

# Привет!

Меня зовут Павел  
и я хочу представить  
свой проект



автоматизация тестирования  
сайта для создания плана тренировок  
с применением BDD подхода.

<https://log.finalsurge.com/>

<https://github.com/PavelNovikau1/FINALSURGE>



# Стек технологий, использованный на проекте:

- ▶ **Java 8**
- ▶ **Apache Maven**
- ▶ **Cucumber**
- ▶ **Selenide**
- ▶ **testNG**
- ▶ **Allure Framework**
- ▶ **Apache Log4j2**
- ▶ **CircleCI**
- ▶ **Git + GitHub**



Java 8 – строго типизированный объектно-ориентированный язык программирования общего назначения, разработанный компанией Sun Microsystems (в последующем приобретённый компанией Oracle).



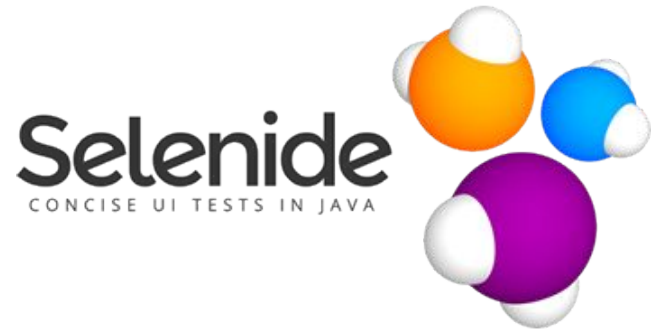
Apache Maven — фреймворк для автоматизации сборки проектов на основе описания их структуры в файлах на языке POM, являющемся подмножеством XML. Проект Maven издаётся сообществом Apache Software Foundation, где формально является частью Jakarta Project.



Cucumber – это программный инструмент, поддерживающий разработку, основанную на поведении. Центральным элементом подхода Cucumber BDD является синтаксический анализатор обычного языка, называемый Gherkin. Это позволяет указать ожидаемое поведение программного обеспечения на понятном для клиентов логическом языке.



Selenide – это обёртка вокруг Selenium WebDriver, позволяющая быстро и просто его использовать при написании тестов, сосредоточившись на логике, а не суете с браузером. Одним словом, цель Selenide – сосредоточиться на бизнес-логике и не заниматься вечными надоедливыми мелкими проблемами.



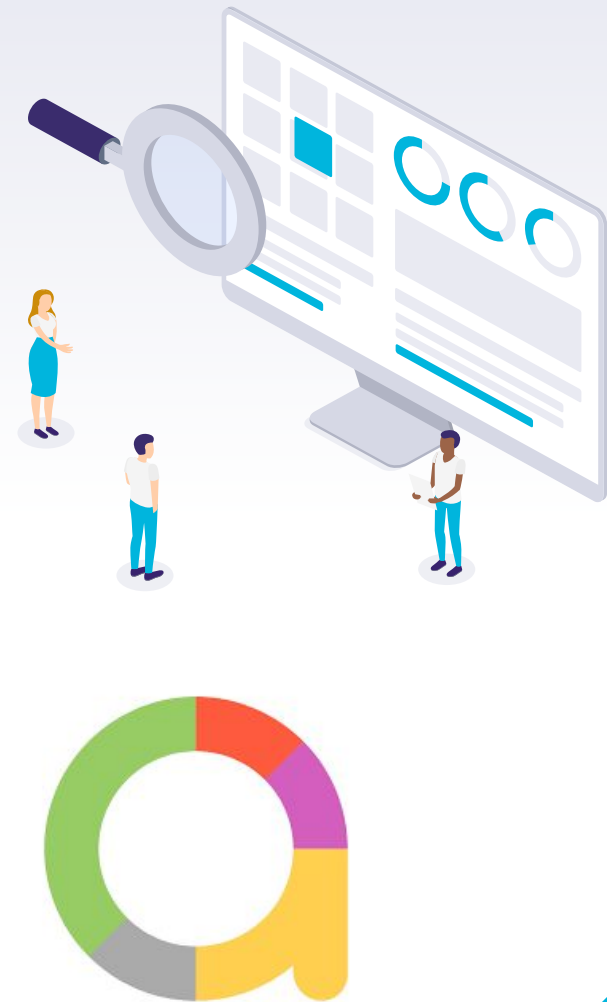
TestNG – это тестовая среда для языка программирования Java, созданная Cédric Beust и вдохновленная JUnit и NUnit. Целью разработки TestNG является охват более широкого диапазона категорий тестов: модульные, функциональные, сквозные, интеграционные и т. д. С более мощными и простыми в использовании функциями.



# TestNG



Allure Framework – популярный инструмент построения отчётов автотестов, упрощающий их анализ. ... Кроме того, Allure-отчеты поддерживают взаимодействие с системами непрерывной интеграции и баг-трекингowymi системами, что позволяет всегда держать под рукой нужную информацию о прохождении тестов и дефектах



Log4j2 представляет собой вторую, улучшенную версию логгера Log4, библиотеку логирования, в которой интерфейс API и реализация выделены отдельно, что позволяет использовать API Log4j 2 совместно с реализацией другого логгера.



Git — распределённая система управления версиями.

GitHub — крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки. Веб-сервис основан на системе контроля версий Git и разработан на Ruby on Rails и Erlang компанией GitHub, Inc.



CircleCI — это continuous integration сервис для веб и мобильных приложений. Одно из преимуществ данного сервиса — сборка билда в облаке, то есть не нужна локальная билд-машина.



# Архитектура фреймворка

В основе архитектуры лежит BDD подход, который позволяет сделать процесс разработки более понятным и управляемым путем налаживания внутренних коммуникаций.



# Немного о BDD подходе:

**BDD (Behaviour Driven Development – «Разработка через поведение»)** – гибкая методология, тесно связанная с TDD (Test Driven Development – "Разработка через тестирование").

В BDD для описания тестов требуется использование естественного языка, понятного каждому участнику проекта, чтобы, фактически, объединить постановку задачи, тесты и документацию воедино.

Другими словами, определяется DSL (специфичный предметно-ориентированный язык), потом составляется стандартный ограниченный набор фраз, описывающих поведение нужных элементов. Затем с их помощью разрабатывается сценарий использования новой функциональности, который будет понятен всем.



# Gherkin

– язык документации, который описывает поведение системы (BDD) при помощи сценариев.

- ▶ **Given**
- ▶ **When**
- ▶ **Then**

**Feature:** Calculators functionality.

As a customer to use the application the user wants to have a possibility to calculate a number steps per workout and daily calorie requirement.

**@SmokeTest**

**Scenario Template:** Calculate the pace of workout.

**Given** User is logged on his account  
**And** User open intensity calculator on the header  
**And** User select event Marathon by clicking on radio button  
**And** User enter event time '<hours>' '<minutes>' '<seconds>'  
**When** User click on calculate passes button  
**Then** Workouts paces result table is open

**Examples:**

hours	minutes	seconds
7	24	36

## ГРАФИЧЕСКИ МОЖНО ПРЕДСТАВИТЬ СЛЕДУЮЩИМ ОБРАЗОМ:





# ШАГИ (STEPS)

```
public class WorkoutSteps extends AbstractSteps {

    @And("User expand Workouts drop down menu on the header")
    public void userExpandWorkoutsDropDownMenuOnTheHeader() {
        headerPage.expandWorkoutDropDownMenu();
    }

    @And("User click on the add workout button")
    public void userClickOnTheAddWorkoutButton() {
        headerPage.clickOnTheAddWorkoutButton();
    }

    @And("User select Activity Type {string}")
    public void userSelectActivity(String activity) {
        workoutAddPage.selectActivityType(activity);
    }

    @And("User filling workout name {string} field")
```

# Автотесты

```
public WorkoutAddPage selectActivityType(String activityType) {  
    log.info(String.format("Select: '%s' activity from list", activityType));  
    $(By.xpath(String.format(ACTIVITY_TYPE, activityType))).click();  
    return this;  
}  
  
public WorkoutAddPage fillingWorkoutNameField(String name) {  
    log.info(String.format("Enter text: '%s' into workout name field", name));  
    workoutNameField.sendKeys(name);  
    return this;  
}  
  
public WorkoutAddPage fillingDescriptionNameField(String description) {  
    log.info(String.format("Enter text: '%s' into description name field", description));  
    descriptionField.sendKeys(description);  
    return this;  
}
```

# Что не получилось?!

<https://docs.gameta.io/allure-report/frameworks/java/cucumber>

## Steps

All scenario steps are automatically translated into allure steps



# Что не получилось?!

## Execution

### ✓ Set up

✓ setUpClass 907ms

### ✓ Test body

✓ \$("open") https://log.finalsurge.com/login.cshtml 5s 230ms

✓ \$("By.id: login\_name") send keys(johnsmit7589@mailinator.com) 290ms

✓ \$("By.id: login\_password") send keys(RrXDVWkUIV7KhIV) 397ms

✓ \$("By.xpath: //button[normalize-space()='Login']") click() 3s 005ms

✓ \$("#LayoutProfilePic") is displayed() 109ms

### ✓ Tear down

✓ tearDownClass 111ms

# Спасибо!

