



Univerza v Mariboru

---

Fakulteta za elektrotehniko,  
računalništvo in informatiko

# Pregled Orodij in Tehnologij pri Razvoju Programskih Sistemov

Avtor: Tilen Gašparič

Univerza v Mariboru

13. april 2025

# Kazalo

<b>1</b>	<b>Uvod</b>	<b>2</b>
<b>2</b>	<b>Koda</b>	<b>2</b>
2.1	Python . . . . .	2
2.1.1	openCV.py . . . . .	2
2.1.2	openCV_test.py . . . . .	2
2.2	Docker . . . . .	3
2.2.1	Dockerfile . . . . .	3
2.2.2	requirements.txt . . . . .	3
2.3	YAML . . . . .	3
2.3.1	28544_test.yml . . . . .	3
2.3.2	28544_deploy.yml . . . . .	5
<b>3</b>	<b>Posnetki zaslona</b>	<b>6</b>
3.1	Github . . . . .	6
3.2	Linux runner strežnik . . . . .	7
3.3	Docker hub . . . . .	8

# Slike

1	Github Actions . . . . .	6
2	Github Secrets . . . . .	6
3	Vzpostavljanje Runner Strežnika 1 . . . . .	7
4	Vzpostavljanje Runner Strežnika 2 . . . . .	7
5	sa3-cicd projekt na DockerHub . . . . .	8

# 1 Uvod

Za osnovo sem se odločil uporabiti 1. nalogo pri predmetu Osnove Računalniškega Vida, v kateri smo na sliki iskali piksele v barvi kože. Po končani nalogi sem ugotovil, da to ni najboljša izbira, saj docker ne podpira grafičnih vmesnikov ampak zdaj nimam časa za ponovitev naloge.

- GitHub repozitorij: [github.com/tilcica/SA3-CICD](https://github.com/tilcica/SA3-CICD)
- DockerHub repozitorij: [hub.docker.com/repository/docker/tilcica/sa3-cicd](https://hub.docker.com/repository/docker/tilcica/sa3-cicd)

## 2 Koda

Koda je v posameznih datotekah komentirana, tukaj pa bi za to bilo preveč besedila v posamezni vrstici.

### 2.1 Python

#### 2.1.1 openCV.py

```
1 def doloci_barvo_koze(slika, spodnjaMeja, zgornjaMeja):
2     pass
3
4 def zmanjsaj_sliko(slika, sirina, visina):
5     pass
6
7 def obdelaj_sliko_s_skatlami(slika, sirina_skatle, visina_skatle, barva_koze):
8     pass
9
10 def prestej_piksle_z_barvo_koze(slika, barva_koze):
11     pass
```

#### 2.1.2 openCV\_test.py

```
1 class TestOpenCVFunctions(unittest.TestCase):
2     def test_prestej_piksle_z_barvo_koze(self):
3         img = cv2.imread("test.png")
4         if img is None:
5             self.fail("test.png not found or could not be loaded.")
6
7         spodnjaMeja = np.array([0, 48, 80], dtype=np.uint8)
8         zgornjaMeja = np.array([20, 255, 255], dtype=np.uint8)
9
10        num_skin_pixels = prestej_piksle_z_barvo_koze(
11            img,
12            (spodnjaMeja, zgornjaMeja))
13
14        mask = cv2.inRange(img, spodnjaMeja, zgornjaMeja)
15        expected_num_skin_pixels = cv2.countNonZero(mask)
16
17        self.assertEqual(num_skin_pixels, expected_num_skin_pixels)
```

```

18
19     def test_zmanjsaj_sliko(self):
20         img = cv2.imread("test.png")
21         if img is None:
22             self.fail("test.png not found or could not be loaded.")
23
24         resized_img = cv2.resize(img, (100, 100))
25
26         self.assertEqual(resized_img.shape[0], 100)
27         self.assertEqual(resized_img.shape[1], 100)

```

## 2.2 Docker

### 2.2.1 Dockerfile

```

1 FROM python:3.10-slim
2
3 WORKDIR /app
4
5 RUN apt-get update && apt-get install -y \
6     libgl1 \
7     libglu1-mesa \
8     libx11-6 \
9     libxext6 \
10    libxrender1 \
11    libsm6 \
12    libxcb1 \
13    && rm -rf /var/lib/apt/lists/*
14
15 COPY requirements.txt .
16 RUN pip install --no-cache-dir -r requirements.txt
17
18 COPY openCV.py .
19 COPY smiley.png .
20 COPY test.png .
21
22 ENV PYTHONUNBUFFERED=1
23
24 CMD ["python", "openCV.py"]

```

### 2.2.2 requirements.txt

```

1 numpy
2 opencv-python

```

## 2.3 YAML

### 2.3.1 28544\_test.yml

```

1 name: 28544_test
2
3 on:

```

```

4   push:
5     branches: [master]
6
7   jobs:
8     checkTests:
9       name: check if test file exists
10      runs-on: ubuntu-latest
11
12      outputs:
13        error_exists: ${ steps.check.outputs.error_exists }
14
15      steps:
16        - name: checkout
17          uses: actions/checkout@v3
18
19        - name: check for openCV_test.py
20          id: check
21          run: |
22            if [ -f openCV_test.py ]; then
23              echo "Test file found."
24              echo "error_exists=false" >> $GITHUB_OUTPUT
25            else
26              echo "Test file 'openCV_test.py' not found!" 1>&2
27              echo "Test file not found!" > napaka.txt
28              echo "error_exists=true" >> $GITHUB_OUTPUT
29            fi
30
31        - name: upload napaka.txt
32          if: always()
33          uses: actions/upload-artifact@v4
34          with:
35            name: napaka
36            path: napaka.txt
37
38      runTests:
39        name: run tests
40        needs: checkTests
41        if: needs.checkTests.outputs.error_exists == 'false'
42        runs-on: ubuntu-latest
43
44        strategy:
45          matrix:
46            pythonVersion: [3.10, 3.13]
47
48        steps:
49          - name: checkout
50            uses: actions/checkout@v3
51
52          - name: download napaka.txt
53            uses: actions/download-artifact@v4
54            with:
55              name: napaka
56              continue-on-error: true
57
58          - name: set up python
59            uses: actions/setup-python@v4

```

```

60         with:
61             pythonVersion: ${ matrix.python-version }
62
63     - name: dependencies
64       run: |
65         if [ -f requirements.txt ]; then pip install -r requirements.txt; fi
66
67     - name: run tests
68       run: |
69         if grep -q "not found" napaka.txt; then
70             echo "Error found in napaka.txt. Exiting."
71             exit 1 # Fail the job if there's an error
72         else
73             echo "Running unit tests..."
74             if ! python -m unittest openCV_test.py; then
75                 echo "Unit tests failed. Exiting."
76                 exit 1 # Fail the job if tests fail
77             fi
78         fi

```

### 2.3.2 28544\_deploy.yml

V primeru, da se 28544\_test ne izvede uspešno, se izvajanje 28544\_deploy ne začne.

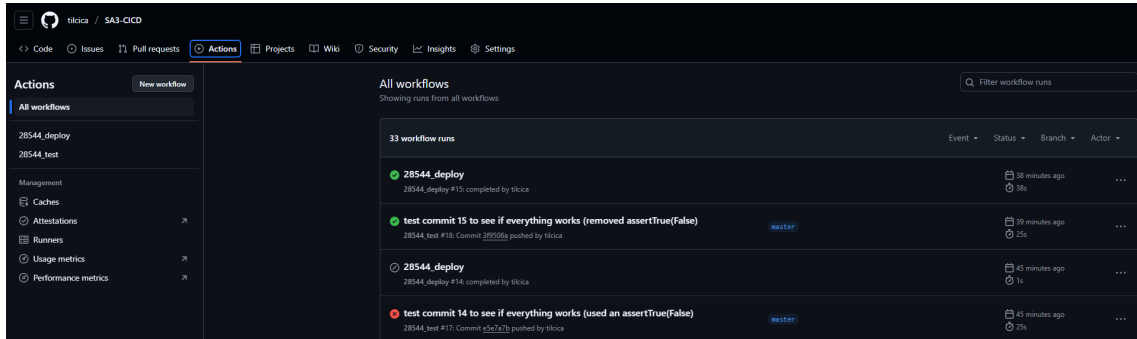
```

1 name: 28544_deploy
2
3 on:
4   workflow_run:
5     workflows: ["28544_test"]
6     types: ["completed"]
7
8 jobs:
9   buildAndPush:
10    if: ${ github.event.workflow_run.conclusion == 'success' }
11    runs-on: ubuntu-latest
12
13    steps:
14      - name: checkout
15        uses: actions/checkout@v3
16
17      - name: docker login
18        uses: docker/login-action@v3
19        with:
20          username: ${ secrets.DOCKER_USERNAME }
21          password: ${ secrets.DOCKER_PASSWORD }
22
23      - name: docker build
24        run: |
25          docker build -t tilcica/sa3-cicd:latest .
26
27      - name: docker push
28        run: |
29          docker push tilcica/sa3-cicd:latest

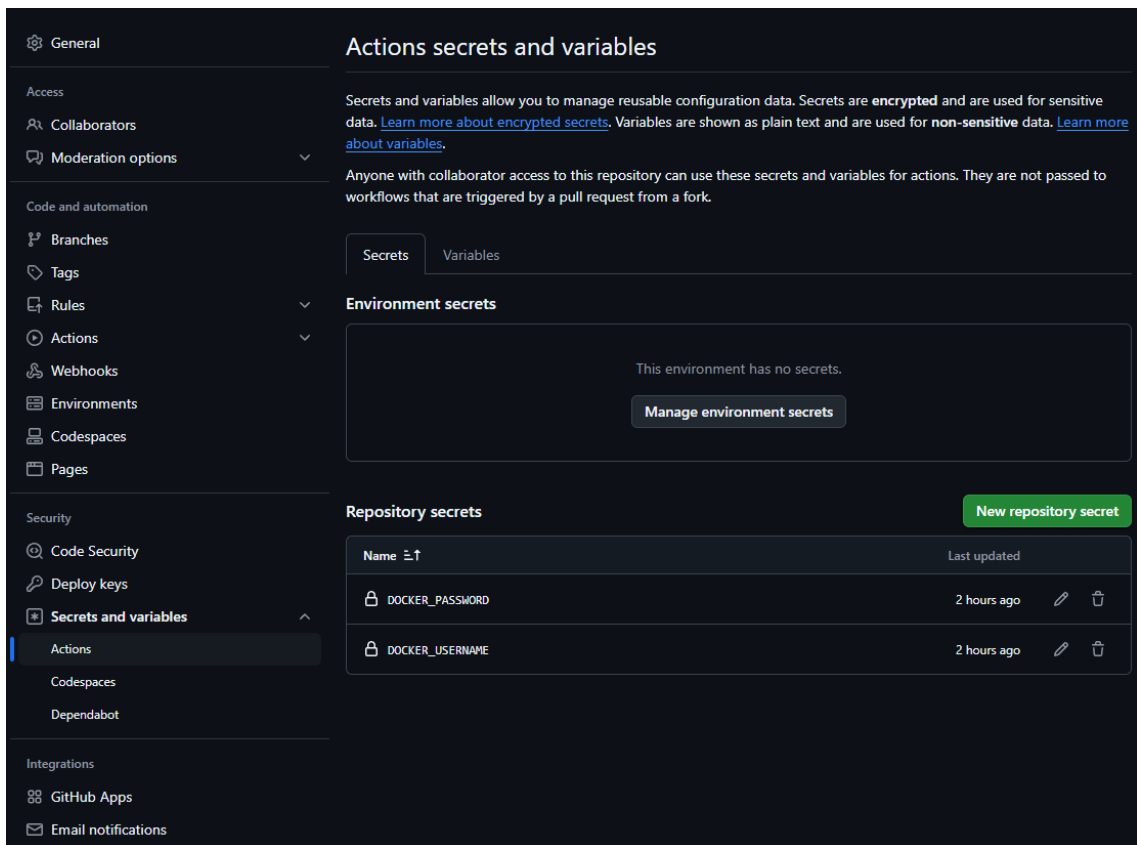
```

## 3 Posnetki zaslona

### 3.1 Github

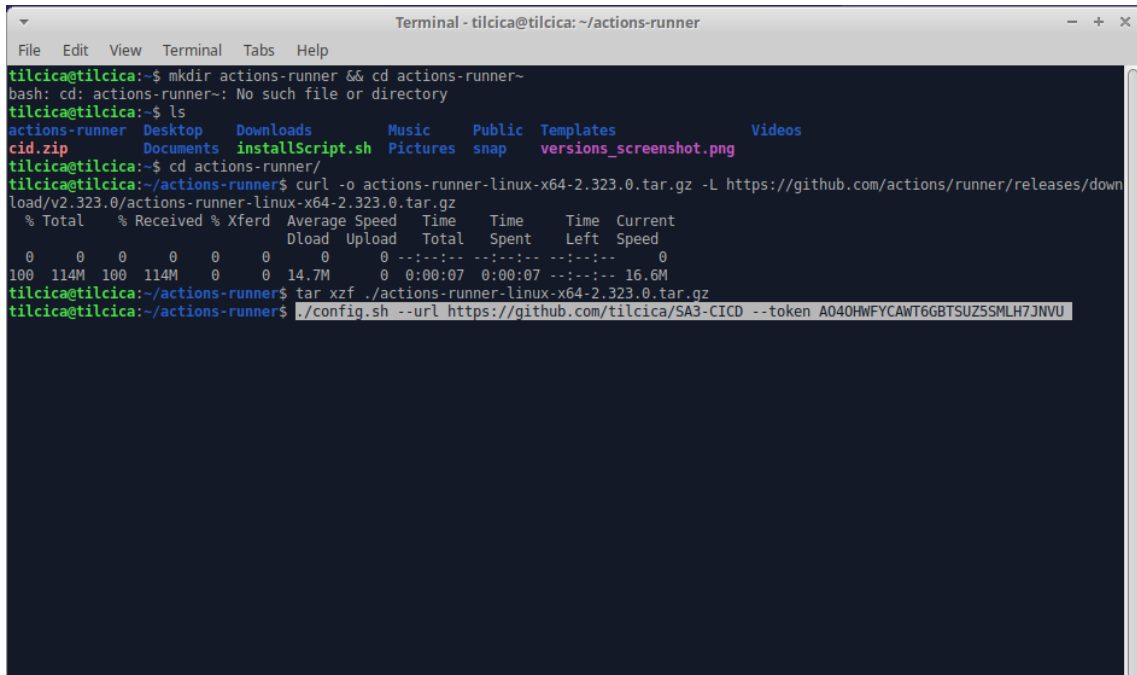


Slika 1: Github Actions



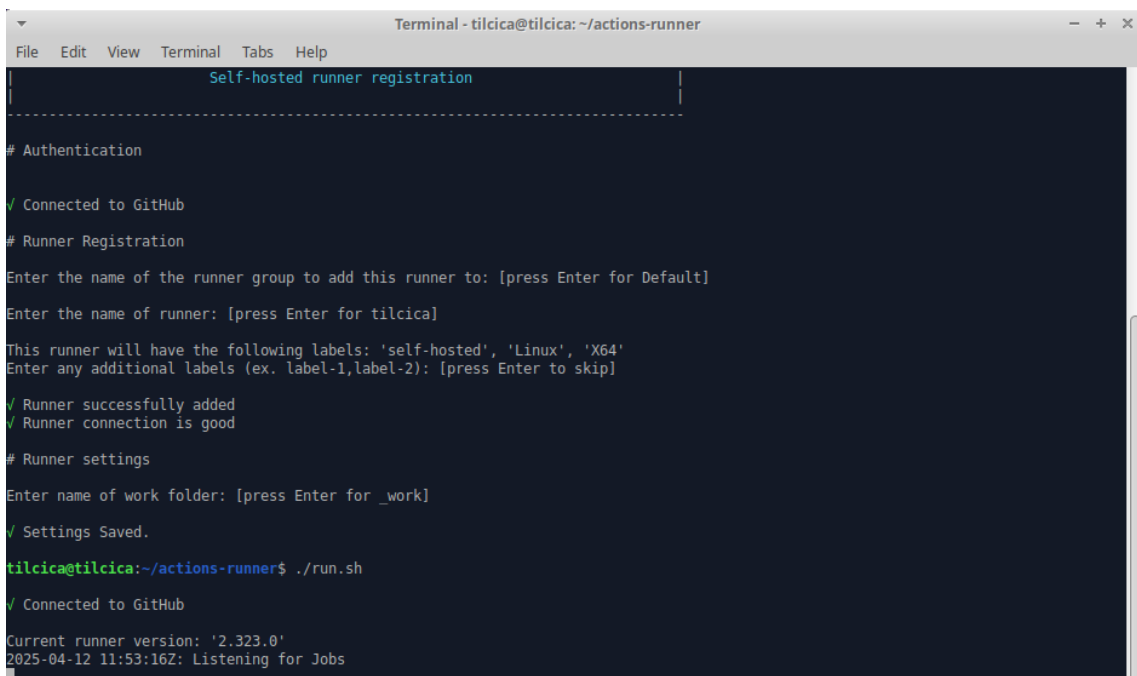
Slika 2: Github Secrets

## 3.2 Linux runner strežnik



```
Terminal - tilcica@tilcica: ~/actions-runner
File Edit View Terminal Tabs Help
tilcica@tilcica:~$ mkdir actions-runner && cd actions-runner~
bash: cd: actions-runner~: No such file or directory
tilcica@tilcica:~$ ls
actions-runner Desktop Downloads Music Public Templates Videos
cid.zip Documents installScript.sh Pictures snap versions_screenshot.png
tilcica@tilcica:~$ cd actions-runner/
tilcica@tilcica:~/actions-runner$ curl -o actions-runner-linux-x64-2.323.0.tar.gz -L https://github.com/actions/runner/releases/download/v2.323.0/actions-runner-linux-x64-2.323.0.tar.gz
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
100 114M 100 114M 0 0 14.7M 0 0:00:07 0:00:07 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
tilcica@tilcica:~/actions-runner$ tar xzf ./actions-runner-linux-x64-2.323.0.tar.gz
tilcica@tilcica:~/actions-runner$ ./config.sh --url https://github.com/tilcica/SA3-CICD --token A040HwFYCAWT6GBT5UZ5SMLH7JNVU
```

Slika 3: Vzpostavljanje Runner Strežnika 1



```
Terminal - tilcica@tilcica: ~/actions-runner
File Edit View Terminal Tabs Help
Self-hosted runner registration
-----
# Authentication
✓ Connected to GitHub

# Runner Registration
Enter the name of the runner group to add this runner to: [press Enter for Default]
Enter the name of runner: [press Enter for tilcica]

This runner will have the following labels: 'self-hosted', 'Linux', 'X64'
Enter any additional labels (ex. label-1,label-2): [press Enter to skip]

✓ Runner successfully added
✓ Runner connection is good

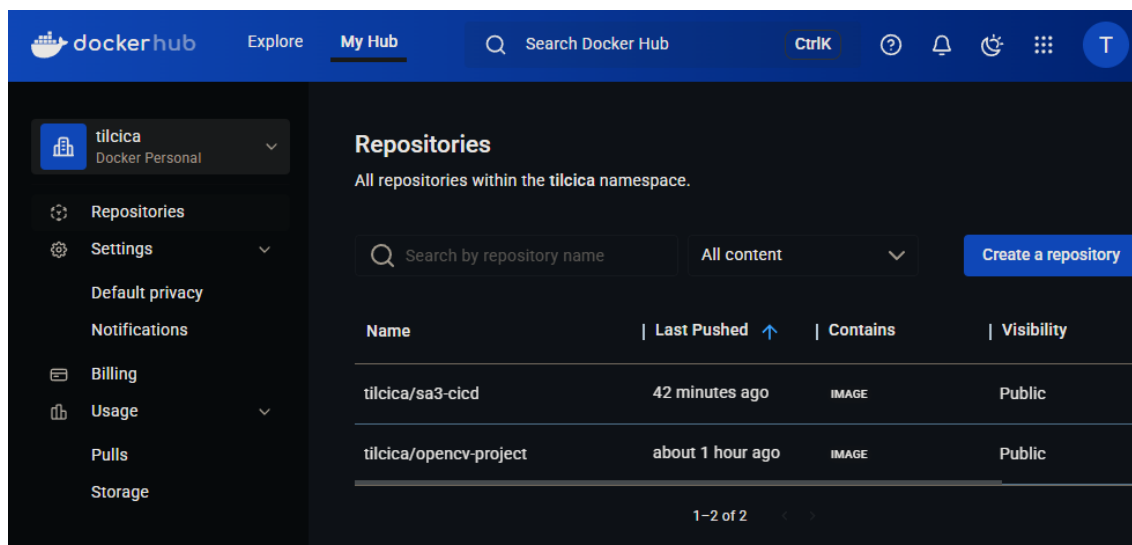
# Runner settings
Enter name of work folder: [press Enter for _work]
✓ Settings Saved.

tilcica@tilcica:~/actions-runner$ ./run.sh
✓ Connected to GitHub
Current runner version: '2.323.0'
2025-04-12 11:53:16Z: Listening for Jobs
```

Slika 4: Vzpostavljanje Runner Strežnika 2



### 3.3 Docker hub



Slika 5: sa3-cicd projekt na DockerHub