**BÁO CÁO QUÁ TRÌNH**

**THIẾT KẾ VÀ THỰC THI KIỂM THỬ HIỆU NĂNG**

**Công cụ hỗ trợ**: Apache JMeter

**Ngôn ngữ:** Java

Cách cài đặt:

Tải JDK mới nhất để có thể sử dụng được ngôn ngữ Java:

Link tải: <https://www.oracle.com/java/technologies/downloads/#java8-windows>

A screenshot of a computer

Description automatically generated

Sau khi tải về, ta cài đặt SDK. Tiếp theo, ta mở Environment Variables, ở mục System variables ta chọn biến Path và nhấn Edit. Ta thêm đường dẫn của JDK mới tải.

Đường dẫn đó ở Windows mặc định thường là **C:\Program Files\java\jdk-21\bin**

A screenshot of a computer

Description automatically generated

Tiếp theo, ta tải JMeter:

Link tải: <https://jmeter.apache.org/download_jmeter.cgi>

Ấn vào file zip ở mục Binaries để tải xuống.

A screenshot of a computer

Description automatically generated

Sau khi tải xong JMeter, ta giải nén và vào thư mục bin, tìm file ApacheJMeter.jar. Ta mở nó lên bằng Java(TM) Platform SE binary:

A screenshot of a computer

Description automatically generated

Ta mở được trang giao diện chính của JMeter.

Sau đây là các bước trong test scenario mà nhóm muốn thiết lập:

# Chức năng đăng ký tài khoản.

Màn hình đăng ký tài khoản như sau:

A screenshot of a login form

Description automatically generated

**Tại mục định nghĩa biến mặc định, cài đặt name và value của URL trang web như sau:**

A screenshot of a computer

Description automatically generated

**Tại thread group có 3 thông số như sau:**

* **Number of Threads (users)**: Đây là số lượng người dùng ảo (luồng) mà bạn muốn mô phỏng. Mỗi người dùng ảo sẽ tạo ra một yêu cầu đến máy chủ trong quá trình thử nghiệm.
* **Ramp-Up Period (thời gian khởi động)**: Đây là khoảng thời gian (tính bằng giây) mà JMeter sẽ dùng để khởi tạo số lượng người dùng ảo đã thiết lập. Nó cho phép bạn kiểm soát tốc độ tăng của các luồng trong quá trình bắt đầu thử nghiệm.
* **Loop Count (số lần lặp lại)**: Đây là số lần mà mỗi luồng (người dùng ảo) sẽ lặp lại chuỗi yêu cầu đã thiết lập. Nếu đặt giá trị này là 1, mỗi luồng sẽ thực hiện chuỗi yêu cầu một lần. Nếu đặt giá trị này lớn hơn 1, mỗi luồng sẽ lặp lại chuỗi yêu cầu theo số lần được chỉ định.

A screenshot of a computer

Description automatically generated

Với màn hình hiện tại, 3 chỉ số lần lượt là: 500 – 1 – 1. Đồng nghĩa với việc khởi tạo 500 người dùng ảo trong 1 giây và mỗi người dùng chỉ gửi request 1 lần.

**Tại màn hình HTTP Request:** Thiết lập 3 thông số tương ứng với các dữ liệu cần nhập trên màn hình đăng ký. Các thông tin cần nhập bao gồm: email, password và password-confirmation.

**A screenshot of a computer

Description automatically generated**

* 1: Giao thức protocol – lấy theo giao thức hiện tại của trang web, hiện tại trang web đang dùng có protocol là http.
* 2: Server name – địa chỉ server hoặc IP của trang web. Ở đây đã được thiết lập ở bước User Defined Variable với name = “BASE\_URL\_1”.
* 3: HTTP Request – định dạng loại yêu cầu HTTP gửi đi cho server xử lý. Khi đăng ký sẽ gửi đi các thông tin đăng ký nên request = “POST”.
* 4: Path – đường dẫn cụ thể tới chức năng, với chức năng đăng ký thì path = “REGISTER”.
* 5: Gửi dữ liệu đi kèm để đăng ký tài khoản. Các dữ liệu đi kèm bao gồm:
  + - Email: Email đăng ký.
    - Password: Mật khẩu.
    - Password-confirmation: Xác nhận mật khẩu.
* Sau khi thiết lập các tham số như trên, bấm nút “Start” màu xanh trên thành công cụ để kiểm thử hiệu suất chức năng đăng ký với 500 người dùng ảo.

# Chức năng tạo hóa đơn.

Giao diện tạo hóa đơn như sau:

A screenshot of a computer

Description automatically generated

Thiết lập các thông số mặc định tương tự chức năng 1. Đăng ký tài khoản. Các thông số trong form hóa đơn như sau:

A screenshot of a computer

Description automatically generated

* Name: Tên của hóa đơn.
* Amount\_min: Số tiền nhỏ nhất.
* Amount\_max: Số tiền lớn nhất.
* Date: Ngày hết hạn thanh toán.
* Repeat\_freq: tần suất hóa đơn (theo tháng, theo năm hoặc theo quý).
* Transaction\_currency\_id: mã tiền tệ, ở đây mặc định là euro.

\*Test hiệu năng với 500 dữ liệu được nhập từ file csv.

Ta có 1 file csv với 500 thông tin đơn hàng với thứ tự sắp xếp các cột tương tự như trên.

A screenshot of a computer

Description automatically generated

Cài đặt jmeter để nhận tham số từ file csv:

* Add thêm CSV Data config vào thread group.
* Thiết lập các thông số đầu vào như sau:

A screenshot of a computer

Description automatically generated

Trong đó:

* + - Name: Tên tên của element config.
    - Filename: Đường dẫn tới file csv.
    - File encoding: Encoding của file dữ liệu (UTF-8).
    - Variable name: Danh sách tất cả các tên biến, ngăn cách bởi dấu phẩy.
    - Delimeter: Tách từng bản ghi trong CSV.
    - \_token: Đoạn mã xác nhận quyền truy cập của tài khoản.
    - Allow quote data ? : Chọn "TRUE" trong trường hợp bạn muốn bỏ qua dấu kép và cho phép các phần tử như vậy chứa dấu phân cách.
    - Recycle on EOF? : Khi số lượng thread lớn hơn số lượng data được sử dụng trong file CSV, chọn "TRUE" nếu bạn muốn tiếp tục thực hiện kiểm thử bằng cách quay lại đọc từ đầu file.
    - Stop thread on EOF? : Nếu chọn "SET", khi chạy đến end of file thì thread sẽ tự động dừng lại.
    - Sharing mode: Định nghĩa hành vi chia sẻ của file CSV. Giá trị mặc định thường là "All threads".
      * All threads: Nếu trong trường hợp kiểm thử với nhiều element CSV config cùng tham chiếu đến một CSV thì CSV Data Set Config kế tiếp sẽ tiếp tục đọc file CSV đã được mở từ CSV data set config trước đó.
      * Current thread group: Nếu trong script có nhiều hơn một element CSV data set config cùng tham chiếu đến một file, CSV data set config kế tiếp sẽ mở lại file cho từng thread group.
      * Current thread: Mỗi file csv được mở riêng biệt cho từng thread khi chọn option này.

Tại màn hình HTTP Request, mapping các tham số vào các biến của file CSV như sau:

A screenshot of a computer

Description automatically generated

Cuối cùng, bấm Start để tiến hành quá trình test.

# Chức năng xóa hóa đơn.

Thiết lập HTTP Request cho chức năng xóa hóa đơn như sau:

A screenshot of a computer

Description automatically generated

Trong đó:

* Protocol, Server name, request không đổi.
* Path: bills/destroy/40.
* Thông số gửi đi:
  + - Submit: Delete permanently (Đồng ý xóa bill).
    - \_token: Đoạn mã xác nhận quyền truy cập của tài khoản.

Bấm Start để tiến hành quá trình test.

Ban đầu, mục tiêu của nhóm:

**Số lượng user: 500**

**Kết quả mong đợi:**

**Average Response-time: <10000**

**Error rate: < 10%**

Tuy nhiên, khi nhóm tiến hành tương tác với API của trang web thì gặp lỗi 419 PAGE EXPIRED của bên backend là PHP với Laravel. Tuy nhiên nhóm vẫn chưa tìm ra cách để khắc phục được lỗi này nên không thể chạy được kết quả.