



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий (ИИТ)

Кафедра практической и прикладной информатики(ППИ)

ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ

по дисциплине «Микросервисная архитектура»

Практическое задание № 7

Студент группы *ИКБО-25-20, Валиков Кирилл*
Юрьевич

подпись)

Преподаватель *Лантев И.А.*

подпись)

Отчет представлен _____ 202__ г.

1 CI в DockerHub

```
Code Blame 46 lines (37 loc) · 1.38 KB Code 55% faster with GitHub Copilot

1 # This is a basic workflow to help you get started with Actions
2
3 name: CI
4 |
5 # Controls when the workflow will run
6 on:
7   # Triggers the workflow on push or pull request events but only for the "main" branch
8   push:
9     branches: [ "main" ]
10  pull_request:
11    branches: [ "main" ]
12
13 # Allows you to run this workflow manually from the Actions tab
14 workflow_dispatch:
15
16 # A workflow run is made up of one or more jobs that can run sequentially or in parallel
17 jobs:
18   # This workflow contains a single job called "build"
19   build:
20     # The type of runner that the job will run on
21     runs-on: ubuntu-latest
22
23     # Steps represent a sequence of tasks that will be executed as part of the job
24     steps:
25       # Checks-out your repository under $GITHUB_WORKSPACE, so your job can access it
26       - name: pulling
27         uses: actions/checkout@v3
28
29       - name: Set up Docker Buildx
30         uses: docker/setup-buildx-action@v1
31         with:
32           version: v0.7.0
33
34       # Runs a single command using the runners shell
35       - name: docker-compose build
36         run: docker-compose build
37
38       - name: docker login
39         uses: docker/login-action@v3.0.0
40         with:
41           username: ${ secrets.DOCKER_USERNAME }
42           password: ${ secrets.DOCKER_PASSWORD }
43
44       # Runs a set of commands using the runners shell
45       - name: pushing
46         run: docker-compose push
```

Рисунок 1. ci.yml

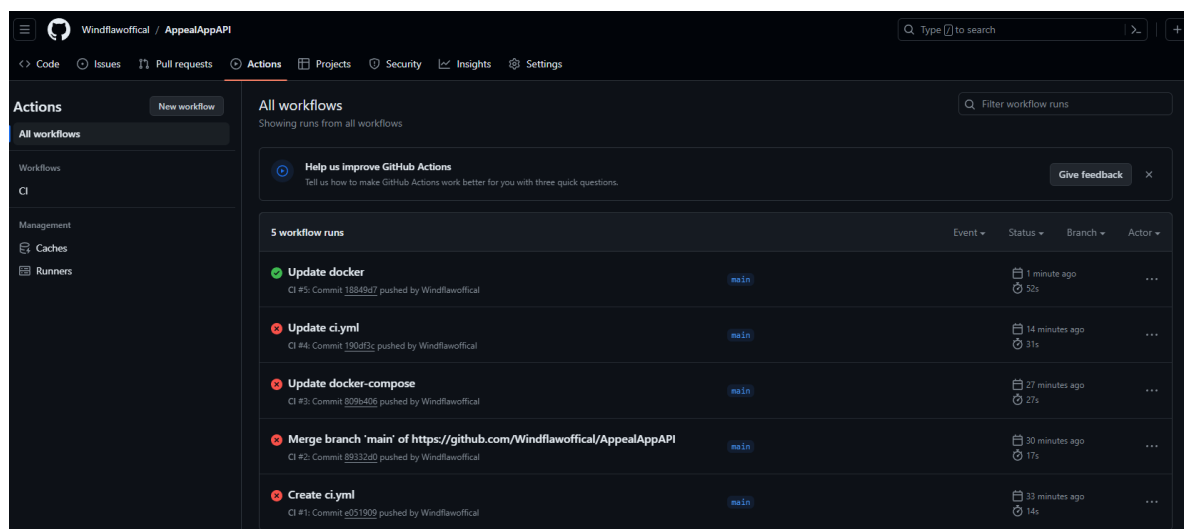


Рисунок 2. Workflow runs

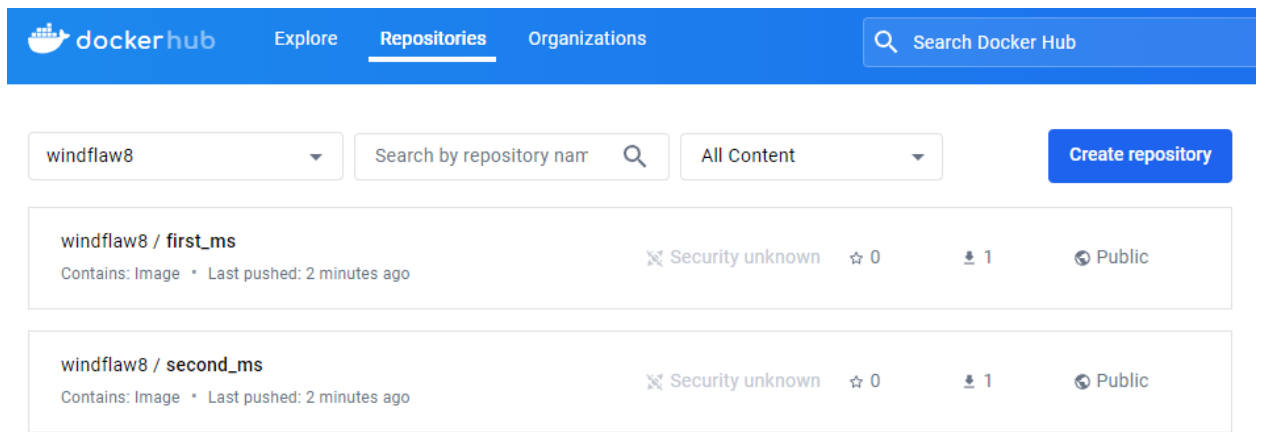


Рисунок 3. Созданные сервисы на dockerhub

2 CD YCloud

```
48   build-and-push-to-yc:
49     runs-on: ubuntu-latest
50     steps:
51       - name: pulling
52         uses: actions/checkout@v3
53
54       - name: docker build
55         run: docker build --file ./pr6_first-microservice/Dockerfile -t cr.yandex/${{ secrets.YC_REGISTRY_ID }}/firstms .
56
57       - name: Login to YC Registry
58         uses: docker/login-action@v3.0.0
59         with:
60           registry: cr.yandex
61           username: json_key
62           password: ${ secrets.YC_KEYS }
63
64       - name: pushing
65         run: docker push cr.yandex/${{ secrets.YC_REGISTRY_ID }}/firstms:latest
66
67   deploy:
68     name: Deploy
69     runs-on: ubuntu-latest
70     needs: [build-and-push-to-yc]
71     steps:
72       - name: deploy serverless container
73         uses: yc-actions/yc-sls-container-deploy@v1
74         with:
75           yc-sa-json-credentials: ${ secrets.YC_KEYS }
76           container-name: ${ secrets.YC_CONTAINER_NAME }
77           folder-id: ${ secrets.YC_FOLDER_ID }
78           revision-image-url: cr.yandex/${{ secrets.YC_REGISTRY_ID }}/firstms:latest
79           revision-service-account-id: ${ secrets.YC_SA_ID }
80           revision-env:
81             POSTGRES_URL=${ secrets.POSTGRES_URL }
82           revision-memory: 4GB
```

Рисунок 4. ci-cd.yml

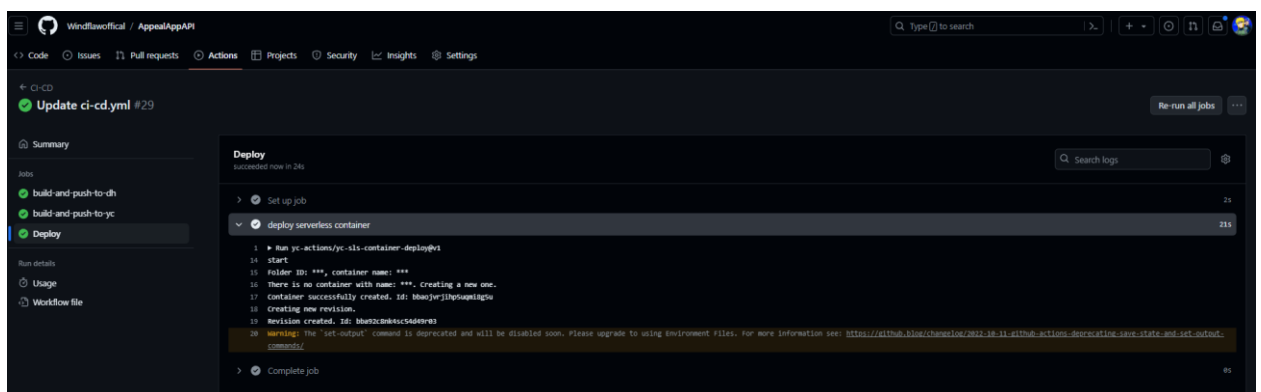


Рисунок 5. Workflows runs

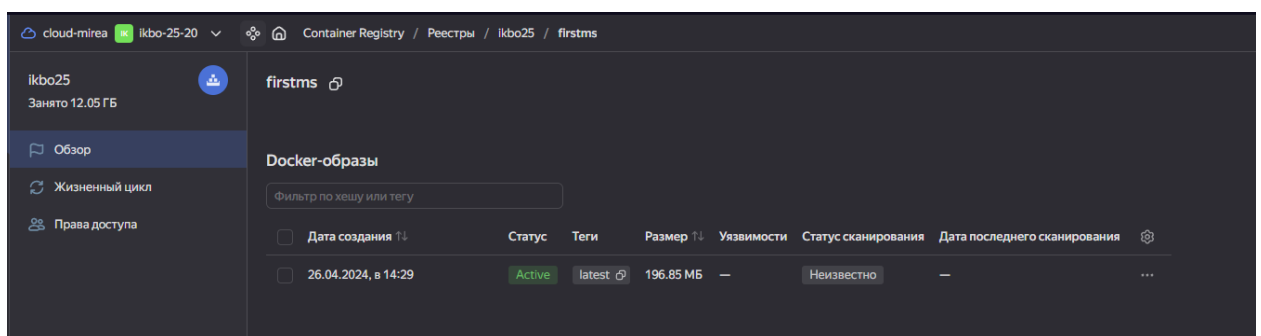


Рисунок 6. Репозиторий в реестре образов

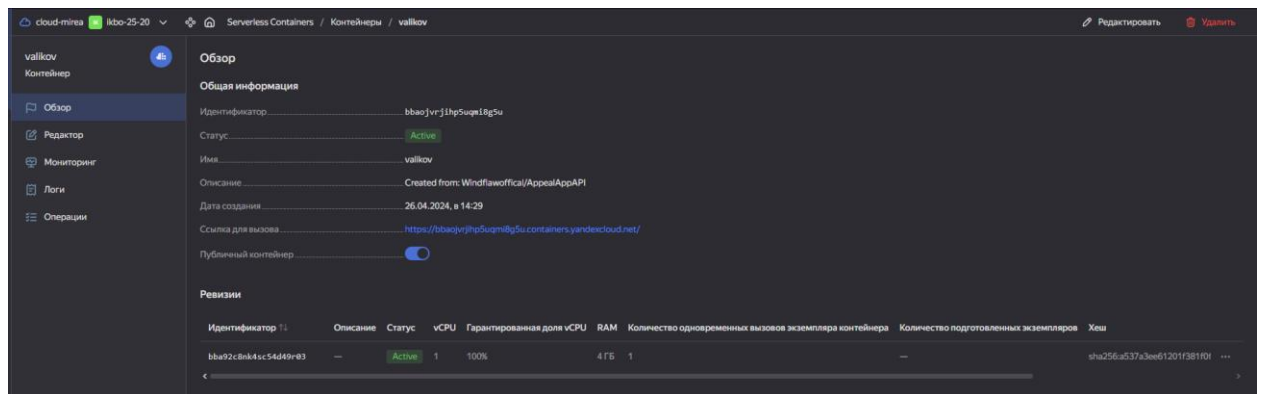


Рисунок 7. Настройки бессерверного контейнера