SELENIUM & JMETER RAPPORT

Management de qualité

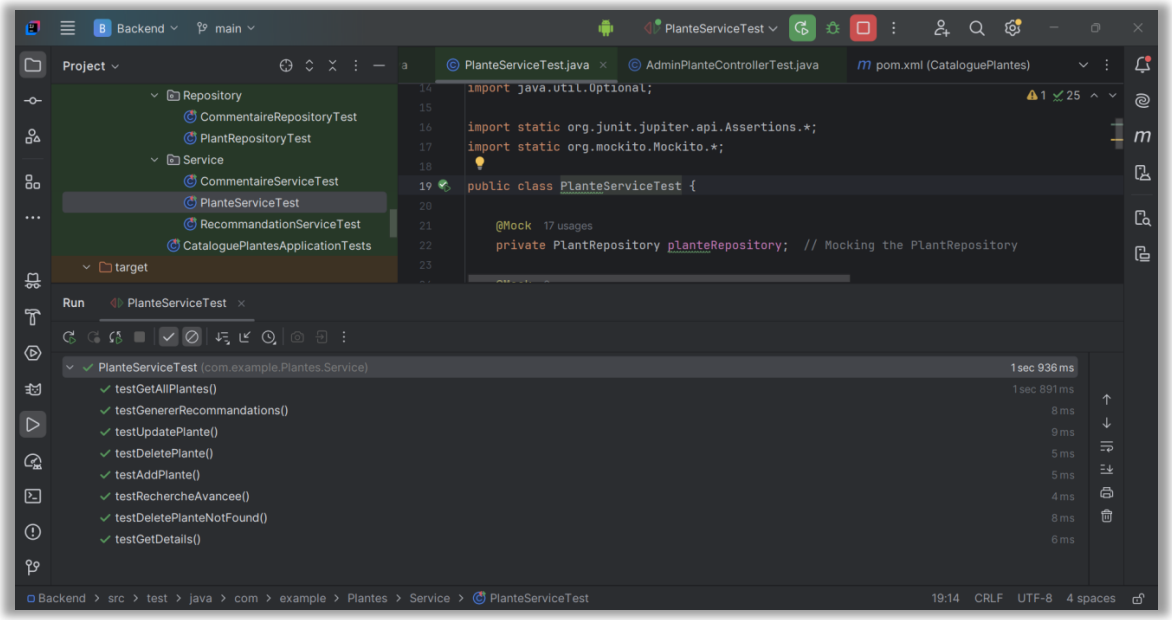
Dr. Driss Essabar

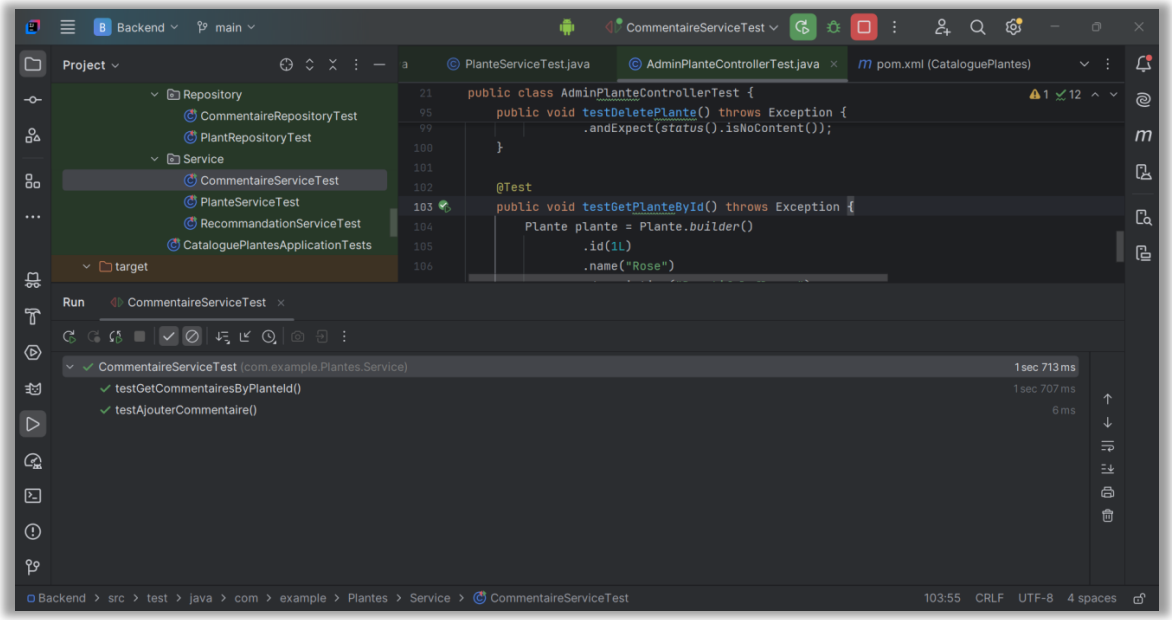
25 DECEMBRE 2024

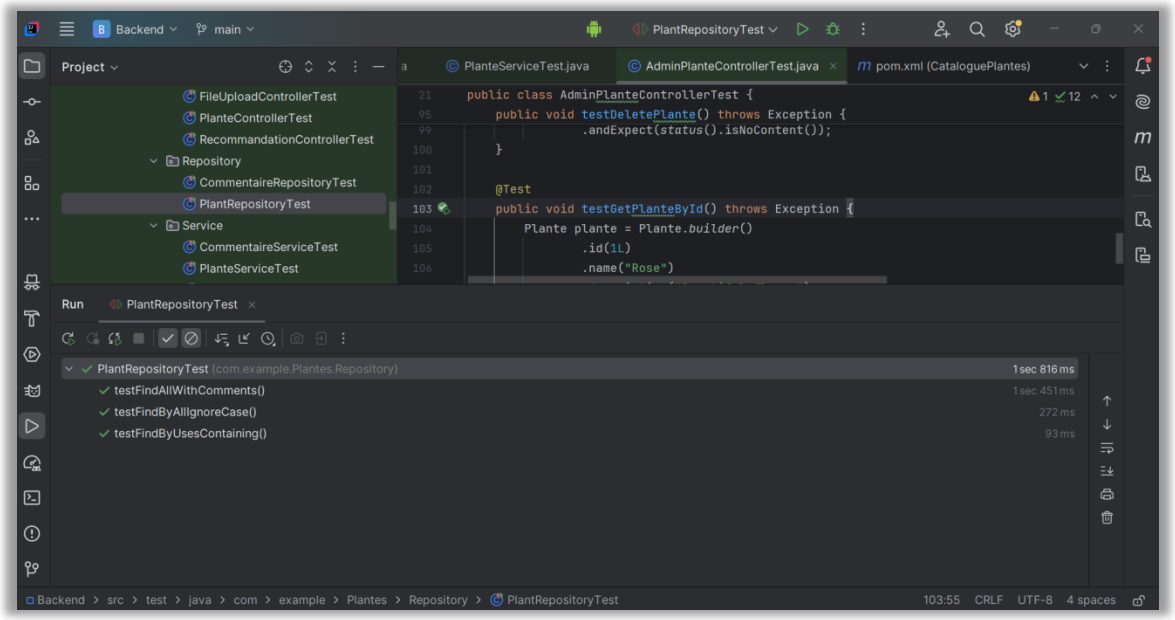
EL ASSOULI NOUHAYLA
DIDI ALAOUI LATIFA
ROUITA CHAIMAE
TIZGHA NIHAL

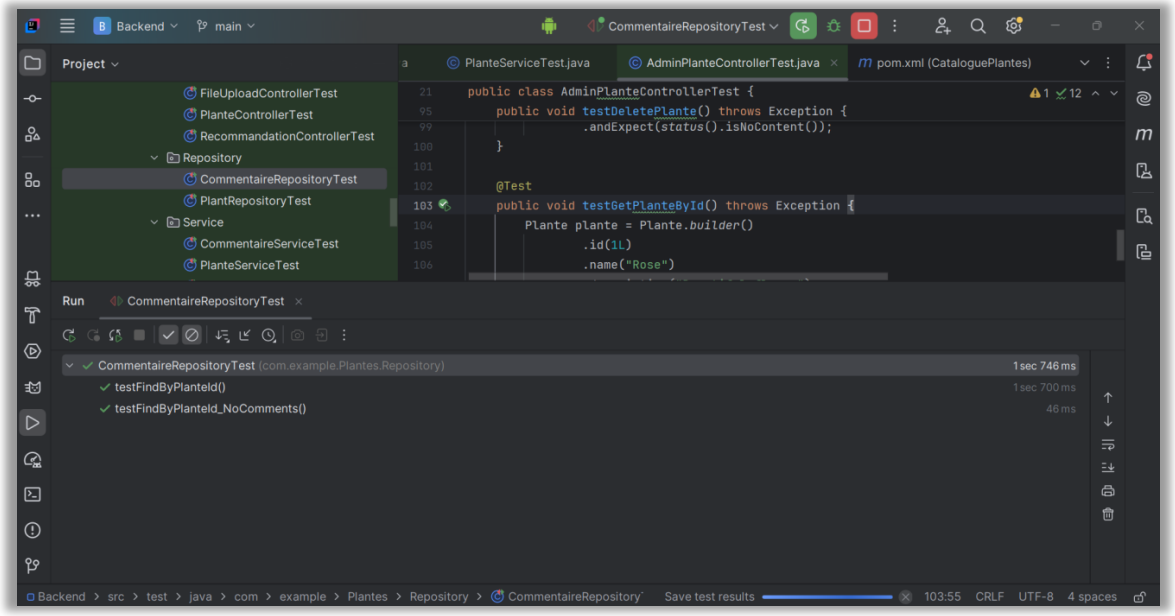
Tests Unitaires

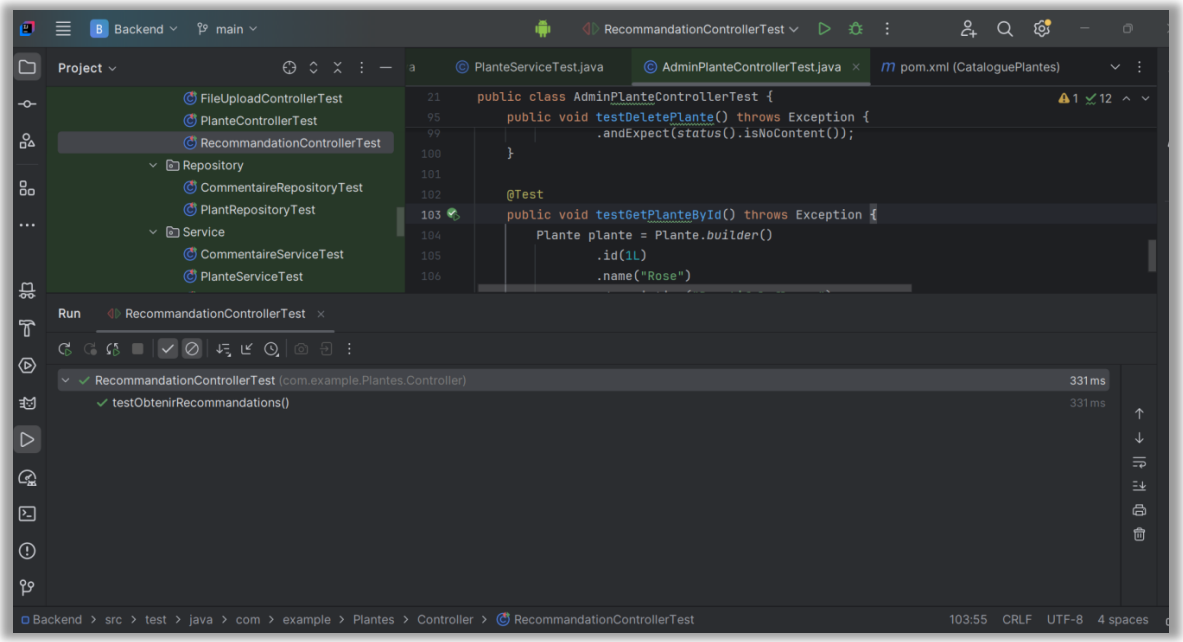
➢ Résultats des Tests JUnit pour le Projet :

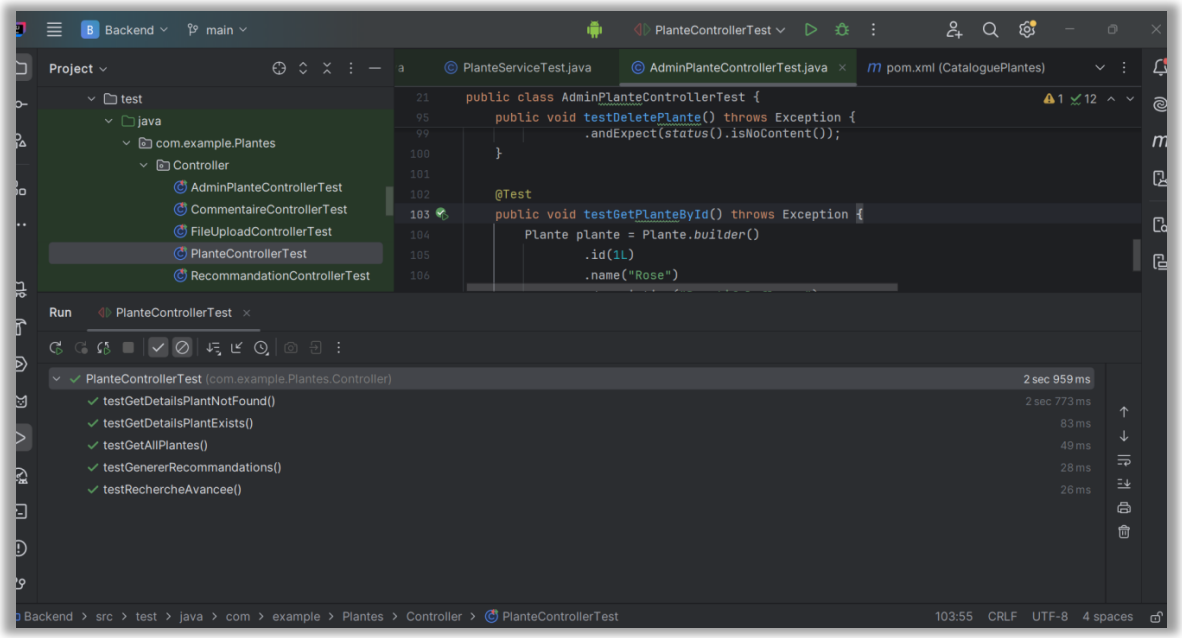


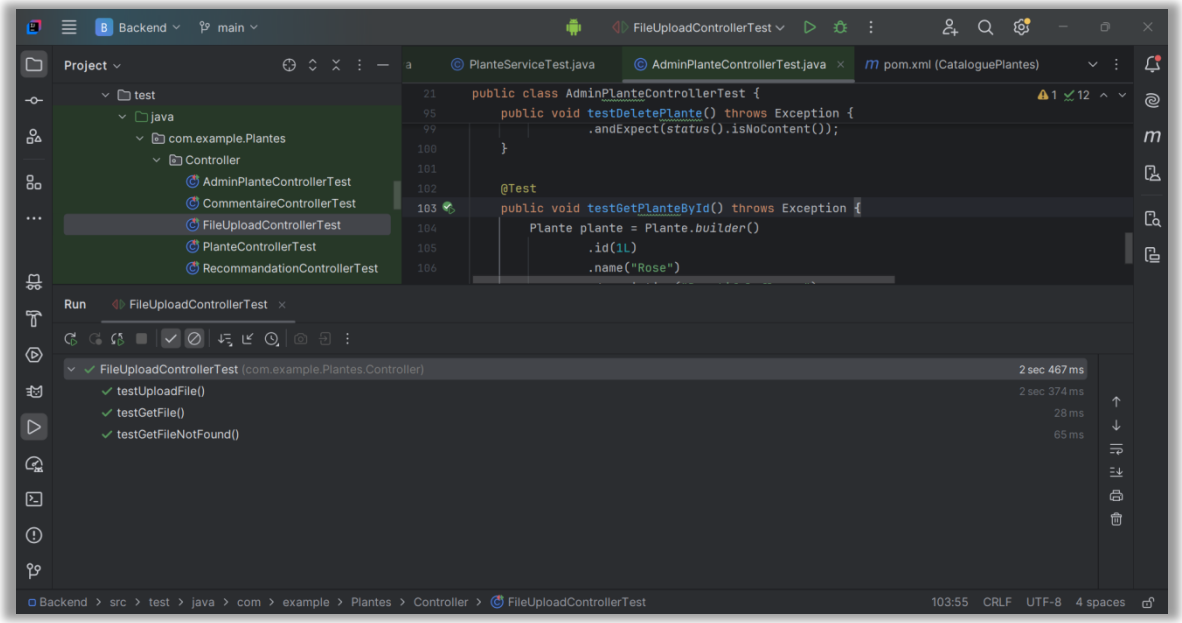


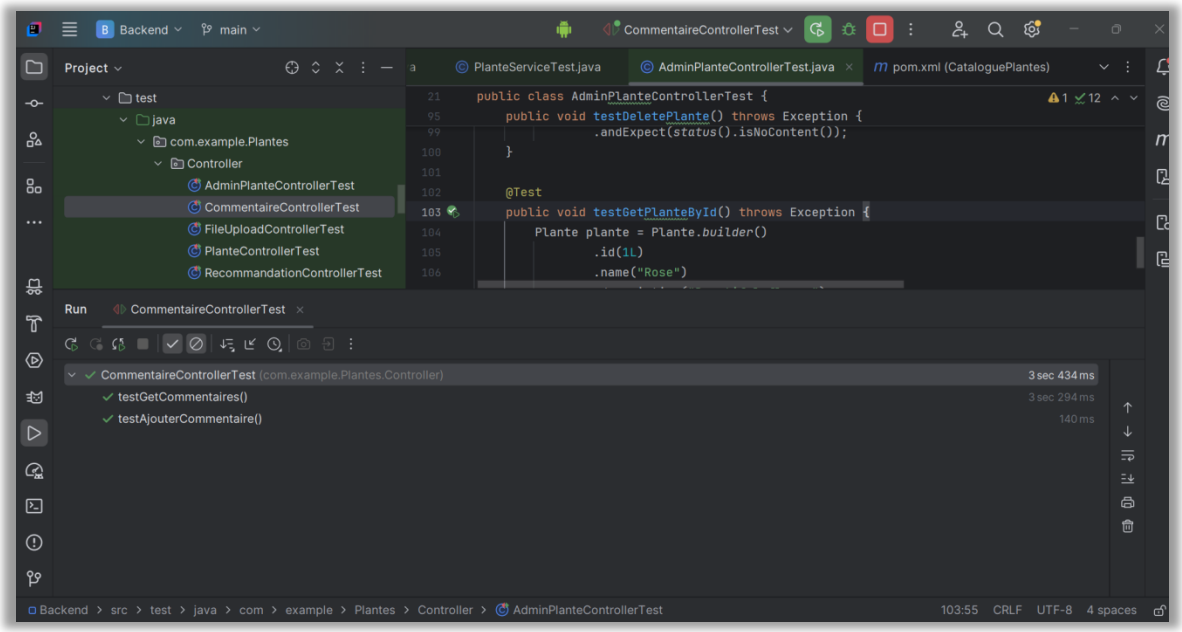

Backend
main
PlantRepositoryTest
Project
FileUploadControllerTest
PlanteControllerTest
RecommandationControllerTest
Repository
CommentaireRepositoryTest
PlantRepositoryTest
Service
CommentaireServiceTest
PlanteServiceTest
PlanteServiceTest.java
AdminPlanteControllerTest.java
pom.xml (CataloguePlantes)
public class AdminPlanteControllerTest {
public void testDeletePlante() throws Exception {
.andExpect(status().isNoContent());
}
@Test
public void testGetPlanteById() throws Exception {
Plante plante = Plante.builder()
.id(1L)
.name("Rose")
Run
PlantRepositoryTest
PlantRepositoryTest (com.example.Plantes.Repository) 1 sec 816 ms
testFindAllWithComments() 1 sec 451 ms
testFindByAllIgnoreCase() 272 ms
testFindByUsesContaining() 93 ms
Backend › src › test › java › com › example › Plantes › Repository › PlantRepositoryTest
103:55 CRLF UTF-8 4 spaces

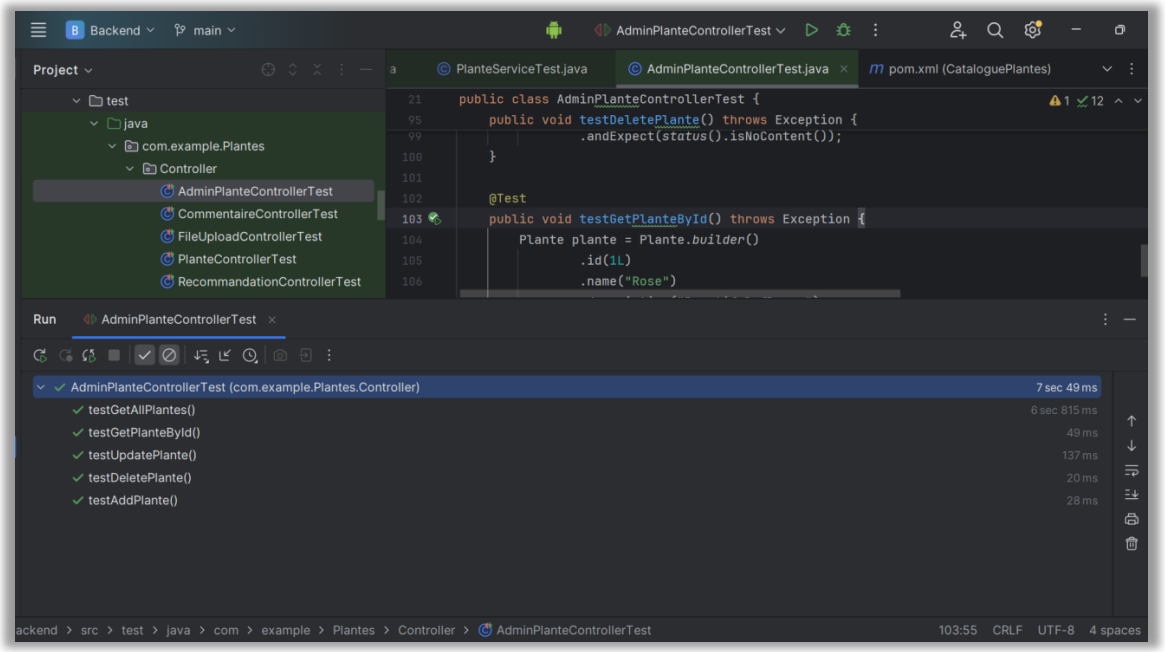

Backend
main
CommentaireRepositoryTest
Project
FileUploadControllerTest
PlanteControllerTest
RecommandationControllerTest
Repository
CommentaireRepositoryTest
PlantRepositoryTest
Service
CommentaireServiceTest
PlanteServiceTest
PlanteServiceTest.java
AdminPlanteControllerTest.java
pom.xml (CataloguePlantes)
public class AdminPlanteControllerTest {
public void testDeletePlante() throws Exception {
.andExpect(status().isNoContent());
}
@Test
public void testGetPlanteById() throws Exception {
Plante plante = Plante.builder()
.id(1L)
.name("Rose")
Run
CommentaireRepositoryTest
CommentaireRepositoryTest (com.example.Plantes.Repository) 1 sec 746 ms
testFindByPlanteId() 1 sec 700 ms
testFindByPlanteId_NoComments() 46 ms
Backend › src › test › java › com › example › Plantes › Repository › CommentaireRepository
Save test results
103:55 CRLF UTF-8 4 spaces

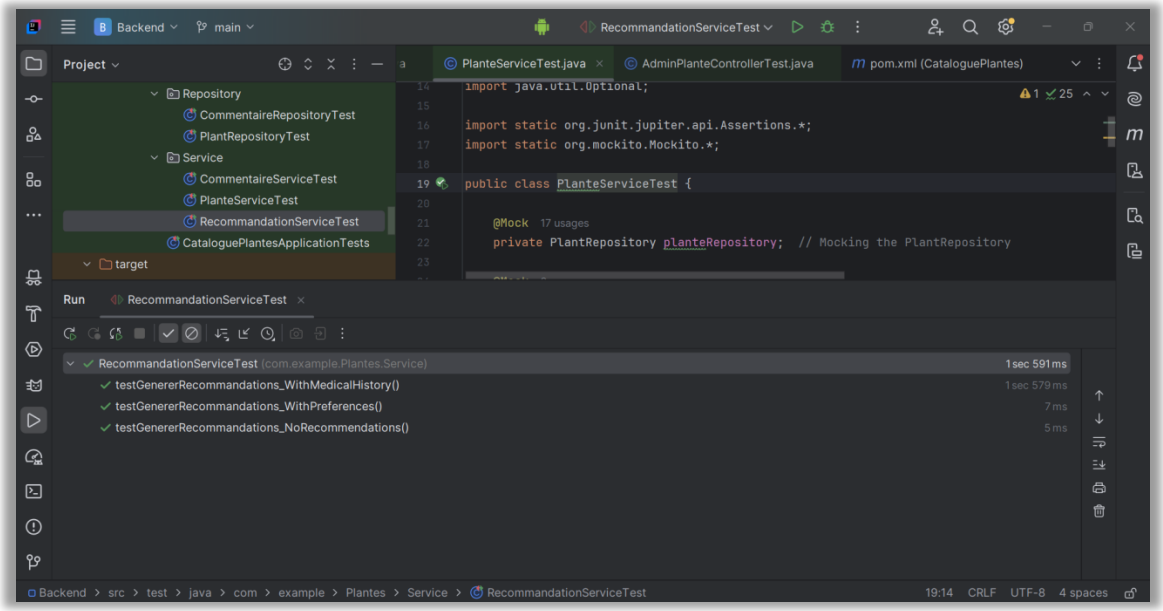

Backend
main
RecommandationControllerTest
Project
FileUploadControllerTest
PlanteControllerTest
RecommandationControllerTest
Repository
CommentaireRepositoryTest
PlantRepositoryTest
Service
CommentaireServiceTest
PlanteServiceTest
PlanteServiceTest.java
AdminPlanteControllerTest.java
pom.xml (CataloguePlantes)
public class AdminPlanteControllerTest {
public void testDeletePlante() throws Exception {
.andExpect(status().isNoContent());
}
@Test
public void testGetPlanteById() throws Exception {
Plante plante = Plante.builder()
.id(1L)
.name("Rose")
Run
RecommandationControllerTest
RecommandationControllerTest (com.example.Plantes.Controller) 331 ms
testObtenirRecommandations() 331 ms
Backend › src › test › java › com › example › Plantes › Controller › RecommandationControllerTest
103:55 CRLF UTF-8 4 spaces


Backend
main
PlanteControllerTest
Project
PlanteServiceTest.java
AdminPlanteControllerTest.java
pom.xml (CataloguePlantes)
test
java
com.example.Plantes
Controller
AdminPlanteControllerTest
CommentaireControllerTest
FileUploadControllerTest
PlanteControllerTest
RecommandationControllerTest
public class AdminPlanteControllerTest {
public void testDeletePlante() throws Exception {
.andExpect(status().isNoContent());
}
@Test
public void testGetPlanteById() throws Exception {
Plante plante = Plante.builder()
.id(1L)
.name("Rose")
Run
PlanteControllerTest
PlanteControllerTest (com.example.Plantes.Controller) 2 sec 959 ms
testGetDetailsPlantNotFound() 2 sec 773 ms
testGetDetailsPlantExists() 83 ms
testGetAllPlantes() 49 ms
testGenererRecommandations() 28 ms
testRechercheAvancee() 26 ms
Backend › src › test › java › com › example › Plantes › Controller › PlanteControllerTest
103:55 CRLF UTF-8 4 spaces


Backend
main
FileUploadControllerTest
Project
PlanteServiceTest.java
AdminPlanteControllerTest.java
pom.xml (CataloguePlantes)
test
java
com.example.Plantes
Controller
AdminPlanteControllerTest
CommentaireControllerTest
FileUploadControllerTest
PlanteControllerTest
RecommandationControllerTest
public class AdminPlanteControllerTest {
public void testDeletePlante() throws Exception {
.andExpect(status().isNoContent());
}
@Test
public void testGetPlanteById() throws Exception {
Plante plante = Plante.builder()
.id(1L)
.name("Rose")
Run
FileUploadControllerTest
FileUploadControllerTest (com.example.Plantes.Controller) 2 sec 467 ms
testUploadFile() 2 sec 374 ms
testGetFile() 28 ms
testGetFileNotFound() 65 ms
Backend › src › test › java › com › example › Plantes › Controller › FileUploadControllerTest
103:55 CRLF UTF-8 4 spaces


Backend
main
CommentaireControllerTest
Project
PlanteServiceTest.java
AdminPlanteControllerTest.java
pom.xml (CataloguePlantes)
test
java
com.example.Plantes
Controller
AdminPlanteControllerTest
CommentaireControllerTest
FileUploadControllerTest
PlanteControllerTest
RecommandationControllerTest
public class AdminPlanteControllerTest {
public void testDeletePlante() throws Exception {
.andExpect(status().isNoContent());
}
@Test
public void testGetPlanteById() throws Exception {
Plante plante = Plante.builder()
.id(1L)
.name("Rose")
Run
CommentaireControllerTest
CommentaireControllerTest (com.example.Plantes.Controller) 3 sec 434 ms
testGetCommentaires() 3 sec 294 ms
testAjouterCommentaire() 140 ms
Backend › src › test › java › com › example › Plantes › Controller › AdminPlanteControllerTest
103:55 CRLF UTF-8 4 spaces


Backend
main
AdminPlanteControllerTest
Project
test
java
com.example.Plantes
Controller
AdminPlanteControllerTest
CommentaireControllerTest
FileUploadControllerTest
PlanteControllerTest
RecommandationControllerTest
PlanteServiceTest.java
AdminPlanteControllerTest.java
pom.xml (CataloguePlantes)
public class AdminPlanteControllerTest {
public void testDeletePlante() throws Exception {
.andExpect(status().isNoContent());
}
@Test
public void testGetPlanteById() throws Exception {
Plante plante = Plante.builder()
.id(1L)
.name("Rose")
Run
AdminPlanteControllerTest
AdminPlanteControllerTest (com.example.Plantes.Controller) 7 sec 49 ms
testGetAllPlantes() 6 sec 815 ms
testGetPlanteById() 49 ms
testUpdatePlante() 137 ms
testDeletePlante() 20 ms
testAddPlante() 28 ms
ackend › src › test › java › com › example › Plantes › Controller › AdminPlanteControllerTest
103:55 CRLF UTF-8 4 spaces

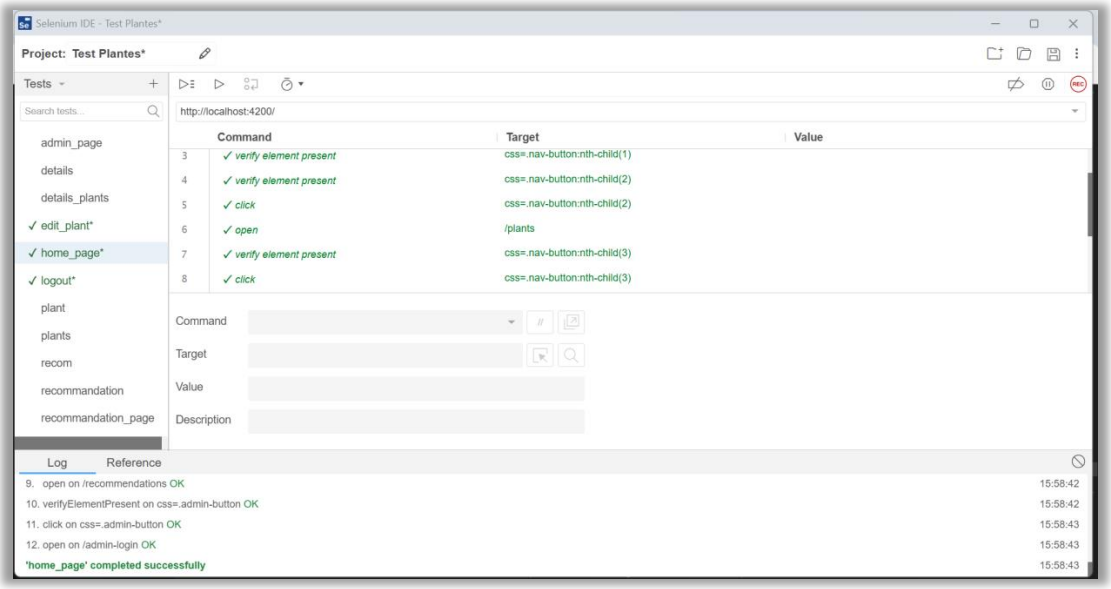

Backend
main
RecommandationServiceTest
Project
Repository
CommentaireRepositoryTest
PlantRepositoryTest
Service
CommentaireServiceTest
PlanteServiceTest
RecommandationServiceTest
CataloguePlantesApplicationTests
target
PlanteServiceTest.java
AdminPlanteControllerTest.java
pom.xml (CataloguePlantes)
import java.util.Optional;
import static org.junit.jupiter.api.Assertions.*;
import static org.mockito.Mockito.*;
public class PlanteServiceTest {
@Mock 17 usages
private PlantRepository planteRepository; // Mocking the PlantRepository
Run
RecommandationServiceTest
RecommandationServiceTest (com.example.Plantes.Service) 1 sec 591 ms
testGenererRecommandations_WithMedicalHistory() 1 sec 579 ms
testGenererRecommandations_WithPreferences() 7 ms
testGenererRecommandations_NoRecommendations() 5 ms
Backend › src › test › java › com › example › Plantes › Service › RecommandationServiceTest
19:14 CRLF UTF-8 4 spaces

- **Scénarios de Test Automatisé avec Selenium IDE :**

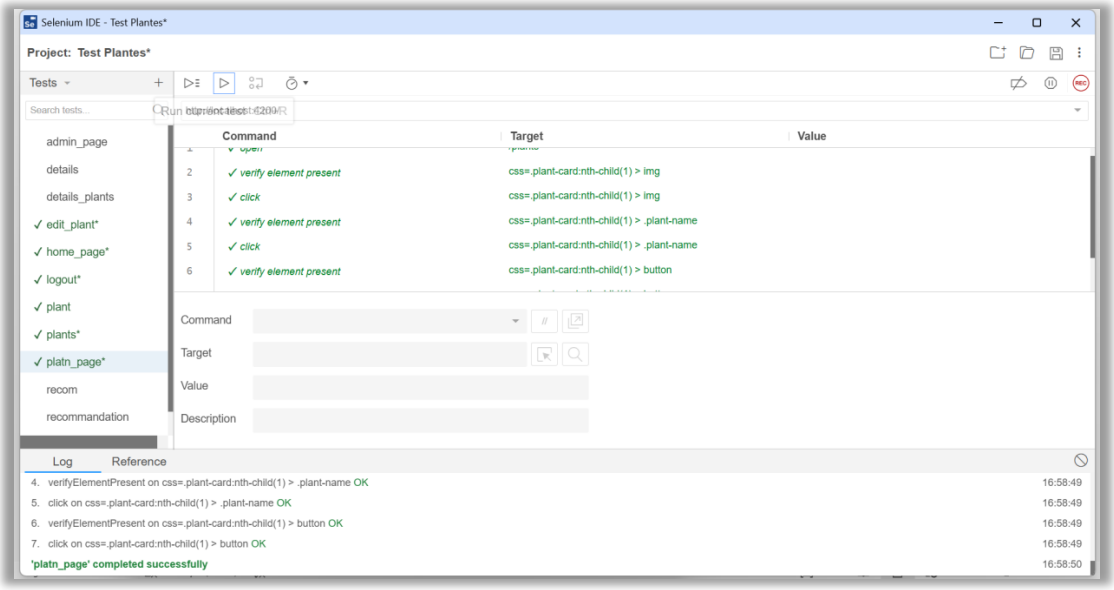
ID	Scénario de Test	Préconditions	Étapes	Résultat Attendu	Vérification
1	Vérifier que les boutons Home, Plants, Recommendations, et Admin sont affichés et fonctionnels.	Accéder à la page d'accueil.	1. Accéder à la page d'accueil. 2. Cliquer sur chaque bouton.	Navigation correcte vers les pages correspondantes	verifyElement
2	Vérifier que la page Plants affiche la liste des plantes avec images et boutons Read More.	La page d'accueil est chargée.	1. Cliquer sur le bouton Plants depuis la page d'accueil.	Chargement de la liste des plantes avec toutes les informations nécessaires.	verifyElement
3	Vérifier que la page Recommendations contient les champs pour les besoins, préférences, et antécédents médicaux.	La page d'accueil est chargée.	1. Cliquer sur le bouton Recommendations depuis la page d'accueil.	Affichage des champs et du bouton Obtenir Résultat.	verifyElement
4	Vérifier que la page Admin redirige vers un formulaire d'authentification	La page d'accueil est chargée.	1. Cliquer sur le bouton Admin depuis la page d'accueil.	Chargement de la page d'authentification.	verifyElement
5	Vérifier que la recherche fonctionne pour des mots-clés valides dans Plants.	La page Plants est chargée.	1. Entrer un mot-clé correspondant à une plante existante. 2. Cliquer sur le bouton Rechercher.	Affichage des plantes correspondantes	assertText
6	Vérifier que la recherche gère les mots-clés non valides dans Plants.	La page Plants est chargée.	1. Entrer un mot-clé inexistant dans la barre de recherche. 2. Cliquer sur Rechercher.	La plante ne s’affiche pas.	verifyElementNotPresent
7	Vérifier que la recherche retourne les plantes selon les besoins de santé dans Recommendations.	La page Recommendations est chargée.	1. Entrer des informations dans le champ Besoins de Santé. 2. Cliquer sur Obtenir Résultat.	Liste des plantes adaptées aux besoins.	verifyElement, assertText
8	Vérifier que la recherche gère les champs vides dans Recommendations.	La page Recommendations est chargée.	1. Laisser tous les champs vides. 2. Cliquer sur Obtenir Résultat.	Aucune plante n'est retournée.	assertElementPresent
9	Vérifier que le bouton Read More redirige vers la bonne page de détails des plantes.	La page Plants est chargée.	1. Cliquer sur le bouton Read More pour une plante spécifique.	Chargement de la page contenant les informations détaillées de la plante.	assertElementPresent
10	Vérifier que toutes	La page de détails	1. Comparer les	Toutes les	assertText,

	les informations des plantes sont correctes sur la page de détails.	d'une plante est chargée.	informations affichées (image, description, etc.) avec les données attendues.	informations sont exactes.	storeAttrubute, assert, assertElementPresent
	Vérifier que l'authentification réussit avec des informations valides.				
11	Vérifier que l'authentification échoue avec des informations incorrectes.	La page Admin est chargée.	1. Entrer des identifiants valides. 2. Cliquer sur Connexion.	Accès au tableau de bord administrateur.	verifyElement, assertalert
12	Vérifier que le bouton d'ajout de plante redirige vers le formulaire d'ajout de plante.	La page Admin est chargée.	1. Entrer un nom d'utilisateur ou mot de passe invalide. 2. Cliquer sur Connexion.	Affichage d'un message "Invalidcredentials!".	assertAlert
13	Vérifier que la suppression d'une plante fonctionne correctement.	Page Admin chargée.	1. Cliquer sur le bouton Ajouter une plante.	Redirection vers la page de formulaire d'ajout de plante.	assertElementPresent
14	Vérifier que la modification d'une plante redirige vers le formulaire de modification.	Page Admin chargée.	1. Cliquer sur le bouton de suppression pour une plante spécifique. 2. Vérifier que la plante a été supprimée de la liste.	La plante est supprimée de la liste.	verifyElementNotPresent
15	Vérifier que l'utilisateur peut se déconnecter avec succès.	Page Admin chargée.	1. Cliquer sur le bouton Editer pour une plante spécifique.	Redirection vers la page de formulaire de modification de plante.	assertElementPresent
16		Page Admin chargée.	1. Cliquer sur le bouton Logout.	Redirection vers la page de connexion.	assertElementPresent

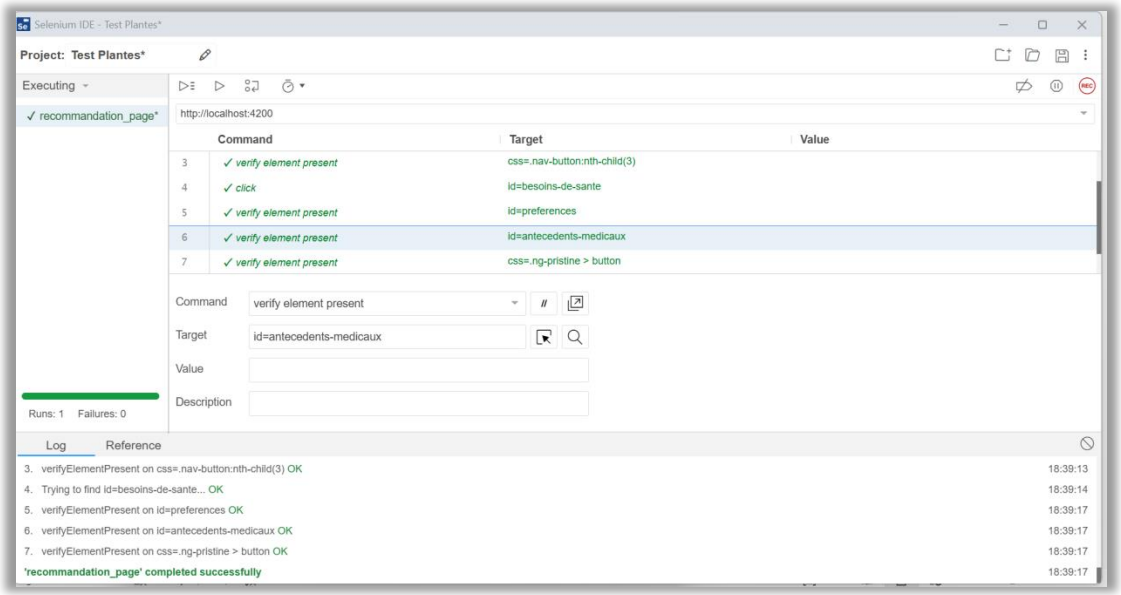
✓ Résultat du Scénario de Test (1) :



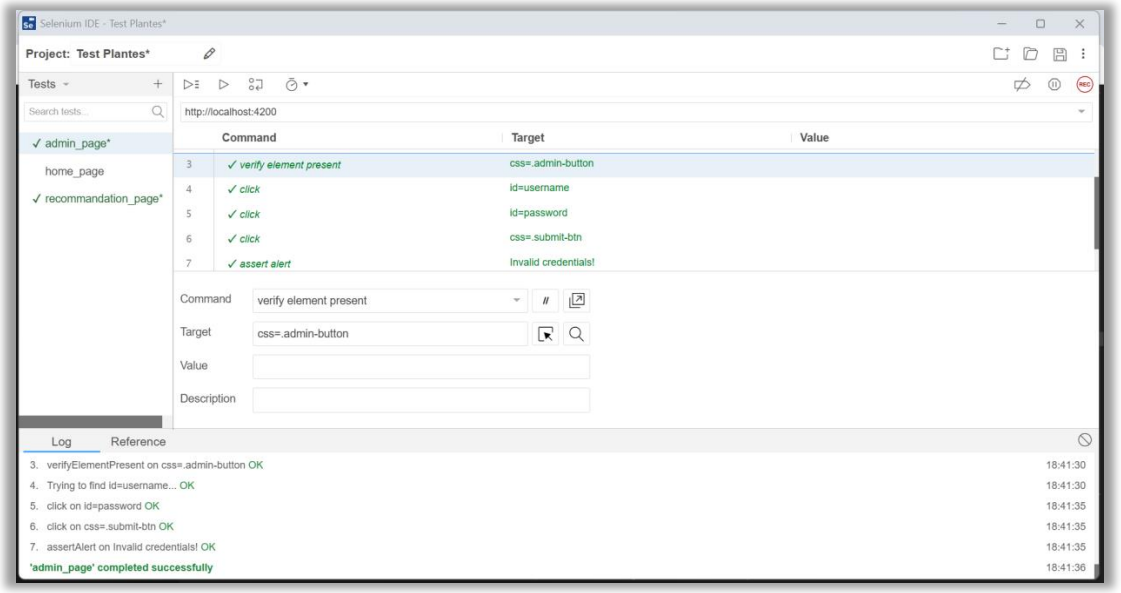
✓ Résultat du Scénario de Test (2) :

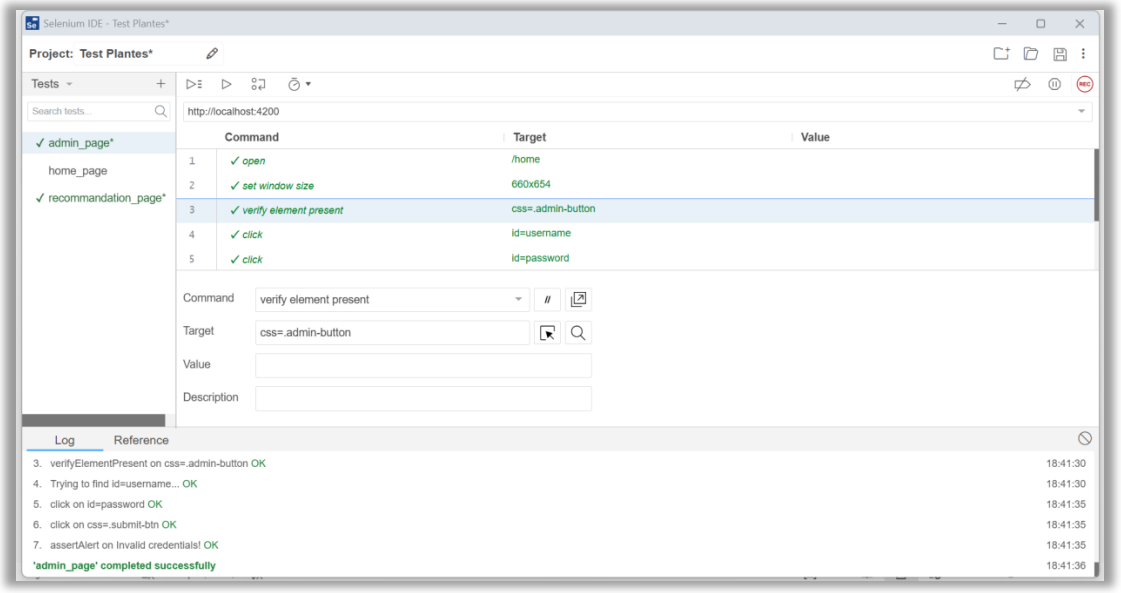


✓ Résultat du Scénario de Test (3) :

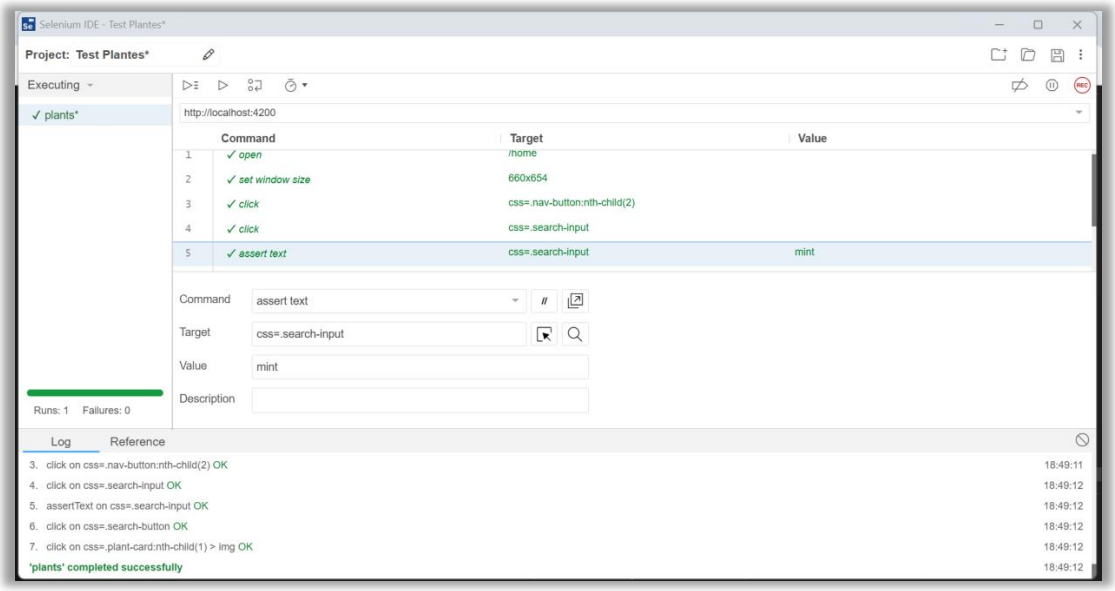


✓ **Résultat du Scénario de Test (4) :**

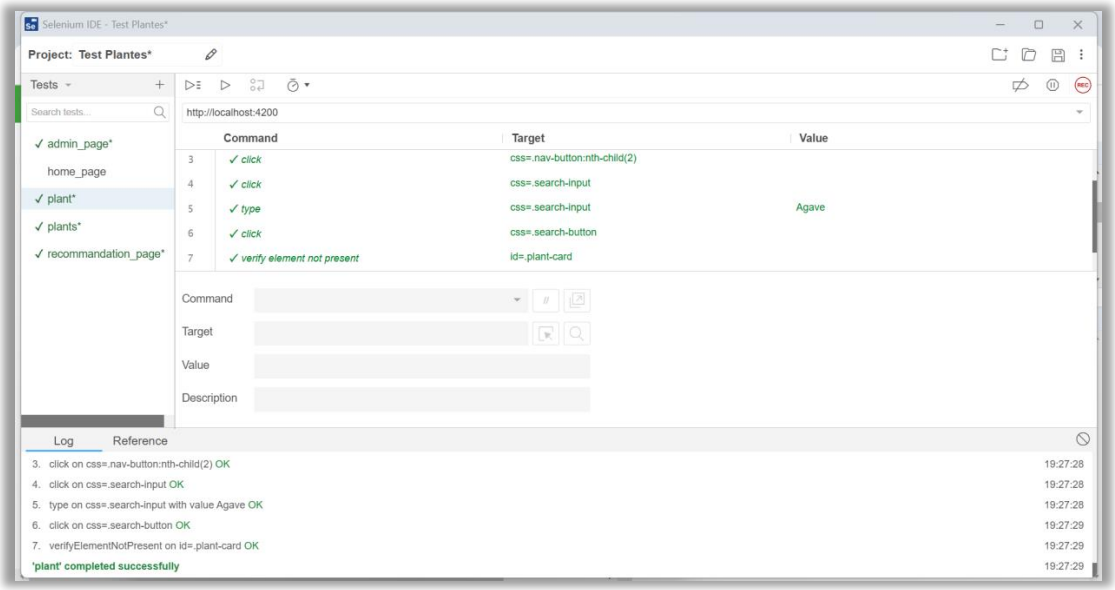




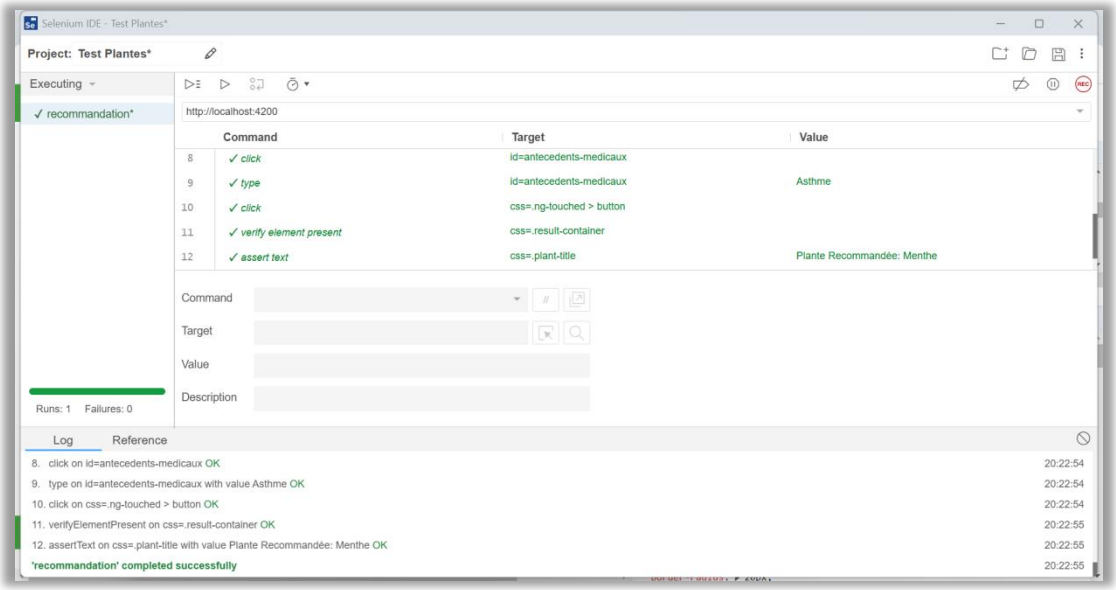
✓ **Résultat du Scénario de Test (5) :**



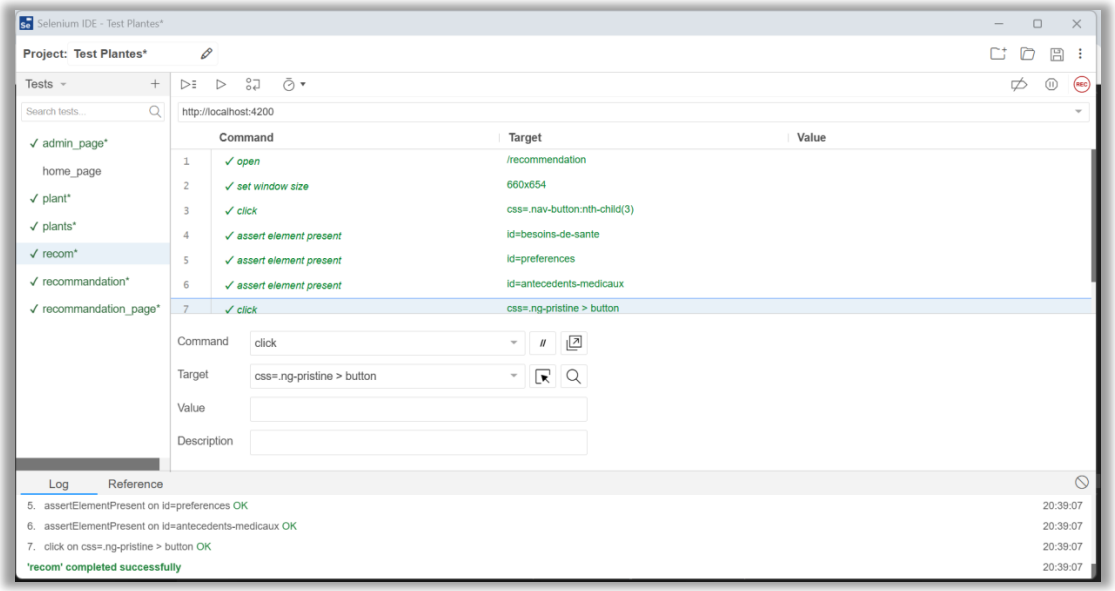
✓ Résultat du Scénario de Test (6) :



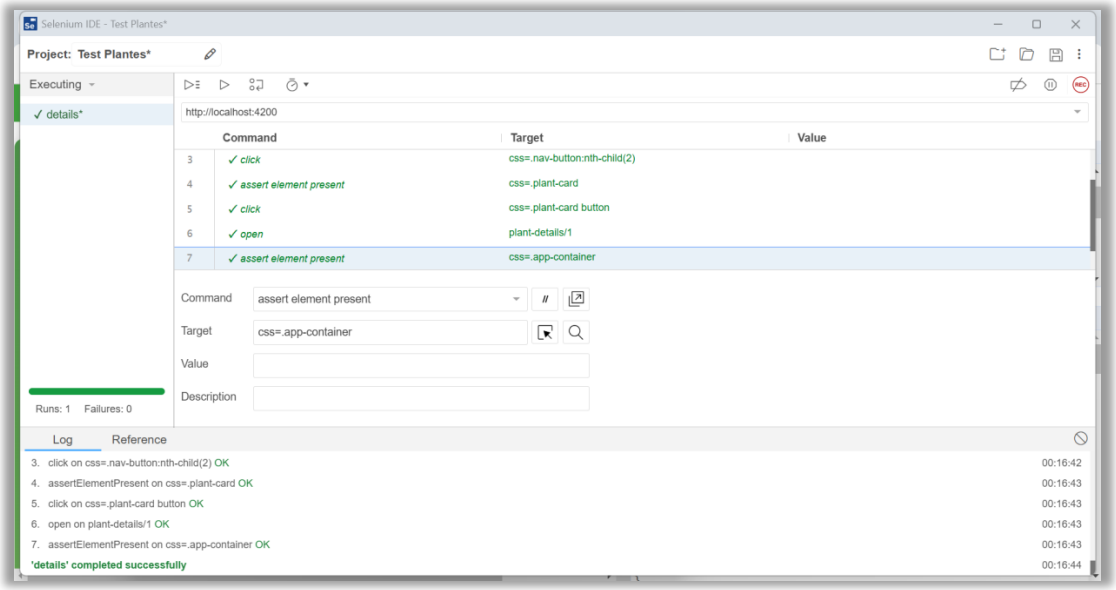
✓ Résultat du Scénario de Test (7) :



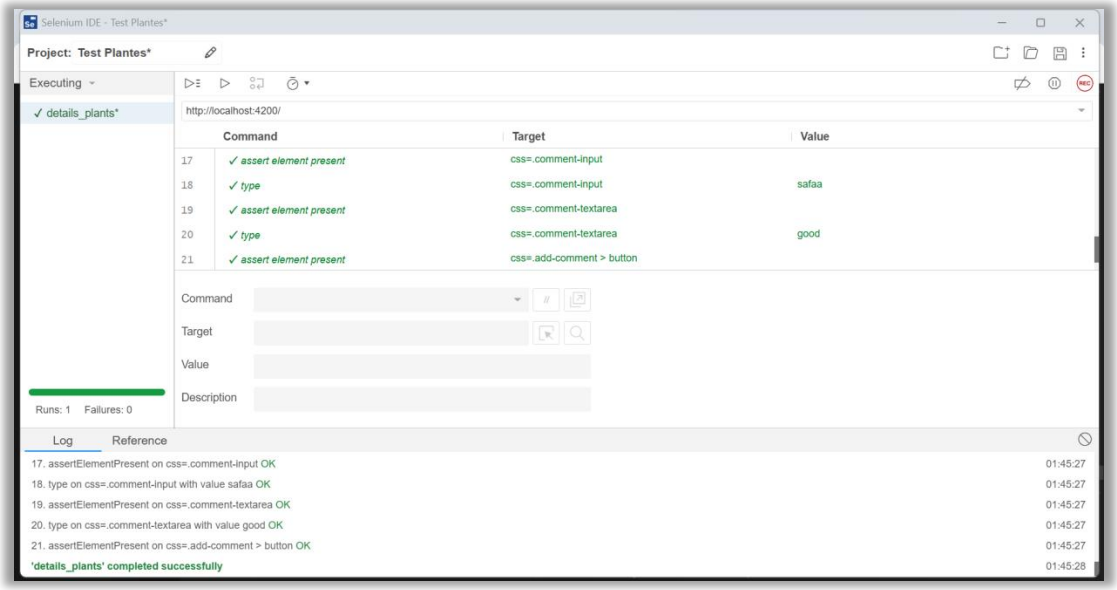
✓ Résultat du Scénario de Test (8) :



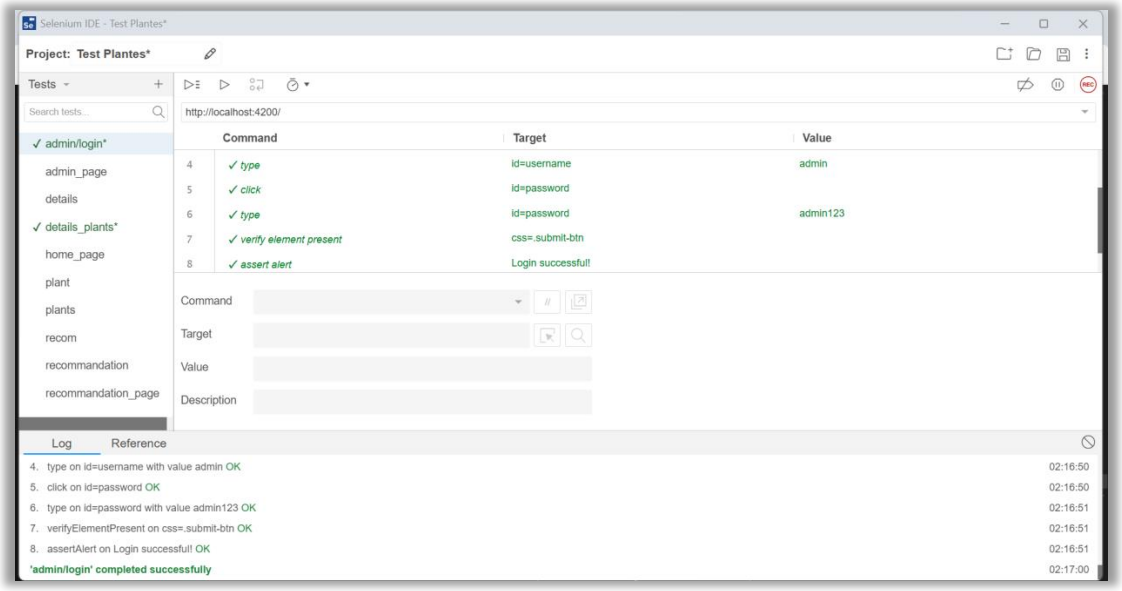
✓ Résultat du Scénario de Test (9) :



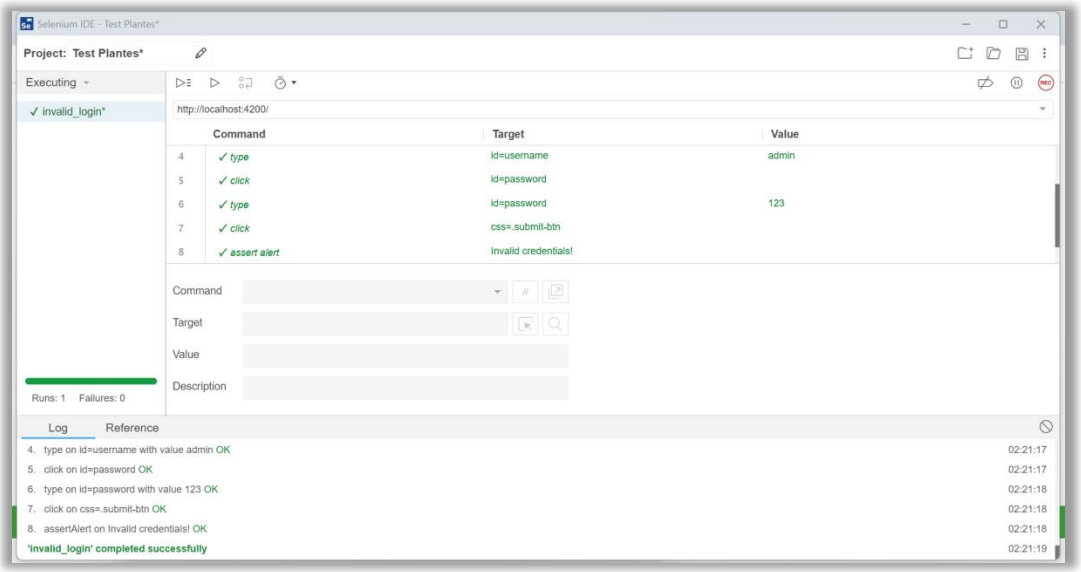
✓ Résultat du Scénario de Test (10) :



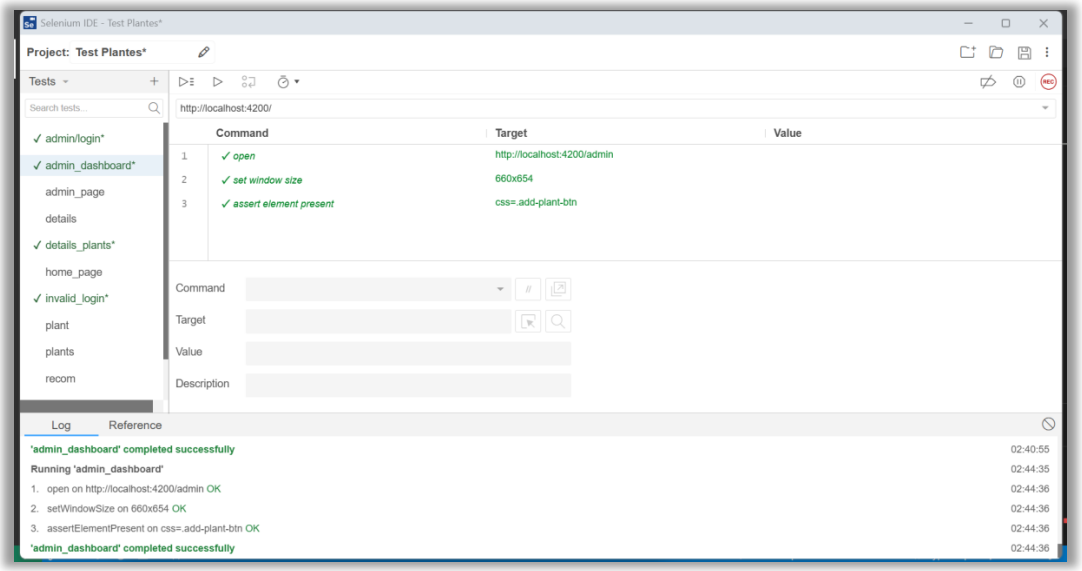
✓ **Résultat du Scénario de Test (11) :**



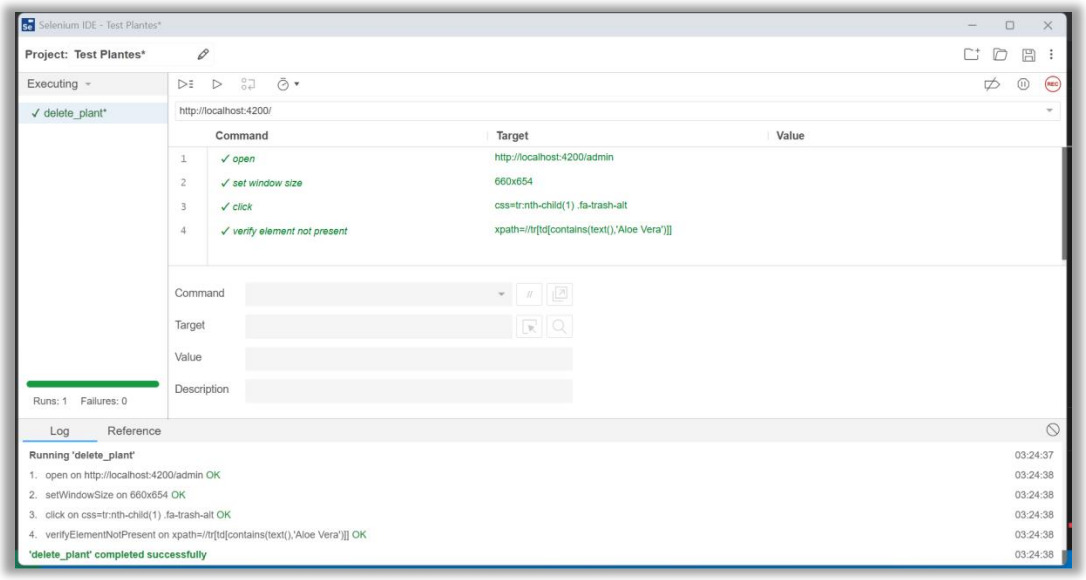
✓ **Résultat du Scénario de Test (12) :**



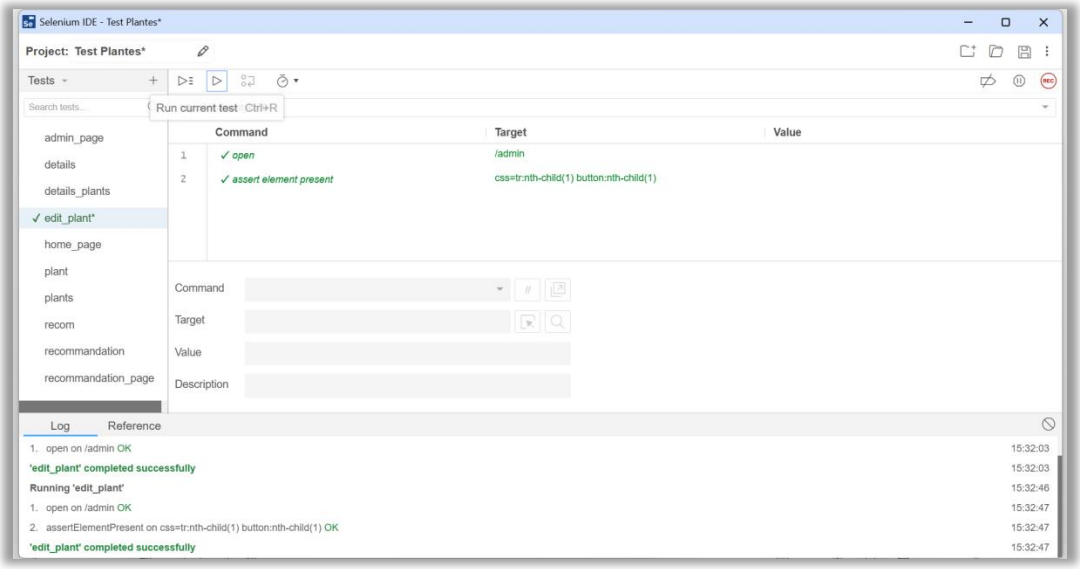
✓ **Résultat du Scénario de Test (13) :**



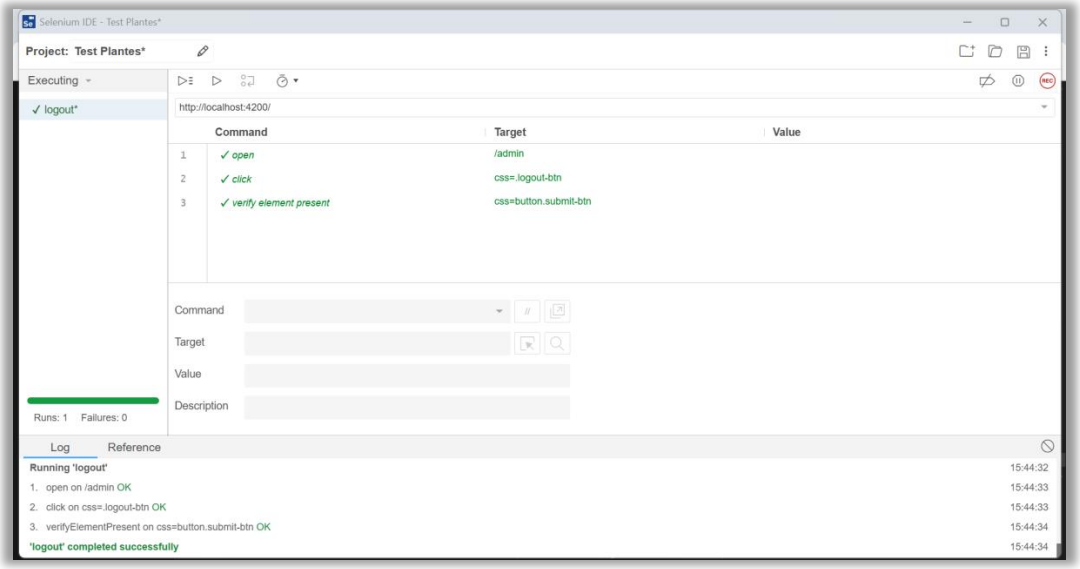
✓ **Résultat du Scénario de Test (14) :**



✓ **Résultat du Scénario de Test (15) :**



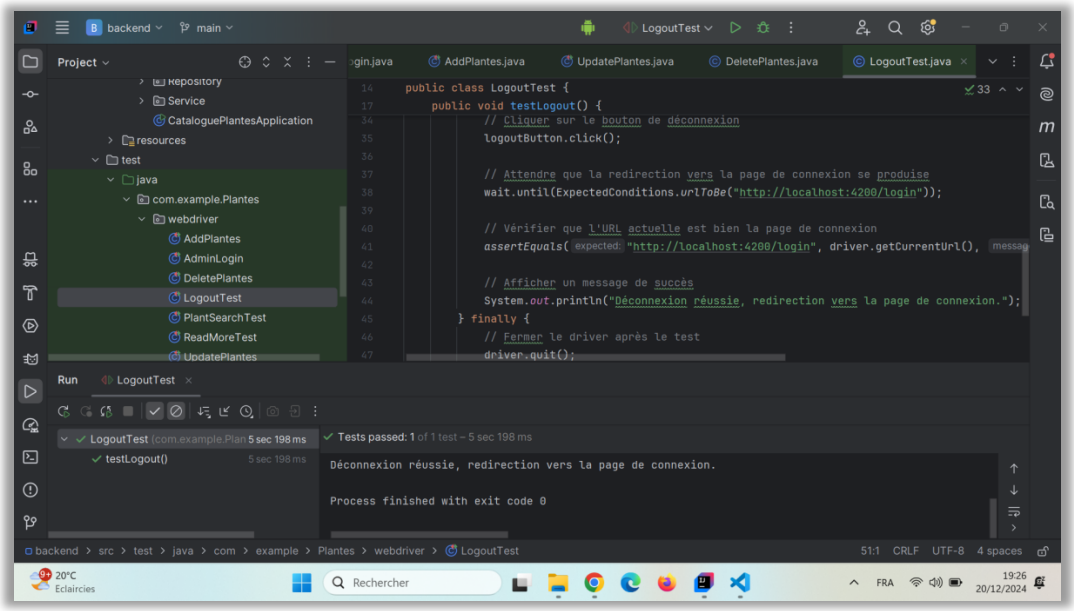
✓ **Résultat du Scénario de Test (16) :**

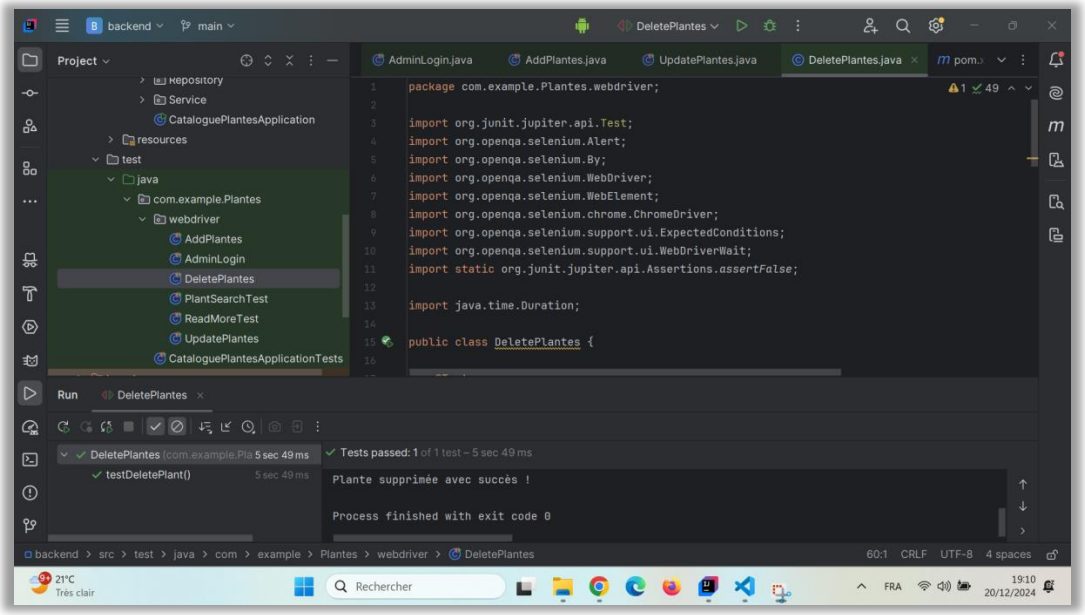


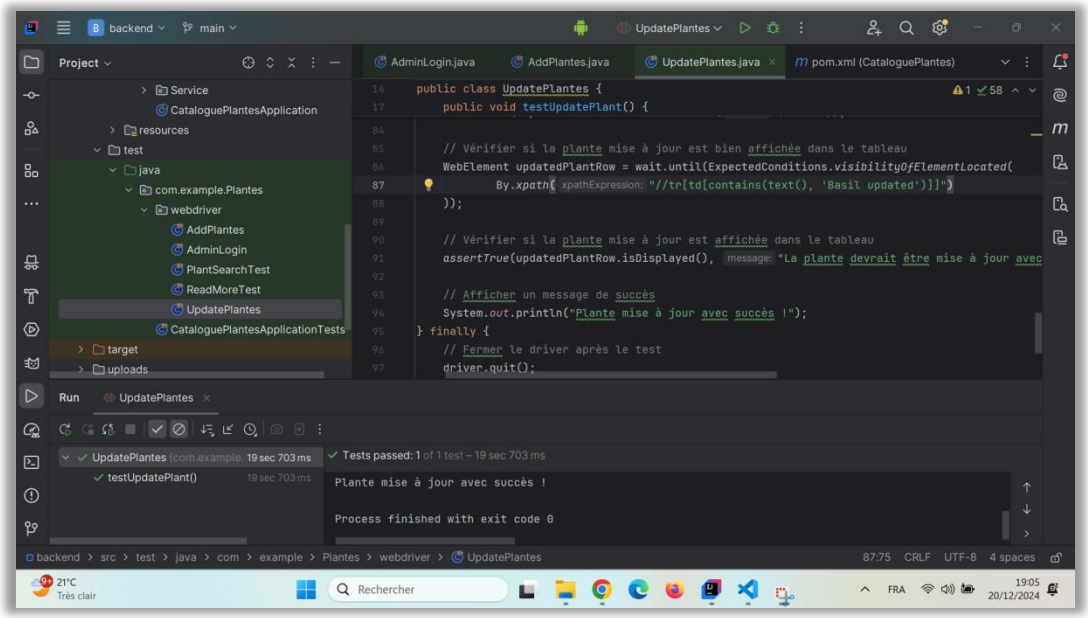
➢ Tests d'Automatisation avec Selenium WebDriver et JUnit :

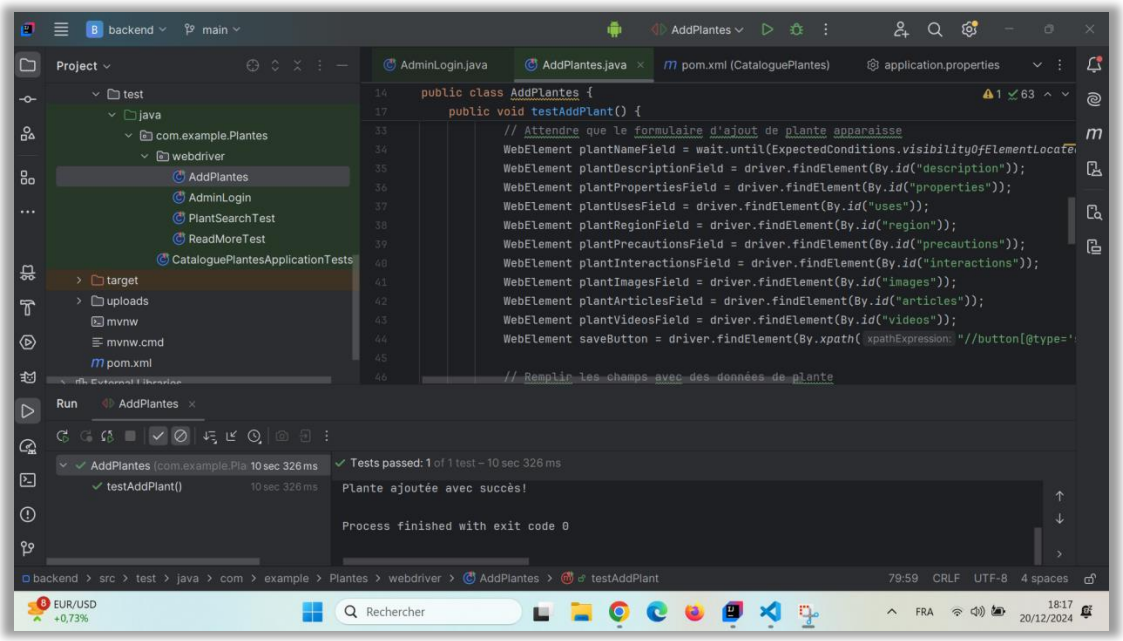
Test Cases with Methods Used :

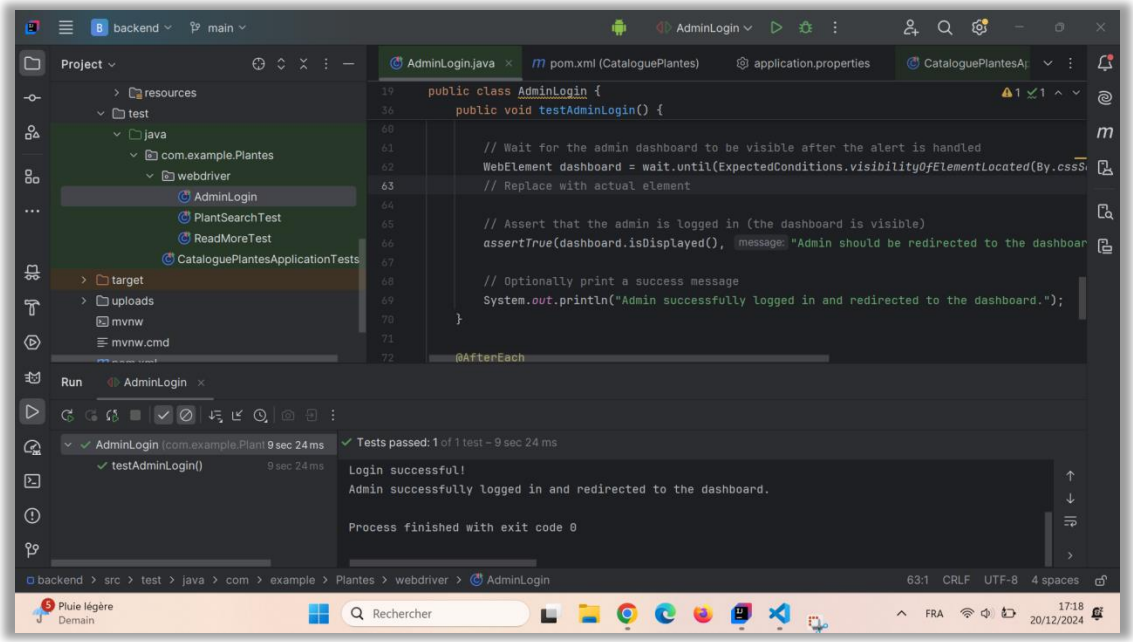
Test Case	Scenario	Steps	Method Used
Logout	Redirect to login page	1. The user is on the admin page (/admin). 2. Click on the 'Logout' button.	driver.findElement(By.xpath("xpath")).click()
Admin Login	Successful login	1. The user is on the login page (/login). 2. Enter valid credentials. 3. Click on 'Login'.	driver.findElement(By.id("username")) driver.findElement(By.id("password")) driver.findElement(By.xpath("xpath")).click()
Add Plant	Add a valid plant	1. The user is on the plant addition page (/admin). 2. Fill in all the required fields. 3. Click on 'Add'.	driver.findElement(By.id("plant-name")).sendKeys("PlantName") driver.findElement(By.id("plant-description")).sendKeys("PlantDescription") driver.findElement(By.xpath("xpath")).click()
Update Plant	Update an existing plant	1. The user is on the admin page (/admin). 2. Select a plant to edit. 3. Modify the required fields. 4. Click on 'Save'.	driver.findElement(By.xpath("xpath")).click() driver.findElement(By.id("edit-name")).sendKeys("UpdatedPlantName") driver.findElement(By.id("edit-description")).clear() driver.findElement(By.xpath("xpath")).click()
Delete Plant	Delete an existing plant	1. The user is on the admin page (/admin). 2. Select a plant to delete. 3. Click on 'Delete'.	driver.findElement(By.xpath("xpath")).click()
Read More	View detailed plant information	1. The user is on the recommendations page (/recommendations). 2. Click on 'Read More' for a specific plant.	driver.findElement(By.xpath("xpath")).click() driver.findElement(By.xpath("xpath")).getText()
Plant Search	Search for an existing plant	1. The user is on the admin page (/admin) with a search bar visible. 2. Enter a valid keyword. 3. Click on 'Search'.	driver.findElement(By.id("search-bar")).sendKeys("PlantName") driver.findElement(By.xpath("xpath")).click() driver.findElement(By.xpath("xpath")).getText()
Recommandations	Recommandation pour "Skin Care	1. Aller à `/recommendations`. 2. Remplir le champ "Besoins de Santé" avec "Skin Care". 3. Cliquer sur "Get Recommendations".	driver.get(URL) new WebDriverWait(driver, Duration.ofSeconds(20)) driver.findElement(By.cssSelector("button[type='submit']"))

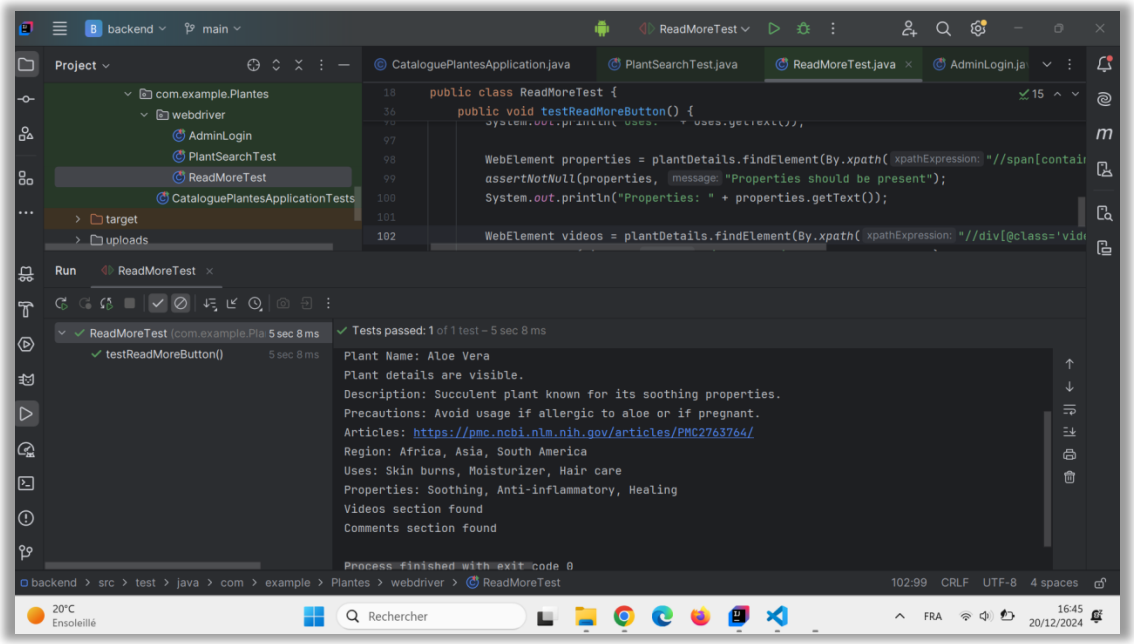

backend
main
LogoutTest
Project
AddPlantes.java
UpdatePlantes.java
DeletePlantes.java
LogoutTest.java
Repository
Service
CataloguePlantesApplication
resources
test
java
com.example.Plantes
webdriver
AddPlantes
AdminLogin
DeletePlantes
LogoutTest
PlantSearchTest
ReadMoreTest
UpdatePlantes
public class LogoutTest {
public void testLogout() {
// Cliquer sur le bouton de déconnexion
logoutButton.click();
// Attendre que la redirection vers la page de connexion se produise
wait.until(ExpectedConditions.urlToBe("http://localhost:4200/login"));
// Vérifier que l'URL actuelle est bien la page de connexion
assertEquals(expected: "http://localhost:4200/login", driver.getCurrentUrl(),
// Afficher un message de succès
System.out.println("Déconnexion réussie, redirection vers la page de connexion.");
} finally {
// Fermer le driver après le test
driver.quit();
Run
LogoutTest
LogoutTest (com.example.Plan 5 sec 198 ms
testLogout() 5 sec 198 ms
Tests passed: 1 of 1 test – 5 sec 198 ms
Déconnexion réussie, redirection vers la page de connexion.
Process finished with exit code 0
backend > src > test > java > com > example > Plantes > webdriver > LogoutTest
51:1 CRLF UTF-8 4 spaces
20°C
Éclaircies
Rechercher
FRA
19:26
20/12/2024

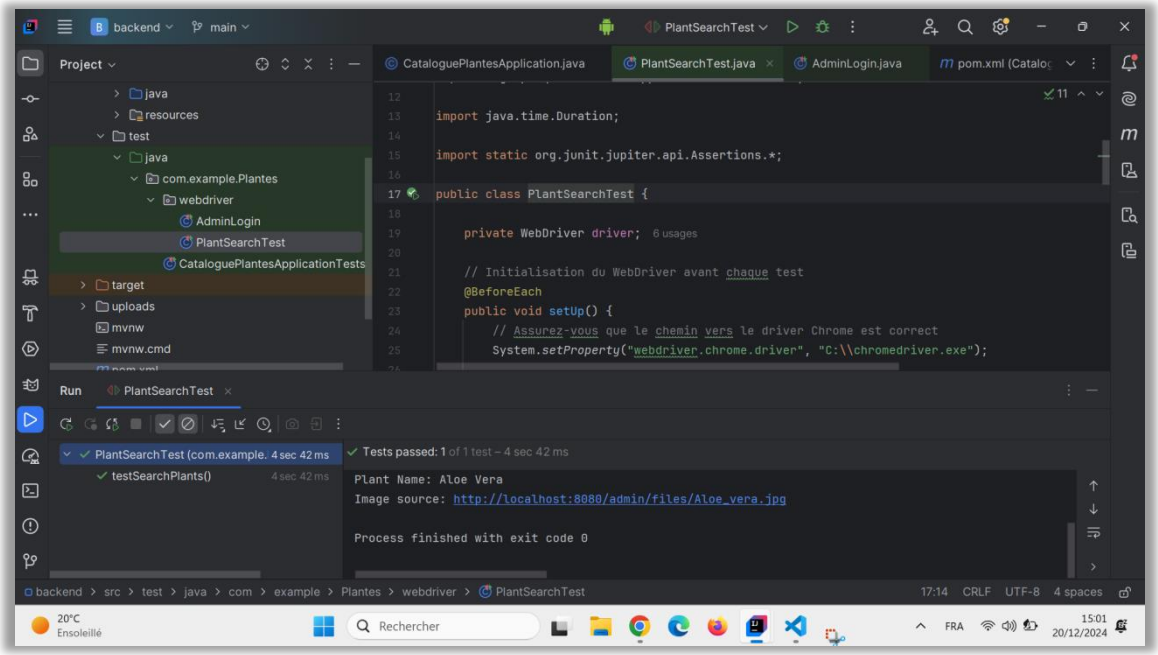

backend
main
DeletePlantes
Project
AdminLogin.java
AddPlantes.java
UpdatePlantes.java
DeletePlantes.java
pom.
Repository
Service
CataloguePlantesApplication
resources
test
java
com.example.Plantes
webdriver
AddPlantes
AdminLogin
DeletePlantes
PlantSearchTest
ReadMoreTest
UpdatePlantes
CataloguePlantesApplicationTests
package com.example.Plantes.webdriver;
import org.junit.jupiter.api.Test;
import org.openqa.selenium.Alert;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import static org.junit.jupiter.api.Assertions.assertFalse;
import java.time.Duration;
public class DeletePlantes {
Run
DeletePlantes
DeletePlantes (com.example.Pla 5 sec 49 ms
testDeletePlant() 5 sec 49 ms
Tests passed: 1 of 1 test – 5 sec 49 ms
Plante supprimée avec succès !
Process finished with exit code 0
backend > src > test > java > com > example > Plantes > webdriver > DeletePlantes
60:1 CRLF UTF-8 4 spaces
21°C
Très clair
Rechercher
FRA
19:10
20/12/2024

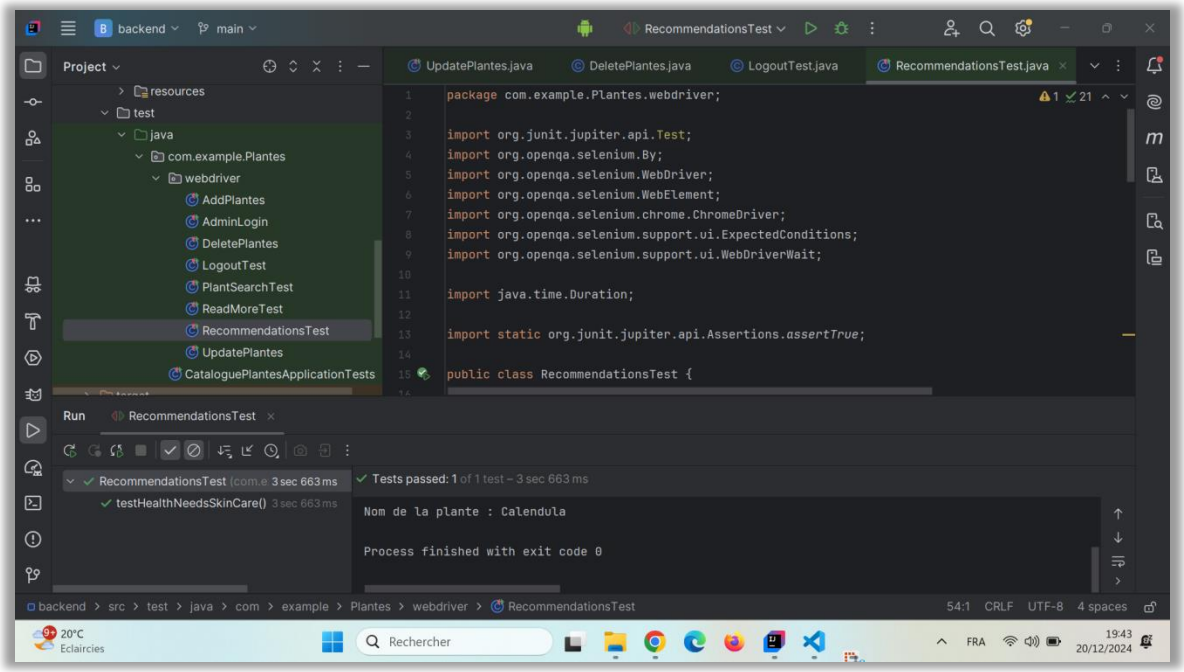

backend
main
UpdatePlantes
Project
AdminLogin.java
AddPlantes.java
UpdatePlantes.java
pom.xml (CataloguePlantes)
public class UpdatePlantes {
public void testUpdatePlant() {
// Vérifier si la plante mise à jour est bien affichée dans le tableau
WebElement updatedPlantRow = wait.until(ExpectedConditions.visibilityOfElementLocated(
By.xpath(xpathExpression: "//tr[td[contains(text(), 'Basil updated')]]")
));
// Vérifier si la plante mise à jour est affichée dans le tableau
assertTrue(updatedPlantRow.isDisplayed(), message: "La plante devrait être mise à jour avec
// Afficher un message de succès
System.out.println("Plante mise à jour avec succès !");
} finally {
// Fermer le driver après le test
driver.quit();
Service
CataloguePlantesApplication
resources
test
java
com.example.Plantes
webdriver
AddPlantes
AdminLogin
PlantSearchTest
ReadMoreTest
UpdatePlantes
CataloguePlantesApplicationTests
target
uploads
Run
UpdatePlantes
Tests passed: 1 of 1 test – 19 sec 703 ms
Plante mise à jour avec succès !
Process finished with exit code 0
backend > src > test > java > com > example > Plantes > webdriver > UpdatePlantes
87:75 CRLF UTF-8 4 spaces
Rechercher
FRA
19:05
20/12/2024


backend
main
AddPlantes
Project
AdminLogin.java
AddPlantes.java
pom.xml (CataloguePlantes)
application.properties
public class AddPlantes {
public void testAddPlant() {
// Attendre que le formulaire d'ajout de plante apparaisse
WebElement plantNameField = wait.until(ExpectedConditions.visibilityOfElementLocate
WebElement plantDescriptionField = driver.findElement(By.id("description"));
WebElement plantPropertiesField = driver.findElement(By.id("properties"));
WebElement plantUsesField = driver.findElement(By.id("uses"));
WebElement plantRegionField = driver.findElement(By.id("region"));
WebElement plantPrecautionsField = driver.findElement(By.id("precautions"));
WebElement plantInteractionsField = driver.findElement(By.id("interactions"));
WebElement plantImagesField = driver.findElement(By.id("images"));
WebElement plantArticlesField = driver.findElement(By.id("articles"));
WebElement plantVideosField = driver.findElement(By.id("videos"));
WebElement saveButton = driver.findElement(By.xpath(xpathExpression: "//button[@type='
// Remplir les champs avec des données de plante
test
java
com.example.Plantes
webdriver
AddPlantes
AdminLogin
PlantSearchTest
ReadMoreTest
CataloguePlantesApplicationTests
target
uploads
mvnw
mvnw.cmd
pom.xml
Run
AddPlantes
Tests passed: 1 of 1 test – 10 sec 326 ms
Plante ajoutée avec succès!
Process finished with exit code 0
backend > src > test > java > com > example > Plantes > webdriver > AddPlantes > testAddPlant
79:59 CRLF UTF-8 4 spaces
EUR/USD
+0,73%
Rechercher
FRA
18:17
20/12/2024


backend
main
AdminLogin
Project
AdminLogin.java
pom.xml (CataloguePlantes)
application.properties
public class AdminLogin {
public void testAdminLogin() {
// Wait for the admin dashboard to be visible after the alert is handled
WebElement dashboard = wait.until(ExpectedConditions.visibilityOfElementLocated(By.cssS
// Replace with actual element
// Assert that the admin is logged in (the dashboard is visible)
assertTrue(dashboard.isDisplayed(), message: "Admin should be redirected to the dashboar
// Optionally print a success message
System.out.println("Admin successfully logged in and redirected to the dashboard.");
}
@AfterEach
resources
test
java
com.example.Plantes
webdriver
AdminLogin
PlantSearchTest
ReadMoreTest
CataloguePlantesApplicationTests
target
uploads
mvnw
mvnw.cmd
Run
AdminLogin
Tests passed: 1 of 1 test – 9 sec 24 ms
Login successful!
Admin successfully logged in and redirected to the dashboard.
Process finished with exit code 0
backend > src > test > java > com > example > Plantes > webdriver > AdminLogin
63:1 CRLF UTF-8 4 spaces
Pluie légère
Demain
Rechercher
FRA
17:18
20/12/2024


backend
main
ReadMoreTest
Project
CataloguePlantesApplication.java
PlantSearchTest.java
ReadMoreTest.java
AdminLogin.ja
com.example.Plantes
webdriver
AdminLogin
PlantSearchTest
ReadMoreTest
CataloguePlantesApplicationTests
target
uploads
public class ReadMoreTest {
public void testReadMoreButton() {
WebElement properties = plantDetails.findElement(By.xpath(xpathExpression: "//span[contair
assertNotNull(properties, message: "Properties should be present");
System.out.println("Properties: " + properties.getText());
WebElement videos = plantDetails.findElement(By.xpath(xpathExpression: "//div[@class='vid
Run
ReadMoreTest
Tests passed: 1 of 1 test – 5 sec 8 ms
testReadMoreButton() 5 sec 8 ms
Plant Name: Aloe Vera
Plant details are visible.
Description: Succulent plant known for its soothing properties.
Precautions: Avoid usage if allergic to aloe or if pregnant.
Articles: https://pmc.ncbi.nlm.nih.gov/articles/PMC2763764/
Region: Africa, Asia, South America
Uses: Skin burns, Moisturizer, Hair care
Properties: Soothing, Anti-inflammatory, Healing
Videos section found
Comments section found
Process finished with exit code 0
backend › src › test › java › com › example › Plantes › webdriver › ReadMoreTest
102:99 CRLF UTF-8 4 spaces
20°C
Ensoleillé
Rechercher
FRA
16:45
20/12/2024


backend
main
PlantSearchTest
Project
CataloguePlantesApplication.java
PlantSearchTest.java
AdminLogin.java
pom.xml (Catalo
java
resources
test
java
com.example.Plantes
webdriver
AdminLogin
PlantSearchTest
CataloguePlantesApplicationTests
target
uploads
mvnw
mvnw.cmd
import java.time.Duration;
import static org.junit.jupiter.api.Assertions.*;
public class PlantSearchTest {
private WebDriver driver; 6 usages
// Initialisation du WebDriver avant chaque test
@BeforeEach
public void setUp() {
// Assurez-vous que le chemin vers le driver Chrome est correct
System.setProperty("webdriver.chrome.driver", "C:\\chromedriver.exe");
Run
PlantSearchTest
Tests passed: 1 of 1 test – 4 sec 42 ms
PlantSearchTest (com.example. 4 sec 42 ms
testSearchPlants() 4 sec 42 ms
Plant Name: Aloe Vera
Image source: http://localhost:8080/admin/files/Aloe_vera.jpg
Process finished with exit code 0
backend › src › test › java › com › example › Plantes › webdriver › PlantSearchTest
17:14 CRLF UTF-8 4 spaces
20°C
Ensoleillé
Rechercher
FRA
15:01
20/12/2024

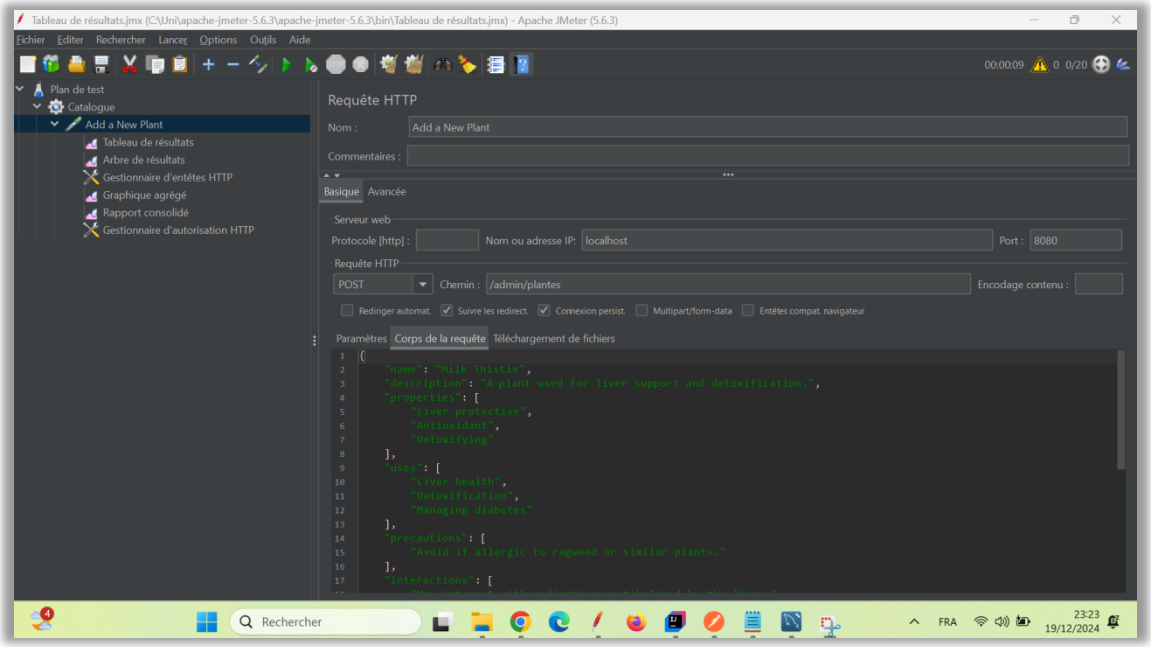

backend
main
RecommendationsTest
Project
UpdatePlantes.java
DeletePlantes.java
LogoutTest.java
RecommendationsTest.java
resources
test
java
com.example.Plantes
webdriver
AddPlantes
AdminLogin
DeletePlantes
LogoutTest
PlantSearchTest
ReadMoreTest
RecommendationsTest
UpdatePlantes
CataloguePlantesApplicationTests
package com.example.Plantes.webdriver;
import org.junit.jupiter.api.Test;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;
import java.time.Duration;
import static org.junit.jupiter.api.Assertions.assertTrue;
public class RecommendationsTest {
Run
RecommendationsTest
Tests passed: 1 of 1 test – 3 sec 663 ms
testHealthNeedsSkinCare() 3 sec 663 ms
Nom de la plante : Calendula
Process finished with exit code 0
backend › src › test › java › com › example › Plantes › webdriver › RecommendationsTest
54:1 CRLF UTF-8 4 spaces
20°C
Eclaircies
Rechercher
FRA
19:43
20/12/2024

Tests de Performance (JMeter)

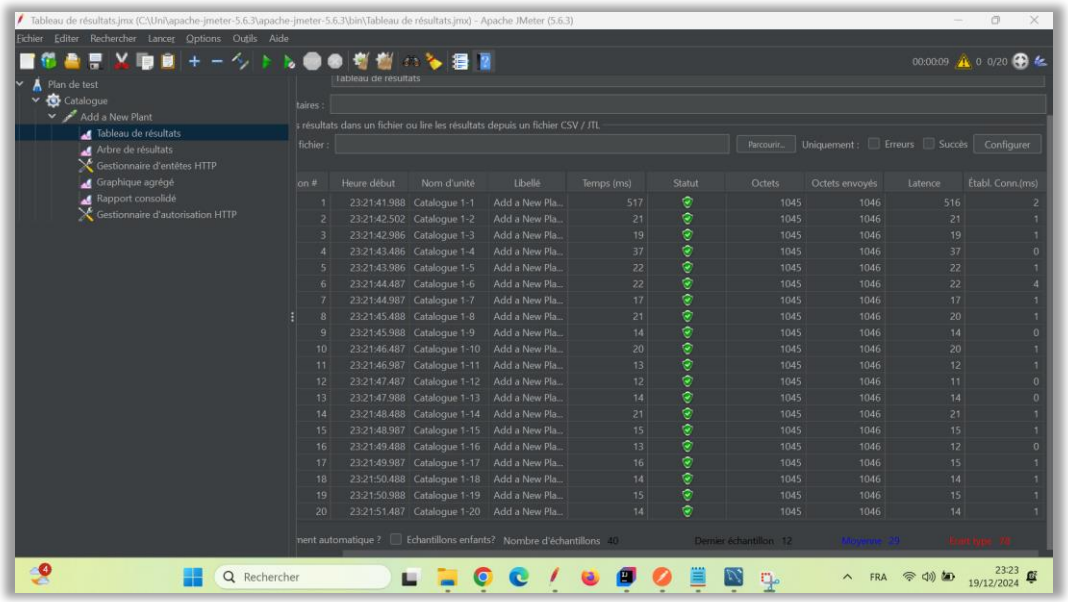
➢ Test Cases with Methods Used

Test Case	Request Type	URL
1. Test Plant Recommendations API	POST	/plantes/recommandations
2. Test Retrieve All Plants API	GET	/plantes
3. Test Add a New Plant	POST	/admin/plantes
4. Test Get Comments for Plant by ID API	GET	/plantes/{id}/commentaires
5. Test File Upload	POST	/admin/files/upload

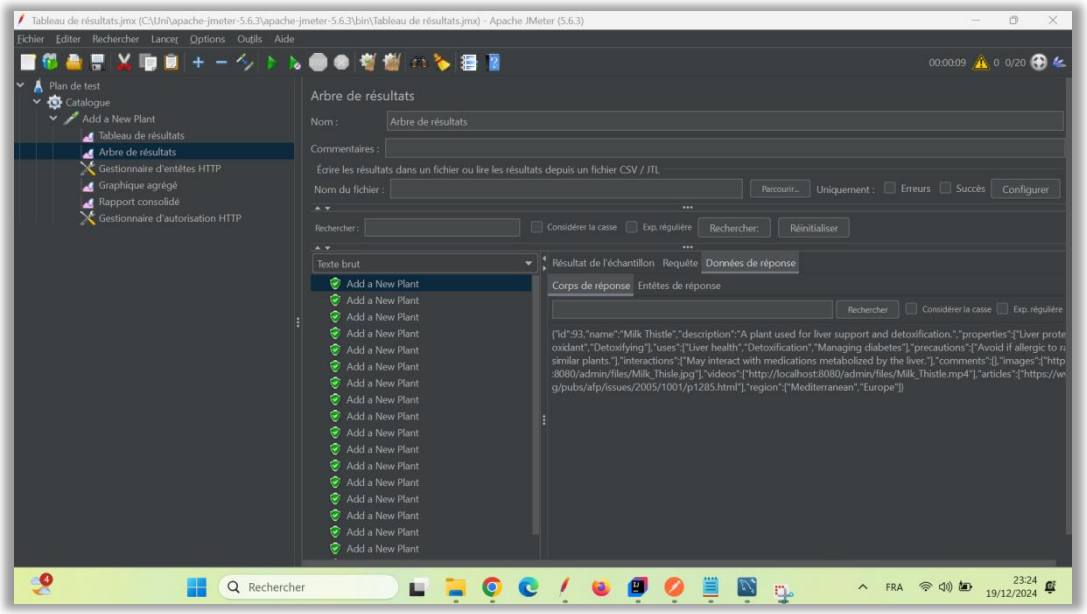
❖ Add New Plant :



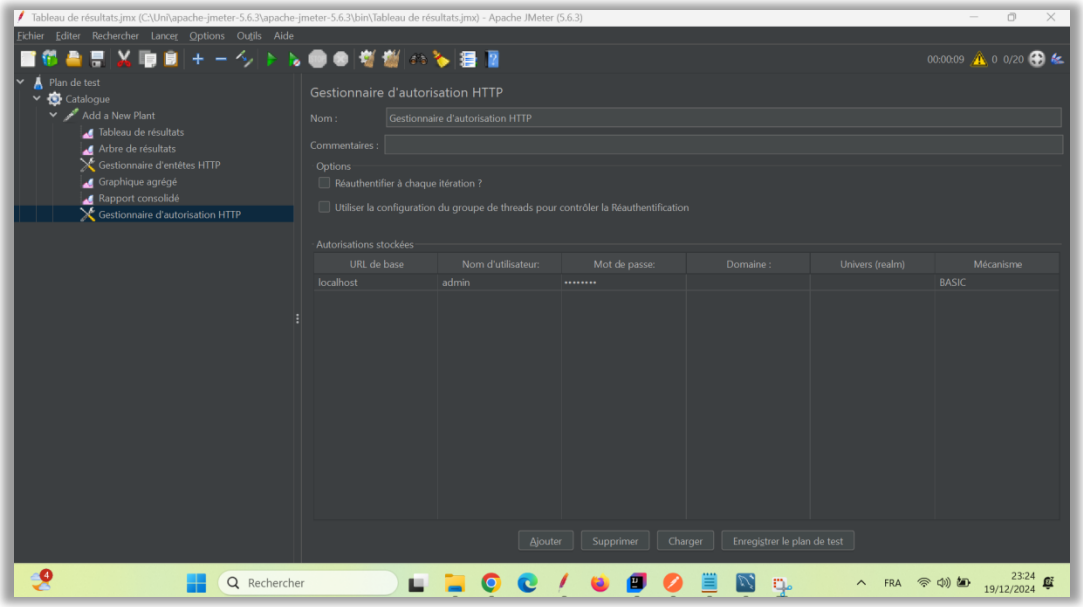
➢ Vue Résultats en Table :



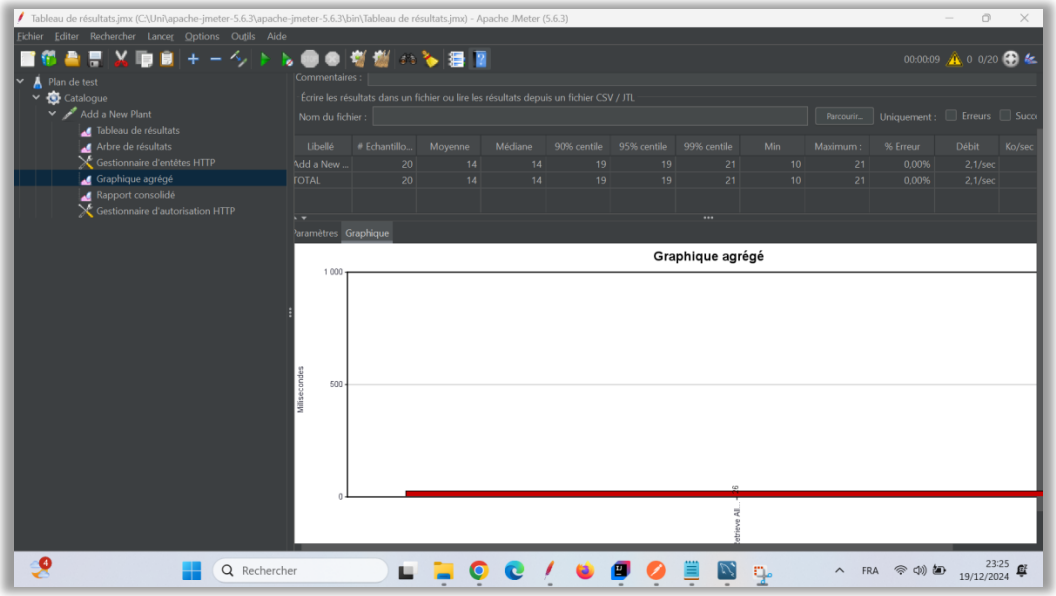
➢ Arbre de Résultats :



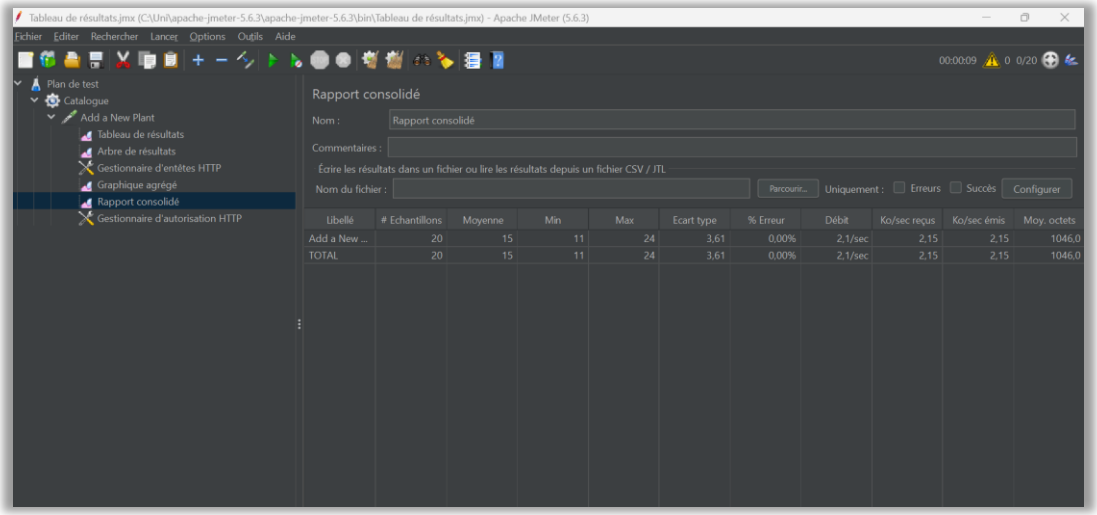
➢ Gestion de l'Autorisation HTTP



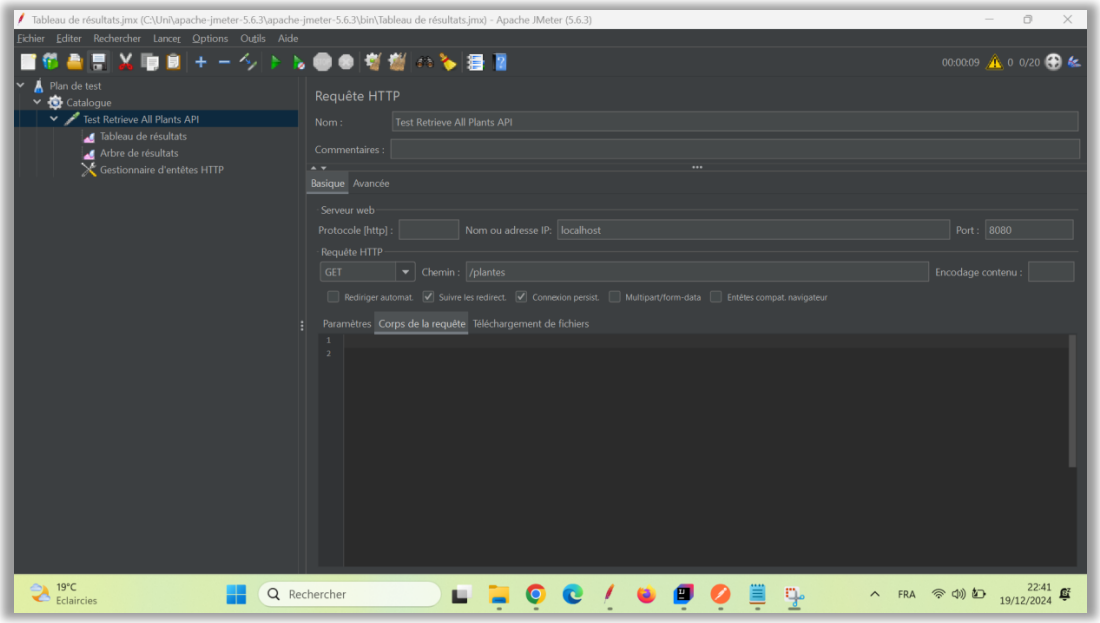
➢ Graphique des Résultats



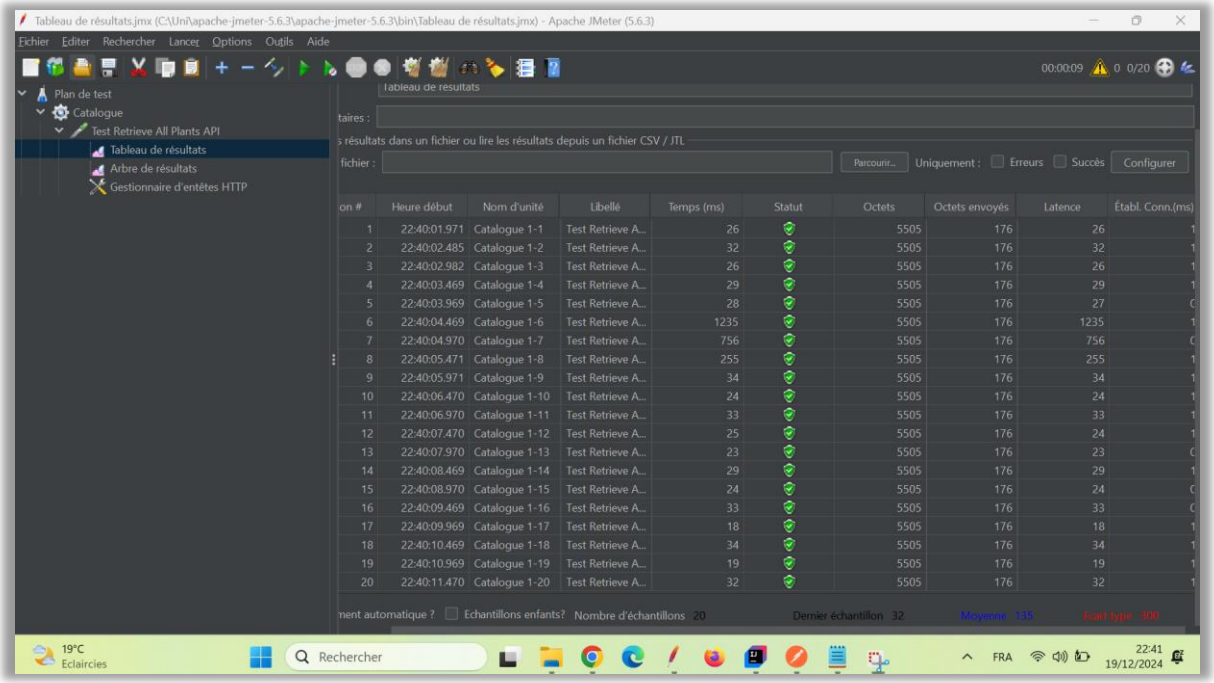
➢ le Rapport Consolidé



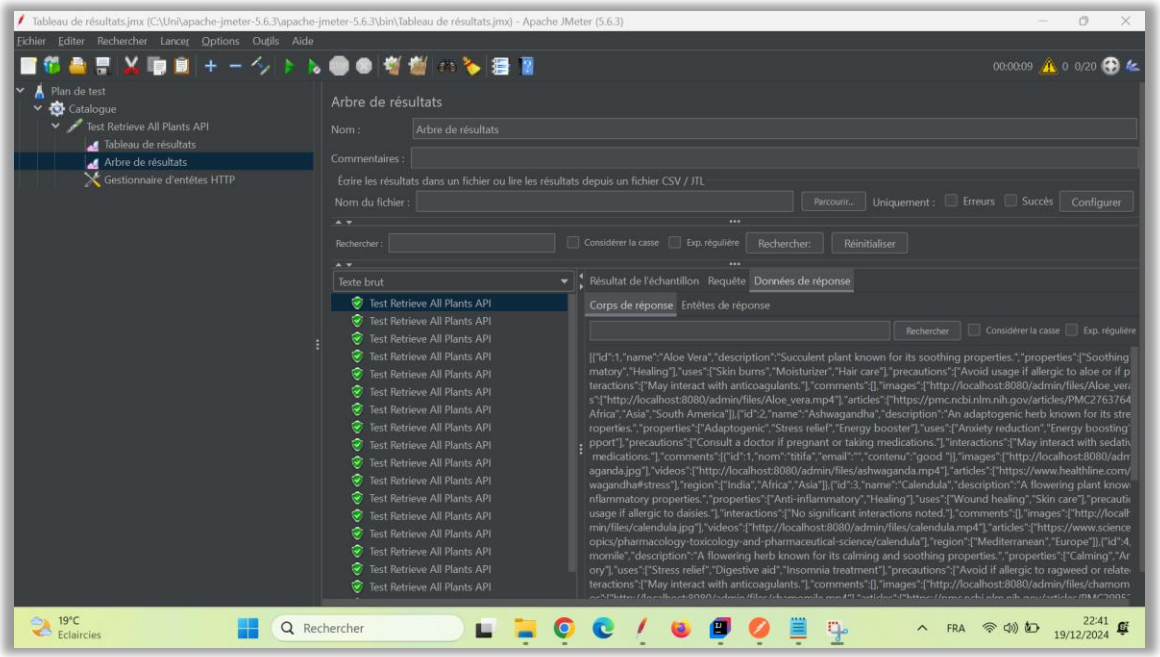
❖ Test Retrieve All Plants API :



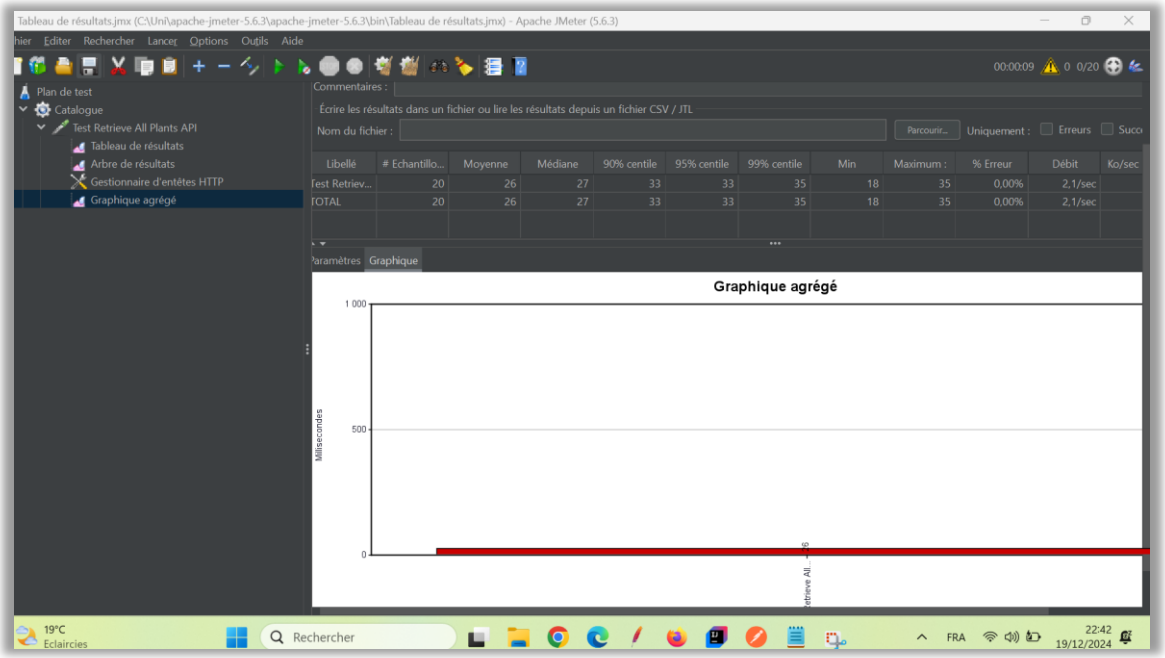
➢ Vue Résultats en Table :



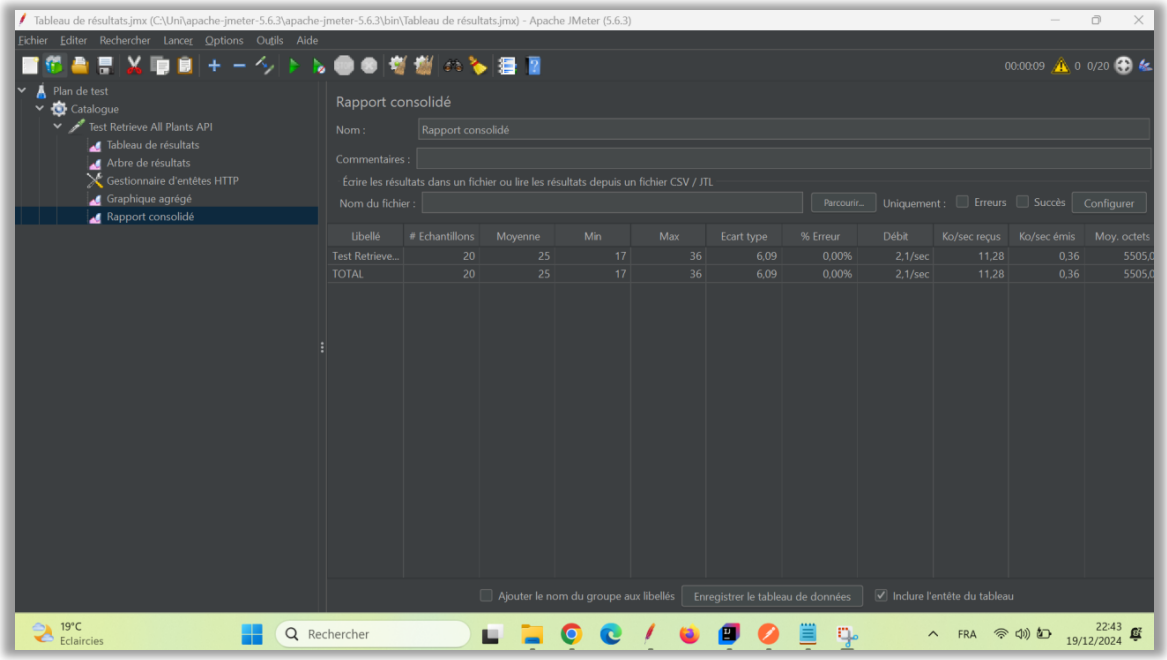
➢ Arbre de Résultats :



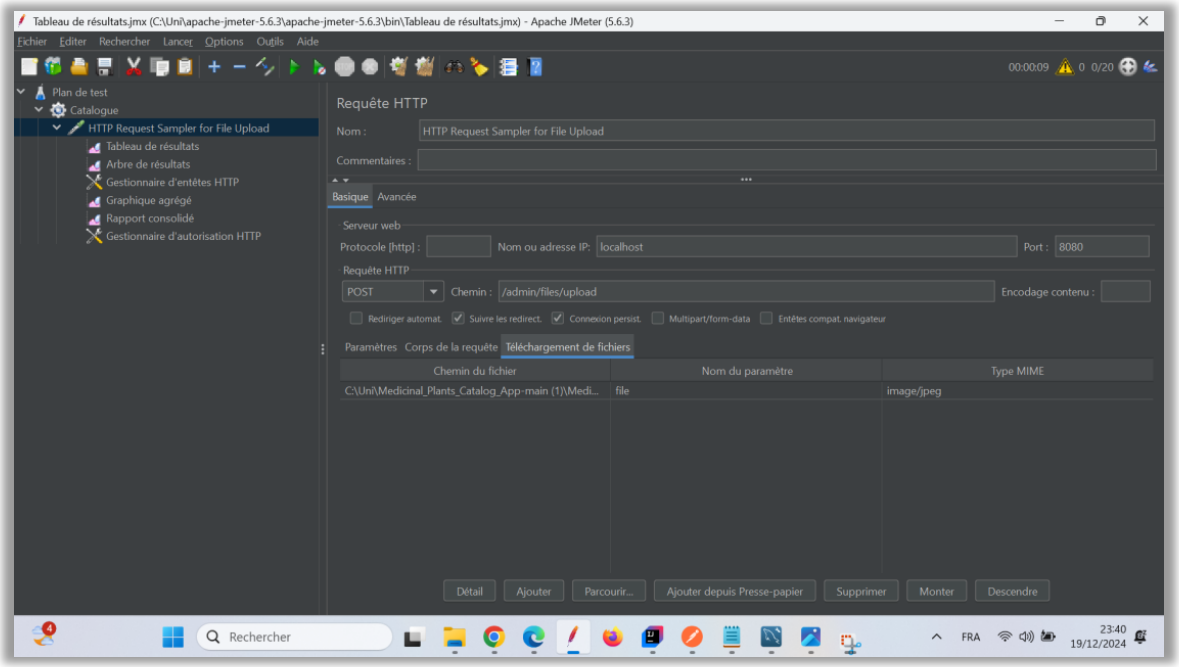
➢ Graphique des Résultats :



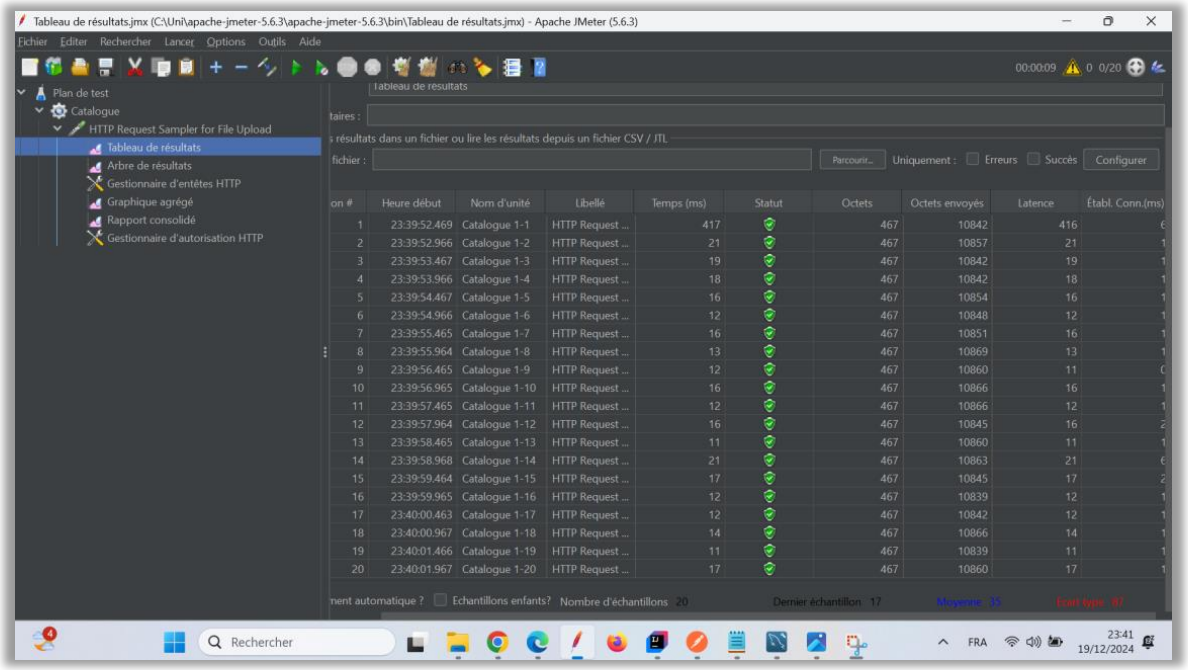
➢ le Rapport Consolidé



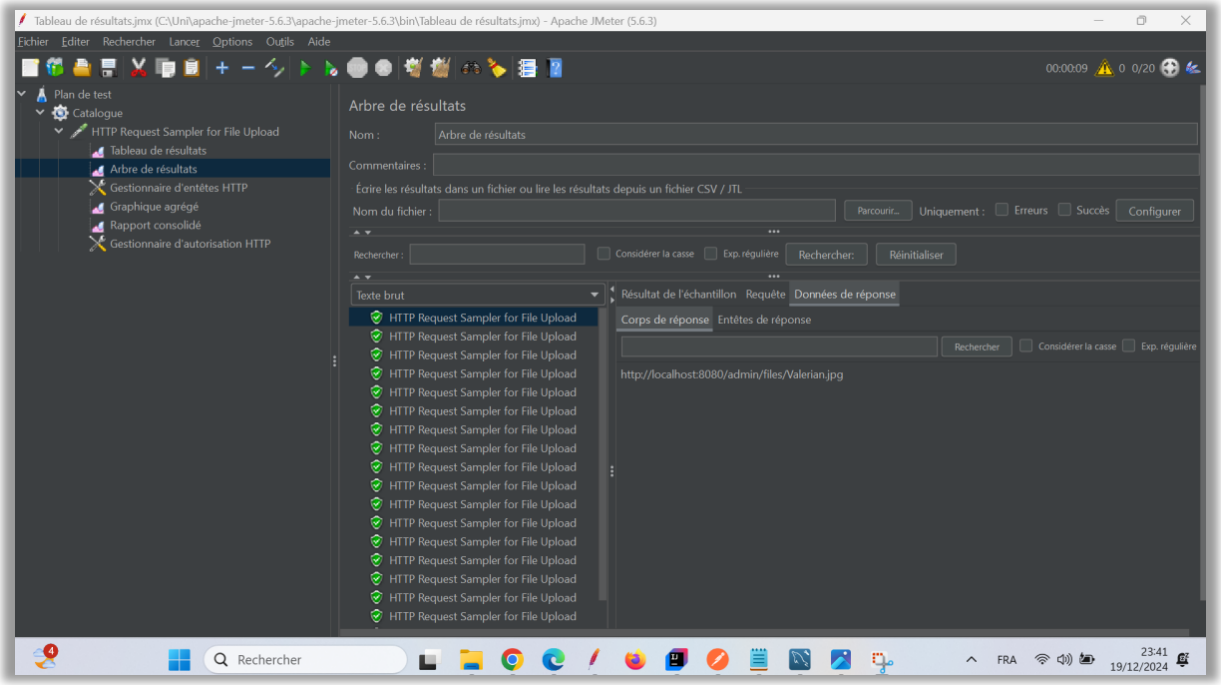
❖ HTTP Request Sampler for File Upload :



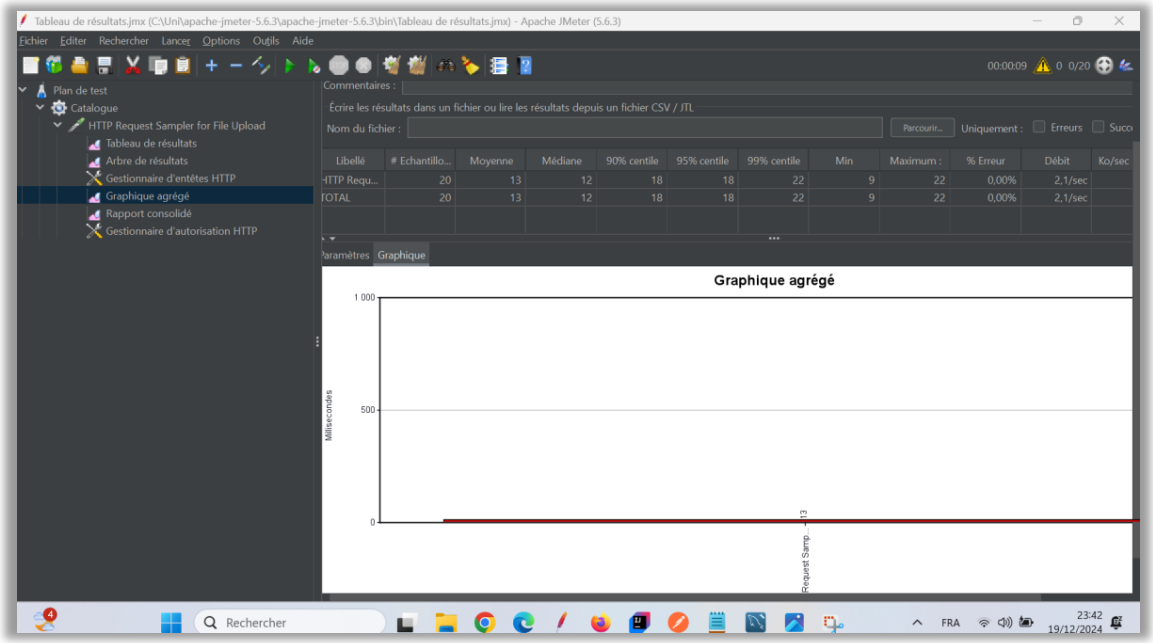
➢ Vue Résultats en Table :



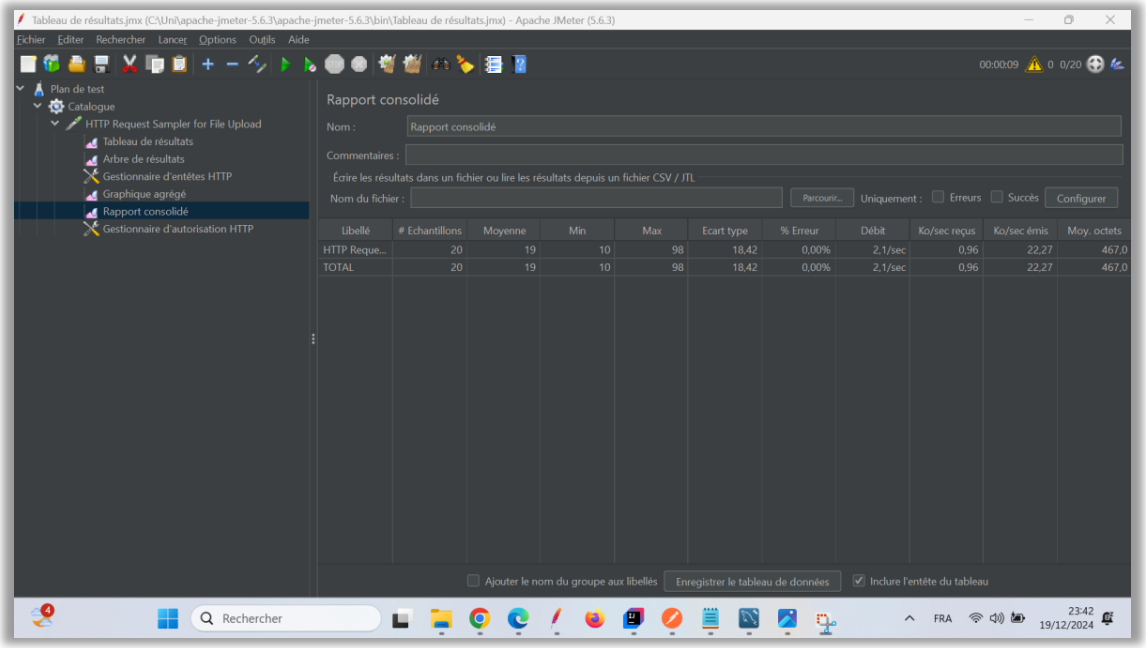
➢ Arbre de Résultats :



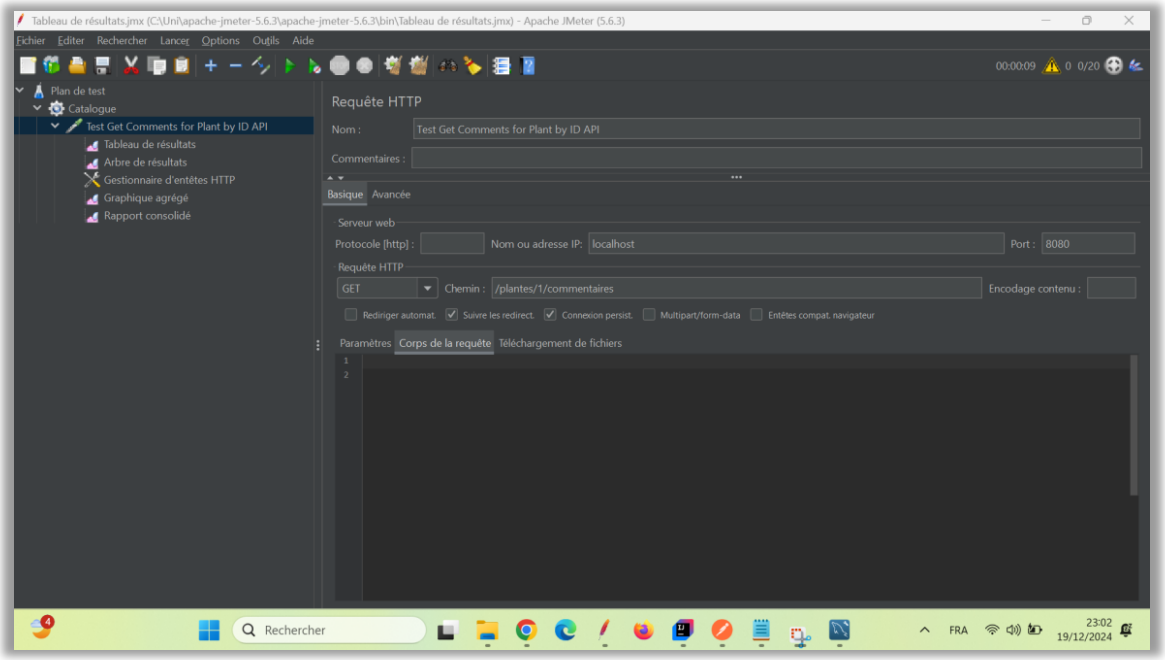
➢ Graphique des Résultats :



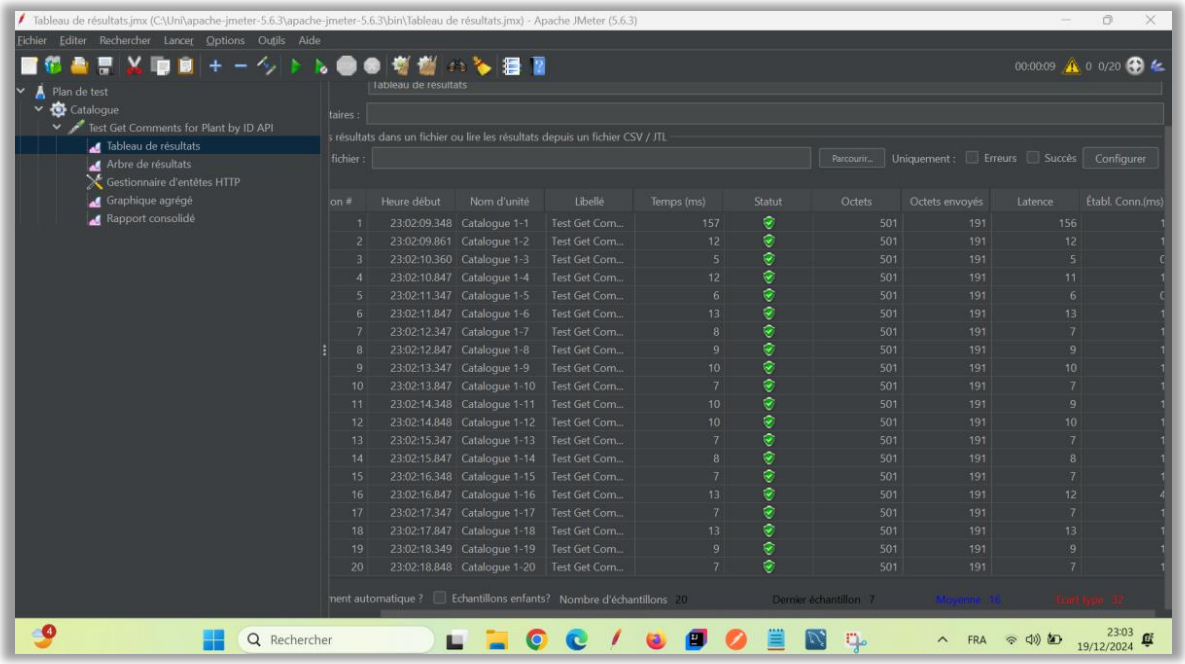
➢ le Rapport Conso



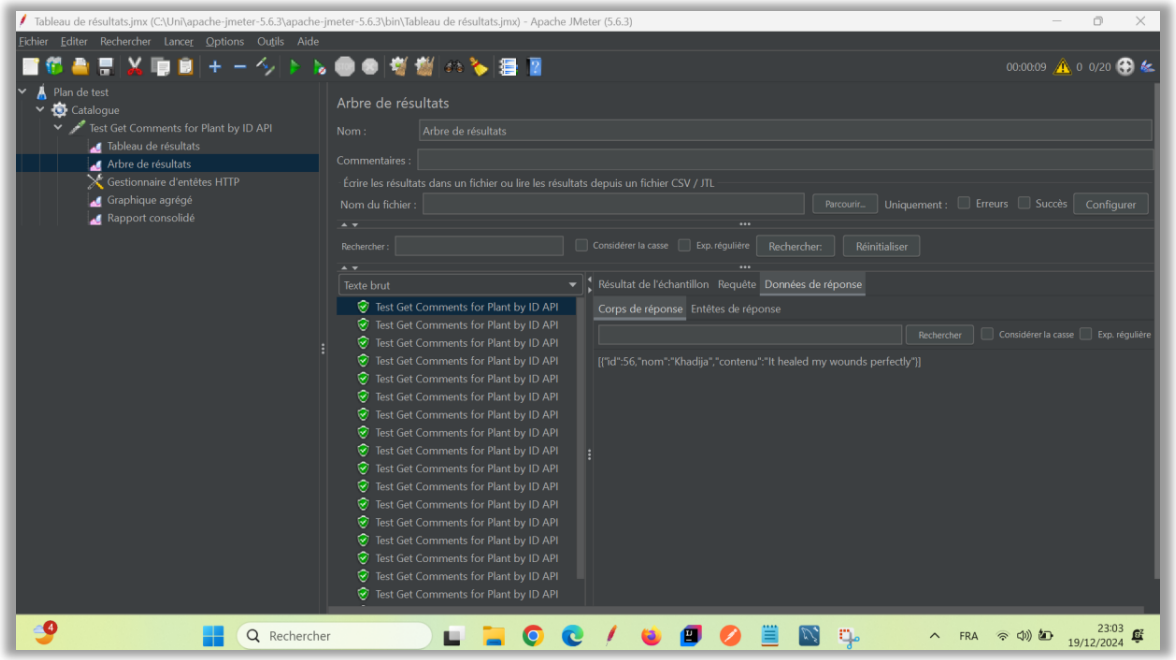
❖ Test Get Comments for Plant by ID API :



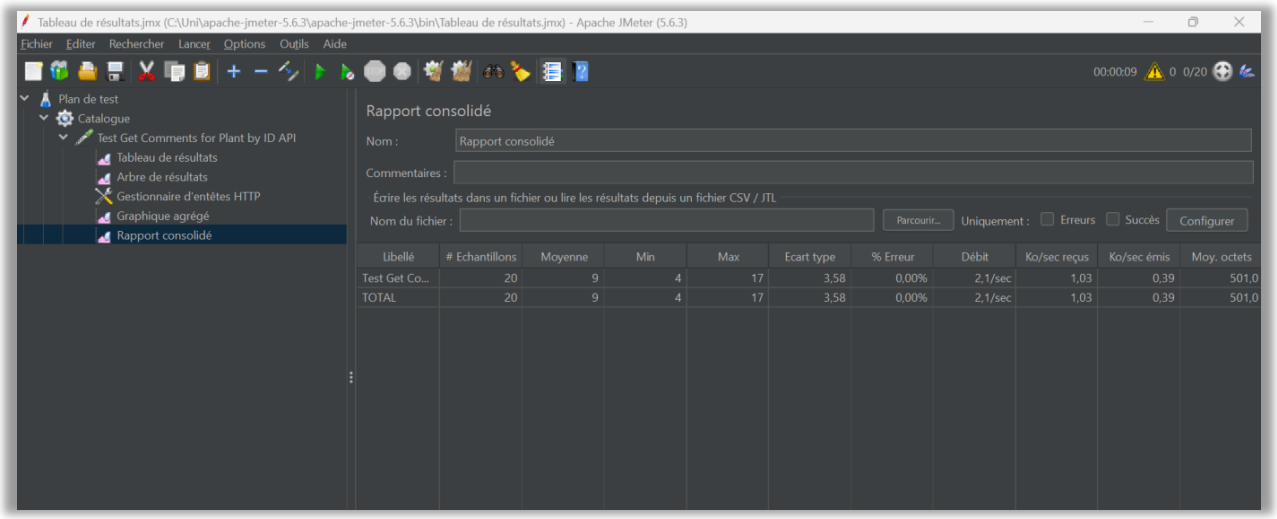
➢ Vue Résultats en Table :



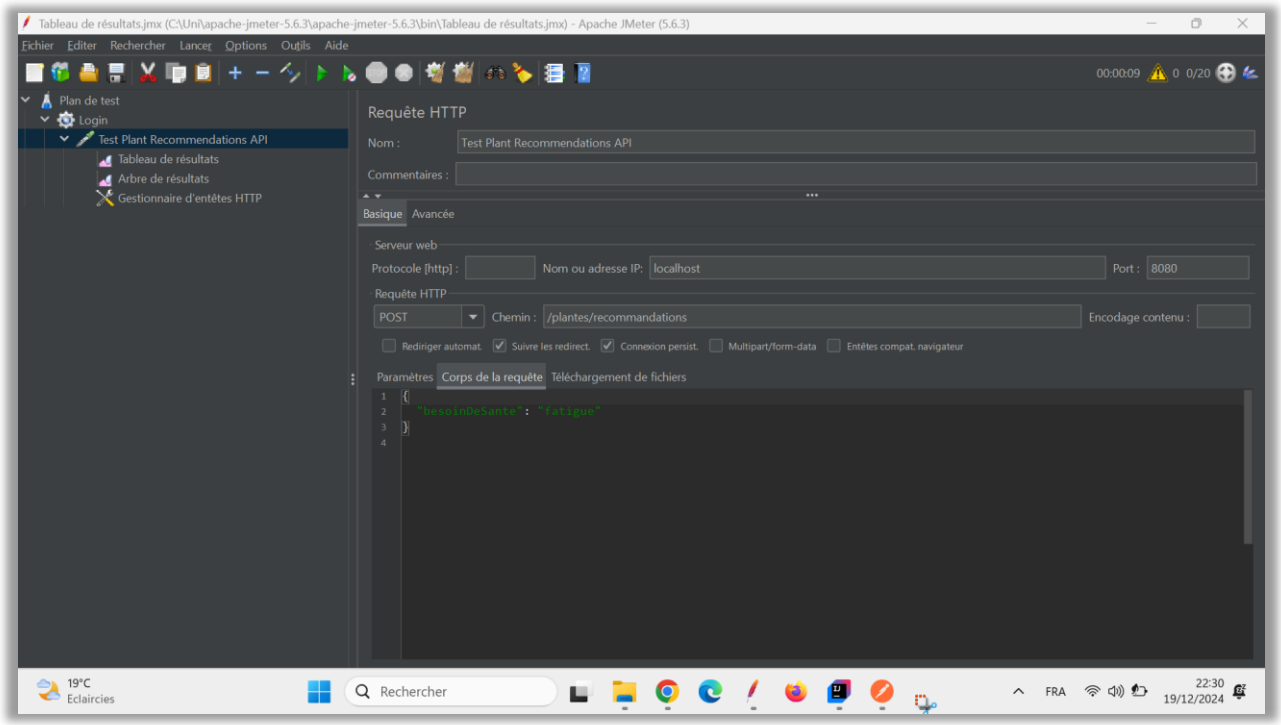
➢ Arbre de Résultats :



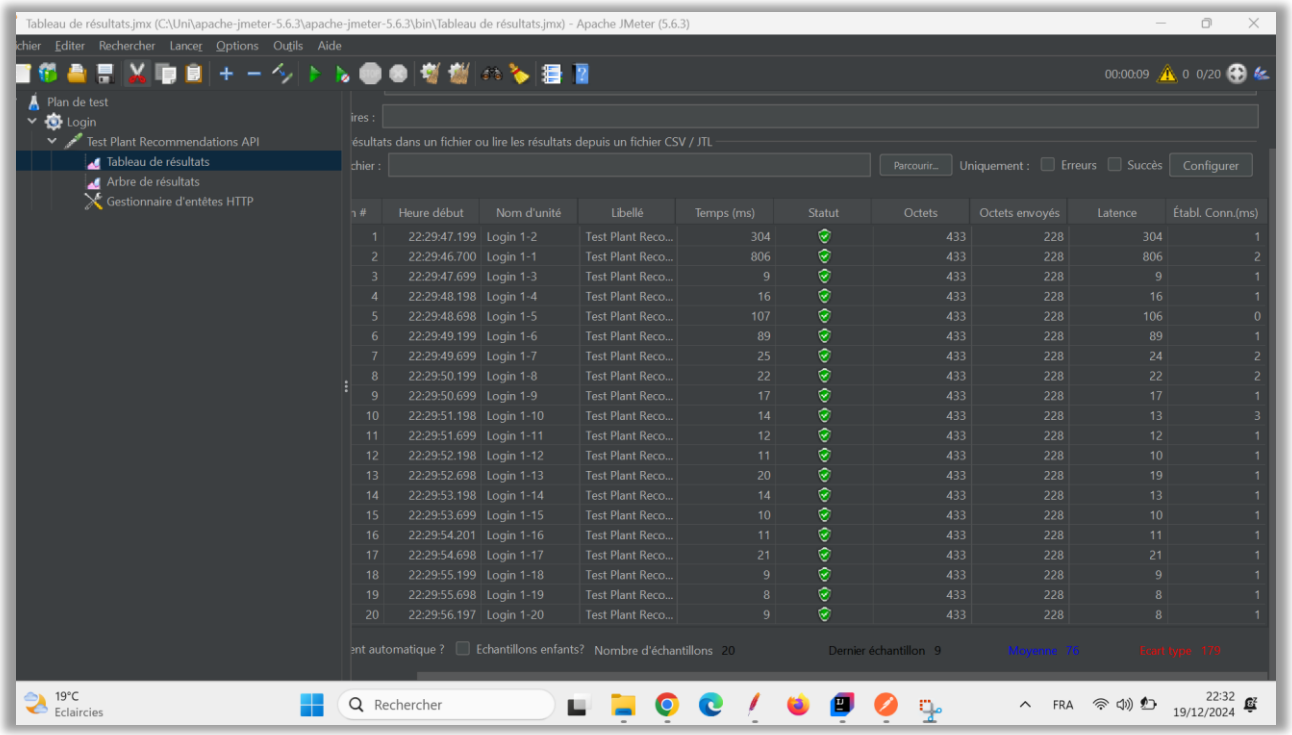
➢ le Rapport Consolidé :



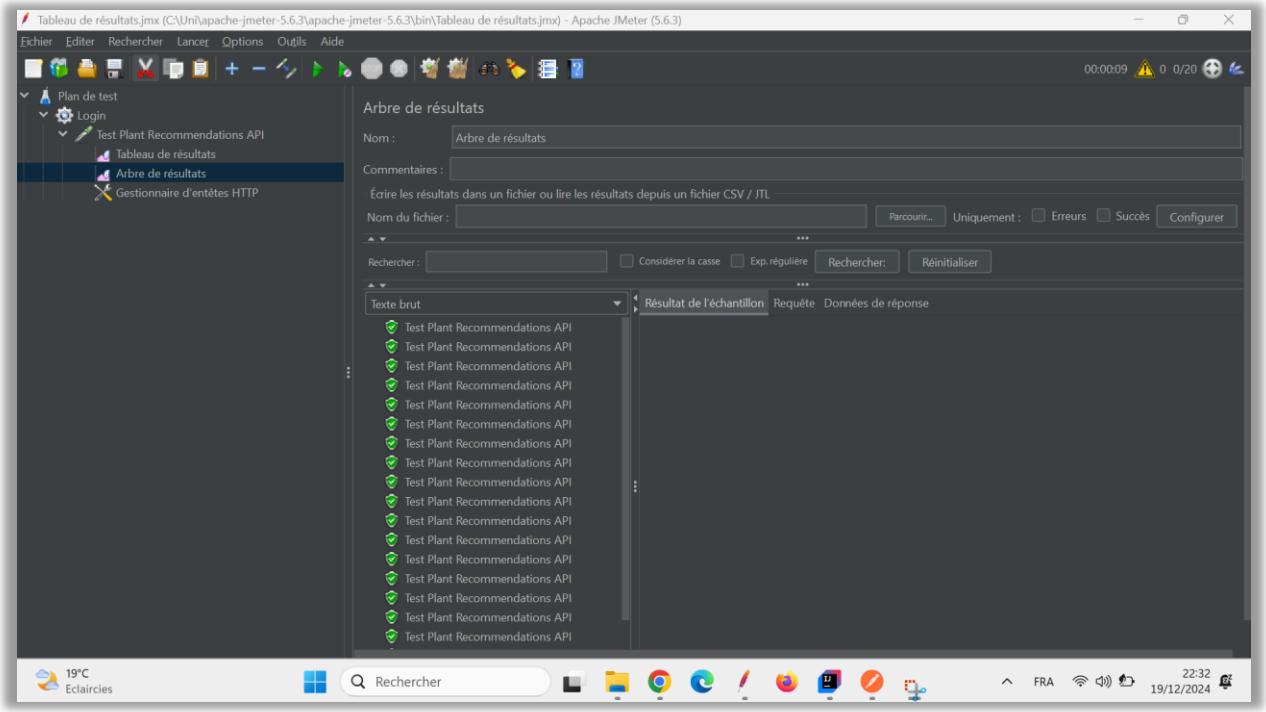
❖ Test Plant Recommendations API :



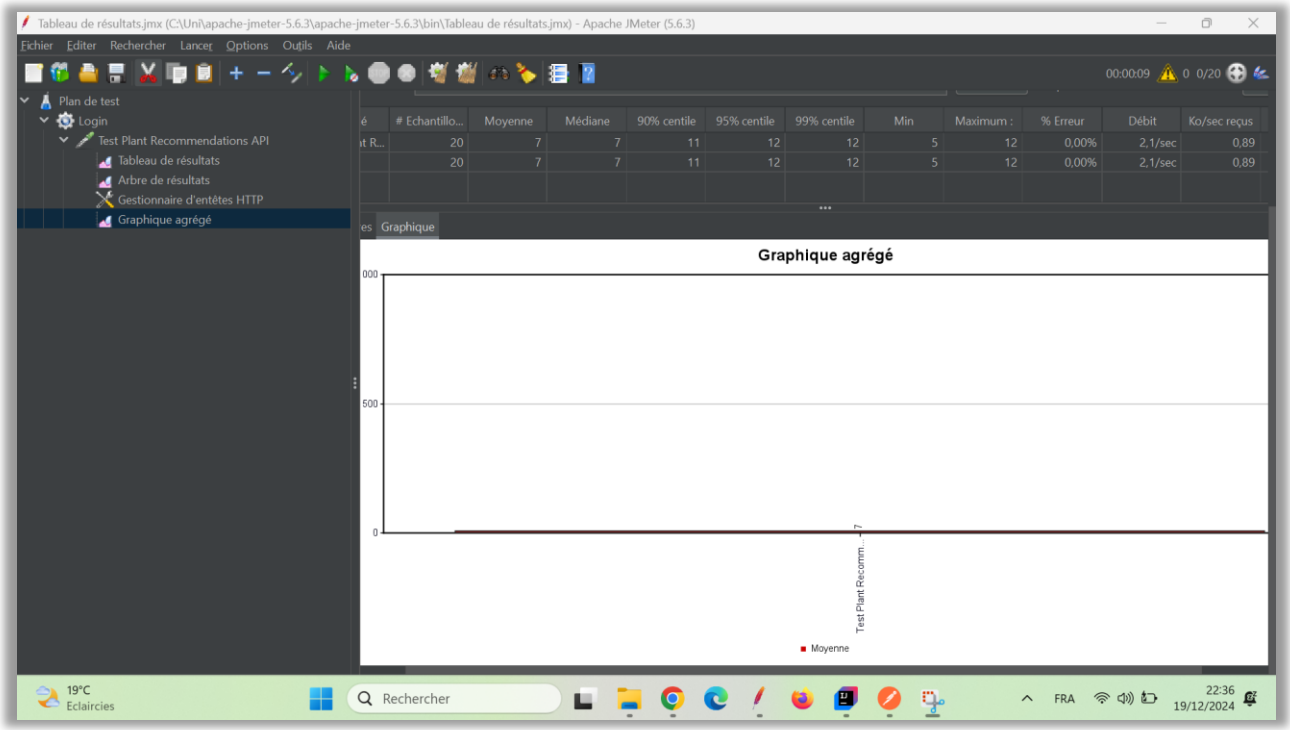
➢ Vue Résultats en Table :



➢ Arbre de Résultats :



➢ Graphique des Résultats :



➢ le Rapport Consolidé :