

Финал истории Rososo: CI в GHA

Дмитрий Тучс

 Heisenbug 2024 Autumn offline





Head of QA @ Dodo
Engineering



@[dtuchs](#) (и @[likeaduck](#))



Преподаю в QA.GURU

Опыт в разработке,
аналитике, менеджменте
более 15 лет





Rococo

127.0.0.1:3000/painting

Finish update


Rococo

Картины Художники Музеи  


Картины

Добавить картину


Искать картины...



Female nude



Утро в сосновом лесу



Над вечным покоем

127.0.0.1:3000/painting/67032d32-5c2c-450a-b360-510e12acb425



<https://github.com/dtuchs/heisenbug-2023-autumn>

```
1 package guru.qa.rococo;
2
3 > import ...
4
5
6
7
8
9
10
11
12
13
14
15
16
17 @WebTest
18 public class DemoWebTest {
19
20     @BeforeEach
21     void addAllureSelenide() {
22         SelenideLogger.addListener(
23             name: "AllureSelenide",
24             new AllureSelenide()
25                 .screenshots( saveScreenshots: true)
26                 .savePageSource( savePageHtml: true)
27         );
28     }
29
30     @ApiLogin(username = "dima", password = "12345")
31     @Test
32     void usernameShouldBeVisibleInProfile() {
33         ${ cssSelector: "[data-testid='avatar']" }.click();
34         ${ cssSelector: "h4.text-center" }.should(text("dima"));
35     }
36
37     @CsvSource({
38         "123, Утро в сосновом лесу",
39         "124, Над вечным покоем"
40     })
41     @ApiLogin(username = "ivan", password = "12345")
42     @ParameterizedTest
43     void paintingNameShouldEditedByAuthorizedUser(@AllureParamId String allureId, String painting) {
44         ${ cssSelector: "nav.list-nav a" }.find(text("Картины")).click();
45         ${ cssSelector: "div.text-center" }.find(text(painting)).parent().click();
46         ${ cssSelector: "[data-testid='edit-painting']" }.click();
47         ${ cssSelector: "input[name='title']" }.setValue("Edited");
48     }
49 }
```

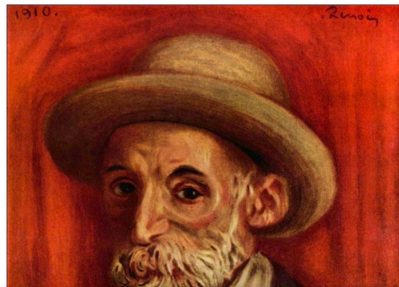


<https://github.com/dtuchs/heisenbug-2023-autumn>



 DODO ENGINEERING

The art of JUnit Extensions



3 Autumn online

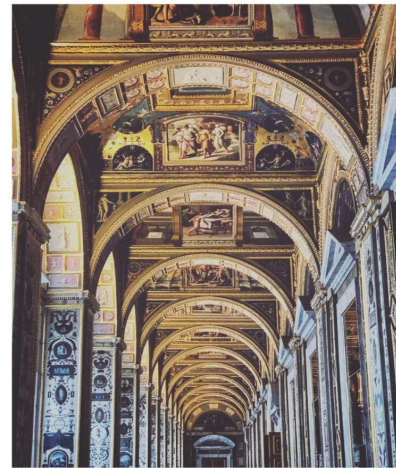
 DODO ENGINEERING

The art of JUnit Extensions

#2

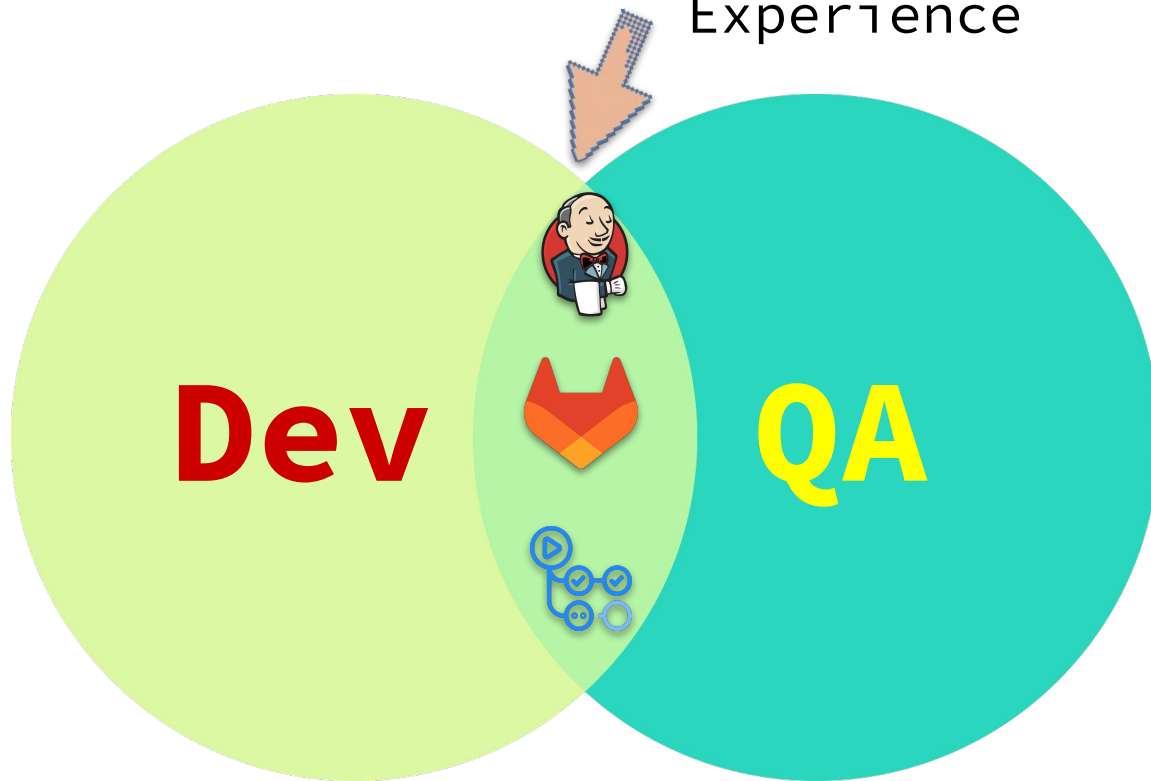
Дмитрий Тучс

 Heisenbug 2024 Spring online



Для чего это все?

Тут мы улучшаем Dev Experience





То есть, в идеале:

1

```
~/IdeaProjects/heisenbug-2023-autumn git:[feature-0]  
git push origin feature-0
```

2



base: master ▾



compare: feature-0 ▾

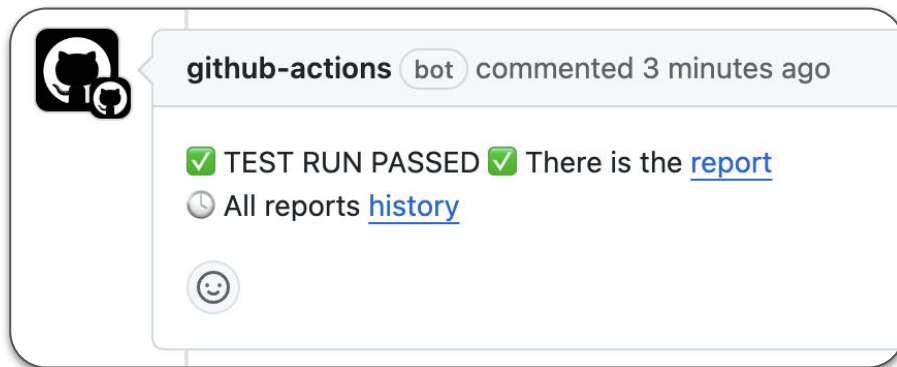
3





То есть, в идеале:

4



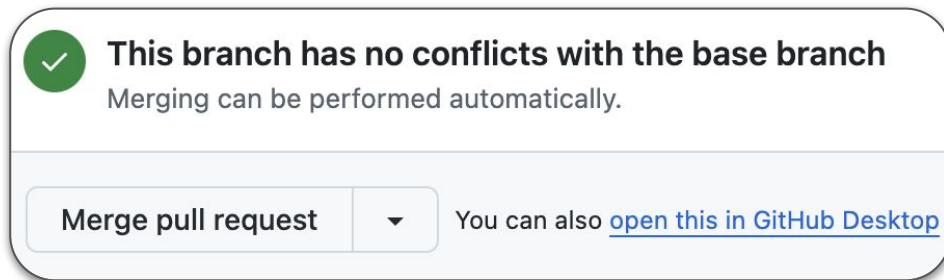
github-actions bot commented 3 minutes ago

✅ TEST RUN PASSED ✅ There is the [report](#)

🕒 All reports [history](#)

😊

5



✅ **This branch has no conflicts with the base branch**
Merging can be performed automatically.

Merge pull request ▾ You can also [open this in GitHub Desktop](#)



Есть
проблема



1

Указать репо с тестами

Запустить (gradle test)

Allure чтоб тоже был



Отправить в slack!



Ночной прогон



У нас же есть Devops!



1





Как учат на курсах “стань АQA”

1

Git ?

Repositories ?

Repository URL ?

`https://github.com/StudentName/tests`

2

Invoke Gradle ?

Gradle Version

Gradle 7.4.1

Use Gradle Wrapper ?

Tasks ?

clean
test



Как учат на курсах “стань АQA”

3

☰ Allure Report

Disabled

Results:

Paths to Allure results directories relative
E.g. **target/allure-results**.

Path

build/allure-results

4

</> Changes

📁 Workspace

▶ Build Now


⚙️ Configure

🗑️ Delete Project




Как учат на курсах “стань АQA”


5

 **Build History** trend ▾

🔍 Filter... /

✓ #11 

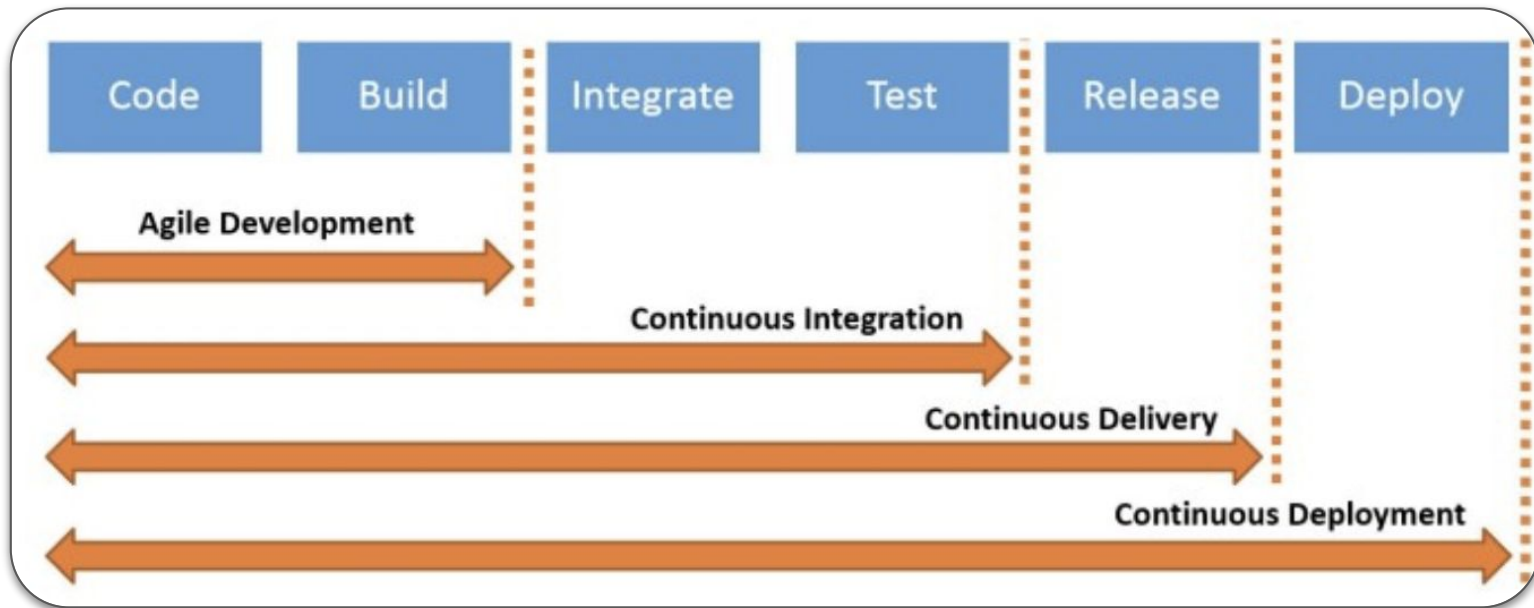
| Apr 19, 2024, 11:43 AM

✓ #10 

| Apr 19, 2024, 11:43 AM



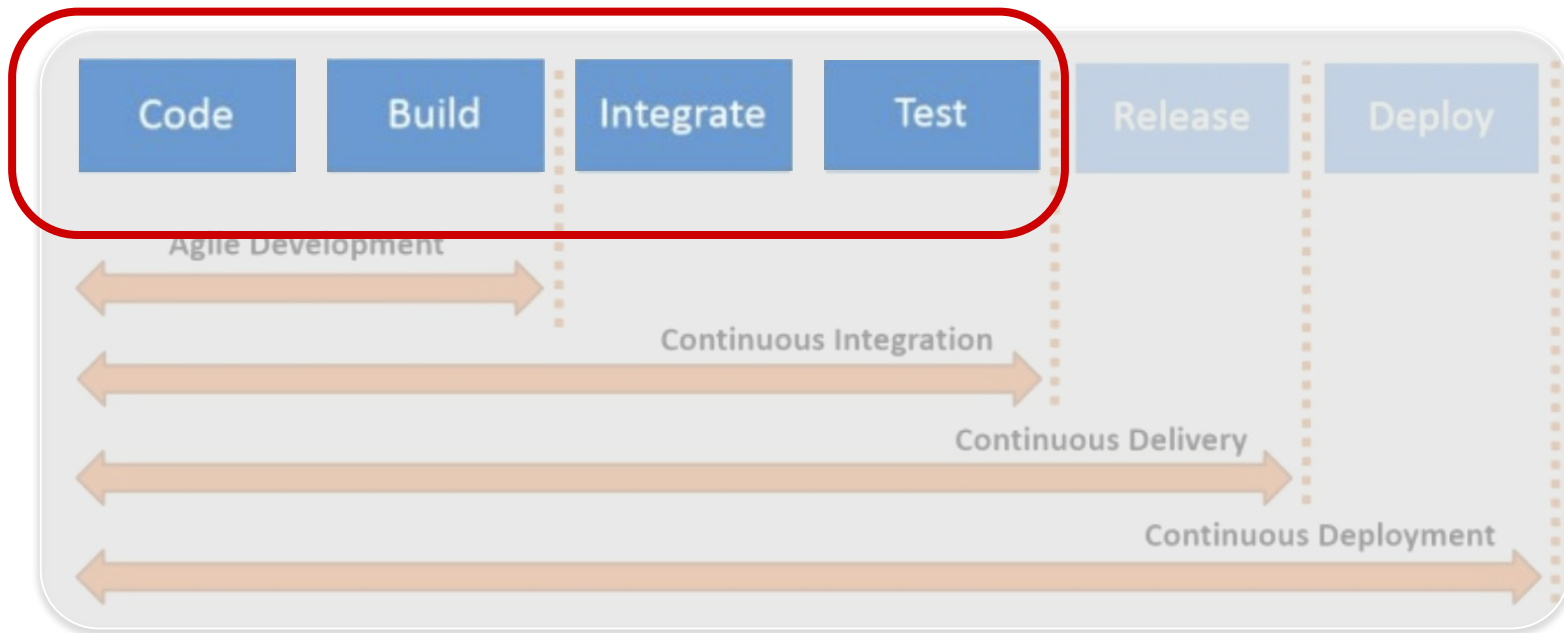
Как учат на курсах “стань АQA”





Как учат на курсах “стань АQA”

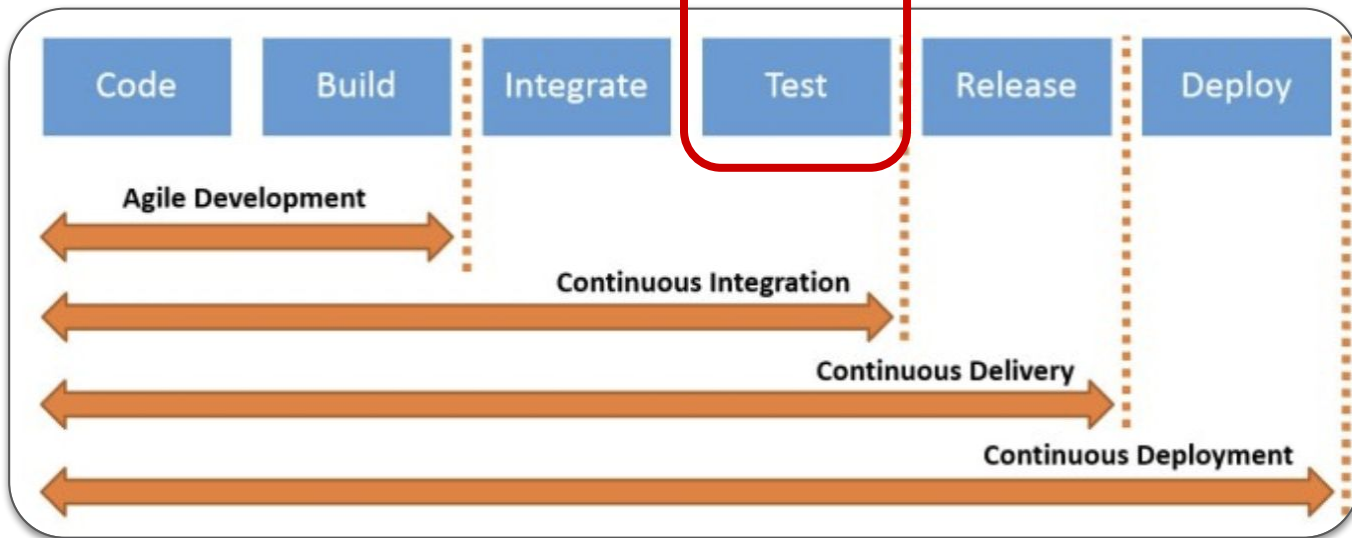
Код тестов - собрать - прогнать





Как учат на курсах “стань АQA”

Код продукта – собрать – тесты – release – deploy



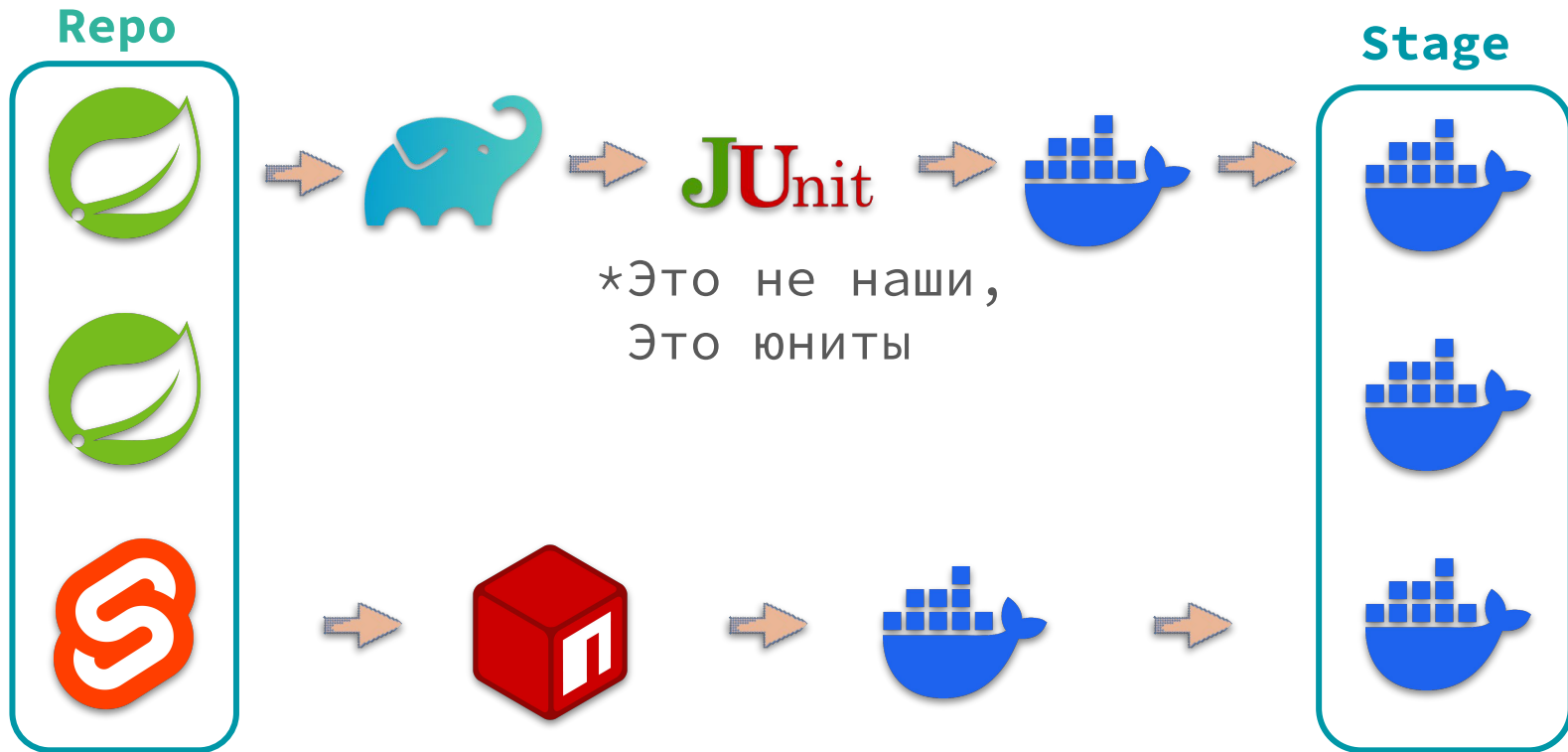


Вместо выводов (если не успеем)

**СИ != указать путь
к геро с тестами и
настроить Allure**



Что делают разработчики



Что будем делать мы

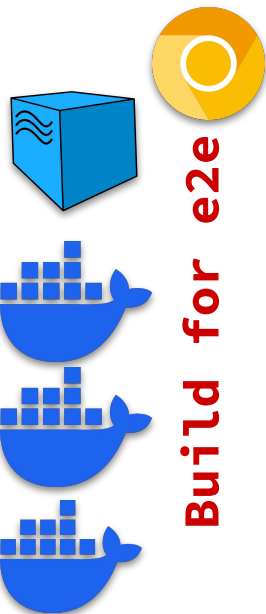
Repo



PR



Comment



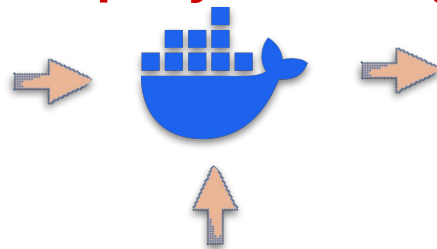
Build for e2e

Наши тесты

Save report

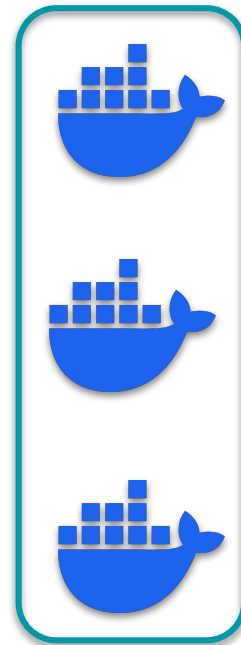


Deploy to Stage*



Only updated!

Stage





Почему тесты тоже в docker?

- 1** **Consistent env**
- 2** **Isolation**
- 3** **CI/CD friendly**
- 4** **Production like**

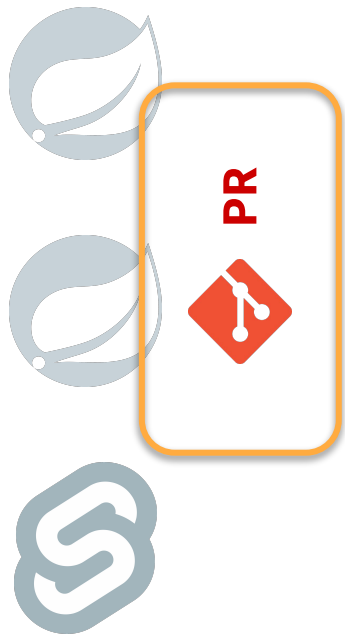


Приступим?

Demo time



Pull Request



```
name: e2e
```

```
on:
```

```
  pull_request:
```

```
    types: [opened, reopened, synchronize]
```



НОВЫЕ КОММИТЫ

<https://docs.github.com/ru/actions/writing-workflows/choosing-when-your-workflow-runs/events-that-trigger-workflows>



Job

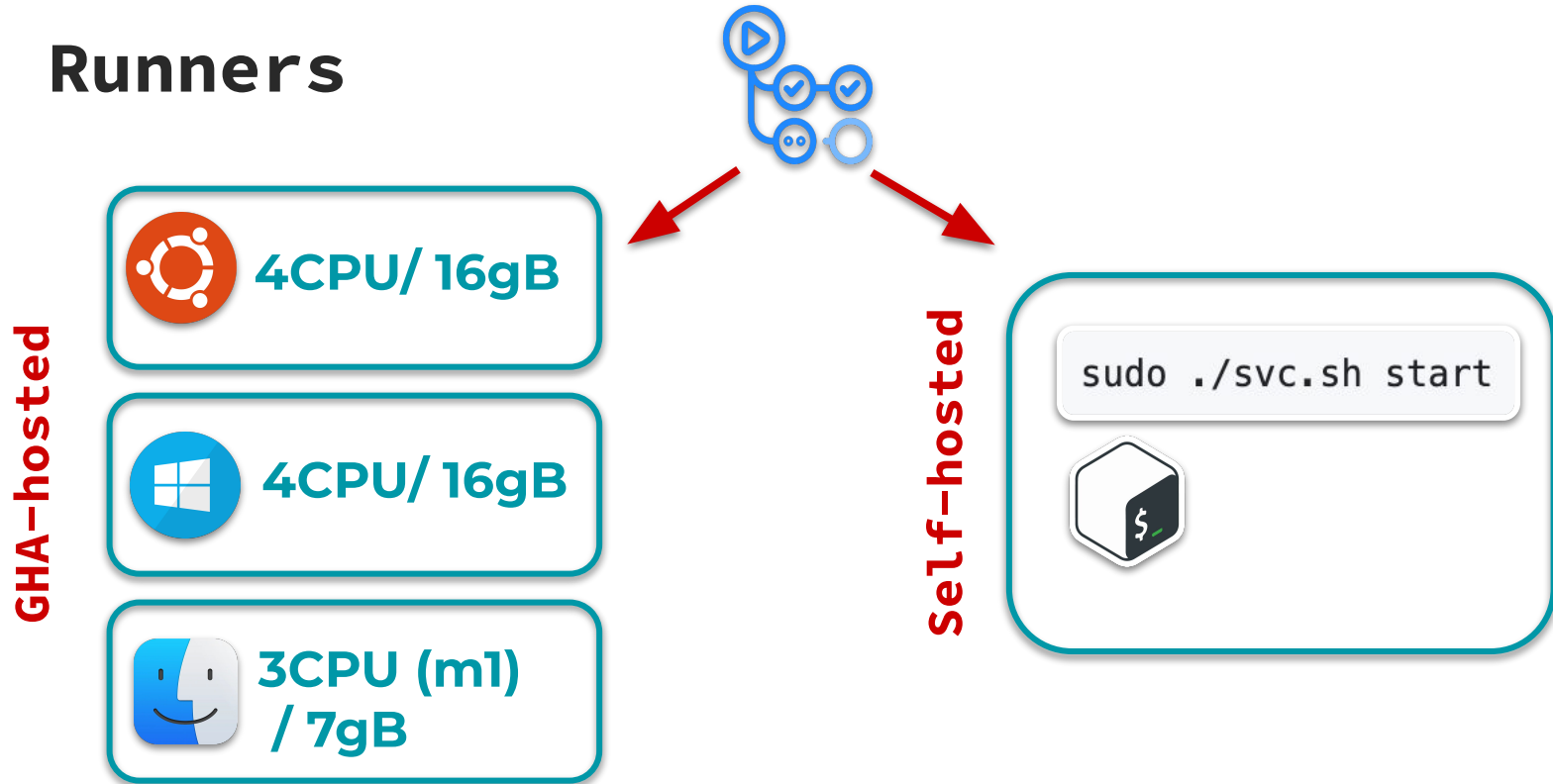
```
jobs:  
  e2e:  
    runs-on: ubuntu-latest  
    env:  
      ARCH: amd64  
    ...  
    steps:  
      - name: Setup JDK  
        uses: actions/setup-java@v4
```



Где выполняется
наш код ворклофа



Runners



<https://docs.github.com/ru/actions/using-github-hosted-runners/using-github-hosted-runners>

<https://docs.github.com/ru/actions/hosting-your-own-runners/managing-self-hosted-runners/about-self-hosted-runners>



Runners



`runs-on: ubuntu-latest`

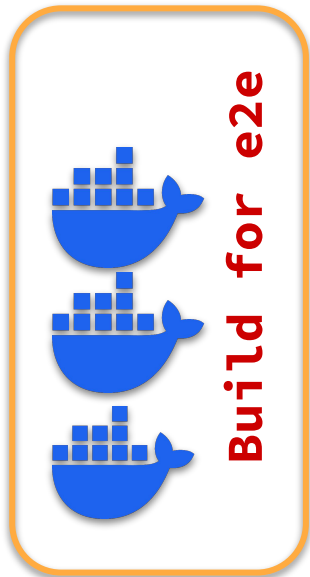


<https://docs.github.com/ru/actions/using-github-hosted-runners/using-github-hosted-runners>

<https://docs.github.com/ru/actions/hosting-your-own-runners/managing-self-hosted-runners/about-self-hosted-runners>



Build project for e2e test



```
client.rococo.dc:  
  container_name: client.rococo.dc  
  image: dtuchs/rococo-client:latest  
  build:  
    context: ./rococo-client  
    dockerfile: ./Dockerfile  
  ports:  
    - 80:80  
  depends_on:  
    api.rococo.dc:  
      condition: service_started  
  networks:  
    - rococo-network
```



Spring boot: no needed Dockerfile

Maven™



Jib



Gradle

```
tasks.jib.dependsOn test
tasks.jibDockerBuild.dependsOn test
```



Иначе теряем юниты :(

Build backend for e2e test

Установим
Java

Сделаем
git fetch

Соберем
проект

Упакуем JAR
в Docker



Используй Action!



```
./gradlew jibDockerBuild -x :rococo-e2e:test
```



Actions

actions / setup-java

<> Code Issues 40 Pull requests

setup-java Public

Languages

Language	Percentage
TypeScript	94.0%
Shell	5.4%
Other	0.6%

<https://github.com/actions/setup-java>

<https://resources.github.com/learn/pathways/automation/advanced/building-your-first-custom-github-action/>



Profiles

 application.yaml

```
spring:
  config:
    activate:
      on-profile: 'docker'

security:
  oauth2:
    resourceserver:
      jwt:
        issuer-uri: http://auth.rococo.dc:9000
```



Profiles

☰ .env

☰ .env.docker

```
VITE_AUTH_URL=http://auth.rococo.dc:9000  
VITE_FRONT_HOST=client.rococo.dc
```



Profiles

- ⓘ Config
- © DockerConfig
- © LocalConfig

```
public class LocalConfig
    implements Config {

    @Override public String frontUrl() {
        return "http://127.0.0.1:3000";
    }
}
```




Profiles

- ⓘ Config
- © DockerConfig
- © LocalConfig

```
public class LocalConfig
    implements Config {

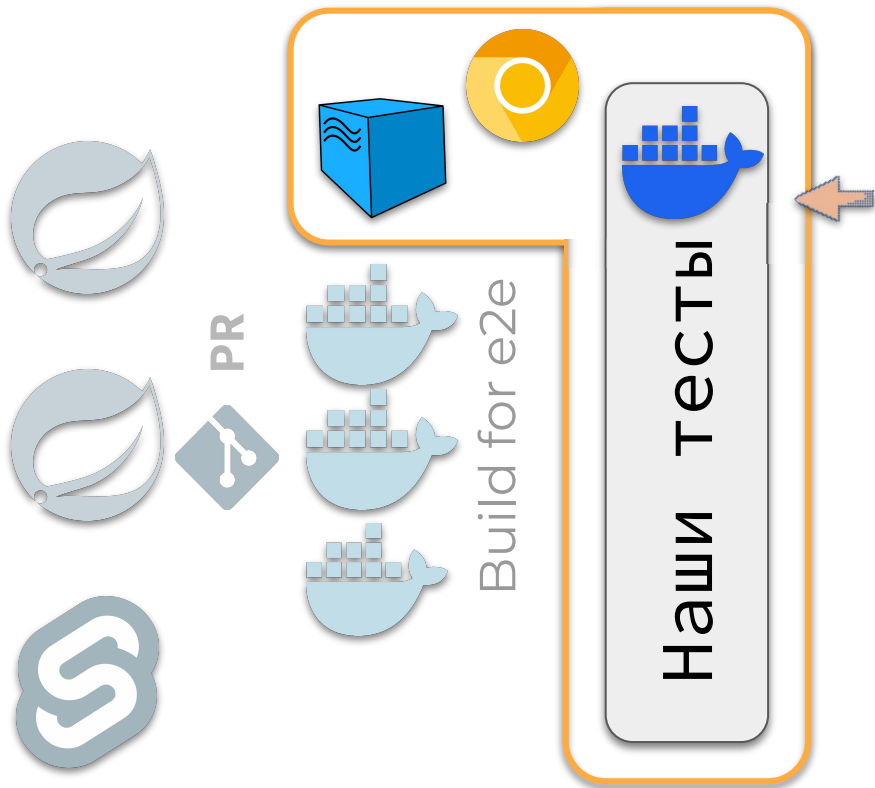
    @Override public String frontUrl() {
        return "http://127.0.0.1:3000";
    }
}
```

```
public class DockerConfig implements Config {
    static {
        Configuration.remote = "http://selenoid:4444/wd/hub";
    }

    @Override public String frontUrl() {
        return "http://client.rococo.dc";
    }
}
```



e2e tests in Docker



```
FROM eclipse-temurin:21-jdk
```

```
WORKDIR /heisenbug-2023-autumn
```

```
COPY ./gradle ./gradle
```

```
COPY ./rococo-e2e ./rococo-e2e
```

```
COPY ./gradlew ./
```

```
COPY ./build.gradle ./
```

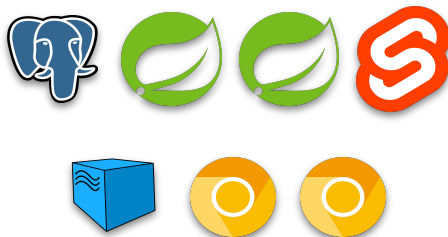
```
COPY ./settings.gradle ./
```

```
COPY ./gradle.properties ./
```

```
CMD ./gradlew test -Dtest.env=docker
```



Selenoid



```
selenoid:  
  container_name: selenoid  
  image: aerokube/selenoid:1.11.3  
  platform: linux/${ARCH}  
  volumes:  
    - ./selenoid:/etc/selenoid  
    - /var/run/docker.sock:/var/run/docker.sock  
  restart: unless-stopped  
  command: [ "-conf", "/etc/selenoid/browsers.json", "-limit", "2" ...  
]  
  ports:  
    - 4444:4444  
  networks:  
    - rococo-network
```

↑
Ограничения не только
в наших головах :(



Run tests!

**Поднять
docker-compose
с тестами**

**Узнать, не
упали ли тесты**

**Написать if-чик
и сообщить о
результатах**



**Дождаться
завершения
тестов**

**Завершить
docker-compose**



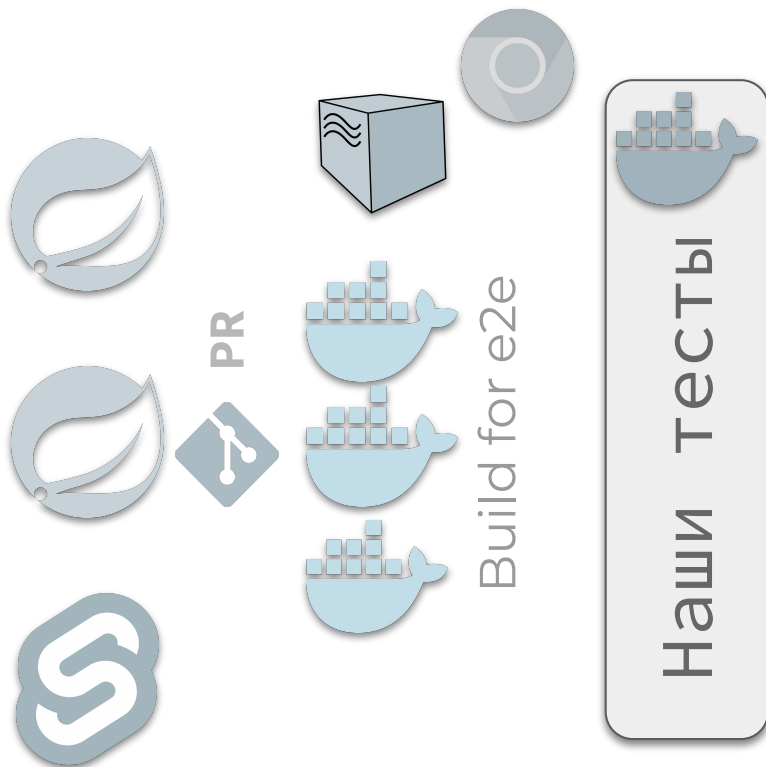
Run tests!

```
docker compose -f docker-compose-e2e.yml up -d
docker wait rococo-e2e
exit_code=$(docker inspect -f '{{.State.ExitCode}}' rococo-e2e)
echo "e2e_exit_code=$exit_code" >> $GITHUB_OUTPUT
docker compose -f docker-compose-e2e.yml down
if [ "$exit_code" -eq "0" ]; then
    echo "Tests passed successfully!"
    exit 0
else
    echo "Tests failed!"
    exit 1
fi
```



**Делимся результатами
с другими step-ами**

e2e tests in Docker



The screenshot shows a GitHub repository page for "ALLURE DOCKER SERVICE". A red arrow points to the "Save report" button. The page displays a passing workflow, 13M docker pulls, and 600 stars.

Save report

ALLURE DOCKER SERVICE

Allure Docker Service Workflow **passing**

docker pulls **13M**

★ Starred **600**



JUnit Extension



И немного магии

```
ALLURE_DOCKER_API: ${ secrets.ALLURE_DOCKER_API }  
BUILD_URL: ${ github.server_url }/${ github.repository  
}/actions/runs/${ github.run_id }  
EXECUTION_TYPE: github
```

- **name**: Get the last commit message

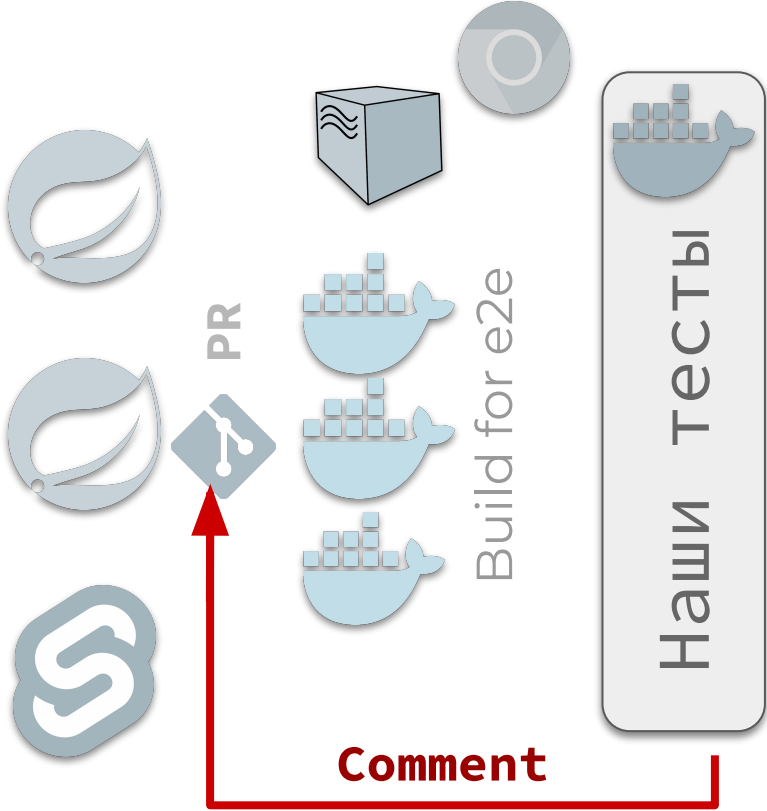
run: |

```
MESSAGE=$(git log -1 --pretty=%B)
```

```
echo "HEAD_COMMIT_MESSAGE=$MESSAGE" >> $GITHUB_ENV
```




Comment



Save report

The screenshot shows a GitHub repository page for **ALLLURE DOCKER SERVICE**. It features the Docker logo and the Docker whale mascot. The repository status is **Allure Docker Service Workflow** *passing*. It also shows **docker pulls: 13M** and a **Starred 600** button.



Comment

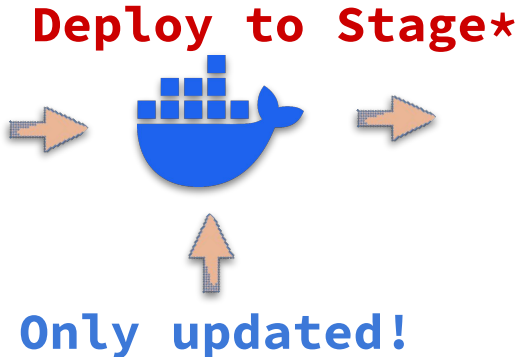
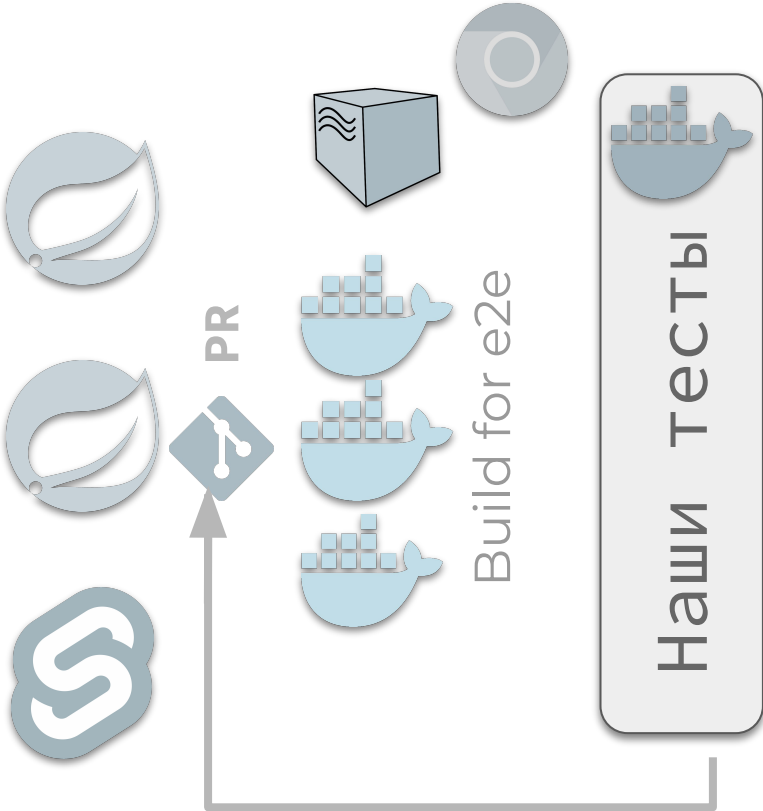
✅ TEST RUN PASSED ✅ There is the [report](#)

🕒 All reports [history](#)





Deploy *





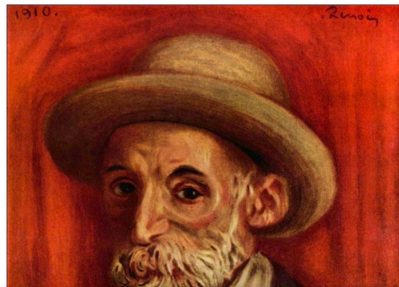
Проверяем, что именно деплоить

```
- name: Check changes path
  uses: dorny/paths-filter@v3
  id: changes
  with:
    filters: |
      auth:
        - 'rococo-auth/**'
      api:
        - 'rococo-api/**'
      front:
        - 'rococo-client/**'
```



 DODO ENGINEERING

The art of JUnit Extensions



3 Autumn online

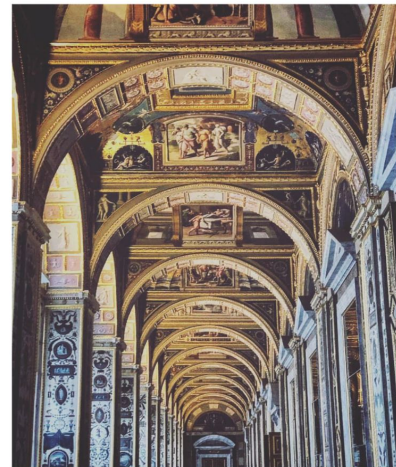
 DODO ENGINEERING

The art of JUnit Extensions

#2

Дмитрий Тучс

 Heisenbug 2024 Spring online





Rococo

127.0.0.1:3000/painting


Finish update


Rococo

Картины Художники Музеи  


Картины

[Добавить картину](#)


Искать картины... 



Female nude



Утро в сосновом лесу



Над вечным покоем

127.0.0.1:3000/painting/67032d32-5c2c-450a-b360-510e12acb425



<https://github.com/dtuchs/heisenbug-2023-autumn>



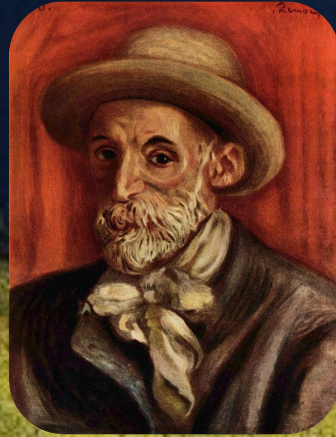
Выводы

**Без хорошего CI
тесты бесполезны**



Выводы

**Никто не сделает
лучше, чем вы**





Мой TG канал <https://t.me/likeaduck>