

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA: CÔNG NGHỆ THÔNG TIN**

-----2022-----



# **BÁO CÁO**

## **Testing Phase**

**LỚP : 19\_22**

**Giáo viên lý thuyết: Th.S. Bùi Tấn Lộc**

**SINH VIÊN THỰC HIỆN:**

**19120544 - Cao Thanh Khiết**

**19120533 - Ninh Duy Huy**

**19120557 - Trần Tuấn Kiệt**

**19120558 - Văn Quý Lâm**

**19120559 - Hà Duy Lãm**

# Testing

## 5.1 User Testing

### 5.1.1 Test Cases

#### Màn hình Login

Test case 1: Test login với đúng tài khoản và mật khẩu

Test case 2: Test login với đúng tài khoản và sai mật khẩu

Test case 3: Test login với sai tài khoản và đúng mật khẩu

Test case 4: Test login với sai tài khoản và mật khẩu nào đó bất kì

- Test với trường hợp crazy input

Test case 5: Test login với bỏ trống tài khoản và mật khẩu bất kỳ

Test case 6: Test login với tài khoản và mật khẩu bị bỏ trống

Test case 7: Test login với bỏ trống cả tài khoản và mật khẩu

Test case 8: Test login với tài khoản và mật khẩu rất dài

Test case 9: Test login với SQL Injection

Test case 10: Test login với tài khoản đã tồn tại

#### Màn hình tra cứu phòng

Test case 1: Test booking phòng có trạng thái “trống”

Test case 2: Test booking phòng có trạng thái “Đang sử dụng”

Test case 3: Test **chỉnh sửa** thông tin của phòng với giá trị cho phép

Test case 4: Test **chỉnh sửa** thông tin của phòng với giá trị vi phạm

Test case 5: Test **chỉnh sửa** thông tin với giá trị bỏ trống

Test case 6: Test **thêm** thông tin của phòng mới với giá trị cho phép

Test case 7: Test **thêm** thông tin của phòng mới với giá trị vi phạm

Test case 8: Test **thêm** thông tin của phòng mới với giá trị bỏ trống

Test case 9: Test **xóa** thông tin phòng

### **Màn hình hóa đơn thanh toán**

Test case 1: Test BillForm với đúng các trị trong khoảng cho phép.

Test case 2: Test BillForm với sai 2 giá trị đầu và đúng 2 giá trị cuối.

Test case 3: Test BillForm với 1 trường hợp bị bỏ trống.

Test case 4: Test BillForm với 1 giá trị âm rất dài.

Test case 5: Test BillForm với Số phòng và số ngày thuê vượt ngoài giá trị cho phép.

### **Màn hình lập phiếu thuê phòng**

Test case 1: Test GuestRentRoom với đúng các giá trị trong khoảng cho phép

Test case 2: Test GuestRentRoom với các giá trị vi phạm

Test case 3: Test GuestRentRoom với với giá trị bỏ trống

Test case 4: Test GuestRentRoom xóa thông tin

Test case 5: Test GuestRentRoom chỉnh sửa thông tin

### **Màn hình loại phòng**

Test case 1: Nhập đúng thông tin trong input để tạo một Room mới

Test case 2: Nhập thông tin Room mới trùng với một Room đã tồn tại trước đó

Test case 3: Test thay đổi thông tin của một Room

## **5.1.2 Test Cases & Report**

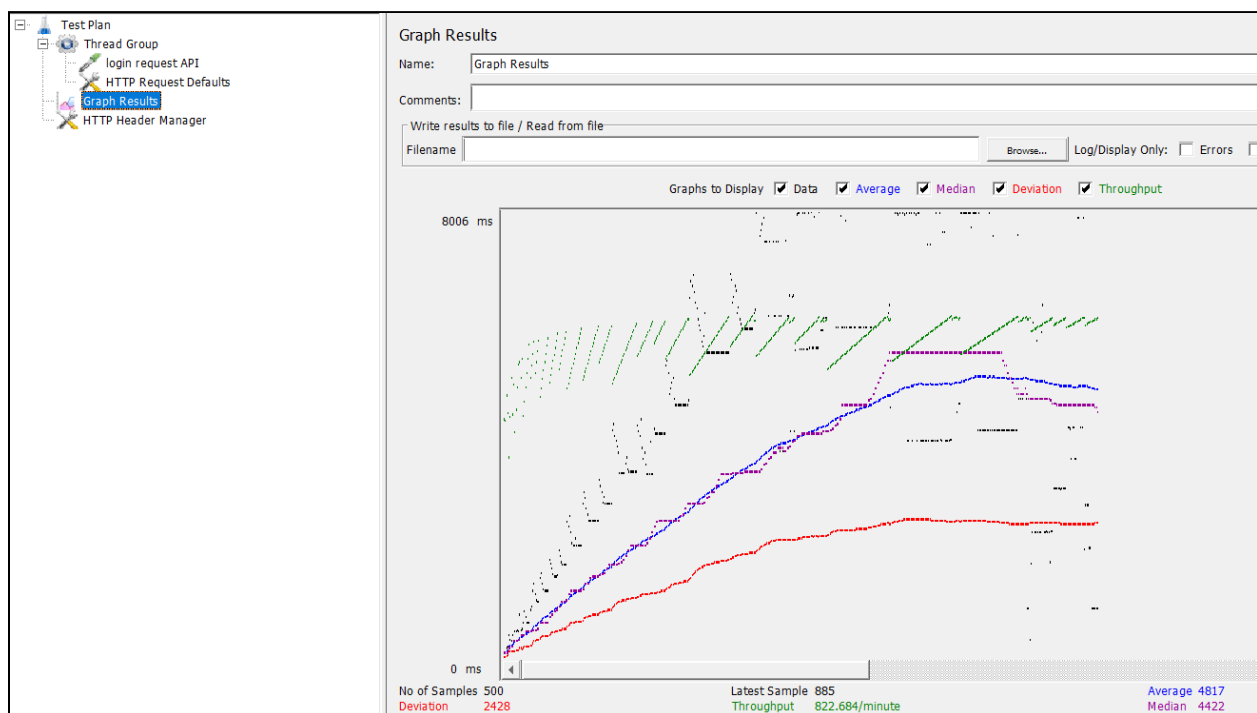
Link: <https://1drv.ms/x/s!AhyYQS6QAzU0giYvcP175K8i3d7I?e=6cRylv>

## 5.2 Performance Testing

### 5.2.1 Performance testing cho API Login

Cấu hình:

- Number of Threads: 100
- Ramp-up period: 10
- Loop count: 5
- Body data gồm có “email” và “password” dùng để đăng nhập
- Request có phương thức Post



Hình 5.2.1.1 Graph Result cho API login

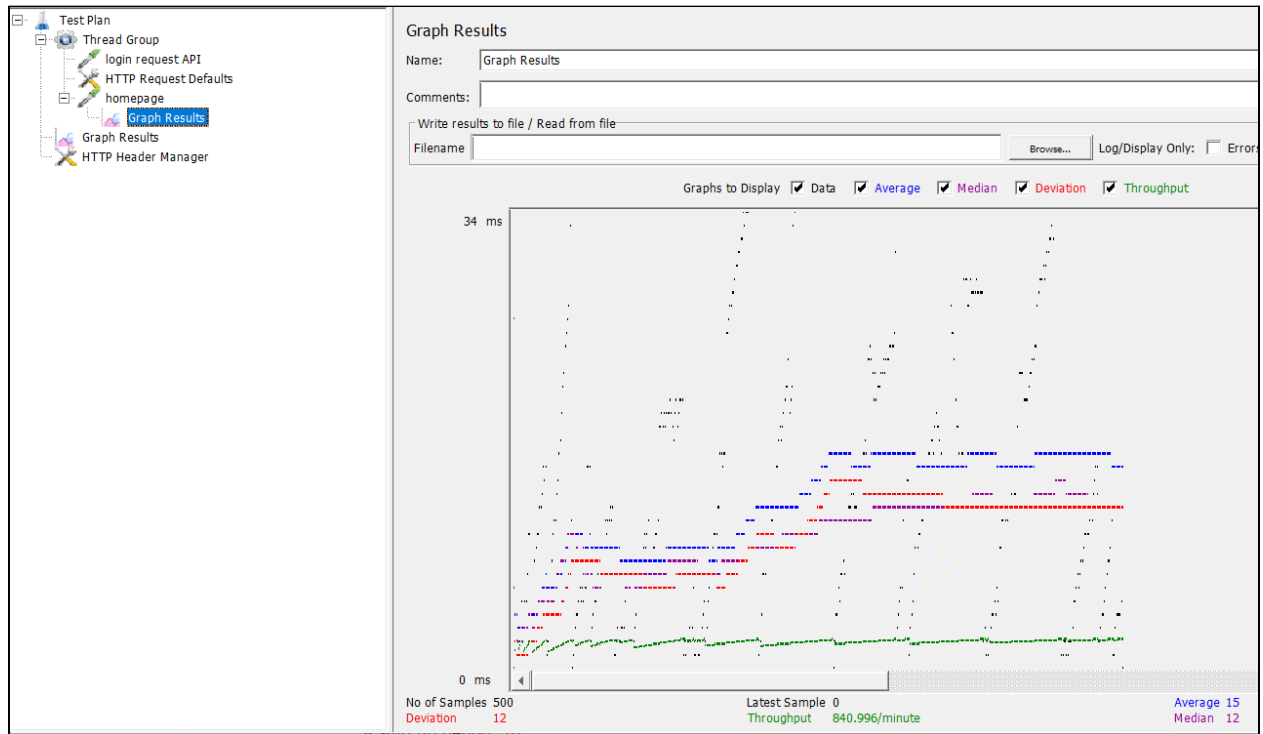
2 tham số quan trọng ta cần quan tâm là Throughput và Deviation

- Throughput: là một tham số rất quan trọng, thể hiện mức độ chịu tải nặng mà server có thể xử lý được, tham số này càng lớn thì hiệu suất server càng lớn. Công thức được tính là: (số lượng requests/ tổng thời gian). Ở hình trên thì server có thể xử lý được 822.684 requests/phút, khá là lớn bởi vì đang được test trên máy localhost.
- Deviation: Thể hiện độ sai lệch với trung tâm, thông số này càng nhỏ thì sẽ càng tốt.

## 5.2.2 Performance testing cho Webpage

Cấu hình:

- Number of Threads: 100
- Ramp-up period: 10
- Loop count: 5
- Request có phương thức GET



Hình 5.2.2.1 Graph Result cho MainPage

Thông số Throughput rất cao và Deviation rất thấp, chứng tỏ khả năng xử lý rất tốt, có thể giải thích rằng bởi vì đối với request này, server chỉ trả về dạng landing page, không cần những chi phí tính toán phức tạp.