



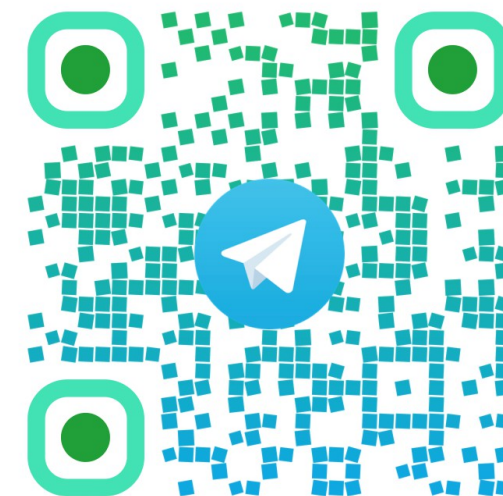
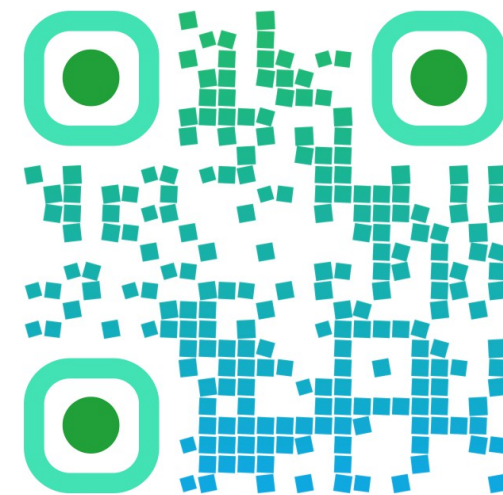
# **Инструменты** автоматизации тестовой модели



# О спикере

Червяков Сергей Геннадьевич

Сбер  
инструменты и методологии  
тестирования СББОЛ  
и целевых сервисов



# О чем это выступление

- Рассмотрение инструментария для автотестирования на проекте. Сервисы, интеграции, данные.
- Какие у нас есть типы тестов и какие метрики по тестированию рассчитываются.
- Test-as-code – как удалось добиться того, что тестовую модель можно вести в коде
- Как показать результаты тестирования руководству так, чтобы не было стыдно

# Что привело к пересмотру тестовой модели



## Микросервисы

Начался процесс миграции с монолитной архитектуры к микросервисной



## Деформация ТМ

В старом монолите в результате развития тестов сформировалась неверная тестовая модель – очень много E2E сценариев по сравнению с низкоуровневыми тестами

# Что получилось сделать при пересмотре ТМ

Выровнялась пирамида тестирования — стала с головы на ноги, E2E сценариев стало мало, а низкоуровневых тестов — много

Тесты стали быстрее, стабильнее и проще

Тестовые проверки стали более атомарными

Поиск причин падений тестов стал отнимать гораздо меньше ресурсов

# Но...

Запрос на визуализацию данных по тестированию никуда не делся.

Единого стандарта тестирования нет

Теперь тесты разрабатывают разные команды

Решение о написании и запусках тестов принимаются на командном уровне, а не централизованно

HOW STANDARDS PROLIFERATE:  
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)

SITUATION:  
THERE ARE  
14 COMPETING  
STANDARDS.

14?! RIDICULOUS!  
WE NEED TO DEVELOP  
ONE UNIVERSAL STANDARD  
THAT COVERS EVERYONE'S  
USE CASES.



SOON:

SITUATION:  
THERE ARE  
15 COMPETING  
STANDARDS.

# Отчетность?



## Дашборд

Самописный сервис с информацией по тестированию легаси монолита



## Allure OPS

Как способ хранения сборок, ведения истории запусков



## Jenkins

Как основной инструмент выполнения сборок с ПРов и релизных сборок



## Проекты

Большое количество проектов, когда-то форкнутых от одного и того же шаблона.

Стек: Spring Boot, Liberica JDK 15, Gradle, Liquibase



# Отчетность!

Бизнес хочет иметь возможность зайти на дашборд и сразу получить все интересующую информацию, чтобы и срезы данных можно было строить, а если это еще и презентабельно...

⚠ Если вашего сервиса нет в списке или информация по нему неактуальна - проверьте настройки в build.gradle.kts, смотрите документацию

Команда \ лидер \ проект \ ФО \ тестовый фреймворк

Информация о проекте	Процент покрытия					
	JiCov	Back	Front	API	CDC	Unit
<b>ufs</b> Проект 329 ФО: ФО 1 ЛГПТ: ФИО 0 Команда 4 ufs v3.3.4 not found	100%	16,4%	96,3%	100%	---	---
<b>sbbol</b> Проект 213 ФО: ФО -2 ЛГПТ: ФИО 3 Команда -8 sbbol v3.3.4 not found	100%	23,1%	---	100%	---	6
<b>comp</b> Проект 304 ФО: ФО 9 ЛГПТ: ФИО 0 Команда 2 comp v3.3.4 not found	100%	73,1%	---	100%	---	12
<b>sbbol</b> Проект 215 ФО: ФО 3 ЛГПТ: ФИО -6 Команда 4 sbbol v3.3.4 not found	100%	53,5%	---	100%	---	8
<b>sbbol</b> Проект 95 ФО: ФО 0 ЛГПТ: ФИО 3 Команда 0 sbbol v3.3.4 not found	100%	85,9%	---	100%	---	13

## Проект 329

ufs (31.03.2023, 12:46)

ufs v3.3.4 not found

Уровень автоматизации проекта (JiCov): 100.00%

Лидер проекта: ФИО 0

Владелец проекта: ФИО 0

Команды: Команда 4

Фреймворки для автотестов: Playwright React Test Library JUnit 5

Axe

Список сборок

BACK prCheck

FRONT prCheck

BACK release

Список Allure отчетов

BACK (release) 31.03.2023, 9:45

BACK (prCheck) 30.03.2023, 7:14

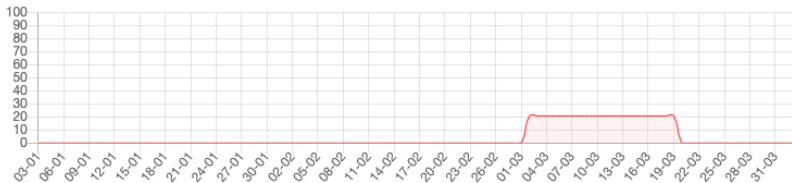
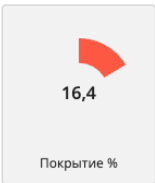
FRONT (prCheck) 24.03.2023, 6:29

### Процент покрытия кода тестами

#### Покрытие по Backend тестам

BACKEND | График обновлен: 31.03.2023, 12:46

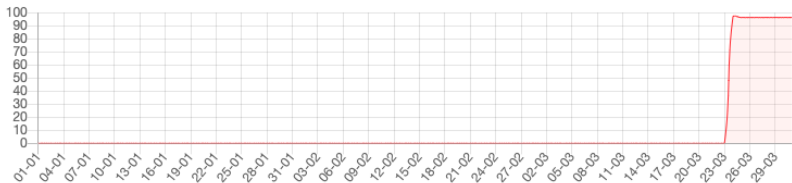
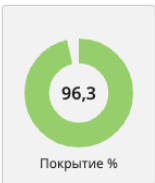
Открыть отчет



#### Покрытие по UI тестам

FRONTEND | График обновлен: 24.03.2023, 9:31

Открыть отчет



### Количество backend тест

#### Количество API тестов

API | График обновлен: 31.03.2023, 12:



#### Количество Unit тестов

BACKEND | График обновлен: 31.03.20





# Какие типы тестов и виды отчетов мы предложили командам «из коробки»?

- Покрытие по строчкам java кода, рассчитываемое с помощью инструмента JaCoCo.
- Покрытие по количеству эндпоинтов, рассчитываемое с помощью библиотеки Swagger Coverage
- Количество тестов с помощью JUnit 5
- Ручные тесты
- Покрытие по строчкам js/ts кода с помощью библиотеки Icov
- Количество js/ts тестов из тестового раннера

# Какие новые фишки удалось добавить

- Поддержка концепции Test-as-code
- Синхронизация тестовых кейсов в коде с TMS
- Бесшовное (почти) встраивание плагина для тестирования со всеми инструментами в командные проекты
- Соккрытие от пользователей большей части настроек для выполнения тестов и выгрузки результатов

С какими проблемами  
столкнулись

# С какими проблемами столкнулись?

JaCoCo

Плагин JaCoCo входящий в поставку самого Gradle не работает с JDK 15 версии как ожидалось.

Возникающая при работе агента ошибка была плавающей и ни разу не информативной.

Проблему удалось исправить повышением версии агента.

```
plugins { this: PluginDependenciesSpecScope
    jacoco
}

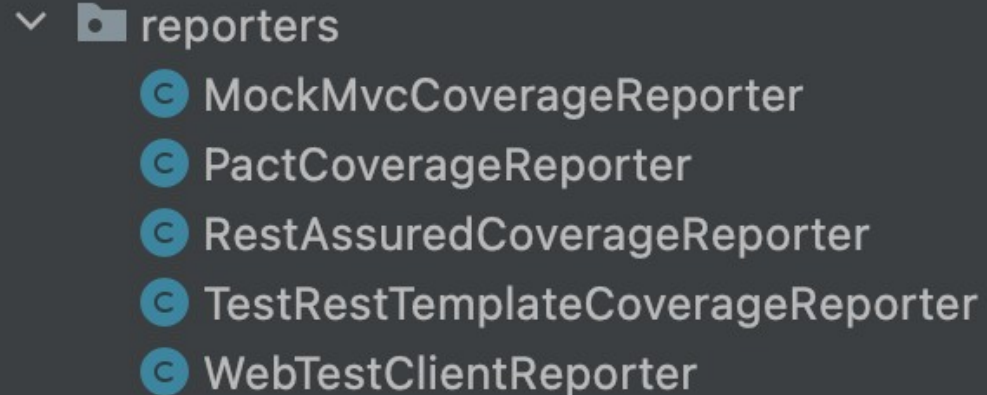
jacoco { this: JacocoPluginExtension
    toolVersion = "0.8.7"
}
```

# С какими проблемами столкнулись?

## Swagger Coverage

Подсчитать количество эндпоинтов в приложении не так сложно. А вот как быть с перечетом эндпоинтов проверенных в тестах?

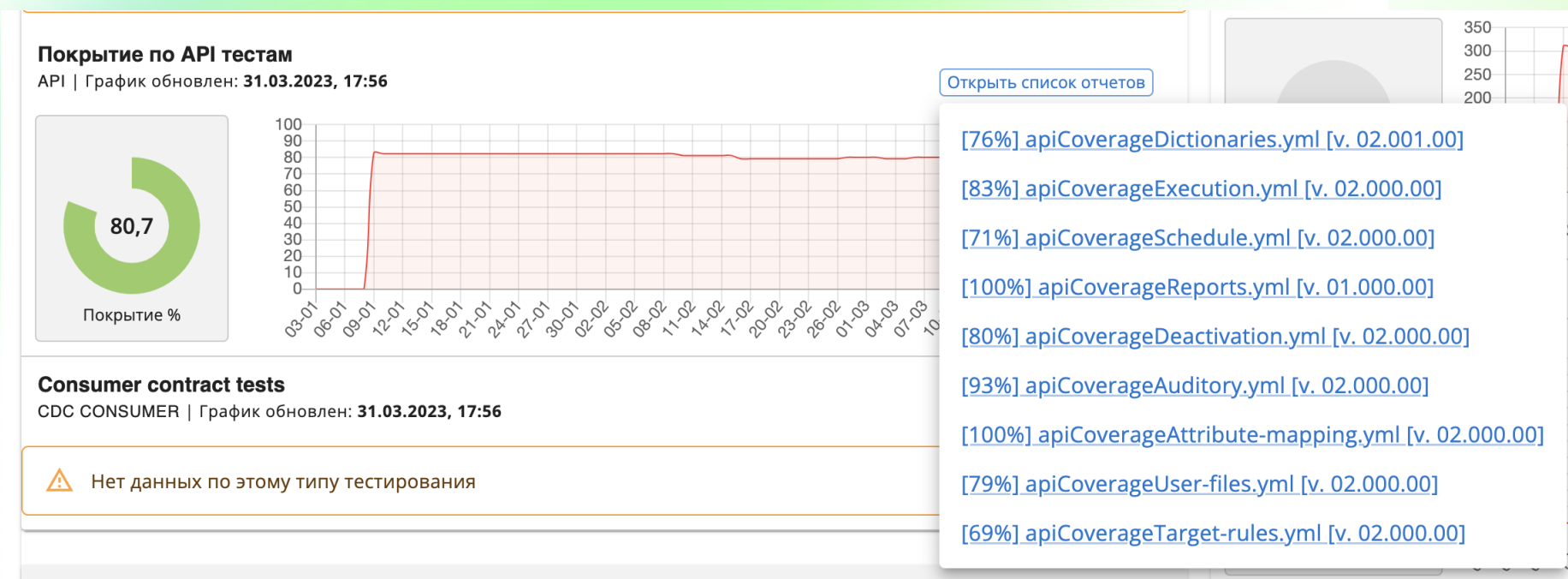
Так как наши команды используют разные инструменты для вызова API в тестах, а Swagger Coverage поддерживает только RestAssured, пришлось допиливать инструмент для работы с другими тестовыми клиентами.



```
▼ [icon] reporters
  [icon] MockMvcCoverageReporter
  [icon] PactCoverageReporter
  [icon] RestAssuredCoverageReporter
  [icon] TestRestTemplateCoverageReporter
  [icon] WebTestClientReporter
```

# Но...

Оказалось, что спецификация API не всегда есть и может быть описана в нескольких местах. Из-за этого из Swagger Coverage формировался не один общий отчет, а несколько частных:

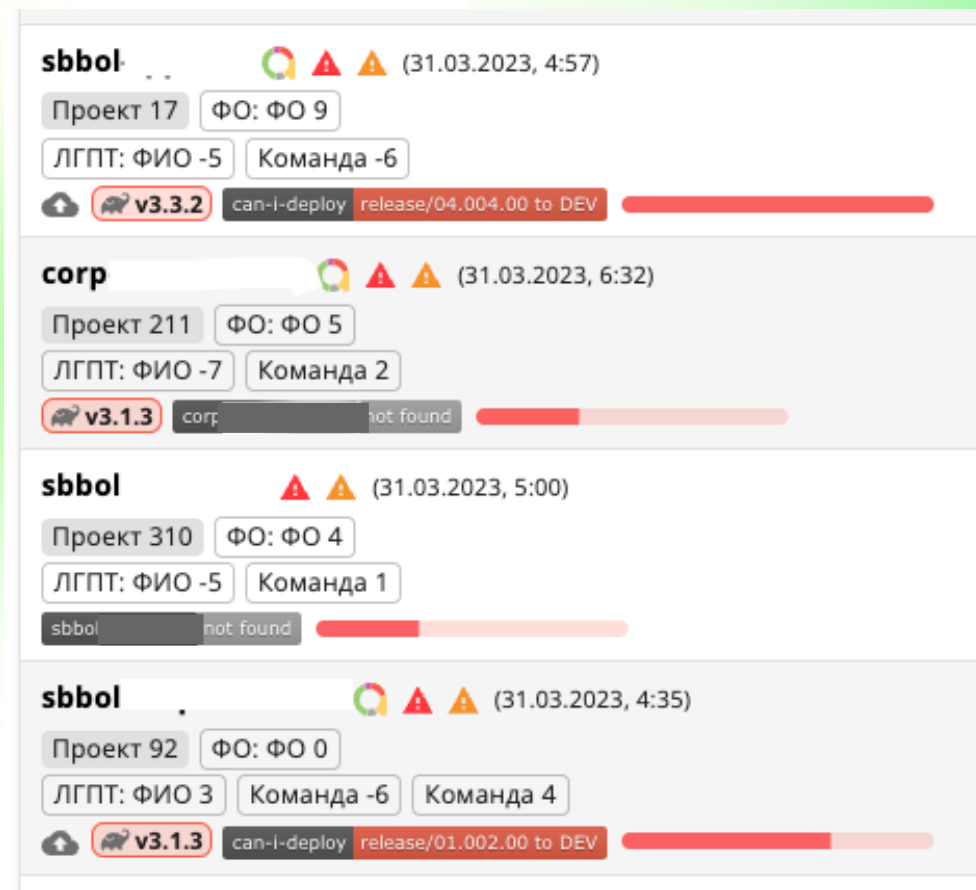


# С какими проблемами столкнулись?

Pact + Pact Broker

С этим инструментом всё хорошо. Контракты формируются в тестах потребителей, отправляются в брокер, провайдеры в своих тестах контракты подтверждают и публикуют статус контракта в брокер.

А на дашборде мы отобразили баджики по проекту, которые отрисовывает сам брокер:





# С какими проблемами столкнулись?

Подсчет количества тестов

Нам требуется подсчитать количество запущенных тестов по типам. Все необходимые данные уже есть в JUnit раннере и их нужно только научиться извлекать. Мы написали расширение, которое логирует данные по тесту в текстовый файл, который затем отправляется на разбор в дашборд

# С какими проблемами столкнулись?

Подсчет количества тестов

Фреймворк Allure подключен к каждому проекту и поддерживает работу с т. н. лейблами. С помощью аннотации

`@io.qameta.allure.LabelAnnotation`

можно создавать и заполнять кастомные лейблы в аллюр отчетах.

Мы завели отдельную лейбл-аннотацию, в которой хранится информация по типу запущенного теста и дополнительно подключаются наши расширения

JUnit с помощью аннотации

`@org.junit.jupiter.api.extension.ExtendWith`

# С какими проблемами столкнулись?

Подсчет количества тестов

```
@MustBeDocumented
@Inherited
@Retention(RUNTIME)
@Target(FUNCTION, PROPERTY_GETTER)
@LabelAnnotation(name = "layer")
@ExtendWith(TestLayerFilterExtensions)
annotation class TestLayer(
    val value: String
) {
```

```
@MustBeDocumented
@Inherited
@Retention(RUNTIME)
@Target(FUNCTION, PROPERTY_GETTER)
@TestLayer("api")
annotation class ApiTestLayer
```

```
@Documented
@Inherited
@Retention(RetentionPolicy.RUNTIME)
@Target({ElementType.METHOD, ElementType.CONSTRUCTOR})
@Repeatable(Epics.class)
@LabelAnnotation(name = EPIC_LABEL_NAME)
public @interface Epic {
```

# С какими проблемами столкнулись?

Подсчет количества тестов

После разметки и настройки маппинга в Allure OPS размеченные тесты стало возможно фильтровать по размеченному меткой layer значению:

The screenshot displays the Allure OPS web interface. The top navigation bar includes tabs for Overview, Tree (which is selected), Errors, Graphs, and Timeline. On the left sidebar, under the 'Layer' section, the 'API Tests' checkbox is checked, while others like 'Manual', 'Automated', 'Android Tests', etc., are unchecked. The main area, titled 'Test results', shows a table of test outcomes. Each row includes a checkbox, a green checkmark icon, a unique ID, a description of the test in Russian, the layer name 'API Tests' in orange, and the execution duration in milliseconds.

<input type="checkbox"/>	Id	Name	Status	Duration
<input type="checkbox"/>	#27210	Получение пользовательских настроек виджетов	API Tests	65ms
<input type="checkbox"/>	#59900	Получение пользовательских настроек виджетов, при не доступной фабрике	API Tests	499ms
<input type="checkbox"/>	#20712	Проверка ssd	API Tests	49ms
<input type="checkbox"/>	#50275	Проверка sup	API Tests	31ms
<input type="checkbox"/>	#50274	Проверка авторизации, рест доступен для пользователя б ез ограничения по ролям	API Tests	30ms

# Но...

Подсчет количества тестов

Теперь просто нужно, чтобы пользователи поставили аннотации над всеми тестами

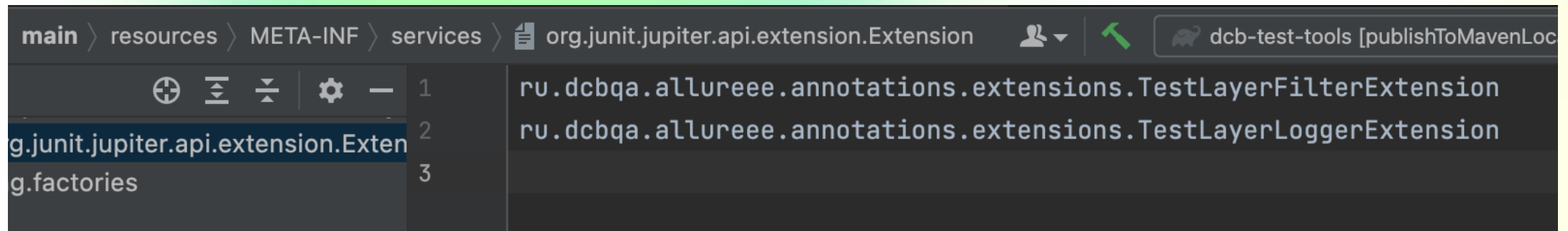




# С какими проблемами столкнулись?

Подсчет количества тестов

Очевидно, что как только пользователи перестанут расставлять аннотации с расширениями — вся статистика развалится. Хорошо, что в JUnit предусмотрели возможность подключать расширения сразу на все тесты:  
- `Djunit.jupiter.extensions.autodetection.enabled=true`  
и дальше `ServiceLoader` сделает всю работу:



The screenshot shows an IDE window with the following details:

- Path:** `main > resources > META-INF > services > org.junit.jupiter.api.extension.Extension`
- File Name:** `org.junit.jupiter.api.extension.Extension`
- Content:**

```
1 ru.dcbqa.allureee.annotations.extensions.TestLayerFilterExtension
2 ru.dcbqa.allureee.annotations.extensions.TestLayerLoggerExtension
3
```
- Left Panel:** Shows a tree view with `g.junit.jupiter.api.extension.Extension` selected and `g.factories` below it.
- Toolbar:** Includes icons for adding, removing, and toggling elements, as well as a settings gear icon.
- Top Right:** A button labeled `dcb-test-tools [publishToMavenLoc`.

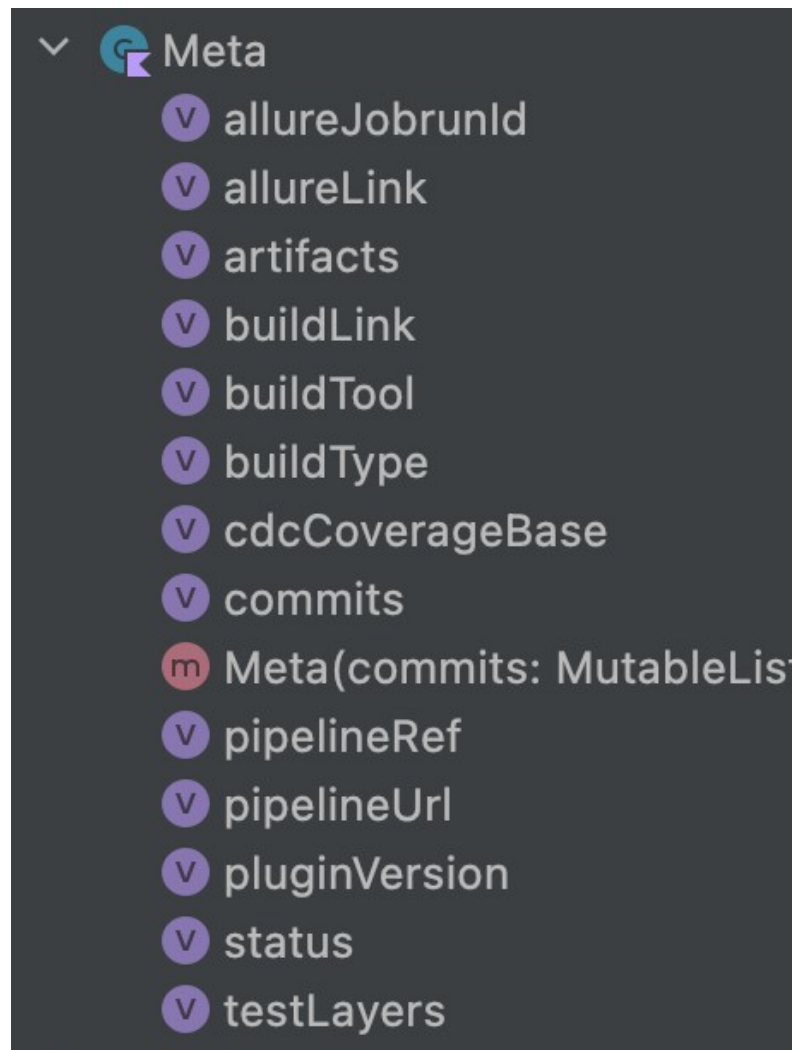
# Все данные собрали. Что дальше?



Первым запросом отправим все метаданные про тестовый запуск, в ответ нам нужен только уникальный идентификатор для разделения загружаемых данных по запускам



Последующими запросами отправим отчеты, к каждому из которых приложим гайд запуска и тип отчета





# Визуализация метаданных

## < Проект 8

sbbol- (31.03.2023, 12:46)



v4.1.3

version not found

can-i-deploy release/06.005.00 to DEV

can-i-deploy release/06.004.00 to DEV

Уровень автоматизации проекта (JiCov): 100.00%

Лидер проекта: ФИО 0

Владелец проекта: ФИО 0

Команды: Команда 6

Фреймворки для автотестов: Playwright React Test Library

JUnit 5

Axe



### Список сборок

BACK prCheck




FRONT prCheck

BACK release

# Визуализация метаданных

 **Проект 131** 

**sbbol** (31.03.2023, 12:03)

  sbbol- not found sbbol- not found sbbol- not found sbbol- not found sbbol- not found sbbol- not found 

Уровень автоматизации проекта (JiCov): 100.00%



Лидер проекта: ФИО -5

Владелец проекта: ФИО -5

Команды: Команда 4

Фреймворки для автотестов: JUnit 5

**Список сборок**  
[BACK prCheck](#)  
[BACK release](#)

**Список Allure отчетов**  
 **BACK** (release) 31.03.2023, 8:41  
 **BACK** (prCheck)

# А что по test-as-code?

Концепция хранения тестовых сценариев в коде понравилась нашей команде тестовой модели, потому мы решили воспользоваться функционалом Allure OPS интегрироваться с другими TMS.

# Но...



Не позволяет раскладывать тестовые результаты по директориям



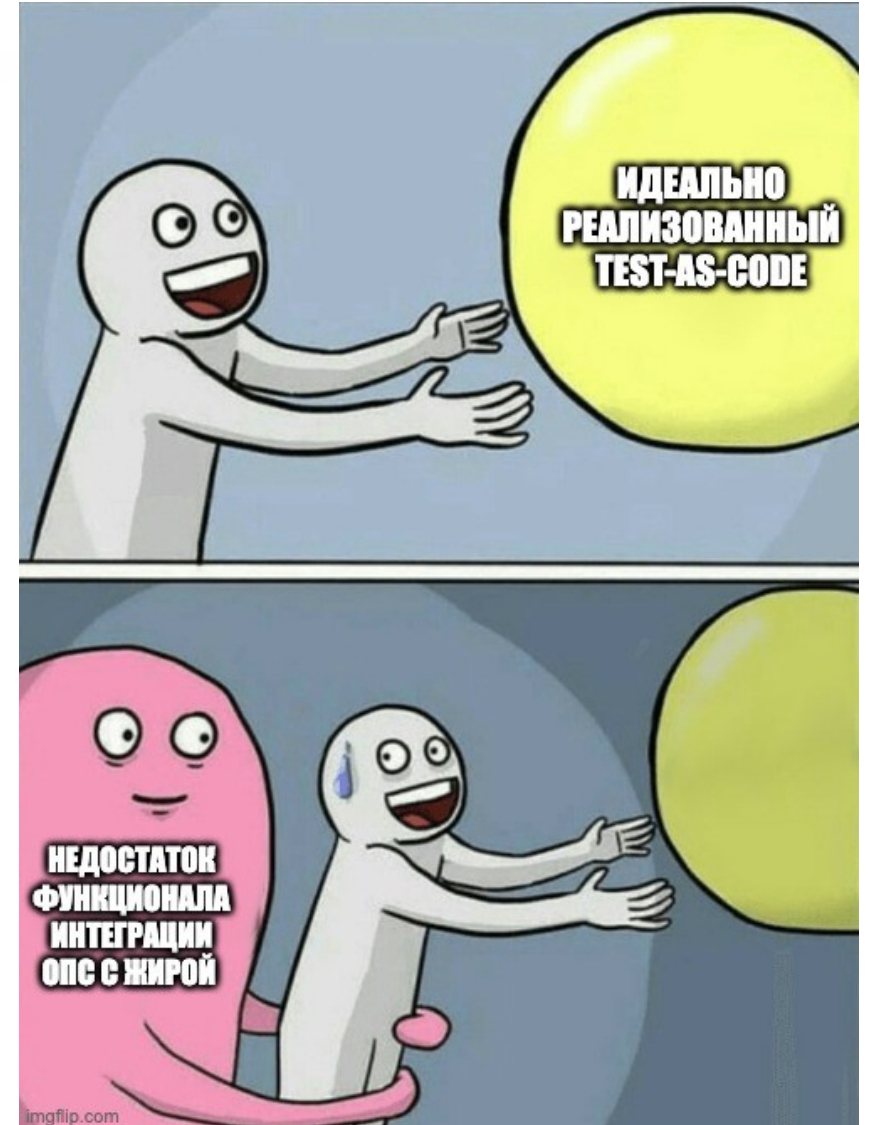
Некорректно работает с кастомными полями в нашей TMS, а также нет валидации отправляемых значений



Нет возможности раскладывать тесты из одного запуска (репозитория) по разным пространствам в TMS

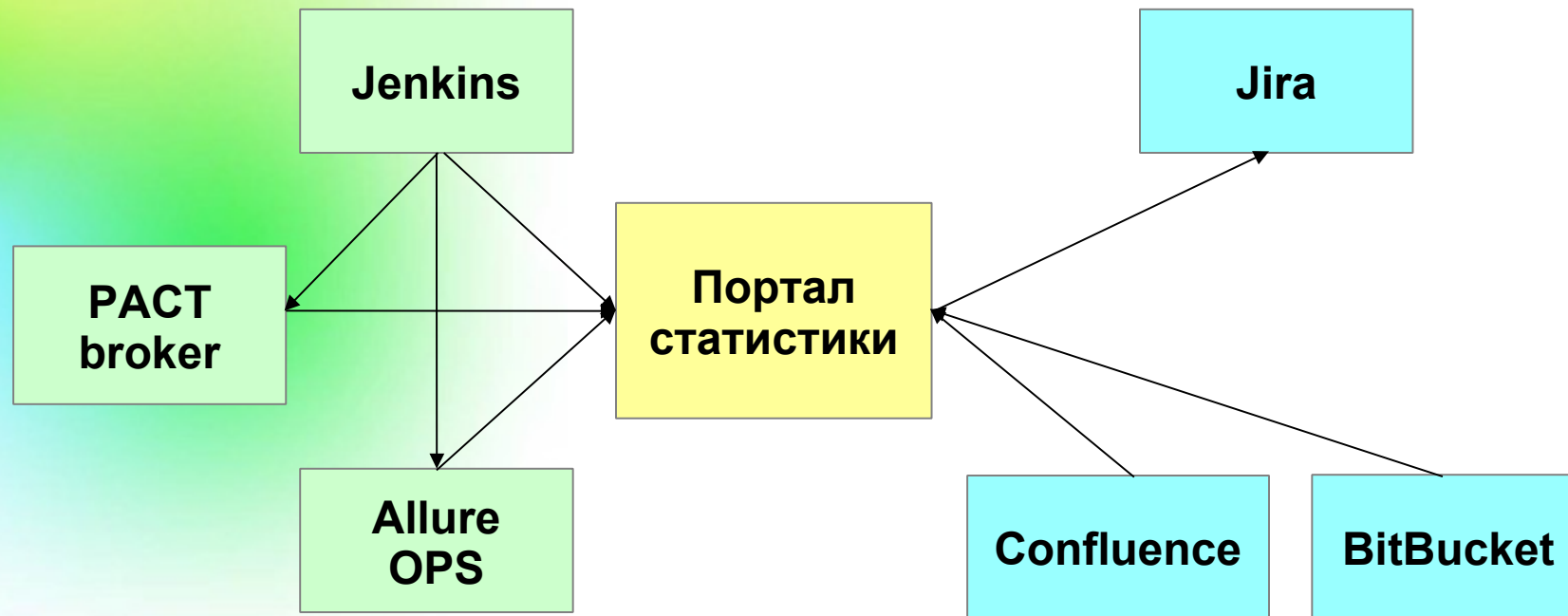


Некоторые дополнительные пожелания команды Тестовой Модели.



# Однако...

Описанные проблемы удалось решить в нашем приложении, реализовав смежные интеграции следующим образом:



# @ManualTestLayer

```
@ManualTestLayer
public class ToBeAddedTest {
    new *
    @Test
    void toBeImplemented() {
        Allure.step( name: "Предусловие для проверки");
        Allure.step( name: "Успешно справляемся с задачей");
    }
}
```

## Test results

<input type="checkbox"/>	Id	Name	Status	Duration
<input type="checkbox"/>	#29491	toBeImplemented()	Manual Tests	218ms

ru.dcbqa.portal.ToBeAddedTest.toBeImplemented

**SKIPPED** #29491 toBeImplemented()

Overview History Retries Attachments

Message

Manual test

## Similar failures

No content

## Description

No description

## Execution

- > 1 Предусловие для проверки
- > 2 Успешно справляемся с задачей



## toBeImplemented()

Details **Test Script** Execution Traceability Attachments Comments

Type: Step-by-Step

## Steps

STEP  
1 Предусловие для проверки

STEP  
2 Успешно справляемся с задачей

META-INF > services > io.qameta.allure.listener.TestLifecycleListener  
1 ru.dcbqa.allureee.annotations.allure.ManualTestLifecycleListener

Project	Team	AC
Проект 272	Команда 2	ЕФС ЮЛ СЕ
Проект 337	Команда 2	ЕФС ЮЛ СЕ
Проект 124	Команда 0	ППРБ ЮЛ СЕ
Проект 195	Команда 0	ППРБ Digit
Проект 198	Команда 0	ППРБ ЮЛ СЕ
Проект 252	Команда 0	ППРБ ЮЛ СЕ
Проект 287	Команда 0	ППРБ ЮЛ СЕ
Проект 256	Команда 4	ППРБ ЮЛ СЕ
Проект 17	Команда -6	ЕФС.ЮЛ СЕ
Проект 93	Команда 6	ЕФС.ЮЛ.СЕ
Проект 152 sbbol-	Команда 6	ЕФС Smth CI01234567
Проект 182	Команда 6	ЕФС ЮЛ СЕ
Проект 68	Команда -5	ЕФС.ЮЛ.СЕ

[Изменить настройку выгрузки](#)

Вид теста по умолчанию

Негативный

Да

Негативный ли тест по умолчанию

Automation status

Автоматизирован API

Статусы автоматизации тестов по умолчанию

Platform \*

UFS API

Нужно указать тестируемые платформы

Test type \*

Full regress

Тип теста по умолчанию

Integration systems \*

БП РПП

Обязательное поле

АС \*

ЕФС Smth [CI012345678]

Обязательное поле

Настройки в виде JSON

Subfolder	Выгрузка
	false
	true
	true
	true
	true
	true
	true
	true
	true
	true
my-folder-in-tms	true
	true
	true



**Несколько примеров**  
того, что получилось

⚠ Если вашего сервиса нет в списке или информация по нему неактуальна - проверьте настройки в [build.gradle.kts](#), смотрите [документацию](#) по расчету покрытия.

Команда \ лидер \ проект \ ФО \ тестовый фреймворк



Группировка



















Информация о проекте	Процент покрытия					Backend тесты				Frontend тесты			Прочее	
	JiCov	Back	Front	API	CDC	Unit	Api ↓	Consumer	Provider	Unit	Web	Axe	E2E	Undefined
<b>sbbol-</b> (31.03.2023, 12:03) Проект 131    ФО: ФО -9 ЛГПТ: ФИО -5    Команда 4 <b>v3.1.3</b> sbbol- not found <div><div></div></div>	100%	35,5%	---	---	---	271	548	---	---	---	---	---	1047	356
<b>digital</b> (31.03.2023, 9:37) Проект 162    ФО: ФО 0 ЛГПТ: ФИО 7    Команда 8 <b>v3.3.4</b> <div><div></div></div>	100%	54,3%	---	3,2%	---	801	514	---	---	---	---	---	732	0
<b>sbbol-</b> (31.03.2023, 11:57) Проект 151    ФО: ФО 4 ЛГПТ: ФИО -9    Команда 7 <b>v3.3.1</b> sbbol- not found <div><div></div></div>	100%	74,6%	---	80,7%	---	303	427	---	---	---	---	---	615	0
<b>sbbol-</b> (31.03.2023, 6:47) Проект 134    ФО: ФО 8 ЛГПТ: ФИО 0    Команда -1 <b>v4.1.3</b> <div><div></div></div>	100%	84,7%	---	90,3%	---	125	315	---	---	---	---	---	282	0
<b>event</b> (31.03.2023, 10:23) Проект 129    ФО: ФО 8 ЛГПТ: ФИО -5    Команда 7 <b>v3.2.14</b> version not found <div><div></div></div>	100%	87,3%	---	---	---	20	219	---	---	---	---	---	311	0
<b>sbbol-</b> (31.03.2023, 4:52)														

Rows per page: 200

1-200 of 236



restassured										
Информация о проекте	Процент покрытия					Backend тесты				
	JiCov ?	Back	Front	API	CDC	Unit	Api	Consumer ?	Provider ?	Un
<b>sbbol-</b>  (31.03.2023, 6:40) Проект 104 <span>ФО: ФО -8</span> ЛГПТ: ФИО -5 <span>Команда 3</span>   v4.1.2 <span>sbbol-</span> <span>not found</span> <div><div></div></div>	<div>100%</div>	<div>68,7%</div>	---	<div>91,9%</div>	<div>100%</div>	<div>260</div>	<div>59</div>	<div>1</div>	---	---
<b>sbbol-</b>  (31.03.2023, 4:19) <a href="#">Проект 200</a>  v4.5.1 <span>sbbol</span> <span>not found</span> <div><div></div></div>	<div>100%</div>	<div>70,5%</div>	<div>100%</div>	<div>86,7%</div>	---	<div>46</div>	<div>3</div>	---	---	---
<b>sbbol-</b>  (31.03.2023, 5:47) Проект 141 <span>ФО: ФО -8</span> ЛГПТ: ФИО -5 <span>Команда 7</span>   v4.1.2 <div><div></div></div>	<div>100%</div>	<div>88,5%</div>	---	<div>94,1%</div>	---	<div>14</div>	<div>13</div>	---	---	---
<b>upg-</b>  (31.03.2023, 6:03) Проект 330 <span>ФО: ФО -3</span> ЛГПТ: ФИО -9 <span>Команда 0</span>  v4.3.0 <span>upg-</span> <span>not found</span> <div><div></div></div>	<div>100%</div>	<div>100%</div>	---	---	---	---	<div>1</div>	---	---	---
<b>sbbol-</b>  (31.03.2023, 5:26) Проект 289 <span>ФО: ФО -7</span> ЛГПТ: ФИО -9 <span>Команда 6</span>   v4.3.0 <span>sbbol</span> <span>not found</span> <div><div></div></div>	<div>100%</div>	<div>65,1%</div>	---	---	---	<div>4</div>	<div>13</div>	---	---	---
<b>sbbol-</b>  (31.03.2023, 12:22) Проект 109 <span>ФО: ФО 0</span> ЛГПТ: ФИО 3 <span>Команда 4</span>   v4.3.0 <span>sbbol</span> <span>not found</span> <div><div></div></div>	<div>100%</div>	<div>89,8%</div>	---	<div>81,3%</div>	---	<div>37</div>	<div>26</div>	---	---	---

Команда \ лидер \ проект \ ФО \ тестовый фреймворк										
Информация о проекте	Процент покрытия					Backend тесты				
	JiCov ?	Back	Front	API	CDC	Unit	Api	Consumer ?	Provider ?	Ur
<b>curcontrol-</b> (31.03.2023, 8:48) <div>Проект 201</div> <div> <div>m v3.1.0</div> <div>curcontrol</div> <div>not found</div> </div>	---	---	---	---	---	2	---	---	---	---

### Критические проблемы

- Нет WEB-тестов для БП/АРМ
- Allure-отчет старше даты обновления (6 дн.)

### Рекомендации

- Покрытие по UNIT-тестам меньше 80%
- Использование устаревшей версии dcb-test-plugin

not found

### БП

Статус эксплуатации (3):

**Промышленная эксплуатация**

Текущий уровень зрелости (3):

**Промышленная эксплуатация**

Покрытие по Backend (ур. 3): **84%**

Покрытие по Api тестам (ур. 3): **75%**

Покрытие по CDC-тестам): **не учитывается**

Уровень развития: **Соответствует**

develop to DEV

### БП

Статус эксплуатации (3):

**Промышленная эксплуатация**

Текущий уровень зрелости (0):

Покрытие по Backend (ур. 2): **58%**

Покрытие по Api тестам (ур. 0): **0%**

Покрытие по CDC-тестам): **не учитывается**

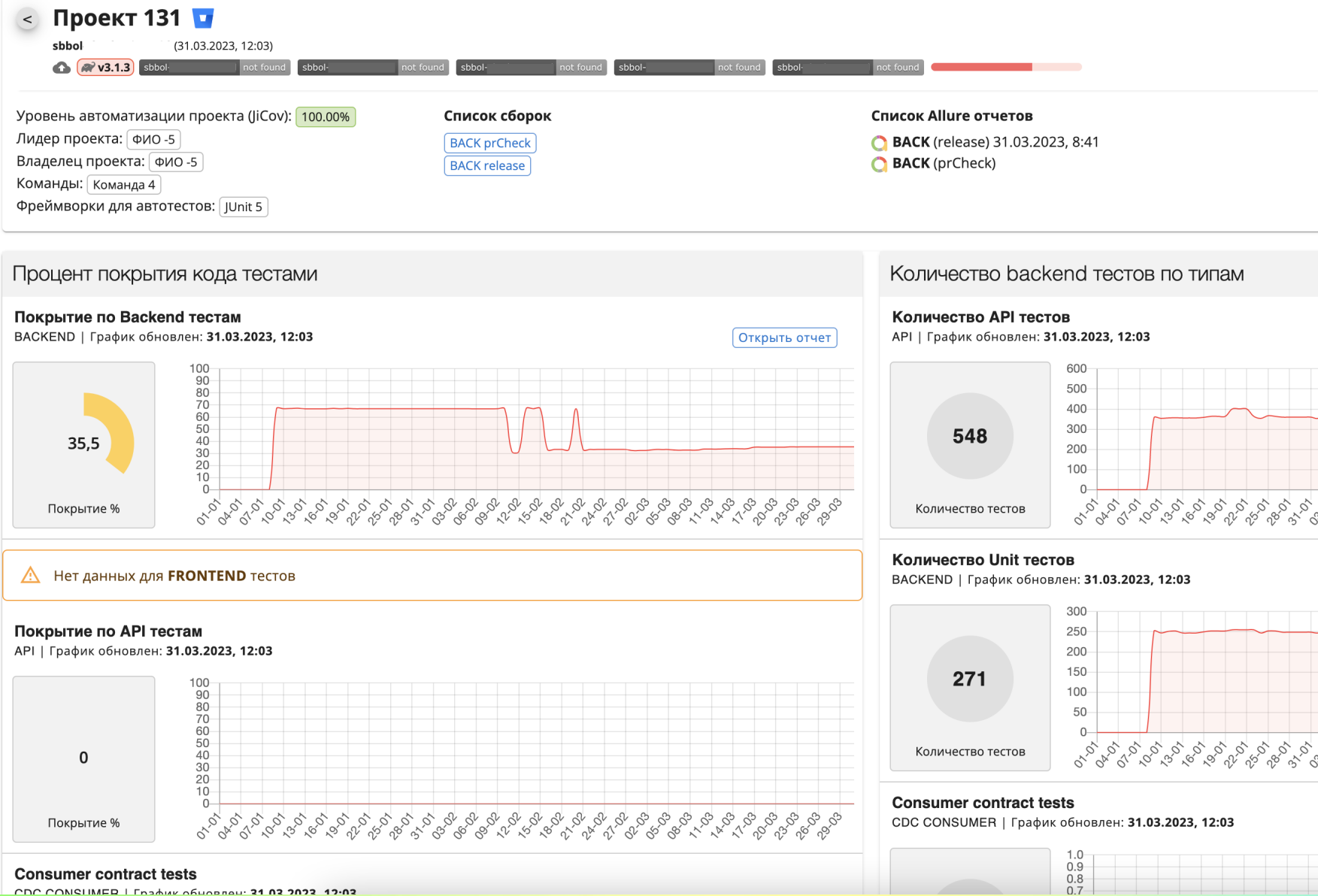
Уровень развития: **Не соответствует**

⚠ Если вашего сервиса нет в списке или информация по нему неактуальна - проверьте настройки в [build.gradle.kts](#), смотрите [документацию](#) по расчету покрытия.

Команда \ лидер \ проект \ ФО \ тестовый фреймворк



Информация о проекте	Процент покрытия					Backend тесты			
	JiCov ?	Back	Front	API	CDC	Unit	Api	Consumer ?	Provider ?
<b>sbbol-</b> (31.03.2023, 12:46) Проект 8    ФО: ФО -9 ЛГПТ: ФИО 0    Команда 6 <b>v4.1.3</b> version not found	100%	84,5%	76,5%	---	---	661	65	---	---
<b>sbbol-</b> (31.03.2023, 5:20) Проект 258    ФО: ФО 6 ЛГПТ: ФИО 3    Команда 0 <b>v3.3.4</b> sbbol not found	100%	72%	96,1%	87,5%	---	9	13	---	---
<b>sbbol-</b> (31.03.2023, 6:39) Проект 136    ФО: ФО 0 ЛГПТ: ФИО -5    Команда 7 	---	---	---	---	---	---	---	---	---
<b>sbbol-</b> (31.03.2023, 6:07) Проект 195    ФО: ФО 0 ЛГПТ: ФИО 0    Команда 0 <b>v3.2.5</b> sbbol- not found	100%	98,9%	---	---	---	---	7	---	---
<b>sbbol-</b> (31.03.2023, 6:38) Проект 63    ФО: ФО 6 ЛГПТ: ФИО -9    Команда 1 <b>v3.3.4</b>	100%	60,2%	81%	---	---	44	11	---	---
<b>sber</b> (31.03.2023, 4:09) Проект 177 	100%	68,7%	---	---	---	32	21	---	---



Группировка	Процент покрытия					Backend тесты			Frontend тесты			Прочее	
Тип сервиса													
<input type="checkbox"/> Среднее количество тестов	Back ↓	Front	API	CDC	Unit	Api	CDC Consumer	CDC Provider	Unit	Web	Axe	E2E	Undefined
<b>Нет данных (null)</b> <a href="#">Список</a> Проектов в группе: 4	39,1%	0%	18,8%	0%	14	21	0	0	0	0	0	704	0
<b>Не определено</b> <a href="#">Список</a> Проектов в группе: 10	37,4%	26,6%	27,4%	0%	75	114	0	0	10	8	6	4853	1
<b>БП</b> <a href="#">Список</a> Проектов в группе: 139	30,1%	18,8%	11,1%	0%	7260	2124	63	0	323	551	35	29054	607
<b>ППРБ</b> <a href="#">Список</a> Проектов в группе: 87	28,9%	0,4%	13,1%	0%	4216	3527	2	0	1	1	0	26345	368
<b>БПС</b> <a href="#">Список</a> Проектов в группе: 9	26,7%	11,1%	21,1%	0%	222	54	1	0	0	0	0	4438	0
<b>СББОЛ</b> <a href="#">Список</a> Проектов в группе: 6	11%	13,6%	11,1%	0%	20	17	0	1	42	0	0	3965	2

Группировка

Тип сервиса ▲

Функциональная область

Лидер группы

Тип сервиса

Владелец продукта

Уровень развития





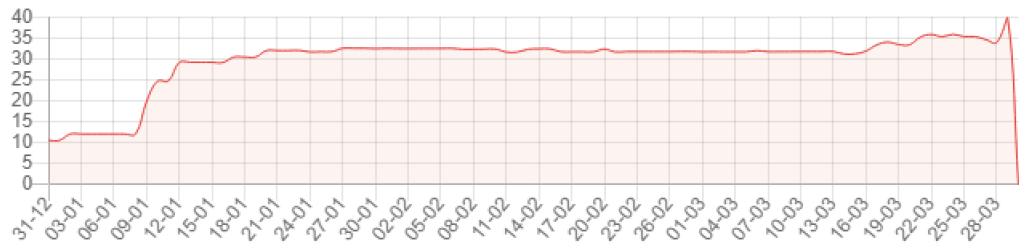
Группировка		Процент покрытия					Backend тесты			Frontend тесты		
Лидер группы		Back ↓	Front	API	CDC	Unit	Api	CDC Consumer	CDC Provider	Unit	Web	Axe
<input type="checkbox"/> Среднее количество тестов												
ФИО -2 <a href="#">Список</a> Проектов в группе: 1		49,8%	0%	66,7%	0%	12	8	0	1	0	0	0
ФИО -3 <a href="#">Список</a> Проектов в группе: 9		43,3%	9,4%	16,8%	0%	277	175	0	0	4	2	0
ФИО -7 <a href="#">Список</a> Проектов в группе: 17		33,3%	25,7%	3,9%	0%	929	306	0	0	10	9	0
ФИО 3 <a href="#">Список</a> Проектов в группе: 37		32,5%	8,1%	16,3%	0%	2137	334	2	0	48	8	4
ФИО 0 <a href="#">Список</a> Проектов в группе: 32		31,8%	14,1%	14,7%	0%	1913	877	17	0	78	296	28
null <a href="#">Список</a> Проектов в группе: 63		27,9%	15%	12,8%	0%	2698	876	2	0	135	234	9
ФИО -6 <a href="#">Список</a> Проектов в группе: 19		27,5%	8,3%	26,5%	0%	234	439	0	0	76	5	0
ФИО -5 <a href="#">Список</a> Проектов в группе: 43		25,7%	10%	7,1%	0%	1625	1451	44	0	23	0	0

Группировка  
Тип сервиса

ППРБ

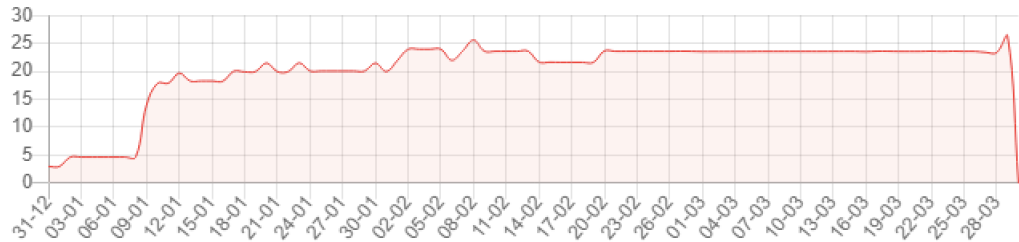
BACKEND

Покрытие по Backend тестам



API

Покрытие по API

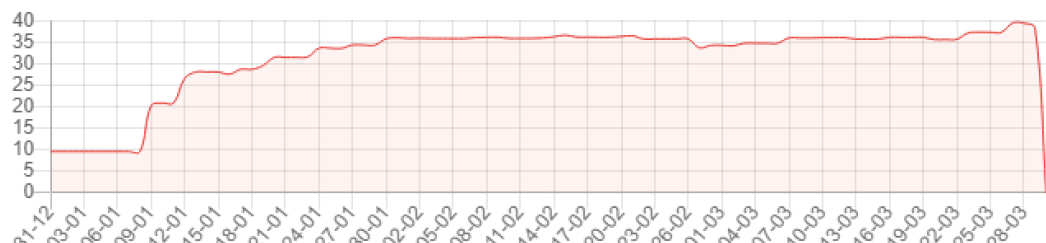


Группировка  
Лидер группы

ФИО 1

BACKEND

Покрытие по Backend тестам



API

Покрытие по API



Проект:



ID	Created at	Project ID	Data key	Build type	Build link	Configuration test	Build tool	Artifacts	CDC coverage base	Test layers
292413	3/29/2023 4:19:01 PM	85	7B33AC38-6BFE-4238-A233-1BA94EC1B284	prCheck	>	true	v 4.5.2	SHOW		SHOW
292409	3/29/2023 4:15:22 PM	130	EFCBA143-94F8-43DE-9782-210B3E02B477	prCheck	>	false		SHOW		SHOW
292406	3/29/2023 4:13:28 PM	19	CF092E46-7399-4E4E-B83E-86724B4E3922	prCheck	>	false		SHOW		SHOW
292402	3/29/2023 4:09:43 PM	184	FCC29195-D9BE-460B-9597-7AE6E01E33AA	prCheck	>	false		SHOW		SHOW
292401	3/29/2023 4:09:19 PM	130	8A0FBFC5-2D17-42C6-A9E1-92468BEC4305	prCheck	>	false		SHOW		SHOW
292397	3/29/2023 4:06:38 PM	19	25996D0E-DBCA-4CAC-82AF-08A030E2C03D	prCheck	>	false		SHOW		SHOW
292396	3/29/2023 4:06:33 PM	66	EBA16DDC-8C80-47C8-813D-E2C36D5C7232	prCheck	>	true	v 3.2.14	SHOW		SHOW
292395	3/29/2023 4:05:15 PM	280	E4343788-B626-4D65-AB78-98CDF3B4A7F0	prCheck	>	true	v 3.1.3	SHOW		SHOW
292394	3/29/2023 4:05:00 PM	130	7642763B-EE88-484D-B0DA-AB913F6CFA1A	prCheck	>	false		SHOW		SHOW
292393	3/29/2023 4:04:27 PM	213	49981A82-8C41-4D6B-8FC3-180A17D465F4	release	>	false	v 3.3.4	SHOW		SHOW

curcontrol-

Element	Missed Instructions	Cov.	Missed Branches	Cov.	M
<a href="#">ru.sbrf.ufs.sbbol</a>	<div><div></div></div>	14%	<div><div></div></div>	0%	
<a href="#">ru.sbrf.ufs.sbbol</a>	<div><div></div></div>	44%	<div><div></div></div>	25%	
<a href="#">ru.sbrf.ufs.sbbol</a>	<div><div></div></div>	29%	<div><div></div></div>		
<a href="#">ru.sbrf.ufs.sbbol</a>	<div><div></div></div>	28%	<div><div></div></div>		
<a href="#">ru.sbrf.ufs.sbbol</a>	<div><div></div></div>	17%	<div><div></div></div>		
<a href="#">ru.sbrf.ufs.sbbol</a>	<div><div></div></div>	76%	<div><div></div></div>		
<a href="#">ru.sbrf.ufs.sbbol</a>	<div><div></div></div>	32%	<div><div></div></div>		
<a href="#">ru.sbrf.ufs.sbbol</a>	<div><div></div></div>	7%	<div><div></div></div>		
<a href="#">ru.sbrf.ufs.sbbol</a>	<div><div></div></div>	38%	<div><div></div></div>		
<a href="#">ru.sbrf.ufs.sbbol</a>	<div><div></div></div>	37%	<div><div></div></div>		

Total: 37

Covered: 94.595% (35/37)

Uncovered: 5.405% (2/37)

All files

56.04% Statements 612/1092 24.02% Branches 122/508 38.15% Functions 66/173 54.29% Lines 468/862

Press *n* or *j* to go to the next uncovered block, *b*, *p* or *k* for the previous block.

File	Statements	B
src	<div><div></div></div> 72.86% 102/140	
src/Common	<div><div></div></div> 38.33% 23/60	
src/Common/ErrorScreen	<div><div></div></div> 81.82% 18/22	
src/Modules/DualDeposit/Components	<div><div></div></div> 62.45% 158/253	
src/Modules/DualDeposit/Components/SliderBlock	<div><div></div></div> 16.55% 24/145	
src/Modules/DualDeposit/LightBoxes	<div><div></div></div> 75.14% 136/181	
src/Store	<div><div></div></div> 100% 16/16	
src/Store/Slices	<div><div></div></div> 64.34% 83/129	
src/Styles	<div><div></div></div> 100% 1/1	
src/Utils	<div><div></div></div> 33.33% 47/141	
src/i18n	<div><div></div></div> 100% 4/4	

Details		
	1 calls	2/2 (100%) condition covered
1/deposit/{id}/{version}	3 calls	6/6 (100%) condition covered
1/deposit/{id}	3 calls	5/5 (100%) condition covered
1/deposit/{id}/{version}/sign	3 calls	7/7 (100%) condition covered
1/deposit/{id}/delete	3 calls	5/5 (100%) condition covered
1/deposit	3 calls	4/5 (80%) condition covered
1/deposit/{id}/{version}/generateOTP	3 calls	6/7 (85.714%) condition covered

# Где можно *подкараулить* спикера

Телеграм и почта в QR-кодах,

а также на стенде Сбера в дискуссионной зоне

