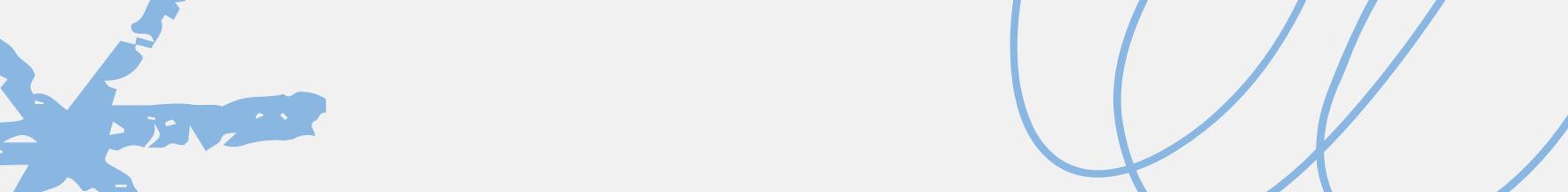


Project presentation

Team Members

- 1-Ahmed Noureldein
- 2-Ahmed Eldokm
- 3- Eman Khairy Elqassas
- 4- Esraa Refaat
- 5- Hany Sameh
- 6- Fatma Abdelazeem
- 7- Mostafa



Agenda

- 1- Project Introduction
- 2- Test Plan
- 3- Test Design
- 4-Test Execution Scope
- 5- DB Testing
- 6- API Testing
- 7- Performance Testing
- 8- Reports and Bug Analysis
- 9- CI/CD Integration
- 10- Automation Testing

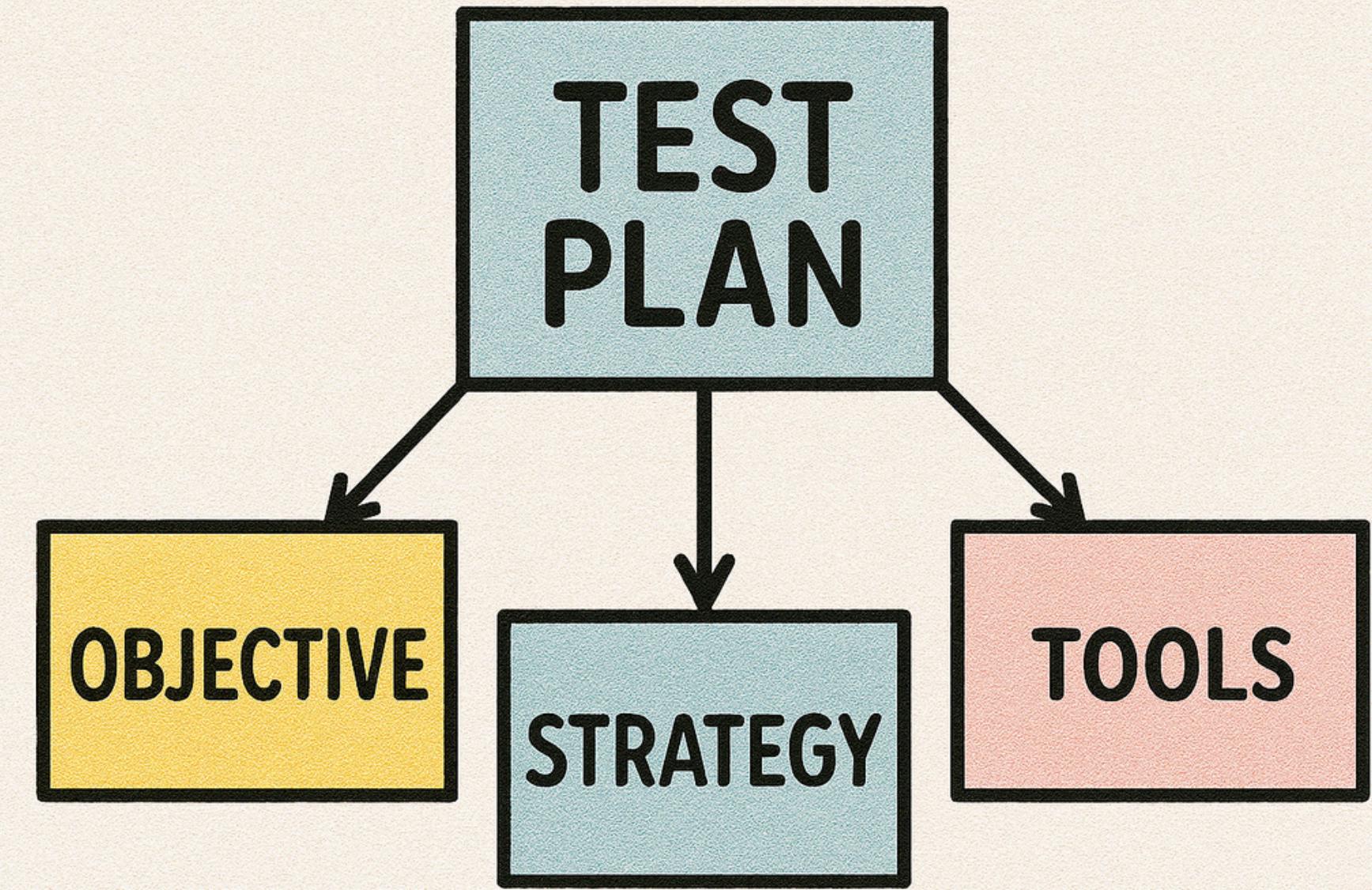
Project Introduction

Fully Optimized Automated Testing Framework

- **Goal:** Ensure quality across applications.
- **Scope:** Demo Blaze Site (Manual, Automation), Trello & Reqres (API, DB).
- **Key Deliverable:** A fully functional framework for end-to-end testing with CI/CD integration.

Test Plan

- **Objectives:** Validate functionality, performance, data integrity, Automate repetitive tests
- **Strategy:** Manual for initial validation, Automated for scale
- Tools: Selenium, Postman, SQL, Jenkins

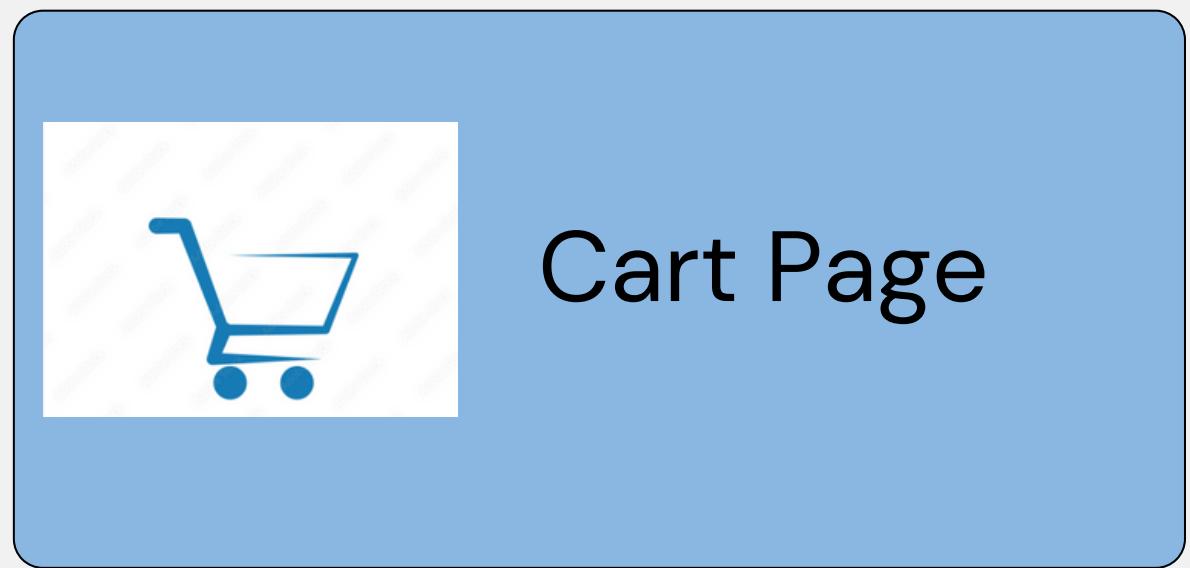
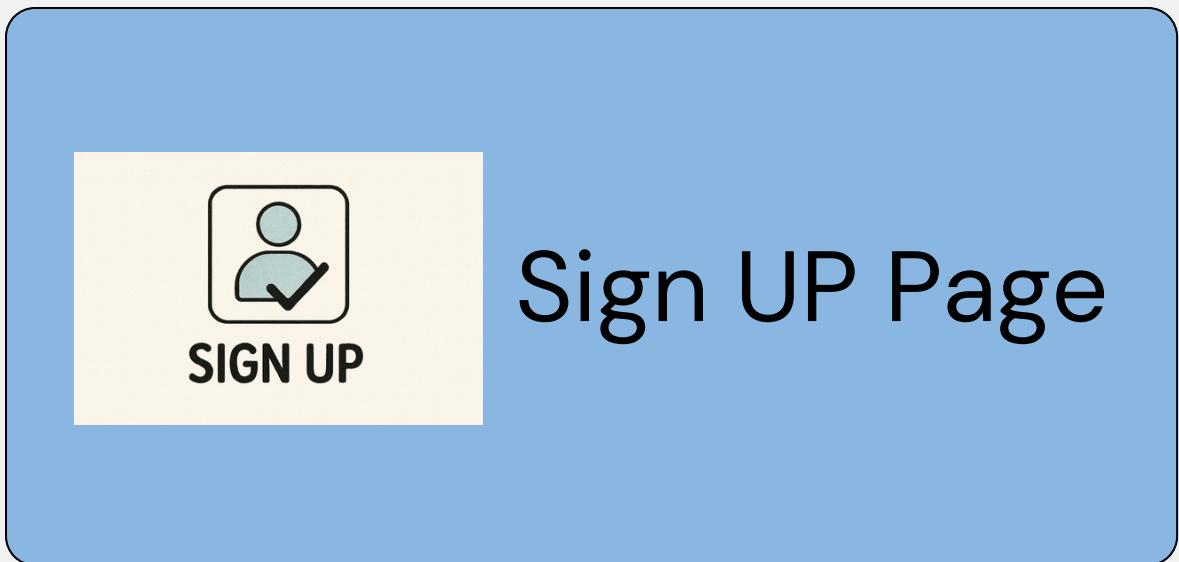


Test Design



- **Test Cases:** UI, functional test cases For Demo Blaze
- **Tool:** Jira , Excell
- Prioritize high-risk areas

Test Execution Scope



DB Testing

- **Approach:** Data integrity & performance
- **Scope:** – Schema validation – Query accuracy
- **Tools:** mySQL
- **Outcome:** Robust data operations

MySQL Workbench Schemas

```
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    email VARCHAR(100) UNIQUE,
    age INT,
    city VARCHAR(100)
);

DESCRIBE users;

Select * from users;

Delete from users where id = 19;
```

Result Grid

Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto_increment
name	varchar(100)	YES		NULL	
email	varchar(100)	YES	UNE	NULL	
age	int	YES		NULL	
city	varchar(100)	YES		NULL	

Action Output

- 1 10:54:57 call 'grad-proj.register_user'; ..
- 2 10:55:11 SELECT * FROM 'grad-proj'.users LIMIT 0, 1000
- 3 10:56:24 DESCRIBE users

Message

Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'DELETE FROM `users` WHERE id = 19' at line 1

19 row(s) returned

5 row(s) returned

MySQL Workbench Schemas

```
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100),
    email VARCHAR(100) UNIQUE,
    age INT,
    city VARCHAR(100)
);

DESCRIBE users;
```

Procedure: register_user

Call stored procedure grad-proj.register_user

Enter values for parameters of your procedure and click <Execute> to create an SQL editor and run the call.

p_name [IN] VARCHAR(100)
p_email [IN] VARCHAR(100)
p_age [IN] INT
p_city [IN] VARCHAR(100)

Execute Cancel

Result Grid

id	name	email	age	city
1	Alice Johnson	alice1@example.com	25	New York
2	Bob Smith	bob2@example.com	30	Los Angeles
3	Charlie Brown	charlie3@example.com	28	Chicago
4	David Clark	david4@example.com	35	Houston
5	Eva Adams	eva5@example.com	22	Phoenix
6	Frank Wright	frank6@example.com	40	Philadelphia
7	Grace Lee	grace7@example.com	27	San Antonio
8	Henry Ford	henry8@example.com	33	San Diego
9	Ivy Moore	ivy9@example.com	29	Dallas
10	Jack Ryan	jack10@example.com	31	San Jose
11	Kate Bell	kate11@example.com	26	Austin
12	Leo King	leo12@example.com	38	Jacksonville
13	Mia Scott	mia13@example.com	24	Fort Worth
14	Nick Wood	nick14@example.com	45	Columbus
15	Olivia Hall	olivia15@example.com	21	Charlotte
16	Paul Allen	paul16@example.com	34	San Francisco
17	Quinton Ross	quinton17@example.com	37	Indianapolis

Action Output

Message

Api Testing

- **Approach:** Endpoint validation
- **Target:** Trello & Reqres
- **Tool:** Postman
- **Outcome:** Reliable API performance

A screenshot of the Postman application interface. The request URL is `https://reqres.in/api/login`. The request method is `POST`. The `Body` tab shows a JSON payload:

```
1 {  
2   "email": "eve.holt@reqres.in",  
3   "password": "cityslicka"  
4 }  
5 
```

The response status is `200 OK`, with a response time of `202 ms` and a size of `1.06 KB`. The response body is:

```
1 {  
2   "token": "QpwL5tke4Pnpja7X4"  
3 }  
4 
```

A screenshot of the Postman application interface. The request URL is `https://mock.codes/403`. The request method is `GET`. The `Body` tab shows a Raw response:

```
1 {"statusCode":403,"description":"Forbidden"}  
2 
```

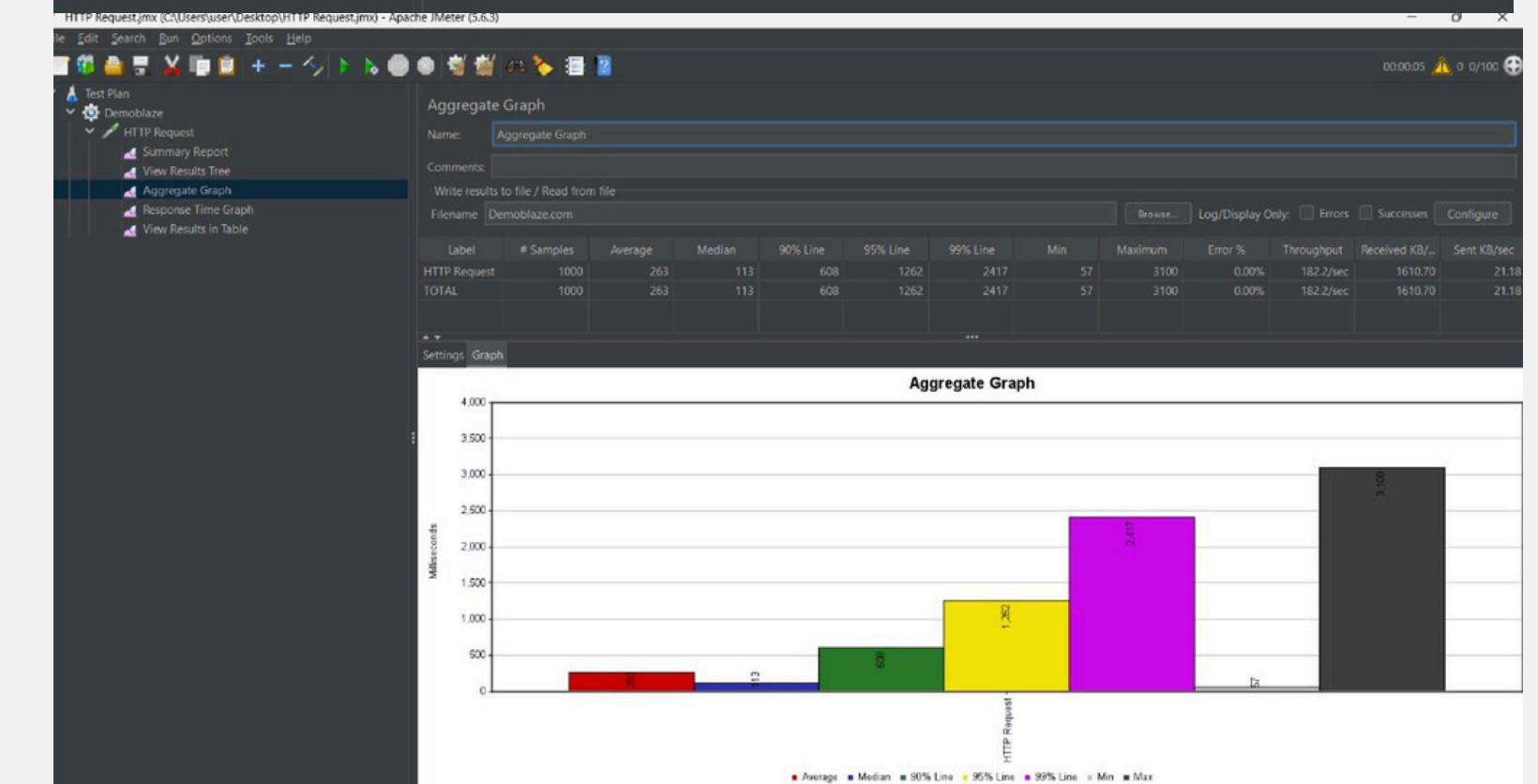
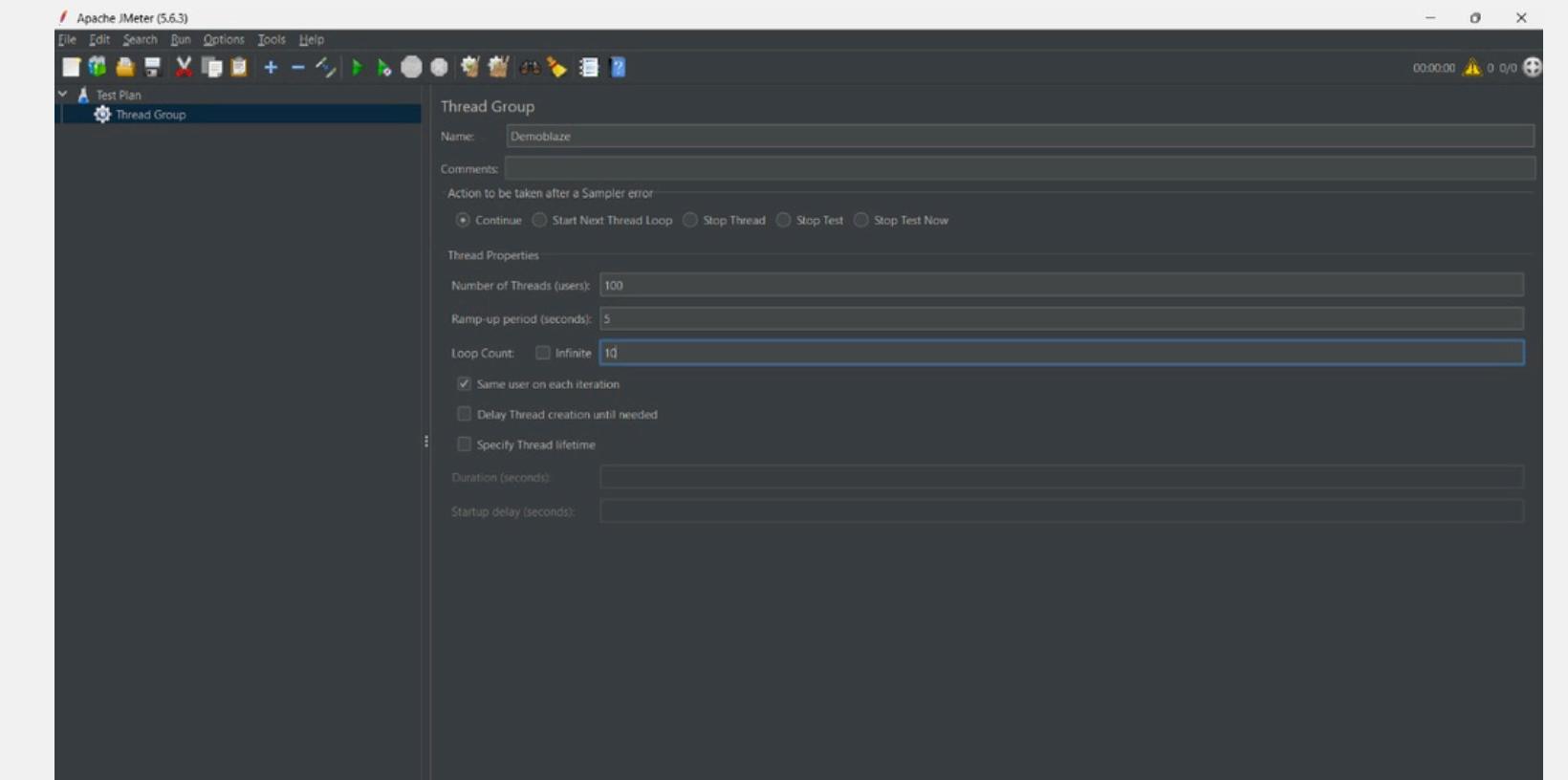
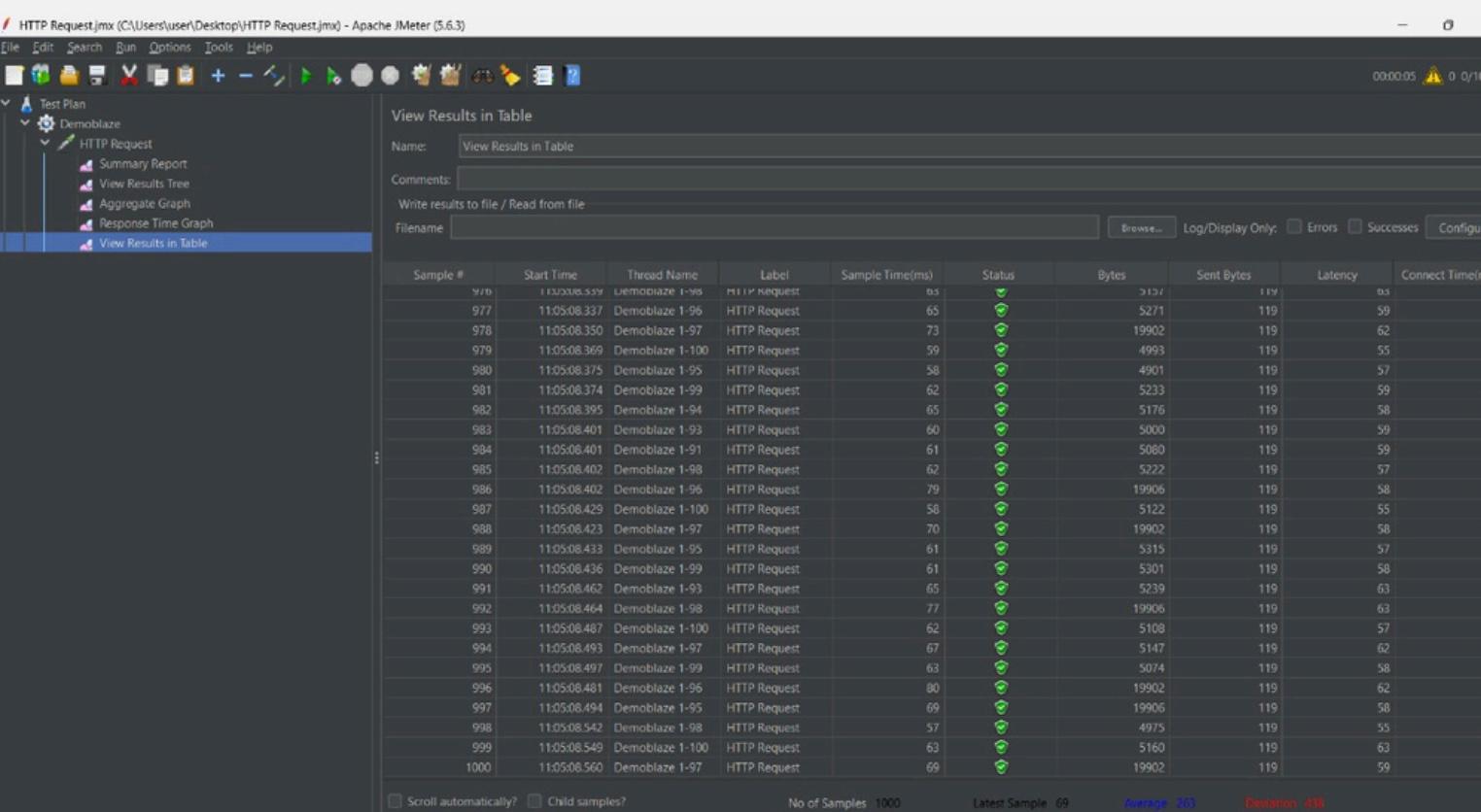
A screenshot of the Postman application interface. The request URL is `https://reqres.in/api/login`. The request method is `POST`. The `Body` tab shows a JSON payload:

```
1 {  
2   "error": "Missing email or username"  
3 }  
4 
```

The response status is `400 Bad Request`, with a response time of `539 ms` and a size of `1.07 KB`.

Performance Testing

- **Approach:** Load & stress testing
- **Scope:** API response times
- **Tool:** JMeter



Reports & Bugs

Metrics: Pass/fail rates – Test coverage –
Tools: Extent Report, Jira –

The screenshot displays a software interface for test automation, specifically an Extent Report. On the left, a sidebar lists various test cases with their execution status (Pass). On the right, a detailed log for the "Add Samsung Galaxy S6 to Cart" test is shown, including timestamped logs and a summary of the process.

Tests

- Add Samsung Galaxy S6 to Cart (Pass)
- Add Nokia Lumia 1520 to Cart (Pass)
- Verify Cart Page (Pass)
- Delete Samsung Galaxy S6 from Cart (Pass)
- Verify Price Update (Pass)
- Delete Nokia Lumia 1520 from Cart (Pass)
- Submit Valid Order (Pass)
- Verify OK Button Functionality (Pass)

Add Samsung Galaxy S6 to Cart

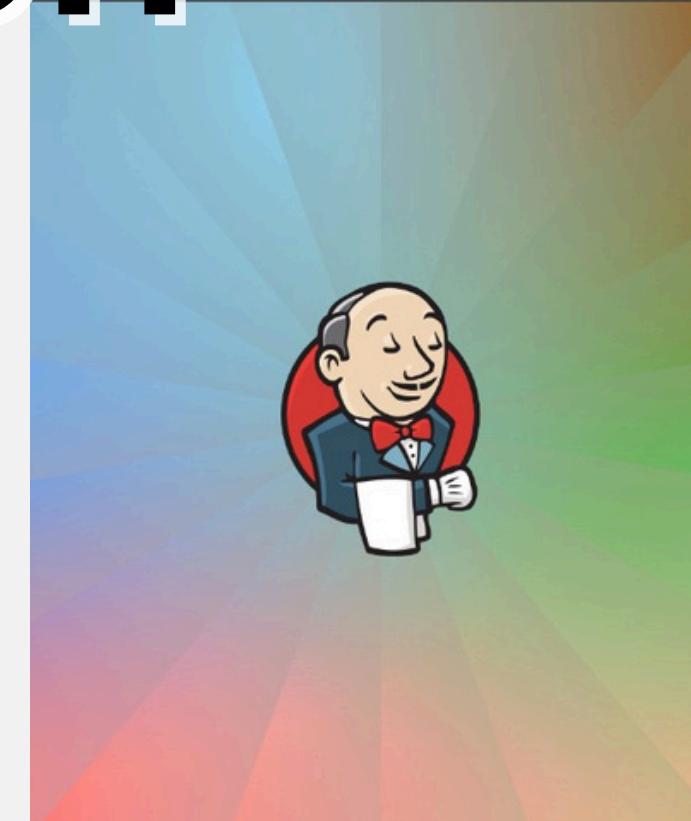
04.13.2025 4:24:52PM 04.13.2025 4:24:58PM 00:00:05:436 · #test-id=1

Verify Samsung Galaxy S6 can be added to cart

STATUS	TIMESTAMP	DETAILS
Info	4:24:52 PM	Navigating to home page
Info	4:24:55 PM	Selecting Samsung Galaxy S6
Info	4:24:56 PM	Verifying Samsung product page
Info	4:24:57 PM	Adding Samsung to cart
Pass	4:24:58 PM	Successfully added Samsung Galaxy S6 to cart

CI/CD Integration

Tools: Jenkins



The screenshot shows the Jenkins login interface. It features a cartoon character in a tuxedo holding a martini glass against a background of colorful radial rays. The URL in the browser bar is `localhost:8080/login?from=%2F`.

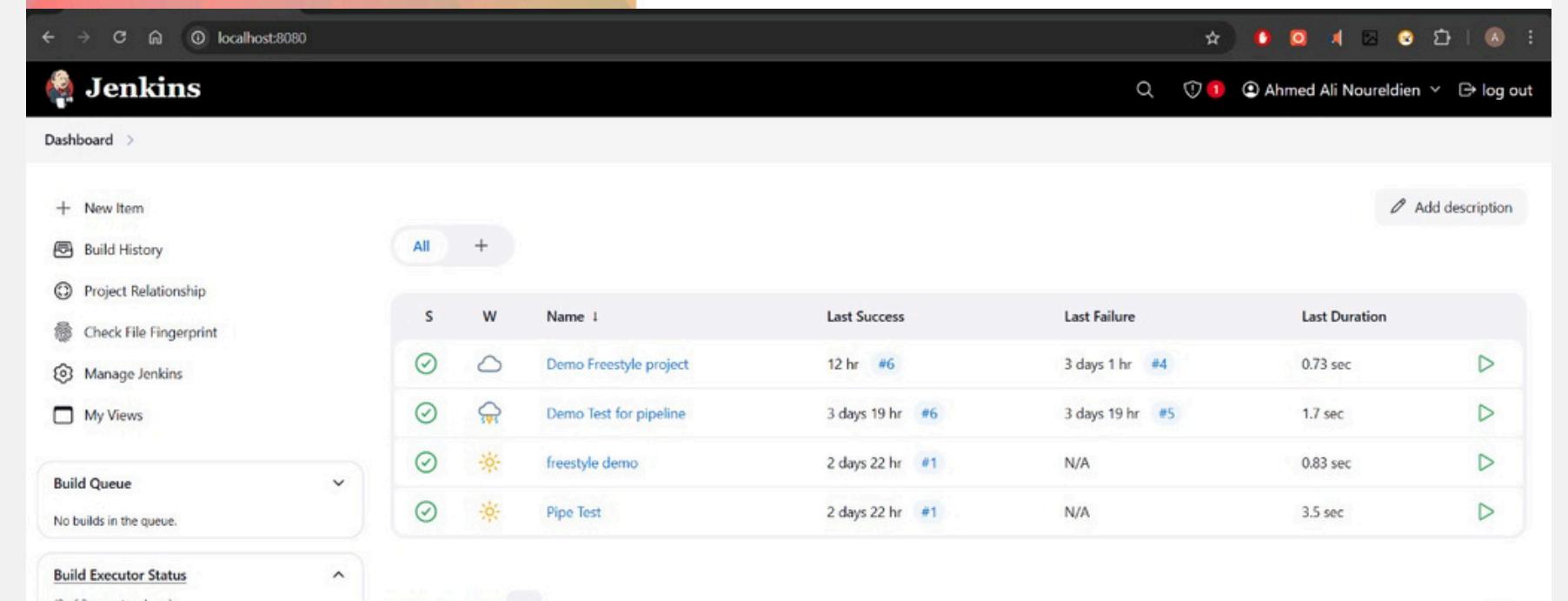
Sign in to Jenkins

Username: Ahmed_Noueldien

Password:

Keep me signed in

Sign in

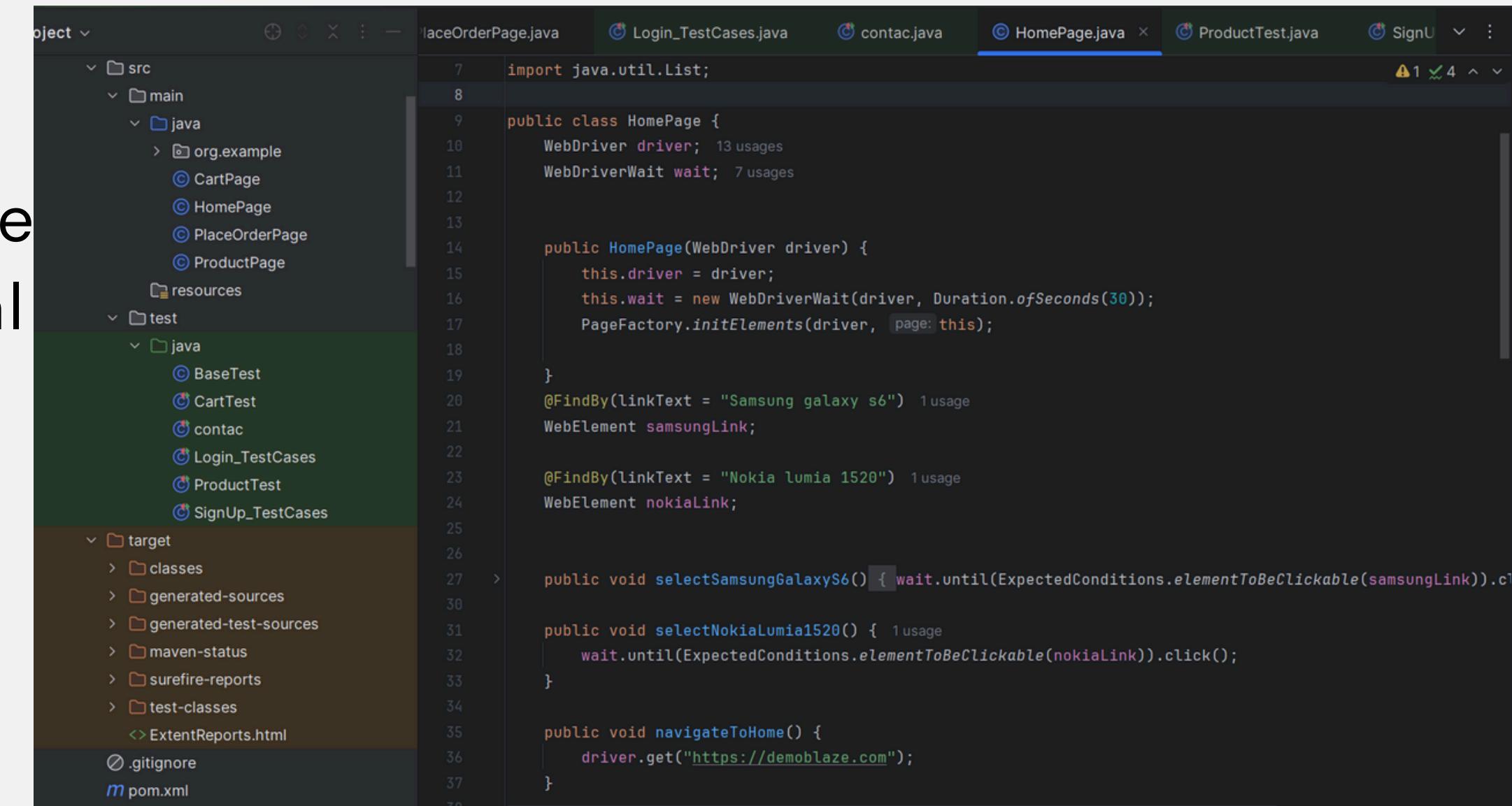


The screenshot shows the Jenkins dashboard. The top navigation bar includes links for search, help, user profile (Ahmed Ali Noueldien), and log out. The dashboard features a sidebar with options like New Item, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins, and My Views. A "Build Queue" section indicates "No builds in the queue". The main area displays a table of projects with columns for Status (S), Workstation (W), Name, Last Success, Last Failure, and Last Duration.

S	W	Name	Last Success	Last Failure	Last Duration
✓	Cloud	Demo Freestyle project	12 hr #6	3 days 1 hr #4	0.73 sec
✓	Cloud	Demo Test for pipeline	3 days 19 hr #6	3 days 19 hr #5	1.7 sec
✓	Sun	freestyle demo	2 days 22 hr #1	N/A	0.83 sec
✓	Sun	Pipe Test	2 days 22 hr #1	N/A	3.5 sec

Automation Testing

- **Approach:** Implementing automated scripts and frameworks to validate software functionalities, reducing manual effort, and improving testing speed.
- **Scope:** Functional Testing: Automating key user flows
- **Tool:** Selenium



The screenshot shows a Java code editor within an IDE. The left pane displays the project structure under 'Project' mode, showing a package named 'org.example' containing several classes: PlaceOrderPage, CartPage, HomePage, ContactPage, ProductPage, BaseTest, CartTest, ContactTest, Login_TestCases, ProductTest, and SignUp_TestCases. The right pane shows the code for the 'HomePage.java' file. The code imports java.util.List and defines a HomePage class that takes a WebDriver driver as a parameter. It initializes a WebDriverWait wait with a duration of 30 seconds and uses PageFactory to initialize elements. The class contains methods for selecting Samsung Galaxy S6 and Nokia Lumia 1520, and a method for navigating to the home page by setting the driver's URL to 'https://demoblaze.com'. The code editor also shows other tabs like PlaceOrderPage.java, Login_TestCases.java, ContactPage.java, ProductTest.java, and SignUp.java.

```
import java.util.List;
public class HomePage {
    WebDriver driver; 13 usages
    WebDriverWait wait; 7 usages
    public HomePage(WebDriver driver) {
        this.driver = driver;
        this.wait = new WebDriverWait(driver, Duration.ofSeconds(30));
        PageFactory.initElements(driver, this);
    }
    @FindBy(linkText = "Samsung galaxy s6") 1 usage
    WebElement samsungLink;
    @FindBy(linkText = "Nokia lumia 1520") 1 usage
    WebElement nokiaLink;
    public void selectSamsungGalaxyS6() { wait.until(ExpectedConditions.elementToBeClickable(samsungLink)).click(); }
    public void selectNokiaLumia1520() { 1 usage
        wait.until(ExpectedConditions.elementToBeClickable(nokiaLink)).click(); }
    public void navigateToHome() {
        driver.get("https://demoblaze.com");
    }
}
```