

1. Налаштування Jenkins

Завантажено і встановлено Jenkins на локальний сервер.

Виконано базове налаштування:

- Встановлено плагіни: **Pipeline**, **Allure Reports**, **Parameterized Trigger**, **HTML Publisher**.
- Створено необхідних користувачів та призначено ролі.

2) А: Завдання для простого методу main

String Parameter ?

Name ?

IN Main for java

Default Value ?

Description ?

для введення тестових даних

Plain text [Попередній перегляд](#)

☐ Trim the string ?

Post-build Actions

Define what happens after a build completes, like sending notifications, archiving artifacts

Execute shell ?

Command

[See the list of available environment variables](#)

```
javac Main.java
java Main ${param1}
```

Allure Report

☐ Disabled

Results:

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

Path

allure-results

В: Виконання API-тестів з параметризованими даними.

Build Steps

Automate your build process with ordered tasks like code compilation, testing, packaging, etc.

Execute shell ?

Command

[See the list of available environment variables](#)

```
mvn test -Dtest="ApiTestSuite" -Dparam1=${param1}
```

Results:

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

Path

allure-results

Name ?

REST Assured

Choices ?

! Requires Choices.

Description ?

For Adding Parametazation

с. Завдання для UI (+Allure)

Definition

Pipeline script

Script ?

```
1 pipeline {  
2   agent any  
3   stages {  
4     stage('Build') {  
5       steps {  
6         sh 'mvn clean compile'  
7       }  
8     }  
9     stage('Test') {  
10      steps {  
11        sh 'mvn test -Dtest="UITestSuite"  
12      }  
13    }  
14    stage('Allure Report') {  
15      steps {  
16        allure([  
17          results: [[path: '**/allure-results']],  
18          reportBuildPolicy: 'ALWAYS'  
19        ])  
20      }  
21    }  
22  }  
23 }
```

d. Завдання для тестування продуктивності

Command

See [the list of available environment variables](#)

jmeter -n -t Test_Scenario.jmx -l results.jtl

Reports

HTML directory to archive ?

C:\Program Files\Jenkins HTML report

Index page[s] ?

index.html

Index page title[s] (Optional) ?

|

Report title ?

HTML Report

e. Завдання для мобільних додатків

Pipeline script

Script ?

```
1 pipeline {
2   agent any
3   stages {
4     stage('Build APK') {
5       steps {
6         sh './gradlew assembleDebug'
7       }
8     }
9     stage('Test') {
10      steps {
11        sh 'mvn test -Dtest="MobileTestSuite"'
12      }
13    }
14    stage('Allure Report') {
15      steps {
16        allure([
17          results: [[path: '**/allure-results']],
18          reportBuildPolicy: 'ALWAYS'
19        ])
20      }
21    }
22  }
23 }
```

f. Завдання для настільних додатків

Definition

Pipeline script

Script ?

```
1 pipeline {
2   agent any
3   stages {
4     stage('Test') {
5       steps {
6         sh 'java -jar DesktopTests.jar'
7       }
8     }
9     stage('Allure Report') {
10      steps {
11        allure([
12          results: [[path: '**/allure-results']],
13          reportBuildPolicy: 'ALWAYS'
14        ])
15      }
16    }
17  }
18 }
```

3. Інтеграція звітів Allure

І основні завдання API, UI, мобільних та настільних додатків додано звіти Allure через плагін **Allure Reports**. Та налаштовано автоматичну генерацію звітів після кожного запуску.