



РЕПОРТИНГ



АРТЕМ РОМАНОВ



АРТЕМ РОМАНОВ

Инженер по обеспечению качества



[romanov-artyom](#)



[@Artem_Telegram](#)



[Романов Артём](#)



ПЛАН ЗАНЯТИЯ

1. [Репортинг](#)
2. [Gradle](#)
3. [Allure](#)
4. [Report Portal](#)
5. [Итоги](#)



РЕПОРТИНГ

РЕПОРТИНГ

Репортинг по результатам прогона авто-тестов - тема, в которой достаточно разных мнений: от того, что достаточно бейджиков сборки



, до того, что отчёты должны быть максимально подробными.

Наш ключевой подход в этом вопросе - репортинг должен решать ваши проблемы. Если вам достаточно бейджика сборки и логов CI, то не нужно накручивать целую систему сверху.

И уж тем более не нужно "тащить" огромную систему репортинга, если ваши отчёты никто не использует (включая вас самих).



РЕПОРТИНГ

Ключевой вопрос: для чего вам нужен репортинг?



CI

Большинство CI систем содержат достаточно подробные логи сборки и прогона авто-тестов, что является в большинстве случаев вполне достаточным для анализа "падающих" тестов.

Мы же рассмотрим альтернативные варианты, начиная с самых простых, заканчивая системами с AI под капотом.

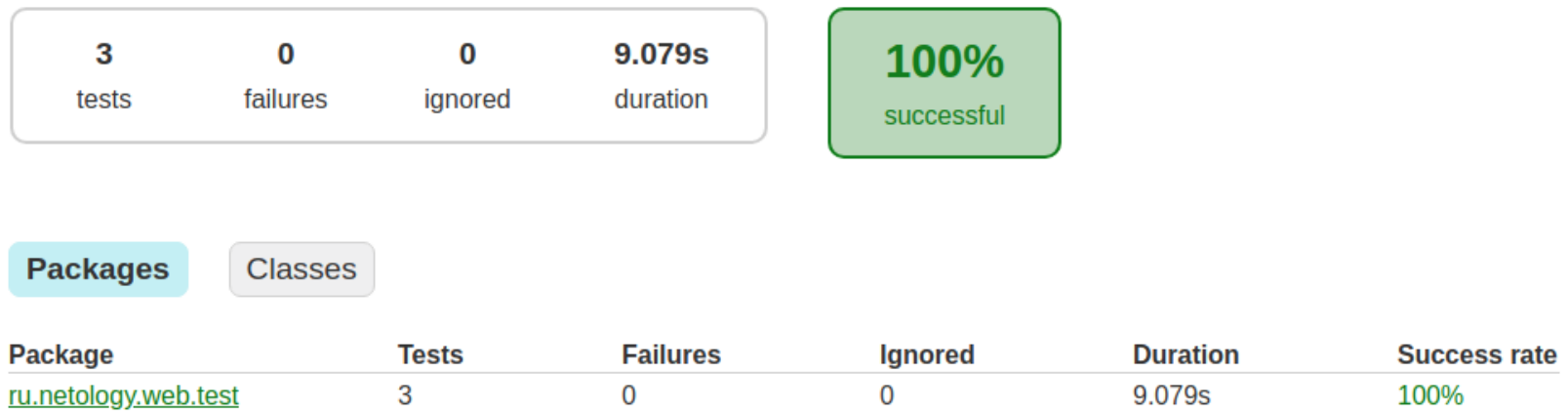


GRADLE

GRADLE

Gradle из коробки включает достаточно хорошо выглядящие отчёты:

Test Summary



GRADLE

При этом можно посмотреть время выполнения каждого теста:

MoneyTransferTest

[all](#) > [ru.netology.web.test](#) > MoneyTransferTest

3	0	0	9.079s
tests	failures	ignored	duration

100%
successful

Tests

Standard error

Test	Duration	Result
shouldTransferMoneyBetweenOwnCardsV1()	7.254s	passed
shouldTransferMoneyBetweenOwnCardsV2()	0.951s	passed
shouldTransferMoneyBetweenOwnCardsV3()	0.874s	passed

GRADLE

Кроме того логируется весь вывод в стандартный поток ошибок:

MoneyTransferTest

[all](#) > [ru.netology.web.test](#) > MoneyTransferTest

3	0	0	9.079s
tests	failures	ignored	duration

100%
successful

Tests

Standard error

```
Nov 17, 2019 12:46:08 AM com.codeborne.selenide.drivercommands.LazyDriver getAndCheckWebDriver
INFO: No webdriver is bound to current thread: 13 - let's create a new webdriver
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
Starting ChromeDriver 78.0.3904.70 (edb9c9f3de0247fd912a77b7f6cae7447f6d3ad5-refs/branch-heads/3904@{#800}) on port 15162
Only local connections are allowed.
Please protect ports used by ChromeDriver and related test frameworks to prevent access by malicious code.
Nov 17, 2019 12:46:13 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
Nov 17, 2019 12:46:13 AM com.codeborne.selenide.webdriver.BrowserResizer adjustBrowserSize
INFO: Set browser size to 1366x768
Nov 17, 2019 12:46:13 AM com.codeborne.selenide.webdriver.WebDriverFactory logBrowserVersion
INFO: BrowserName=chrome Version=78.0.3904.97 Platform=LINUX
Nov 17, 2019 12:46:13 AM com.codeborne.selenide.webdriver.WebDriverFactory createWebDriver
INFO: Selenide v. 5.3.1
Nov 17, 2019 12:46:13 AM com.codeborne.selenide.webdriver.WebDriverFactory logSeleniumInfo
INFO: Selenium WebDriver v. 3.141.59 build time: 2018-11-14T08:17:03
Nov 17, 2019 12:46:13 AM com.codeborne.selenide.drivercommands.CreateDriverCommand createDriver
INFO: Create webdriver in current thread 13: ChromeDriver -> ChromeDriver: chrome on LINUX (7434cbac8e865540d06c342c660043f5)
```

GRADLE

Есть фильтрация по упавшим тестам:

Test Summary

3	3	0	16.391s
tests	failures	ignored	duration

0%
successful

Failed tests

Packages

Classes

MoneyTransferTest. shouldTransferMoneyBetweenOwnCardsV1()

MoneyTransferTest. shouldTransferMoneyBetweenOwnCardsV2()

MoneyTransferTest. shouldTransferMoneyBetweenOwnCardsV3()

GRADLE

И, самое главное, стектрейсы при эксепшенах падающих тестов:

shouldTransferMoneyBetweenOwnCardsV1()

```
com.codeborne.selenium.ex.ElementNotFound: Element not found {[data-test-id=code] inputx}
Expected: visible
Screenshot: file:/aqa-code/bdd/page-object/build/reports/tests/1573940886474.0.png
Page source: file:/aqa-code/bdd/page-object/build/reports/tests/1573940886474.0.html
Timeout: 4 s.
Caused by: NoSuchElementException: no such element: Unable to locate element: {"method":"css selector","selector":"[data-test-id=code] inputx"}
    at com.codeborne.selenium.impl.WebElementSource.createElementNotFoundError(WebElementSource.java:37)
    at com.codeborne.selenium.impl.ElementFinder.createElementNotFoundError(ElementFinder.java:100)
    at com.codeborne.selenium.impl.WebElementSource.checkCondition(WebElementSource.java:65)
    at com.codeborne.selenium.commands.Should.execute(Should.java:24)
    at com.codeborne.selenium.commands.Should.execute(Should.java:10)
    at com.codeborne.selenium.commands.Commands.execute(Commands.java:145)
    at com.codeborne.selenium.impl.SelenideElementProxy.dispatchAndRetry(SelenideElementProxy.java:99)
    at com.codeborne.selenium.impl.SelenideElementProxy.invoke(SelenideElementProxy.java:65)
    at com.sun.proxy.$Proxy13.shouldBe(Unknown Source)
    at ru.netology.web.page.VerificationPage.<init>(VerificationPage.java:14)
    at ru.netology.web.page.LoginPageV1.validLogin(LoginPageV1.java:12)
    at ru.netology.web.test.MoneyTransferTest.shouldTransferMoneyBetweenOwnCardsV1(MoneyTransferTest.java:20)
```



ALLURE



ALLURE

[Allure](#) - фреймворк, предназначенный для создания визуально наглядной картины выполнения тестов.

Работает Allure в два этапа:

1. Подключается к тестовому фреймворку
2. Генерируются отчёты

ПОКЛЮЧАЕМ

```
1  plugins {
2      id 'java'
3      id 'io.qameta.allure' version '2.8.1'
4  }
5
6  ...
7
8  allure {
9      autoconfigure = true
10     version = '2.13.0' // Latest Allure Version
11
12     useJUnit5 {
13         version = '2.13.0' // Latest Allure Version
14     }
15 }
16
17 repositories {
18     jcenter()
19     mavenCentral()
20 }
```

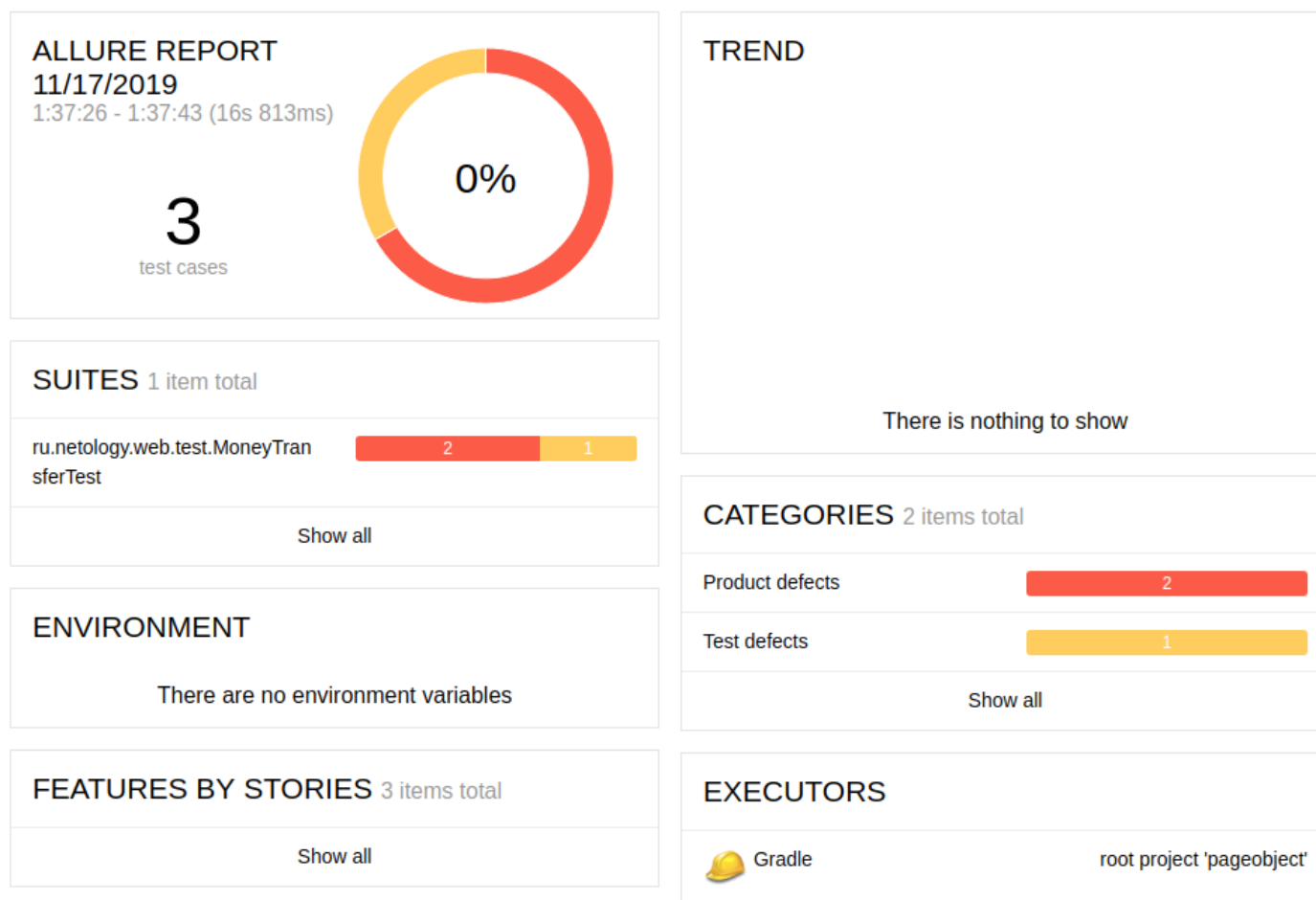



ЗАПУСКАЕМ


```
gradlew clean test allureReport  
gradlew allureServe
```


ALLURE


Allure генерирует более интересные отчёты, заодно пытаясь классифицировать дефекты как относящиеся к продукту или к авто-тестам:





ALLURE


 Allure


 Overview


 Categories

 Suites

 Graphs

 Timeline

 Behaviors

 Packages

Categories



order 

name 

duration 

status 

Status:

2

1

0

0

0

 Marks:  

Product defects

2

> Element not found {[data-test-id=code] inputx} Expected: visible

1

Screenshot: file:/aqa-code/bdd/page-object/build/reports/tests/1573943853907.0.png Page source: file:/aqa-code/bdd/page-object/build/reports/tests/1573943853907.0.html Timeout: 4 s. Caused by: NoSuchElementException: no such element: Unable to locate element: {"method":"css selector","selector":"[data-test-id=code] inputx"}

> Element not found {[data-test-id=code] inputx} Expected: visible

1

Screenshot: file:/aqa-code/bdd/page-object/build/reports/tests/1573943858879.1.png Page source: file:/aqa-code/bdd/page-object/build/reports/tests/1573943858879.1.html Timeout: 4 s. Caused by: NoSuchElementException: no such element: Unable to locate element: {"method":"css selector","selector":"[data-test-id=code] inputx"}

Test defects

1

> Failed to create new instance of class ru.netology.web.page.VerificationPage

1

ИНТЕГРАЦИЯ С SELENIDE

```
1 dependencies {  
2     ...  
3  
4     testImplementation 'io.qameta.allure:allure-selenide:2.13.0'  
5 }
```

В тестах:


```
1 @BeforeAll  
2 static void setUpAll() {  
3     SelenideLogger.addListener("allure", new AllureSelenide());  
4 }  
5  
6 @AfterAll  
7 static void tearDownAll() {  
8     SelenideLogger.removeListener("allure");  
9 }
```

ИНТЕГРАЦИЯ С SELENIDE

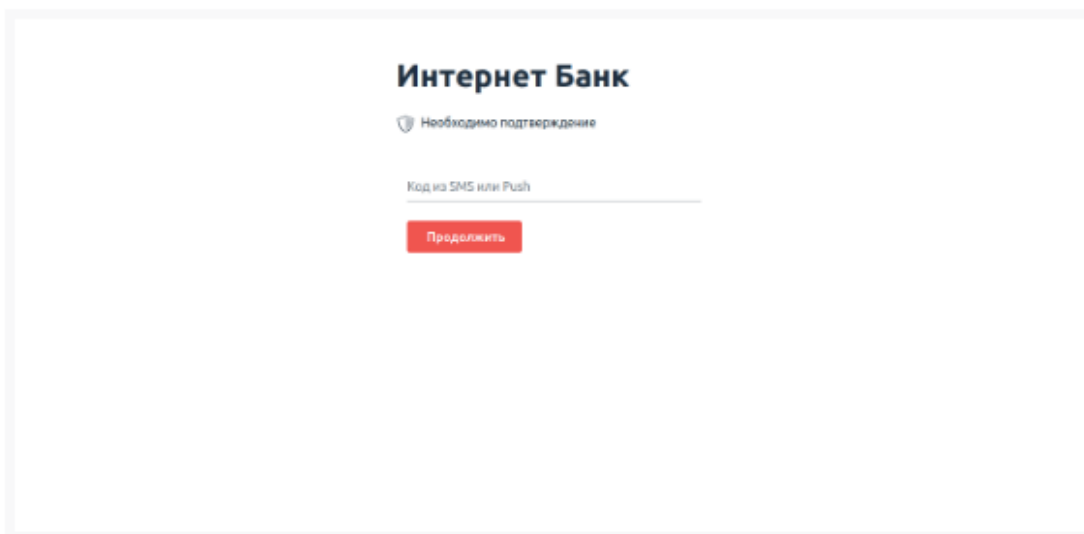
Execution

✓ Test body

- ✓ \$(open) http://localhost:9999 3s 508ms
- ✓ \$([data-test-id=login] input) set value(vasya) 237ms
- ✓ \$([data-test-id=password] input) set value(qwerty123) 170ms
- ✓ \$([data-test-id=action-login]) click() 95ms
- ✓ \$([data-test-id=code] inputx) should be(visible) 2 attachments 4s 374ms

✓  Screenshot

 20.2 KB 



>  Page source

 4.3 KB 



РАСШИРЕННАЯ ИНТЕГРАЦИЯ

Allure позволяет гораздо больше, чем "просто логгировать" - на самом деле, вы можете отмечать нужные события прямо в тестах, прикладывать скриншоты, файлы и помечать тесты.

Обратитесь к документации за более подробной информацией.



REPORT PORTAL



REPORT PORTAL

[Report Portal](#) - сервис, предоставляющий возможности по аналитике и репортигу.

Поскольку Report Portal требует значительных ресурсов для своего запуска, мы ознакомимся с [демо-версией](#), представленной на сайте.



REPORT PORTAL

Разработчики уже подготовили файл [Docker Compose](#).

При желании (и наличии технической возможности) вы можете развернуть Report Portal сами.

Полную инструкцию можно найти на сайте [Report Portal](#).

REPORT PORTAL

All

Import

Actions

Refresh

NAME	START TIME	TOTAL	PASSED	FAILED	SKIPPED	PRODUCT BUG	AUTO BUG	SYSTEM ISSUE	TO INVESTIGATE
<div><div></div><div>Demo Api Tests_Demo #10</div><div><div>9s</div><div>default</div><div>desktop demo build:3.0.1.10</div></div><div><div>Demonstration launch.</div><div>A typical Launch structure comprises the following elements: Suite > Test > Step > Log. Launch contains randomly generated suites, tests, steps with:</div><div>.....</div></div></div> <td>3 hours ago</td> <td>138</td> <td>80</td> <td>32</td> <td>26</td> <td><div>12</div></td> <td><div>11</div></td> <td><div>8</div></td> <td><div>27</div></td>	3 hours ago	138	80	32	26	<div>12</div>	<div>11</div>	<div>8</div>	<div>27</div>
<div><div></div><div>Demo Api Tests_Demo #9</div><div><div>10s</div><div>default</div><div>desktop demo build:3.0.1.9</div></div><div><div>Demonstration launch.</div><div>A typical Launch structure comprises the following elements: Suite > Test > Step > Log. Launch contains randomly generated suites, tests, steps with:</div><div>.....</div></div></div> <td>3 hours ago</td> <td>137</td> <td>79</td> <td>37</td> <td>21</td> <td><div>15</div></td> <td><div>13</div></td> <td><div>4</div></td> <td><div>39</div></td>	3 hours ago	137	79	37	21	<div>15</div>	<div>13</div>	<div>4</div>	<div>39</div>

REPORT PORTAL

[All](#) > [Demo Api Tests_Demo #10](#) > [Launch Tests \(Launch Tests > LaunchStatusTest \)](#) > [testMixedItemsStatusAfterDeletingStep](#)
< > Refresh

#1 #2 #3 #4 #5 #6 #7 #8 #9 LAUNCH #10

Product Bug
Copy defect from #8
Post Issue
Link Issue

[STACK TRACE](#) | [ATTACHMENTS](#) | [ITEM DETAILS](#) | [PARAMETERS](#) | [HISTORY OF ACTIONS](#)

LOG LEVEL [Fatal](#) [Error](#) [Warn](#) [Info](#) [Debug](#) [Trace](#) ☐ Logs with attachments CONSOLE VIEW Next error > < 1 of 1 >

LOG MESSAGE	TIME
09:55:37.264 [main] INFO com.epam.ta.utils.logger.Logger - [STEP] Open 'Launches' page.	2019-11-17 00:25:41
10:21:12.642 [main] INFO com.epam.ta.utils.logger.Logger - Click 'Submit' button.	2019-11-17 00:25:41
09:53:16.412 [main] INFO com.epam.ta.utils.logger.Logger - Set user email.	2019-11-17 00:25:41
10:21:09.918 [main] INFO com.epam.ta.utils.logger.Logger - Select 'Bug Tracking System' from list: 'JIRA'	2019-11-17 00:25:41
10:14:36.146 [main] INFO com.epam.ta.utils.logger.Logger - [STEP] Open 'Project Settingstest_project_8fguvwbxk' page	2019-11-17 00:25:41
09:55:21.743 [main] INFO com.epam.ta.utils.logger.Logger - [STEP] Click History button.	2019-11-17 00:25:41
09:55:18.609 [main] INFO com.epam.ta.utils.logger.Logger - [STEP] Open 'Launches' page.	2019-11-17 00:25:41
10:21:00.335 [main] INFO com.epam.ta.utils.logger.Logger - [STEP] Add new email rule EmailRule{launches=[], recipients=[], tags=[tag1, tag2], sendCase=null}	2019-11-17 00:25:41



ИТОГИ



ИТОГИ

Сегодня мы рассмотрели системы репоринга - от самых простых, не требующих особой настройки, до продвинутых, требующих целых серверов и использующих алгоритмы Machine Learning.

Ключевое - системы репоринга не должны быть самоцелью, они должны решать существующие проблемы.

ДОМАШНЕЕ ЗАДАНИЕ

Давайте посмотрим ваше [домашнее задание](#).

- Вопросы по домашней работе задаем в чате Slack!
- Задачи можно сдавать по частям.
- Зачет по домашней работе проставляется после того, как приняты **все задачи**.

Важно: если ваш компьютер не поддерживает виртуализацию или Docker, вам нужно будет попробовать выполнить ДЗ через Play With Docker.



Задавайте вопросы и напишите отзыв о лекции!

АРТЕМ РОМАНОВ

 [romanov-artyom](#)

 [@Artem_Telegram](#)

 [Романов Артём](#)