



UNIVERZITET U NOVOM SADU  
FAKULTET TEHNIČKIH NAUKA  
KATEDRA ZA PRIMENJENE RAČUNARSKE NAUKE

# Računarstvo u oblaku

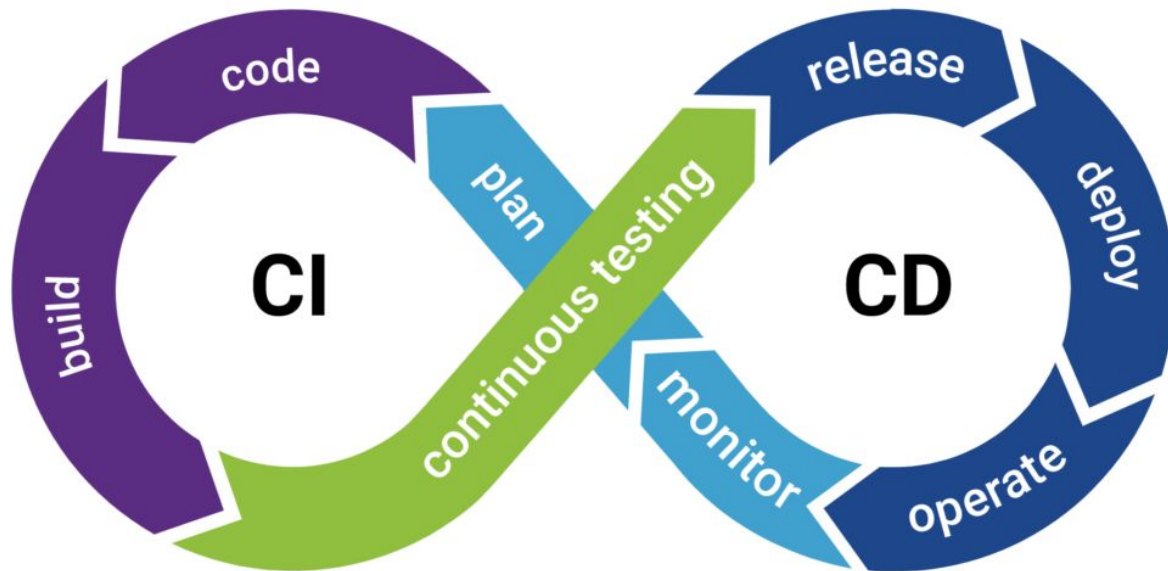
ms Helena Anišić

Zimski semester 2022/2023.

Studijski program: Računarstvo i automatika

Modul: Računarstvo visokih performansi

## CI/CD



# CI/CD

- CI/CD pripada oblasti DevOps-a
- Kombinuje praksu kontinualne integracije i kontinualne isporuke
- CI/CD automatizuje većinu ili čak sve postupke koje su ljudi radili ručno kako bi novi kod prebacili u produkciju
  - build
  - test
  - deploy
  - ..
- Kada programer komituje neku izmenu u kodu CI/CD mu omogućava da tu izmenu automatski testira (sa ostatkom koda) te da isporuči novi kod u produkciju

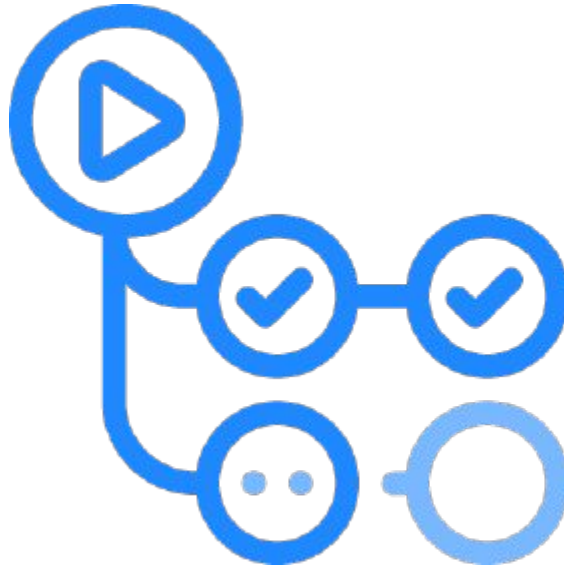
## Prednosti CI/CD

- Brža i češća isporuka softvera korisnicima
  - Popravka bug-a
  - Novi feature
  - itd
- Bolji kvalitet koda
  - Automatsko testiranje
- Lakši rollback
  - Vraćanje na prethodnu verziju softvera koja radi

## CI/CD alati

- Jenkins
- CircleCI
- TeamCity
- Bamboo
- GitLab
- Buddy
- Travis CI
- Codeship
- GoCD
- ...

## Github actions



## Šta su github akcije?

- Github akcije su vrsta CI/CD platforme koji omogućava kreiranje automatskog pipeline-a za build, testiranje i deployment.
  - primer: workflow koji izvršava testove nad kodom koji je pull request-ovan
  - Primer: workflow koji izvršava deploy koda u produkciju
- Github akcije prevazilaze opseg običnog DevOps-a i omogućavaju reagovanje i na neke druge događaje.
  - primer: kreiranje workflow-a koji automatski dodaje odgovarajuću labelu kad god neko kreira novi *issue* u repozitorijumu.

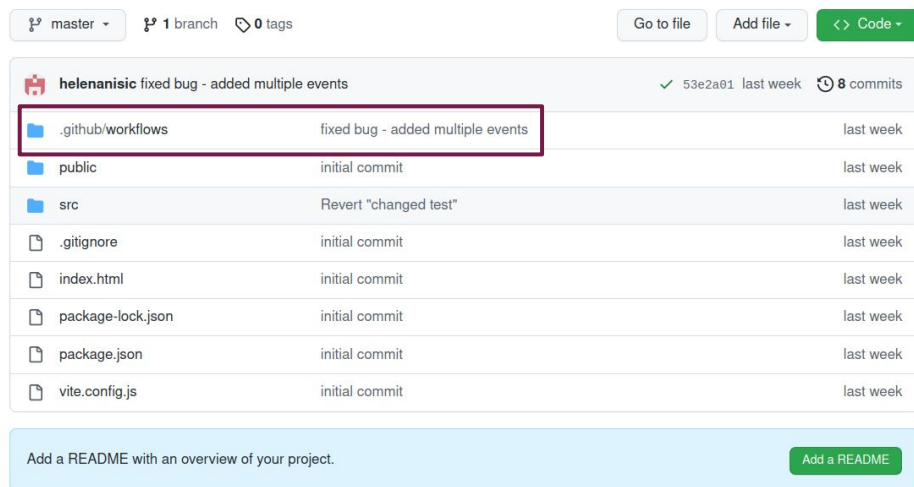
# Komponente github akcija

- Workflows
- Events
- Jobs
- Actions
- Runners



## Github akcija [workflow]

- Github akcija koristi YAML sintaksu za definisanje workflow-a
- Svaki workflow predstavlja pojedinačni YAML fajl u okviru repozitorijuma
  - `./github/workflows`



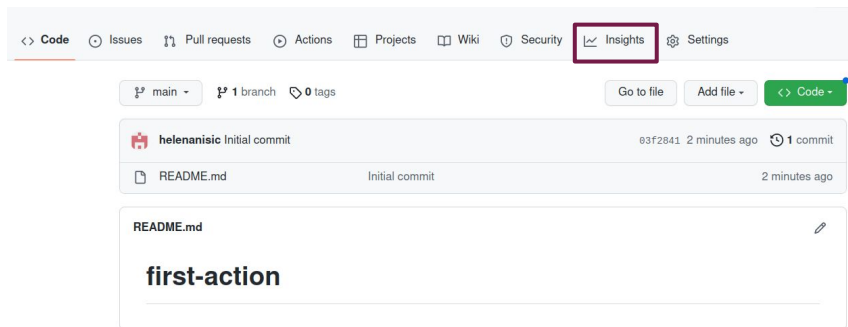
The screenshot shows the GitHub web interface for a repository. At the top, there are navigation elements: 'master' branch selected, '1 branch', '0 tags', and buttons for 'Go to file', 'Add file', and 'Code'. Below this is a table of files and folders. The first row, representing the '.github/workflows' directory, is highlighted with a red box. The table lists the following items:

File/Folder	Commit Message	Last Commit
<b>.github/workflows</b>	fixed bug - added multiple events	last week
public	initial commit	last week
src	Revert "changed test"	last week
.gitignore	initial commit	last week
index.html	initial commit	last week
package-lock.json	initial commit	last week
package.json	initial commit	last week
vite.config.js	initial commit	last week

At the bottom of the repository view, there is a light blue banner with the text 'Add a README with an overview of your project.' and a green button labeled 'Add a README'.

## Primer 1 [workflow]

1. Kreiranje novog javnog repozitorijuma
2. U okviru **actions tab**-a kreiranje novog **workflow**-a



actions tab u okviru repozitorijuma

# Github akcija [workflow]

- Postoje dve vrsta workflow-a
  - Unapred kreirani
    - Od strane drugih korisnika Github-a
      - besplatni
      - za određenu cenu
  - Prazni - koje vi kreirate

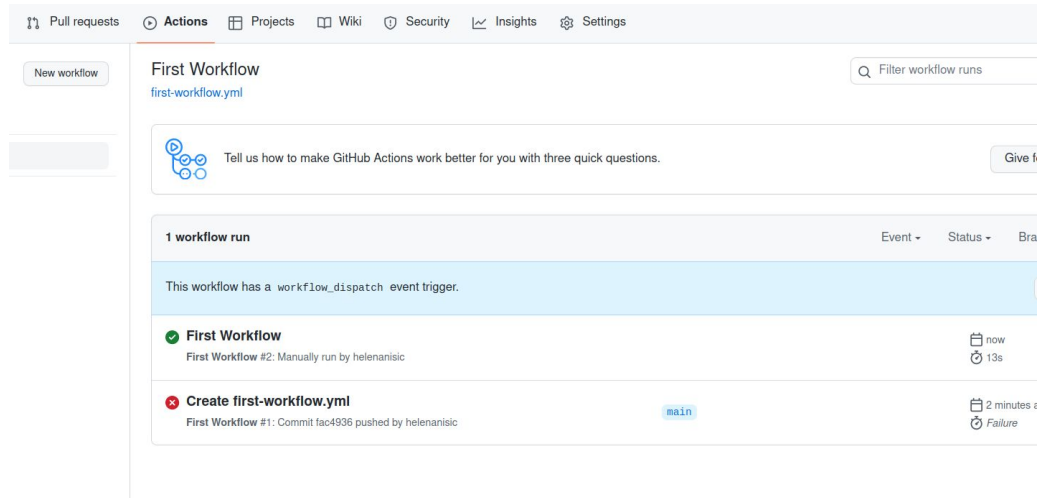
## Primer 1 [workflow]

```
name: First Workflow
on: workflow_dispatch
jobs:
  first-job:
    runs-on: ubuntu-22.04
    steps:
      - name: Print greeting
        run: echo "Hello World"
      - name: Print goodbye
        run: echo "Done - bye!"
```

Workflow first-action.yml

# Primer 1 [workflow]

- Workflow
  - Uspešno izvršavanje
  - Neuspešno izvršavanje



The screenshot displays the GitHub Actions interface for a repository. The top navigation bar includes links for Pull requests, Actions (selected), Projects, Wiki, Security, Insights, and Settings. On the left sidebar, there is a 'New workflow' button. The main content area is titled 'First Workflow' with a link to 'first-workflow.yml'. Below this, there is a section for '1 workflow run' with a search bar 'Filter workflow runs'. The workflow run list shows two runs:

Run Name	Description	Status	Branch
First Workflow	First Workflow #2: Manually run by helenanistic	Success (green checkmark)	now
Create first-workflow.yml	First Workflow #1: Commit fac4936 pushed by helenanistic	Failure (red X)	main

## Primer 1 [workflow] - analiza delova

- `name: First Workflow`
  - opciono - naziv workflow-a pojavljuje se u okviru Actions taba na Github repozitorijumu
- `on: workflow_dispatch`
  - Obavezno - određuje okidač za dati workflow
- `jobs:`
  - obavezno - grupiše sve poslove koji se obavljaju u okviru datog workflowa
    - minimalno jedan posao
    - Svaki posao se deklariše svojim imenom (`first-job:`)
    - Tipom virtuelne mašine na kojoj se izvršava (`runs-on: ubuntu-22.04`)
    - Koracima koji se izvršavaju u okviru datog posla (`steps:`)
      - Svaki korak ima naziv i
      - Naredbu koja se pokreće

## Github akcija [events]

- **on** naredba u workflow-u služi za definisanje okidača za dati workflow
- može biti jedan okidač ili više
  - on: push
  - on: [push, fork]
- Primer okidača:
  - push, pull\_request, push, workflow\_dispatch, create, delete,...
  - dokumentacija postojećih okidača:  
<https://docs.github.com/en/actions/using-workflows/events-that-trigger-workflows#available-events>

# Github akcija [events]

- Događaji (push, pull\_request, ...) mogu da imaju:
  - Tipove aktivnosti (Activity Types)
    - Obezbeđuje veću kontrolu nad time kada će se neki workflow okinuti
      - pull\_request događaj - opened, closed, edited, ...
  - Filtere (Filters)
    - Obezbeđuje veću kontrolu nad time kada će se neki workflow okinuti
      - push događaj - filter na osnovu grane (main, branch\_x, branch\_y, ...)



## Github akcija [jobs]

- Workflow je sačinjen od jednog ili više poslova koji se po default-u izvršavaju paralelno
  - dodatnom opcijom može da se podesi sekvencijalno izvršavanje
- Poslovi se izvršavaju u definisanom okruženje (runner)
  - runs-on:
- Svaka slika virtuelne mašine ima određene instalacije u okviru sebe

# Github akcija [runners]

## Choosing GitHub-hosted runners

If you use a GitHub-hosted runner, each job runs in a fresh instance of a runner image specified by `runs-on`.

Available GitHub-hosted runner types are:

Runner image	YAML workflow label	Notes
Windows Server 2022	<code>windows-latest</code> or <code>windows-2022</code>	The <code>windows-latest</code> label currently uses the Windows Server 2022 runner image.
Windows Server 2019	<code>windows-2019</code>	
Ubuntu 22.04	<code>ubuntu-latest</code> or <code>ubuntu-22.04</code>	The <code>ubuntu-latest</code> label currently uses the Ubuntu 22.04 runner image.
Ubuntu 20.04	<code>ubuntu-20.04</code>	
Ubuntu 18.04 <small>[deprecated]</small>	<code>ubuntu-18.04</code>	Migrate to <code>ubuntu-20.04</code> or <code>ubuntu-22.04</code> . For more information, see <a href="#">this GitHub blog post</a> .
macOS Monterey 12	<code>macos-12</code>	
macOS Big Sur 11	<code>macos-latest</code> or <code>macos-11</code>	The <code>macos-latest</code> label is currently transitioning to the macOS Monterey 12 runner image. During the transition, the label might refer to the runner image for either macOS 11 or 12. For more information, see <a href="#">this GitHub blog post</a> .
macOS Catalina 10.15 <small>[deprecated]</small>	<code>macos-10.15</code>	Migrate to <code>macos-11</code> or <code>macos-12</code> . For more information, see <a href="#">this GitHub blog post</a> .

## Github akcija [jobs] - steps

- Posao sadrži niz zadataka koji se zovu `steps`
- Koraci mogu da:
  - Izvršavaju komande
  - Pokreću zadatke za podešavanje ili
  - Da izvršavaju akcija u privatnom repozitorijumu / javnom repozitorijumu ili Docker registru.
- Svaki korak pokreće sopstveni proces u okviru izvršnog okruženja i ima sopstveni radni prostor i fajl sistem.
  - Zbog toga se izmene u environment varijablama ne čuvaju između koraka
- Nisu svi koraci akcije, ali svaka akcija jeste korak
- Github nudi ugrađene korake za podešavanje i izvršavanje posla

## Github akcija [actions]

- Akcija je aplikacija Github Actions platforme koja izvršava kompleksne, ali često ponavljane zadatke.
- Koristiti akciju kada treba da se umanju količina repetitivnog koda koji se piše u workflow fajlu.
- Akcije mogu da:
  - Povuku (pull) kod sa github repozitorijuma,
  - Podese odgovarajuće alate za izvršno okruženje ili
  - Podese autentifikaciju za cloud provajder.
- Akcije mogu da se:
  - Pišu od strane programera ili
  - Da se koriste gotove akcije sa Github Marketplace-a (<https://github.com/marketplace?type=> )

## Github akcija [actions]

```
steps:
  - name: Get code
    uses: actions/checkout@v3
  - name: Install NodeJS
    uses: actions/setup-node@v3
    with:
      node-version: 18
  - name: Install dependencies
    run: npm ci      #komanda slična kao npm install, ali se koristi u CI/CD okruženjima
  - name: Run tests
    run: npm test
```

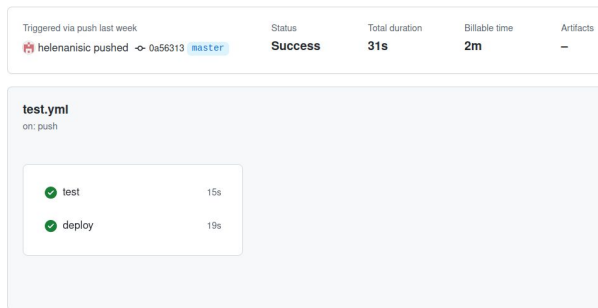
## Github akcija [jobs]

- Predefinisano je da se poslovi izvršavaju paralelno, međutim po potrebi se može definisati da se izvršavaju sekvencijalno - `needs` naredba

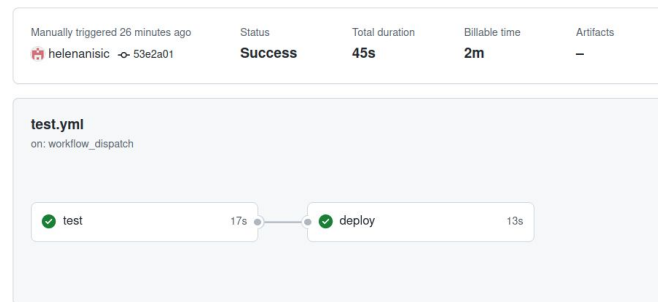
```
jobs:
  test:
    runs-on: ubuntu-22.04
    steps:
      - name: Get code
        uses: actions/checkout@v3
  deploy:
    needs: test
    runs-on: ubuntu-22.04
    steps:
      - name: Get code
        uses: actions/checkout@v3
```

# Github akcija [jobs]

- Vizuelni prikaz poslova koji se izvršavaju sekvencijalno ili paralelno



Paralelno izvršavanje poslova



Sekvencijalno izvršavanje poslova

# Github akcije sa Docker kontejnerom

- Zašto bismo koristili Docker kontejnere u workflow-u?
  - Skraćivanje vremena izvršavanja, broj koraka i potencijalno cene izvršavanja
  - Pokretanje servisnih kontejnera za dodatne poslove
    - Primer: testiranje rada sa bazom podataka



# Github akcije sa Docker kontejnerom

- Posao zajedno sa pojedinačnim koracima može da se izvršava:
  - Samo u okviru izvršnog okruženja (runner) ili
    - Sadrži određene instalacije alata
      - Ukoliko su potrebni dodatni alati, mogu se posebno instalirati u okviru koraka u datom poslu
        - svaki korak je poseban zahtev -> podiže cenu usluge
  - U kontejneru koji se nalazi u datom izvršnom okruženju (runner).
    - Korisnik pravi sliku kontejnera po svojoj potrebi
      - pokretanje kontejnera je samo jedan zahtev

# Github akcije sa Docker kontejnerom

- Servisni kontejneri
  - Primer upotrebe:
    - Kontejner u kome će biti pokrenuta baza podataka koja služi samo za izvršavanje testova
- Servisni kontejneri se nalaze unutar kontejnera koji je pokrenut u odabranom izvršnom okruženju datog posla
- Koraci posla (steps) mogu da komuniciraju sa servisnim kontejnerom

# Github akcije sa Docker kontejnerom

```
jobs:
  test:
    environment: testing
    runs-on: ubuntu-latest
    container:
      image: node:16
    env:
      MONGODB_CONNECTION_PROTOCOL: mongodb
      MONGODB_CLUSTER_ADDRESS: mongodb
      MONGODB_USERNAME: root
      MONGODB_PASSWORD: example
      PORT: 8080
  services:
    mongodb:
      image: mongo
      env:
        MONGO_INITDB_ROOT_USERNAME: root
        MONGO_INITDB_ROOT_PASSWORD: example
```

## Primer 3

- Kreiranje workflow-a koji svakim push-om build-uje novu sliku i postavlja je na DockerHub

```
name: Docker Image CI
on:
  push:
    branches: [ "master" ]
jobs:
  build:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v3
      - name: docker login
        env:
          DOCKER_USER: ${secrets.DOCKER_USER}
          DOCKER_PASSWORD: ${secrets.DOCKER_PASSWORD}
        run: |
          docker login -u $DOCKER_USER -p $DOCKER_PASSWORD
      - name: Build the Docker image
        run: docker build . --file Dockerfile --tag hanisic/mysite:django_image
      - name: Docker Push
        run: docker push ${secrets.DOCKER_USER}/mysite:django_image
```

## Github akcije - secrets

- Neke environment varijable ne smeju da se nalaze u plain text-u na github repozitorijumu
  - Zbog potencijalnih malicioznih korisnika koji mogu da se iskoriste naše podatke
    - primer: username i password za neki servis
- Iz ovog razloga se koriste takozvane tajne (secrets)
  - U okviru podešavanja (settings) datog repozitorijuma moguće je postaviti tajne koje će se koristiti u izvršavanju akciju

# Github akcije - secrets

helenanasic / mysite Public

Pin Unwatch 1 Fork 0

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Actions

Webhooks

Environments

Codespaces

Pages

Security

Code security and analysis

Deploy keys

Secrets

Actions

Actions secrets

New repository secret

Secrets are environment variables that are **encrypted**. Anyone with **collaborator** access to this repository can use these secrets for Actions.





Secrets are not passed to workflows that are triggered by a pull request from a fork. [Learn more about encrypted secrets.](#)

Environment secrets

Manage environments

There are no secrets for this repository's environments.

Repository secrets

DOCKER_PASSWORD	Updated last week		
DOCKER_USER	Updated last week		

## Github akcije - secrets

```
name: Docker Image CI
on:
  push:
    branches: [ "master" ]
jobs:
  build:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v3
      - name: docker login
        env:
          DOCKER_USER: ${secrets.DOCKER_USER}
          DOCKER_PASSWORD: ${secrets.DOCKER_PASSWORD}
        run: |
          docker login -u $DOCKER_USER -p $DOCKER_PASSWORD
      - name: Build the Docker image
        run: docker build . --file Dockerfile --tag hanisic/mysite:django_image
      - name: Docker Push
        run: docker push ${secrets.DOCKER_USER}/mysite:django_image
```