă a aplicațiilor**

**Teza de master**

**Student:** \_\_\_\_\_ **Chifa Eugen, TI-201M**

**Conducător:** \_\_\_\_\_ **Chirev Pavel, lect. univ.**

**Consultant:** \_\_\_\_\_ **Cojocaru Svetlana, lect. univ.**

**Chișinău, 2022**

## АННОТАЦИЯ

DevOps является перспективным и быстрорастущим направлением в сфере разработки программного обеспечения и администрирования информационных систем. DevOps-инженеры стараются автоматизировать процессы интеграции и поставки программного кода или различных конфигурационных файлов в инфраструктуру, поэтому появились такие понятия как Continuous Integration (непрерывная интеграция) и Continuous Delivery (непрерывная поставка).

**Continuous Integration & Continuous Delivery** – это концепция, которая позволяет облегчить внедрение изменений программного кода в существующий репозиторий. DevOps-инженеры настраивают свои репозитории так, чтоб при внедрении изменений срабатывал pipeline, который выполняет проверку кода на ошибки, то есть выполняет процедуру интеграции кода и завершается это развертыванием измененного кода в новый или существующий продукт, тем самым происходит процедура поставки кода в конечный продукт, который будет доступен пользователям.

Цель данной работы – анализ рынка на наличие инструментов для непрерывных процессов интеграции и поставки приложений и практическое сравнение их между собой для выявления лучшего решения для DevOps-инженера.

## **REZUMAT**

DevOps este un domeniu promițător și în creștere rapidă al dezvoltării de software și al administrării sistemelor informatice. Inginerii DevOps încearcă să automatizeze procesele de integrare și livrare a codului software sau a diverselor fișiere de configurare în infrastructură, astfel că au apărut concepte precum Continuous Integration și Continuous Delivery.

Integrarea continuă și livrarea continuă este un concept care facilitează implementarea modificărilor de cod software într-un depozit existent. Inginerii DevOps își configurează depozitele astfel încât, atunci când modificările sunt implementate, să fie declanșată o conductă care efectuează verificarea erorilor de cod, adică execută o procedură de integrare a codului și o finalizează prin implementarea codului modificat într-un produs nou sau existent, livrând astfel codul în produsul final care va fi disponibil utilizatorilor.

Scopul acestei lucrări este de a analiza piața în ceea ce privește disponibilitatea instrumentelor pentru procesele de integrare și livrare continuă a aplicațiilor și de a le compara între ele în practică pentru a identifica cea mai bună soluție pentru inginerul DevOps.

## **ABSTRACT**

DevOps is a promising and fast-growing area of software development and information systems administration. DevOps engineers are trying to automate the processes of integration and delivery of software code or various configuration files into the infrastructure, so such concepts as Continuous Integration and Continuous Delivery appeared.

Continuous Integration & Continuous Delivery is a concept that makes it easier to implement software code changes into an existing repository. DevOps engineers set up their repositories so that when changes are implemented, a pipeline is triggered that checks the code for errors, that is, it performs a code integration procedure and ends up deploying the modified code into a new or existing product, thus the procedure of code delivery into the final product that will be available to users.

The purpose of this paper is to analyze the market for the availability of tools for continuous application integration and delivery processes and practical comparison between them to identify the best solution for the DevOps engineer.

# ОГЛАВЛЕНИЕ

<i>АННОТАЦИЯ</i> .....	5
<i>REZUMAT</i> .....	6
<i>ABSTRACT</i> .....	7
<i>ВВЕДЕНИЕ</i> .....	10
<b>1 ОБЛАСТЬ ИССЛЕДОВАНИЯ – КУЛЬТУРА НЕПРЕРЫВНОЙ ИНТЕГРАЦИИ И ПОСТАВКИ КОДА</b> .....	11
1.1 Описание общих понятий .....	11
1.2 Концепция CI/CD .....	12
1.3 Основные принципы CI/CD.....	12
1.4 Этапы CI/CD .....	13
1.5 Преимущества и недостатки CI/CD .....	15
<b>2 ОБЛАСТЬ ИССЛЕДОВАНИЯ – АНАЛИЗ РЫНКА НА НАЛИЧИЕ CI/CD-ИНСТРУМЕНТОВ</b> .....	16
2.1 GitLab.....	16
2.2 Jenkins .....	17
2.3 Travis CI.....	17
2.4 CircleCI.....	18
2.5 GoCD .....	18
2.6 Concourse.....	19
<b>3 АНАЛИЗ И СРАВНЕНИЕ ИССЛЕДУЕМЫХ CI/CD ИНСТРУМЕНТОВ</b> .....	20
3.1 GitLab и Jenkins .....	20
3.1.1 Особенности GitLab и GitLab CI/CD .....	20
3.1.2 Преимущества и недостатки GitLab.....	22
3.1.3 Особенности Jenkins .....	22
3.1.4 Преимущества и недостатки Jenkins .....	24
3.1.5 Сравнение Jenkins и GitLab CI/CD .....	25
3.1.6 Различия между Jenkins и GitLab CI/CD .....	26
3.1.7 Пример настройки конвейера CI/CD в GitLab .....	26
3.1.8 Пример настройки конвейера CI/CD в Jenkins.....	32
3.2 TravisCI и CircleCI .....	37
3.2.1 Особенности Travis CI.....	37
3.2.2 Преимущества и недостатки Travis CI.....	38
3.2.3 Особенности CircleCI.....	38

3.2.4	Преимущества и недостатки CircleCI .....	39
3.2.5	Что есть в TravisCI и нет в CircleCI? .....	39
3.2.6	Сравнение TravisCI и CircleCI.....	40
3.2.7	Пример настройки конвейера CI/CD в TravisCI .....	41
3.2.8	Пример настройки конвейера CI/CD в CircleCI .....	45
3.3	GoCD и ConcourseCI .....	46
3.3.1	Особенности GoCD .....	46
3.3.2	Преимущества и недостатки GoCD .....	48
3.3.3	Особенности ConcourseCI .....	48
3.3.4	Преимущества и недостатки ConcourseCI .....	49
3.3.5	Сравнение GoCD и ConcourseCI.....	50
3.3.6	Пример настройки конвейера CI/CD в ConcourseCI.....	50
<b>ВЫВОД</b>	.....	<b>54</b>
<b>СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ</b>	.....	<b>55</b>

## ВВЕДЕНИЕ

DevOps является перспективным и быстрорастущим направлением в сфере разработки программного обеспечения и администрирования информационных систем. DevOps-инженеры стараются автоматизировать процессы интеграции и поставки программного кода или различных конфигурационных файлов в инфраструктуру, поэтому появились такие понятия как Continuous Integration (непрерывная интеграция) и Continuous Delivery (непрерывная поставка).

**Continuous Integration & Continuous Delivery** – это концепция, которая позволяет облегчить внедрение изменений программного кода в существующий репозиторий. DevOps-инженеры настраивают свои репозитории так, чтоб при внедрении изменений срабатывал pipeline, который выполняет проверку кода на ошибки, то есть выполняет процедуру интеграции кода и завершается это развертыванием измененного кода в новый или существующий продукт, тем самым происходит процедура поставки кода в конечный продукт, который будет доступен пользователям.

Цель данной работы – анализ рынка на наличие инструментов для непрерывных процессов интеграции и поставки приложений и практическое сравнение их между собой для выявления лучшего решения для DevOps-инженера.

## **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

[1] – Habr: Знакомство с Concourse CI [citat 11.11.21]. Disponibil: <https://habr.com/ru/post/347186/>

[2] – GoCD: GoCD Features [citat 11.11.21]. Disponibil: <https://www.gocd.org/why-gocd/>