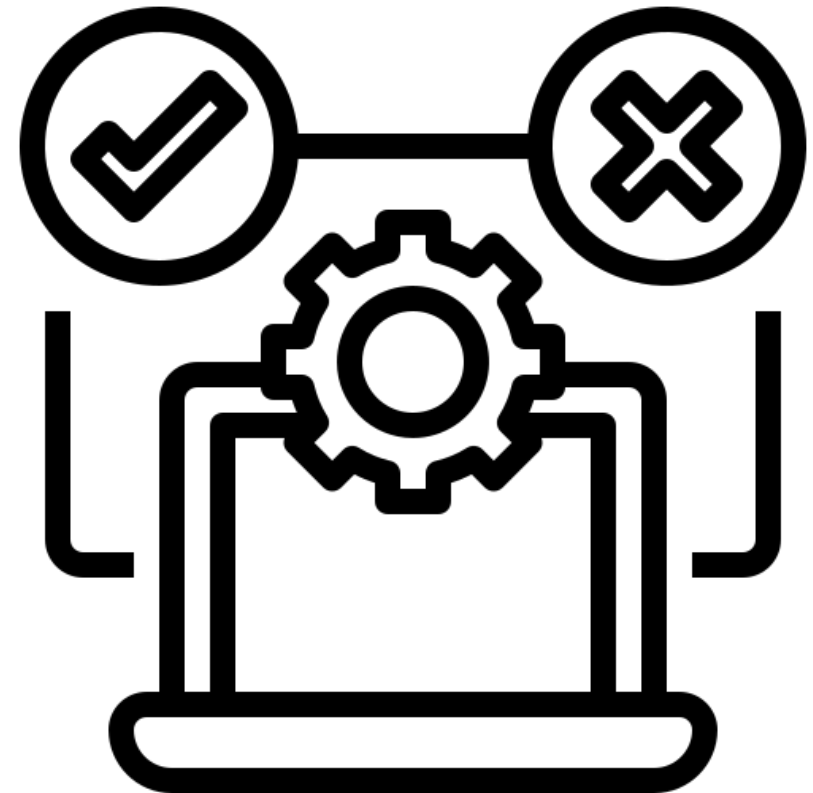


# АВТОМАТИЗИРОВАННОЕ ТЕСТИРОВАНИЕ

## Лекция 5: Jenkins и Allure

БАШАРИНА ЕКАТЕРИНА АЛЕКСАНДРОВНА



# Шаблоны (паттерны)

*- наборы повторно используемых подходов и методов создания тестов*

- Упрощение процесса тестирования
- Улучшение качества тестов
- Повышение производительности тестирования



# Паттерны взаимодействия

- Приспосабливание тестов к восприятию неподготовленным человеком
- Переводим код в человеческий язык
- Разделяем код на атомарные элементы

**Feature:** Addition

In order to avoid silly mistakes

As a math idiot

I want to be told the sum of two numbers

**Scenario:** Add two numbers

**Given** I have entered 50 into the calculator

**And** I have entered 70 into the calculator

**When** I press add

**Then** the result should be 120 on the screen

# Основная идея

- При написании тестов составляется логический сценарий, состоящий из понятных человеку шагов
- На каждом шаге описывается действие, которое необходимо совершить

```
@Step("Navigate driver to {url}")
public void navigate(String url) { home = new Home(driver, url); }

@Step("Perform login with {name} and {password}")
public void login(String name, String password) { home.login(name, password); }

@Step("Switch to default content")
public void switchToDefaultContent() { driver.switchTo().defaultContent(); }
```

# Какие ещё аннотации могут быть?

@Title

@Description

@Step

@Attachment

...

```
@Title("This is our cool test suite")
@Description("In this cool suite we will test only cool features")
public class MyTest {

    @Title("First cool check")
    @Description("In this cool test we will check cool thing")
    @Test
    public void testNumberOne() {
        assertTrue(...);
    }
}
```

# Allure

- **Фреймворк для составления отчётов о прохождении тестов в виде веб-страниц**
- **Отчёты должны быть понятны любому человеку!**
- **Для этого используются человеко-читаемые аннотации (например, Step)**



# Как установить Allure?

- Скачиваем и устанавливаем:  
[https://docs.qameta.io/allure/#installing\\_a\\_commandline](https://docs.qameta.io/allure/#installing_a_commandline)
- Инструкция по настройке:  
<https://docs.qameta.io/allure/#testng>
- Необходимо добавить в pom.xml зависимость и плагин

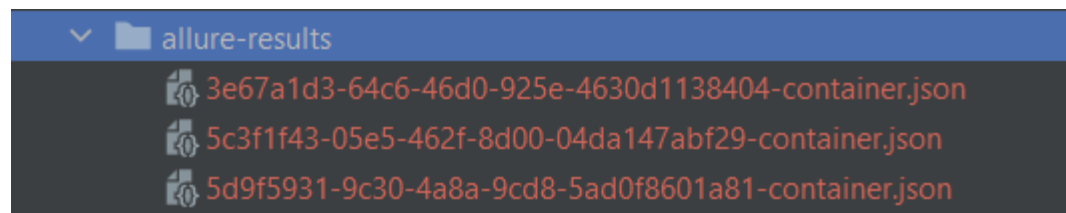
```
<properties>
    <aspectj.version>1.8.10</aspectj.version>
</properties>

<dependencies>
    <dependency>
        <groupId>io.qameta.allure</groupId>
        <artifactId>allure-testng</artifactId>
        <version>LAST_VERSION</version>
        <scope>test</scope>
    </dependency>
</dependencies>

<build>
    <plugins>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-surefire-plugin</artifactId>
            <version>2.20</version> <configuration>
            <argLine> -
                javaagent:"${settings.localRepository}/org/aspect
                j/aspectjweaver/${aspectj.version}/aspectjweaver
                -${aspectj.version}.jar" </argLine>
            </configuration> <dependencies>
            <dependency>
                <groupId>org.aspectj</groupId>
                <artifactId>aspectjweaver</artifactId>
                <version>${aspectj.version}</version>
            </dependency> </dependencies>
            </plugin>
        </plugins>
    </build>
```

# Как пользоваться Allure?

- После запуска тестов появляется папка allure-results

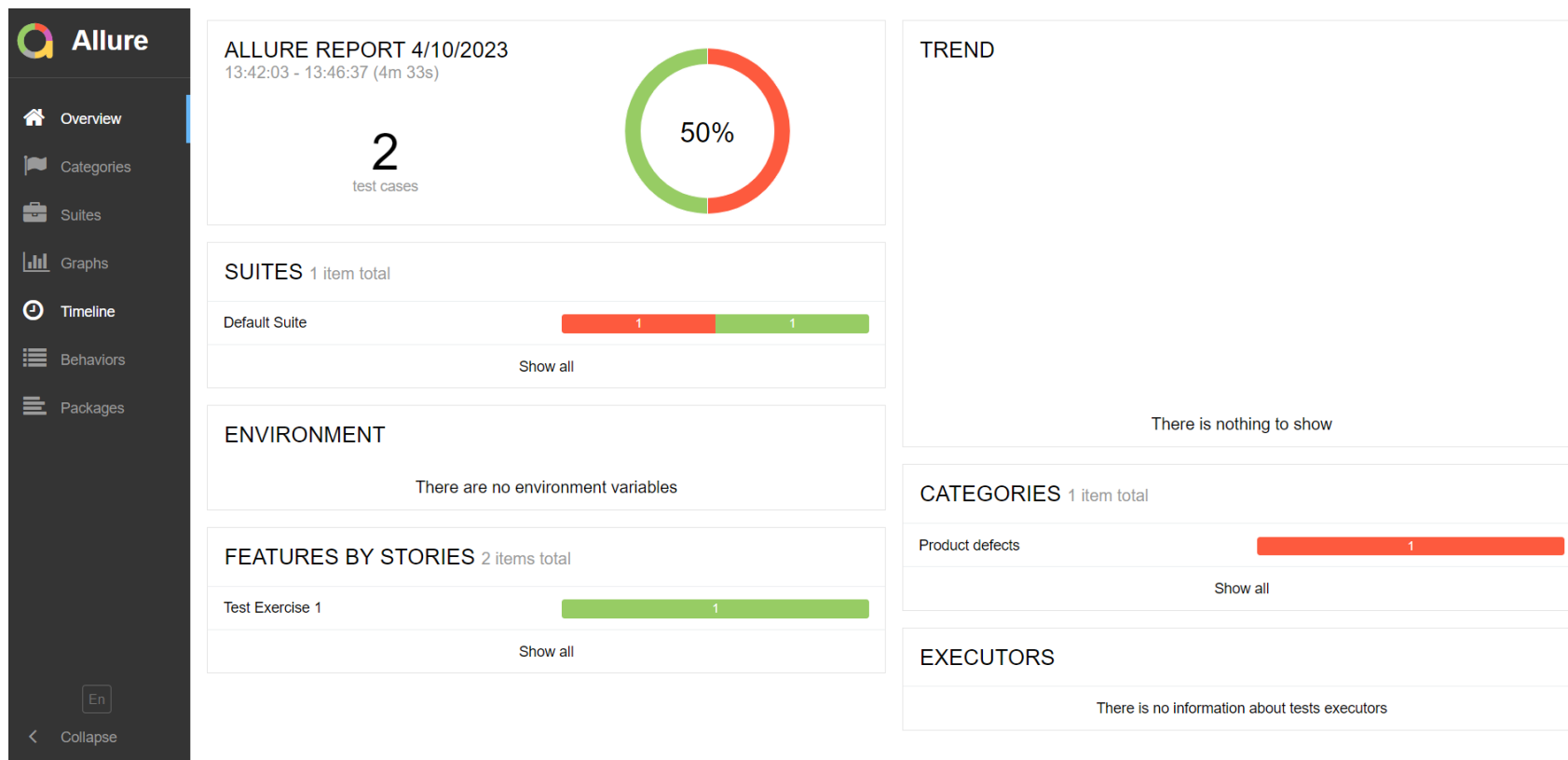


- Из корня проекта запускаем команду `allure serve` (либо передаём в аргументах путь к `allure-results`)

```
C:\Users\Admin\Java\test-app2>allure serve
Generating report to temp directory...
Report successfully generated to C:\Users\Admin\AppData\Local\Temp\12108101811842825976\allure-report
Starting web server...
2023-04-10 13:46:44.641:INFO::main: Logging initialized @2510ms to org.eclipse.jetty.util.log.StdErrLog
Server started at <http://192.168.56.1:5778/>. Press <Ctrl+C> to exit
```



# Allure-отчёт создан!



# Что есть в Allure-репорте?

- Статус запущенных тестов:

**Failed**, **broken**, **passed**, **skipped**, **unknown**

Status:     

- Типы новых тестов:

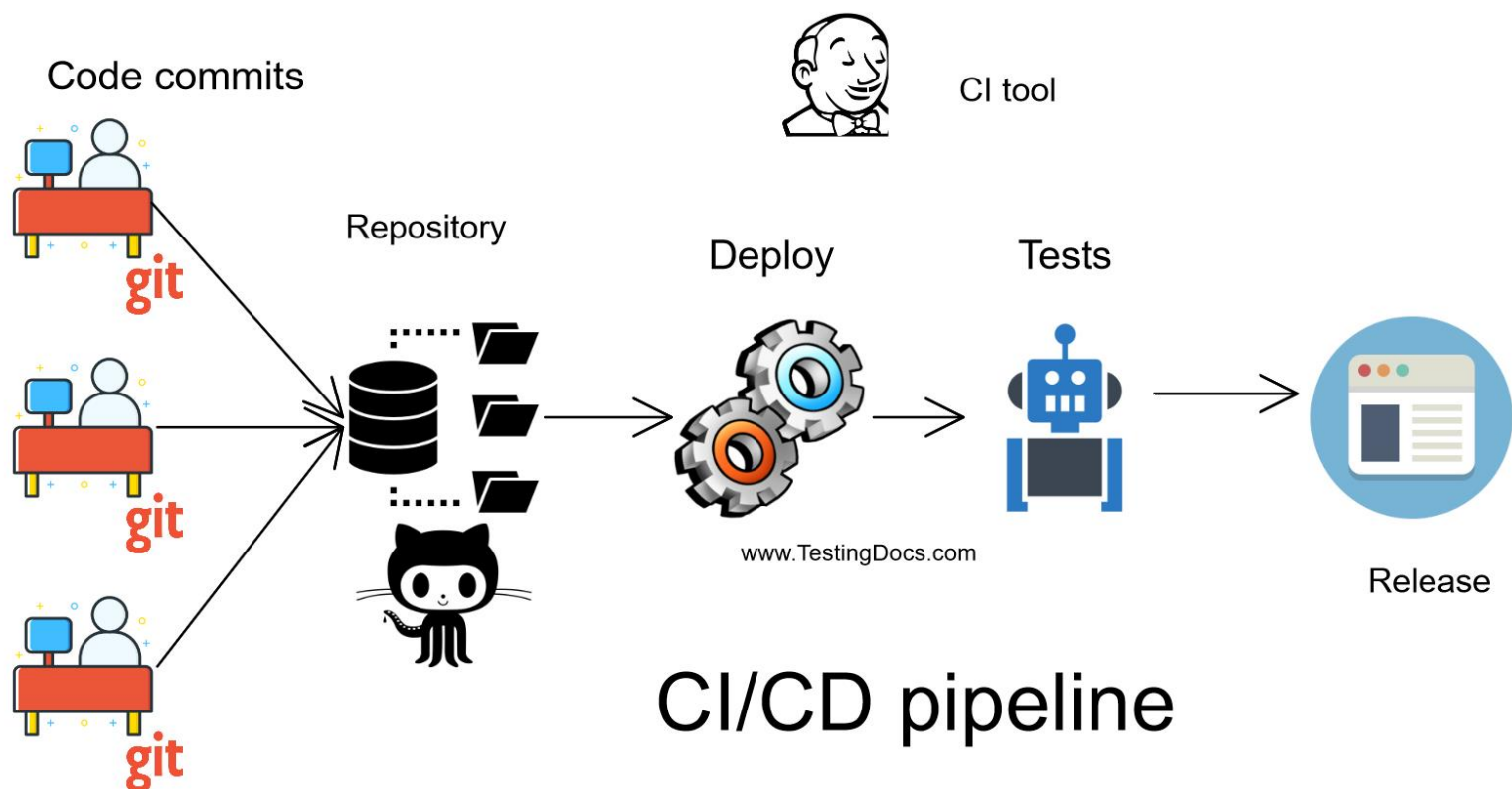
**ненадежный**, **провален**, **пройден**, **сломан**, **изменён**

Marks:     

- И многое другое

# Continuous integration

- Недостаточно просто написать код тестов. Надо автоматизировать даже сам их запуск



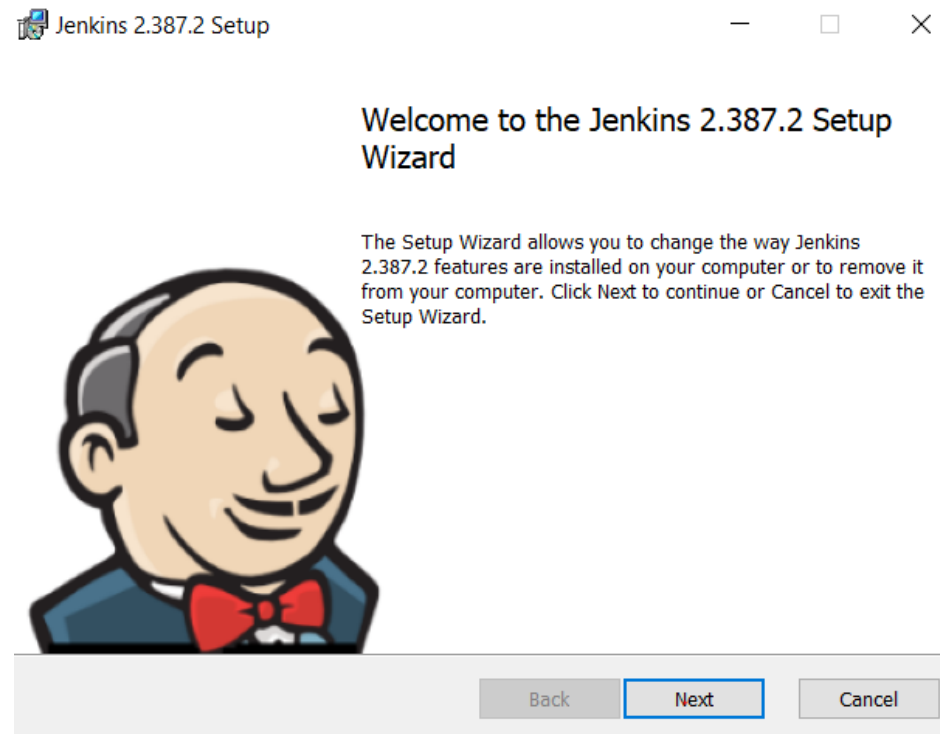
# Jenkins

- Программная система, предназначенная для обеспечения процесса непрерывной интеграции ПО.
- Автоматизирует процесс разработки



# Как установить Jenkins?

- **Скачиваем:**  
<https://www.jenkins.io/download/#downloading-jenkins>
- **Распаковываем и устанавливаем по шагам**
- **Открываем**  
<http://localhost:8080/>



# Как установить Jenkins?

- Видим следующее:

## Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

`C:\Program Files (x86)\Jenkins\secrets\initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

- Вводим пароль из файла `jenkins.err.log` ЛИБО из файла `\.jenkins\secrets\initialAdminPassword`

\*\*\*\*\*

Jenkins initial setup is required. An admin user has been created and a password generated.  
Please use the following password to proceed to installation:

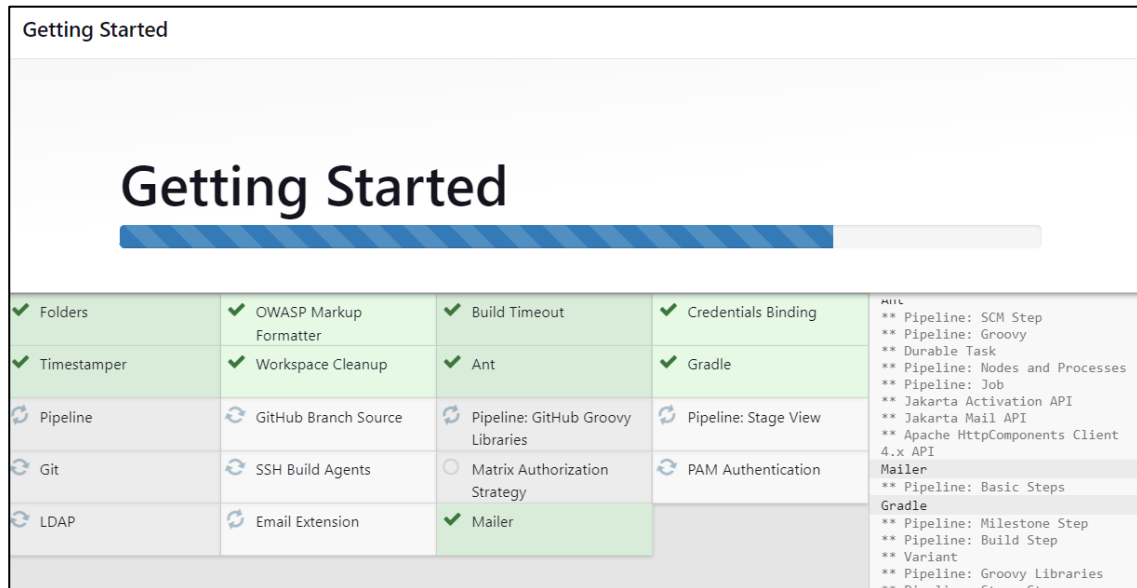
`3f0a53f9bfaa4ad2b89d8aa850181781`

This may also be found at: `C:\ProgramData\Jenkins\.jenkins\secrets\initialAdminPassword`

\*\*\*\*\*

# Как установить Jenkins?

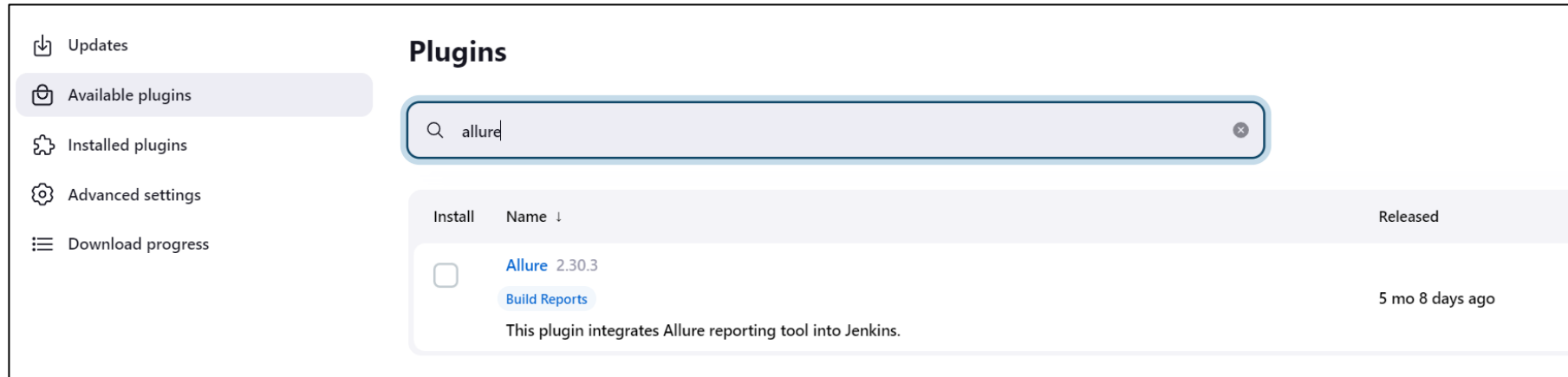
- Ждём долгую загрузку плагинов:



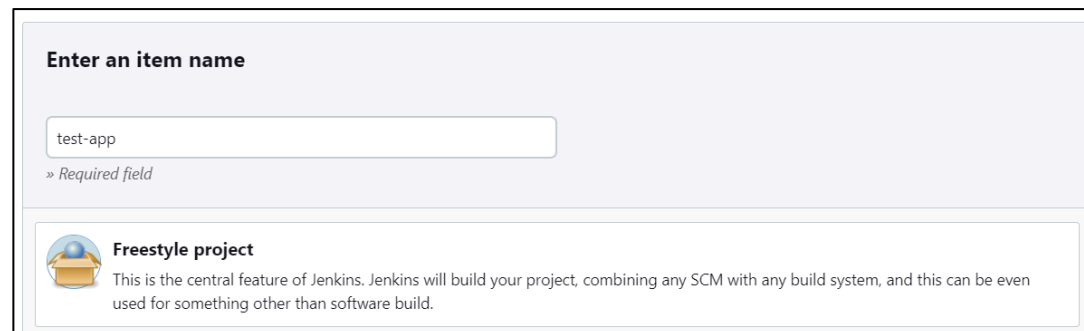
- Вводим свои данные и завершаем настройку

# Как настроить Jenkins?

- В настройках (Manage plugins) добавляем плагин allure



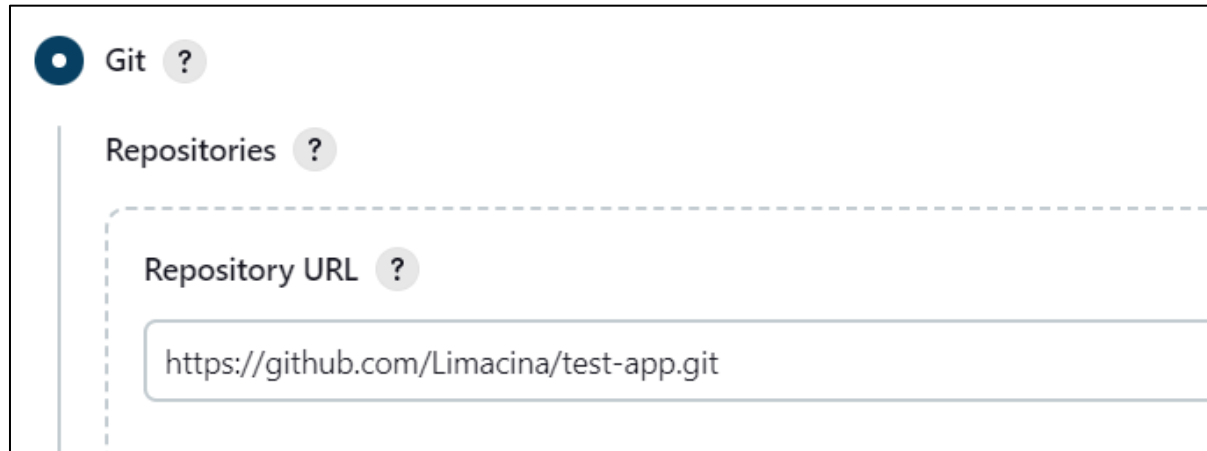
- Создаём новый проект со свободной конфигурацией





# Как настроить проект в Jenkins?

- Подключаем гит-репозиторий:



The screenshot shows the Jenkins configuration interface for a Git repository. At the top, there is a radio button labeled "Git" with a question mark icon. Below it, the "Repositories" section is expanded, showing a "Repository URL" field with a question mark icon. The URL entered is "https://github.com/Limacina/test-app.git".

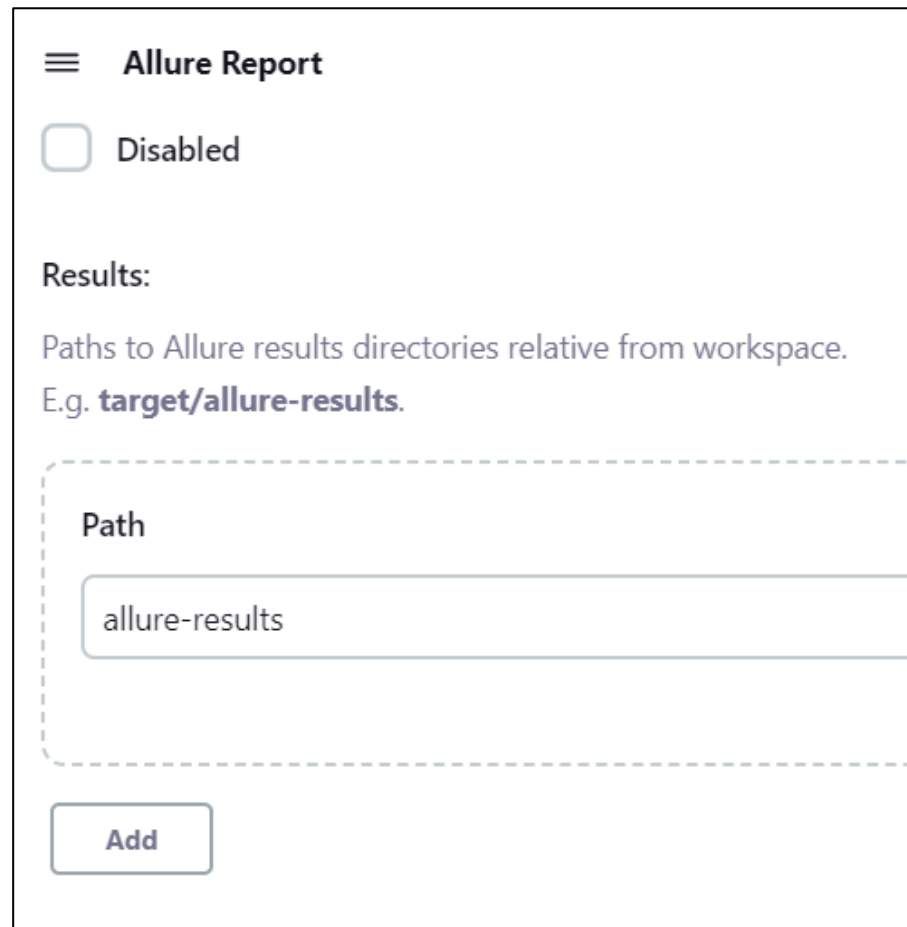
- Настраиваем запуск maven-сьюта



The screenshot shows the Jenkins configuration interface for Maven goals. At the top, there is a section titled "Invoke top-level Maven targets" with a question mark icon. Below it, the "Goals" section is expanded, showing a text input field with the goal "clean test -P AllureSuite".

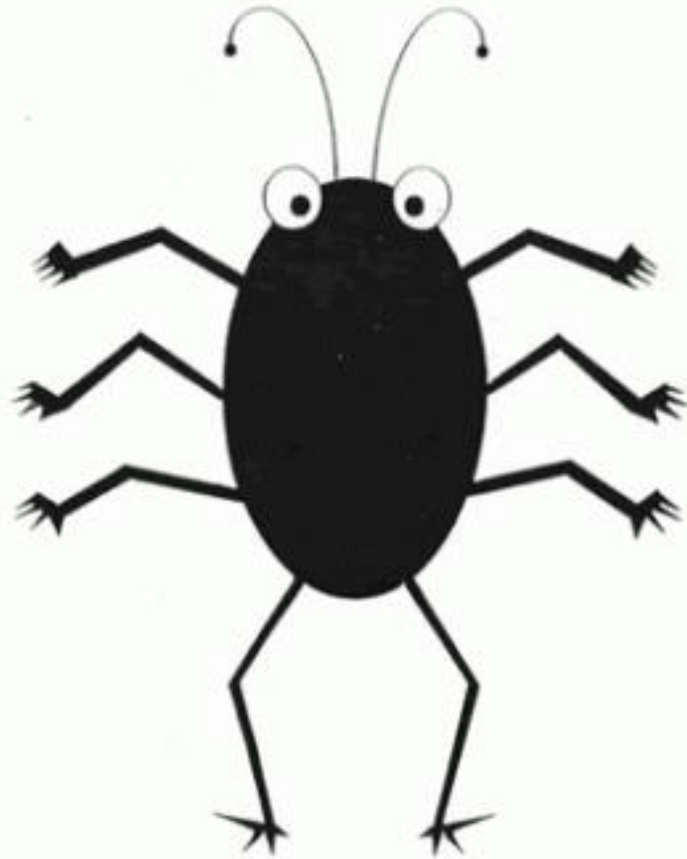
# Как настроить проект в Jenkins?

- После сборки формируем Allure-отчёт:

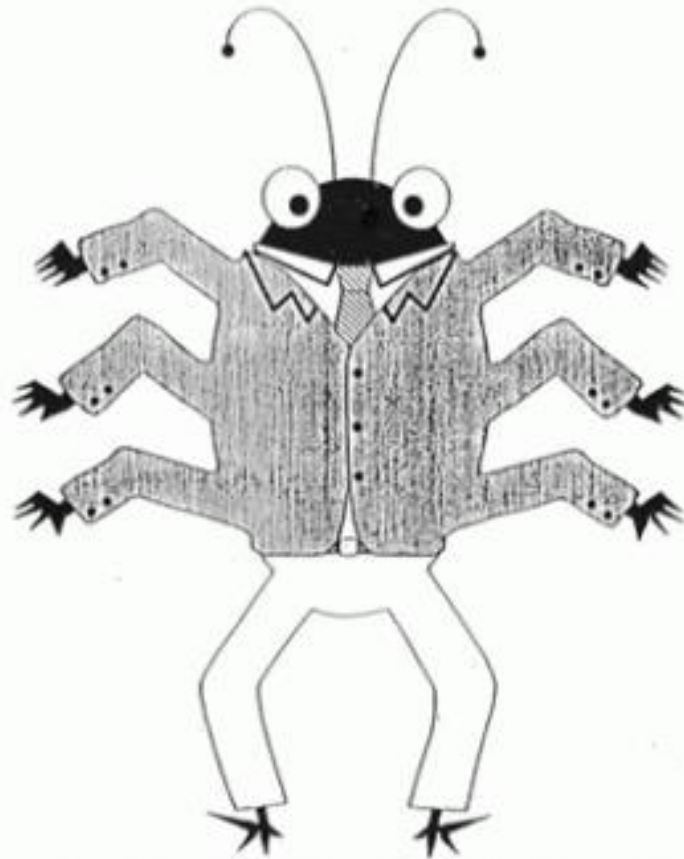


The screenshot shows the 'Allure Report' configuration panel in Jenkins. At the top, there is a hamburger menu icon followed by the text 'Allure Report'. Below this is a checkbox labeled 'Disabled', which is currently unchecked. Underneath the checkbox is the section 'Results:'. This section contains a descriptive text: 'Paths to Allure results directories relative from workspace. E.g. **target/allure-results**.' Below the text is a dashed-line box representing a list of paths. Inside this box, the label 'Path' is positioned above a text input field that contains the value 'allure-results'. At the bottom left of the dashed box is an 'Add' button.

# Спасибо за внимание!



**BUG**



**FEATURE**