**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**======\*\*\*======**

****

BÁO CÁO BÀI TẬP LỚN HỌC PHẦN:

KIỂM THỬ PHẦN MỀM

**KIỂM THỬ PHẦN MỀM – WEBSITE BÁN SÁCH ADCBOOK.NET.VN**

|  |  |
| --- | --- |
| Giáo viên: | ThS. Nguyễn Thái Cường |
| Nhóm: | 7 |
| Thành viên: | Nguyễn Mỹ Hạnh |
|  | Nguyễn Hồng Quân |
|  | Nguyễn Trọng Trường |
|  | Nguyễn Quang Vũ |
| Lớp: | 20232IT6084003 – K16 |

Hà Nội, năm 2024

LỜI CẢM ƠN

Ngày nay, với sự phát triển như vũ bão của công nghệ thông tin, những ứng dụng phần mềm đã trở thành một phần quan trọng của cuộc sống hàng ngày. Các phần mềm được áp dụng phổ biến trong mọi ngành nghề và cả trong đời sống. Để có những phần mềm, ứng dụng chất lượng không thể thiếu việc kiểm thử phần mềm trong quá trình phát triển phần mềm. Kiểm thử có thể coi là một khâu không thể thiếu trong mỗi giai đoạn phát triển mỗi chức năng của một phần mềm hay ứng dụng di động. Do đó, chúng em đã quyết định thực hiện đề tài: “Kiểm thử phần mềm – website bán sách adcbook.net.vn”.

Với lòng biết ơn sâu sắc nhất, chúng em xin gửi đến quý thầy cô khoa Công nghệ thông tin – trường Đại học Công nghiệp Hà Nội đã truyền đạt vốn kiến thức quý báu cho chúng em trong suốt thời gian diễn ra môn học. Nhờ có những lời hướng dẫn, dạy bảo của các thầy cô nên đề tài nghiên cứu của chúng em mới có thể hoàn thiện tốt đẹp.

Đặc biệt, chúng em xin chân thành cảm ơn thầy Nguyễn Thái Cường – người thầy đã trực tiếp dạy bảo, giúp đỡ tận tình, quan tâm, hướng dẫn hết mình và tạo mọi điều kiện thuận lợi cho nhóm trong suốt quá trình học tập và hoàn thiện đề tài nghiên cứu.

Kiến thức của chúng em còn hạn chế nên không tránh khỏi những thiếu sót, chúng em rất mong nhận được những ý kiến đóng góp quý báu của thầy để kiến thức của chúng em trong lĩnh vực này được hoàn thiện hơn đồng thời có điều kiện bổ sung, nâng cao kiến thức của mình.

*Chúng em xin chân thành cảm ơn!*

MỤC LỤC

[Chương 1. Giới thiệu kiểm thử phần mềm 3](#_Toc167742169)

[1.1 Tổng quan về kiểm thử phần mềm 3](#_Toc167742170)

[1.2 Quy trình kiểm thử phần mềm 3](#_Toc167742171)

[1.3 Các cấp độ kiểm thử 4](#_Toc167742172)

[1.4 Các loại hình kiểm thử 6](#_Toc167742173)

[1.5 Các kỹ thuật kiểm thử 7](#_Toc167742174)

[Chương 2. Giới thiệu công cụ LoadForcus 10](#_Toc167742175)

[2.1 Giới thiệu công cụ 10](#_Toc167742176)

[2.2 Lý do chọn đề tài 10](#_Toc167742177)

[2.3 Đặc điểm 11](#_Toc167742178)

[2.4 Cài đặt và sử dụng công cụ LoadForcus 12](#_Toc167742179)

[Chương 3. Giới thiệu phần mềm adcbook.net.vn 13](#_Toc167742180)

[3.1 Đặc tả hệ thống 13](#_Toc167742181)

[Chương 4. Kiểm thử phần mềm adcbook.net.vn 14](#_Toc167742182)

[4.1 Thực hiện kiểm thử 14](#_Toc167742183)

[4.1.1 Kiểm thử tải- Load Testing (Nguyễn Mỹ Hạnh) 14](#_Toc167742184)

[4.1.2 Kiểm thử giám sát tốc độ trang web- Page speed Monitoring (Nguyễn Trọng Trường) 25](#_Toc167742185)

[4.1.3 Kiểm thử API với jmeter bằng LoadFocus (Nguyễn Quang Vũ) 31](#_Toc167742186)

[Phụ lục 43](#_Toc167742187)

[Phụ lục hình ảnh 43](#_Toc167742188)

[Phụ lục bảng biểu 43](#_Toc167742189)

[Tài liệu tham khảo 44](#_Toc167742190)

# Giới thiệu kiểm thử phần mềm

## Tổng quan về kiểm thử phần mềm

Kiểm thử phần mềm là quá trình đánh giá một phần mềm hoặc một hệ thống để xác định xem chúng hoạt động như mong đợi hay không.

Trong quy trình phát triển phần mềm xuất hiện 3 yếu tố chính ảnh hưởng đến chất lượng phần mềm (con người, quy trình, chất lượng) gồm: khoảng cách giữa yêu cầu người dùng và bản đặc tả, khoảng cách giữa bản đặc tả và sản phẩm, và khoảng cách giữa yêu cầu người dùng và sản phẩm. Ví dụ, khi có sự thay đổi yêu cầu trong quá trình phát triển nhưng không được cập nhật vào bản đặc tả. Điều này có thể dẫn đến việc phần mềm không đáp ứng đúng yêu cầu của khách hàng. Để đảm bảo chất lượng phần mềm, kiểm thử phần mềm là rất quan trọng:

* Kiểm thử giúp tìm ra nhiều lỗi trong giới hạn thời gian và nguồn lực có sẵn.
* Nó chứng minh tính phù hợp của sản phẩm với đặc tả và đảm bảo chất lượng với chi phí và nỗ lực tối thiểu.
* Thiết kế tài liệu kiểm thử có hệ thống và thực hiện một cách hiệu quả và tiết kiệm thời gian.

## Quy trình kiểm thử phần mềm

Quy trình kiểm thử phần mềm một tập các hoạt động, các phương thức mà con người phải làm để thực hiện việc kiểm thử cho một phần mềm hay một hệ thống phần mềm.

Quy trình kiểm thử phần mềm được chia thành 4 giai đoạn:

* Lập kế hoạch kiểm thử (Test planning): là quá trình tạo ra bản kế hoạch kiểm thử.
* Phân tích và thiết kế kiểm thử (Test analysis and design): là hoạt động chuyển các mục tiêu của kiểm thử thành các trường hợp kiểm thử cụ thể.
* Thực hiện kiểm thử (Test executing): là quá trình mà các bộ kiểm thử được thực hiện trên phần mềm thực tế. Điều này bao gồm chạy các kịch bản kiểm thử, ghi lại kết quả kiểm thử, xác minh các kết quả và ghi lại các lỗi và vấn đề phát hiện được.
* Báo cáo và đánh giá kiểm thử (Test report and evaluation): là hoạt động đánh giá kết quả kiểm thử so với mục tiêu đã đề ra.

Cấu trúc của bản kế hoạch kiểm thử cơ bản bao gồm 7 thành phần chính:

* Giới thiệu chung (Introduction).
* Các tiêu chí chấp nhận sản phẩm (Acceotance criteria).
* Các yêu cầu cần kiểm thử (Requirements for test).
* Chiến lược kiểm thử (Test strategy).
* Nguồn lực dành cho kiểm thử (Resources for testing).
* Các mốc kiểm thử (Test milestones).
* Sản phẩm cần bàn giao (Deliverables of test).

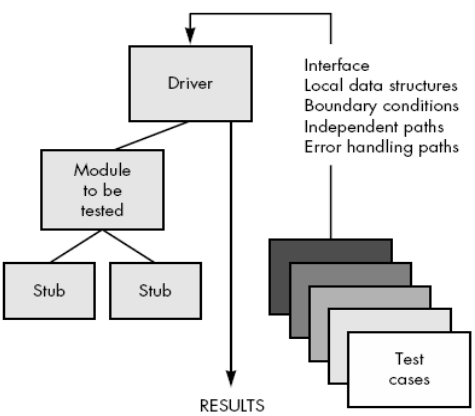
## Các cấp độ kiểm thử

Có 4 cấp độ kiểm thử, gồm:

* Kiểm thử đơn vị (Unit testing).
* Kiểm thử tích hợp (Integration testing).
* Kiểm thử hệ thống (System testing).
* Kiểm thử chấp nhận (Acceptance testing).

*Kiểm thử đơn vị (Unit testing):*

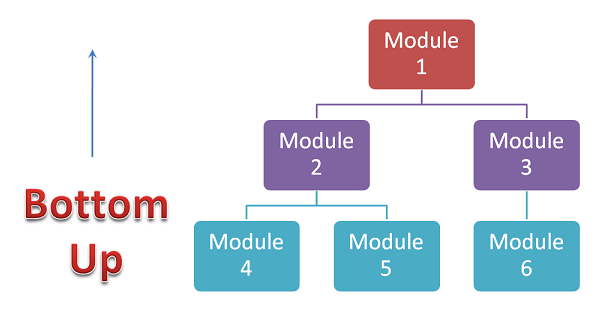
- Nhằm kiểm tra đơn vị thiết kế nhỏ nhất- một module phần mềm. Một module hoạt động thường có trao đổi thông tin với module mức dưới và mức trên nó, do đó phạm vi phát hiện lỗi liên quan chặt chẽ tới module này.



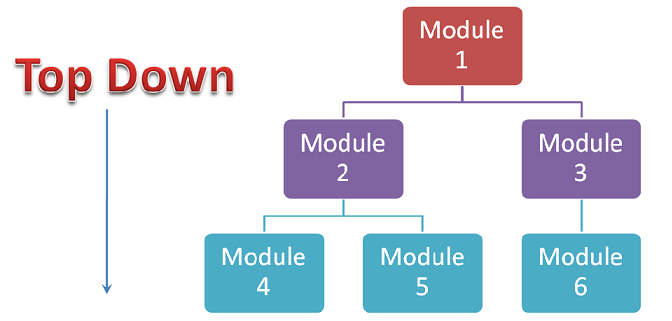
Hình 1.1 Mô hình kiểm thử đơn vị

*Kiểm thử tích hợp (Integration testing):*

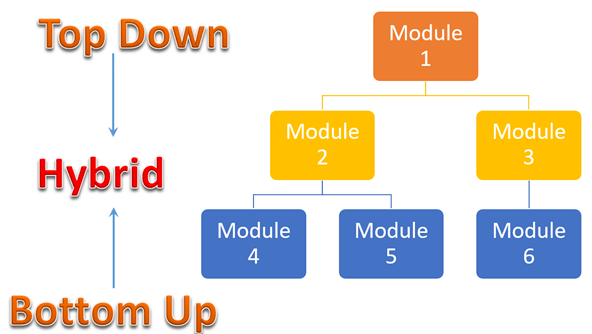
- Nhận các môđun đầu vào đã được kiểm thử đơn vị, nhóm chúng vào những tập hợp lớn hơn, áp dụng các ca kiểm thử đã được định nghĩa trong kế hoạch kiểm thử tích hợp vào tập hợp đó, và cung cấp đầu ra cho hệ thống tích hợp.



Hình 1.2 Mô hình chiến lược tích hợp từ dưới lên (Bottom-up testing)

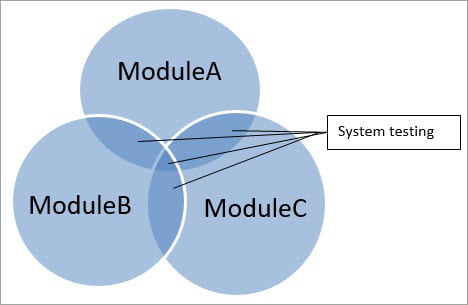


Hình 1.3 Mô hình chiến lược tích hợp từ trên xuống (Top-down testing)



Hình 1.4 Mô hình chiến lược kết hợp (Sandwich testing)

*Kiểm thử hệ thống (System testing):* Kiểm thử thiết kế và toàn bộ hệ thống (sau khi tích hợp) có thỏa mãn yêu cầu đặt ra hay không tìm.



Hình 1.5 Mô hình kiểm thử hệ thống

*Kiểm thử chấp nhận (Acceptance testing):* Vận hành hệ thống trong môi trường của người sử dụng.

A diagram of a flowchart

Description automatically generated

Hình 1.6 Quy trình các bước thực hiện kiểm thử chấp nhận

## Các loại hình kiểm thử

*Kiểm thử chức năng:*

- Là quy trình cố gắng tìm ra các khác biệt giữa đặc tả bên ngoài của phần mềm và thực tế mà phần mềm cung cấp. Đặc tả về bên ngoài của phần mềm là đặc tả chính xác về hành vi của phần mềm theo góc nhìn của người dùng thấy.

- Mục tiêu: đảm bảo đúng mục tiêu của kiểm thử chức năng nhập dữ liệu – xử lý – lấy và kiểm tra kết quả trả về.

- Các loại kiểm thử chức năng:

* Kiểm thử chức năng của hệ thống.
* Kiểm thử tích hợp dữ liệu và cơ sở dữ liệu.
* Kiểm thử vòng lặp công việc.
* Kiểm thử truy cập.
* Kiểm thử giao diện.

*Kiểm thử phi chức năng:*

- Tập trung vào kiểm thử sản phẩm, hệ thống phần mềm cần kiểm thử có những đặc tính tốt như thế nào.

- Sử dụng hiệu quả nhất trong kiểm thử hệ thống và kiểm thử chấp nhận sản phẩm.

- Có 4 loại kiểm thử phi chức năng thường dùng:

* Kiểm thử hiệu năng.
* Kiểm thử tải trọng.
* Kiểm thử tập trung.
* Kiểm thử với dữ liệu lớn.

*Kiểm thử liên quan đến sự thay đổi:*

- Thực hiện hoạt động kiểm thử khi có sự thay đổi trên hoặc trong sản phẩm phần mềm.

- Sự thay đổi của sản phẩm phần mềm có thể là:

* Sửa chữa các lỗi tìm được.
* *S*ản phẩm được nâng cấp được thay đổi về chức năng.

- Gồm: kiểm thử lại và kiểm thử hồi quy.

## Các kỹ thuật kiểm thử

*Kiểm thử hộp đen:*

- Kiểm thử hộp đen (Black box testing) là một phương pháp kiểm thử phần mềm mà việc kiểm tra các chức năng của một ứng dụng không cần quan tâm vào cấu trúc nội bộ hoặc hoạt động của nó.

- Đối tượng được kiểm thử*:* Là thành phần phần mền (TPPM) có thể là 1 hàm chức năng, 1 modul chức năng, 1 phân hệ chức năng…

- Phương pháp thử nghiệm*:* Dựa vào chức năng Kiểm thử hộp đen (Black box test) có thể được áp dụng hầu như đến mọi cấp độ của kiểm thử phần mềm:

* Kiểm thử đơn vị (Unit testing).
* Kiểm thử tích hợp (Intergration testing).
* Kiểm thử hệ thống (System testing).
* Kiểm thử chấp nhận (Acceptance test).

- Đặc điểm:

* Là chiến lược kiểm thử TPPM dựa vào thông tin duy nhất là các đặc tả về yêu cầu chức năng của TPPM tương ứng.
* Người kiểm thử không cần thiết phải có kiến thức về việc mã hoá, cấu trúc bên trong của TPPM, cũng như không yêu cầu phải biết lâp trình phần mềm.
* Việc kiểm thử được tiến hành dựa vào việc kiểm thử TPPM làm được gì, có phù hợp với yêu cầu của người dùng hay không. Các tester nhập số liệu vào phần mềm và chỉ cần xem kết quả của phần mềm và các mục tiêu kiểm tra.
* Tester phải viết test case đầy đủ trước khi test; khi test, đơn giản chỉ cần thực hiện theo các bước mô tả trong test case thao tác và nhập data vào, sau đó xem kết quả trả về hoặc hành vi của phần mềm, rồi so sánh với kết quả mong đọi được viết trong testcase.
* Khi viết test case: Dựa vào yêu cầu và giao diện bên ngoài của chương trình (Không can thiệp vào bên trong code của chương trình).
* Khi thực hiện test: Thực hiện trên giao diện của chương trình (yêu cầu chương trình phải chạy được mới test được, không can thiệp vào code).

*Kiểm thử hộp trắng:*

- Kiểm thử hộp trắng (While box testing) là phương pháp thử nghiệm phần mềm, trong đó các thiết kế, cấu trúc giải thuật bên trong, và việc thực hiện các công việc đều được biết đến.

- Đối tượng kiểm thử: Là 1 thành phần của phần mềm (1 chức năng, 1 module chức năng, 1 phân hệ chức năng....).

- Phương pháp thử nghiệm: Dựa vào thuật giải Kiểm thử hộp trắng dựa vào thuật giải cụ thể, vào cấu trúc dữ liệu bên trong của đơn vị phần mềm cần kiểm thử để xác Định đơn vị phần mềm đó có thực hiện đúng không. Kỹ thuật white box test thích hợp dùng để kiểm thử đơn vị (Unit test).

- Đặc điểm:

* Là chiến lược kiểm thử TPPM dựa vào giải thuật, cấu trúc bên trong chức năng của TPPM tương ứng.
* Người kiểm thử phải có kiến thức nhất định về việc mã hoá, cấu trúc bên trong của chức năng, biết lâp trình phần mềm.
* Việc kiểm thử được tiến hành dựa vào việc kiểm xem giải thuật, mã lệnh đã làm có đúng không.
* Mức test này thường yêu cầu các tester phải viết test case đầy đủ các nhánh trong code; khi test, sẽ set điều kiện và data để chạy vào đủ tất cả các nhánh trong giải thuật, đảm bảo thực hiện đầy đủ.
* Khi viết test case: Dựa vào yêu cầu và nội dung Source Code (can thiệp vào bên trong Code của chương trình).
* Khi thực hiện test: Thực thi test trong code (không cần thực thi chương trình, vì thực hiện test white box sẽ sử dụng framework nào đó hỗ trợ (Ví dụ như test kiểu debug).
* Trong kiểm tra này, đòi hỏi người tester phải có kiến thức và kỹ năng nhất định về ngôn ngữ lập trình được dùng, hiểu thuật giải trong thành phần phần mềm, để có thể hiểu được chi tiết về đoạn code cần kiểm thử.

# Giới thiệu công cụ LoadForcus

## Giới thiệu công cụ

Kiểm thử tự động: là xử lý một cách tự động các bước thực hiện các testcase, kiểm thử tự động bằng một công cụ nhằm rút ngắn thời gian kiểm thử.

Kiểm thử tự động: là một kỹ thuật tự động trong đó người kiểm thử tự viết các tập lệnh và sử dụng phần mềm phù hợp để kiểm thử phần mềm. Nó về cơ bản là một quá trình tự động hóa của một quy trình kiểm thử thủ công. Giống như kiểm thử hồi quy, kiểm thử tự động cũng được sử dụng để kiểm thử ứng dụng theo quan điểm tải, hiệu năng và ứng suất. Kiểm thử tự động giúp giảm chi phí kiểm thử bằng cách hỗ trợ quá trình kiểm thử thông qua các công cụ phần mềm.

Kiểm thử tự động hay sử dụng phần mềm để kiểm thử với các ưu điểm:

* Có thể thực hiện các kiểm thử một cách liên tục, lặp lại và giảm chi phí cho nhân lực kiểm thử.
* Luôn đảm bảo hoạt động theo một kịch bản duy nhất – không bị ảnh hưởng như với kiểm thử viên.

Công cụ kiểm thử LoadFocus là nền tảng thử nghiệm đám mây tất cả trong một dành cho các trang web và API với chức năng kiểm tra tải Apache JMeter và theo dõi tốc độ trang.

Công cụ LoadFocus cung cấp khả năng tạo và chạy các kịch bản kiểm thử tự động, giúp chúng ta kiểm tra tính năng, hiệu suất và độ tin cậy của trang web. Chúng ta có thể tùy chỉnh các kịch bản kiểm thử để mô phỏng các tác vụ như điều hướng, nhập liệu và kiểm tra kết quả.

Tóm lại, công cụ LoadFocus là một công cụ kiểm thử tự động đáng tin cậy và linh hoạt, giúp chúng ta thực hiện kiểm thử tự động và đo lường hiệu suất trang web của mình. Với tính năng và chức năng đa dạng, nó là một lựa chọn phù hợp cho việc đảm bảo chất lượng và tối ưu hóa hiệu suất của ứng dụng web.

## Lý do chọn đề tài

Việc kiểm thử phần mềm là bước không thể thiếu trong quá trình phát triển sản phẩm. Đảm bảo chất lượng và độ tin cậy của phần mềm là một mục tiêu cốt lõi, và việc thực hiện kiểm thử một cách cẩn thận đóng vai trò quan trọng trong việc đạt được điều này Hiện nay, ngành kiểm thử phần mêm đã vô cùng phát triển, đi kèm với đó là hàng loạt các công cụ kiểm thử phần mềm được sinh ra với mục đích đơn giản hoá, cũng như nâng cao chất lượng của phần mềm.

Do đó, chúng em quyết định lựa chọn công cụ LoadForcus để thực hiện đề tài “Kiểm thử phần mềm – website bán sách adcbook.net.vn”. Công cụ kiểm thử Loadfocus được chọn để nghiên cứu và sử dụng trong đề tài này vì tính linh hoạt và hiệu quả của nó. Loadfocus cung cấp các tính năng mạnh mẽ để kiểm thử hiệu suất, kiểm thử tải và kiểm thử chức năng của các ứng dụng web và API.

Qua việc nghiên cứu và sử dụng các kỹ thuật kiểm thử được hỗ trợ bởi công cụ Loadfocus, nhóm em mong muốn nắm vững các phương pháp và tiến trình kiểm thử phần mềm để đảm bảo rằng sản phẩm cuối cùng sẽ đáp ứng được mong đợi của người dùng, mang lại trải nghiệm mua sắm trực tuyến tốt nhất cho người dùng.

## Đặc điểm

LoadFocus là một công cụ mã nguồn mở hỗ trợ ba loại kiểm thử chính:

*1. Load Testing*

- Load testing là kiểm thử hiệu năng để đánh giá hiệu suất của trang web, ứng dụng, hoặc API dưới điều kiện tải bình thường và tải cao. Mục tiêu là xác định hành vi và thời gian phản hồi của hệ thống khi xử lý số lượng người dùng hoặc yêu cầu tải lớn, đảm bảo tính ổn định, đáng tin cậy và khả năng mở rộng.

- Quá trình kiểm thử bao gồm mô phỏng nhiều người dùng đồng thời để đánh giá các thông số như thời gian phản hồi, khả năng xử lý, tài nguyên sử dụng và tính ổn định của hệ thống.

- Các bước thực hiện kiểm thử load testing bao gồm:

* Lập kế hoạch kiểm thử
* Thiết kế kiểm thử
* Thực hiện kiểm thử
* Phân tích hiệu suất
* Tối ưu hóa và kiểm thử lại

- Lợi ích của kiểm thử Load Testing:

* Xác nhận hiệu suất: Đảm bảo thời gian phản hồi và khả năng xử lý của hệ thống.
* Đánh giá khả năng mở rộng: Xác định khả năng tối đa của hệ thống và khi nào cần thêm tài nguyên hoặc nâng cấp.
* Xác định điểm hạn chế: Phát hiện các vấn đề hiệu suất như truy vấn cơ sở dữ liệu chậm, thuật toán không hiệu quả.
* Tính đáng tin cậy và ổn định: Phát hiện các vấn đề ổn định như rò rỉ bộ nhớ, sự cố, sử dụng tài nguyên quá mức.

*2. JMeter Load Testing*

- JMeter Load Testing sử dụng công cụ Apache JMeter để kiểm thử hiệu năng và khả năng mở rộng của ứng dụng web hoặc API. JMeter cho phép tạo và thực thi các kịch bản kiểm thử mô phỏng hành động của người dùng và tạo ra lượng lớn yêu cầu.

- Tính năng: Chạy kịch bản từ nhiều địa điểm khác nhau: Giúp đánh giá hiệu suất và thời gian phản hồi từ các địa điểm địa lý đa dạng.

- Quy trình hoạt động:

* Tạo kịch bản kiểm thử: Mô phỏng hành động của người dùng như điều hướng, gửi biểu mẫu, tương tác với API.
* Cấu hình kiểm thử: Xác định số lượng người dùng đồng thời, thời gian tăng dần, thời lượng kiểm thử.
* Kiểm thử phân tán: Chạy kịch bản từ nhiều máy chủ.
* Thực thi kiểm thử: JMeter thực thi kịch bản và tạo ra tải mong muốn.
* Phân tích kết quả: JMeter cung cấp các chỉ số hiệu suất và kết quả chi tiết.

*3. Page Speed Monitoring*

- Page speed monitoring là quá trình theo dõi liên tục và đo lường hiệu suất tải trang của một trang web, sử dụng công cụ LoadFocus.

- Tổng quan hoạt động:

* Performance Metrics: Đo lường các chỉ số hiệu suất quan trọng như thời gian tải trang, FCP, LCP, TTI.
* Test Execution: Chạy kiểm tra trên các trang web cụ thể hoặc thiết lập giám sát tự động để đánh giá hiệu suất theo thời gian.
* Insights and Recommendations: Cung cấp thông tin và đề xuất để cải thiện hiệu suất trang.
* Historical Performance Tracking: Theo dõi hiệu suất lịch sử của trang web.

- Lợi ích của Page Speed Monitoring:

* Cải thiện trải nghiệm người dùng: Giúp giảm tỷ lệ thoát, tăng cường tương tác và cải thiện sự hài lòng của người dùng.
* Tối ưu hóa SEO: Đảm bảo trang web đáp ứng yêu cầu hiệu suất của công cụ tìm kiếm, cải thiện khả năng tìm thấy và thứ hạng.
* Tối ưu hóa hiệu suất: Cung cấp thông tin chi tiết và đề xuất để cải thiện tốc độ và hiệu suất trang web, như giảm kích thước tệp, tối ưu hóa tải tài nguyên và chiến lược bộ nhớ cache.

## Cài đặt và sử dụng công cụ LoadForcus

* Công cụ kiểm thử Loadforcus là một nền tảng thử nghiệm đám mây tất cả trong một dành cho các trang web và API. Công cụ được xây dựng trang web có tên là loadforcus.com.
* Công cụ Loadforcus có thể được truy cập với mọi thiết bị với điều kiện thiết bị đó có kết nối mạng.
* Cách sử dụng cũng rất đơn giản: Người dùng chỉ cần truy cập vào trang loadforcus.com sau đó đăng ký 1 tài khoản. Sau đó chọn 1 bài test cho trang web của mình, cập nhật các thông số. Nhập địa chỉ các trang web và bắt đầu. Sau đó Loadforcus sẽ tự động chạy các bài test.

# Giới thiệu phần mềm adcbook.net.vn

## Đặc tả hệ thống

* Mục đích: Trang web được xây dựng với mục đích bán sách trực tuyến với nhiều thể loại sách đa dạng.
* Phạm vi: Website sẽ được truy cập qua trình duyệt web trên máy tính và các thiết bị di động có kết nối Internet.
* Đối tượng sử dụng: người chủ tiệm sách, nhân viên, khách hàng.
* Yêu cầu về chức năng: Trang web có đầy đủ tất cả các chức năng thiết yếu mà nhân viên và khách hàng cần.

*Yêu cầu hiệu năng:*

* Thời gian phản hồi: Website cần đáp ứng nhanh chóng khi người dùng thực hiện tìm kiếm, xem thông tin sản phẩm, đặt hàng và thanh toán.
* Tải cao: Trang web có thể chịu tải được số lượng người dùng truy cập vào cùng một lúc lớn.
* Độ tin cậy: Website cần hoạt động ổn định và không có lỗi khi có đồng thời rất nhiều người dùng cùng truy cập.

*Yêu cầu giao diện người dùng:*

* Giao diện đơn giản, đẹp, dễ sử dụng và thân thiện với người dùng.
* Hiển thị danh sách đầy đủ thông tin, chi tiết, danh mục rõ dàng, dễ dàng tìm kiếm sản phẩm và dễ sử dụng cho bất kỳ người dùng nào.
* Cung cấp các bộ lọc và tùy chọn sắp xếp thông minh để giúp người dùng tìm kiếm dễ dàng hơn.
* Giao diện đáp ứng để hiển thị tốt trên các thiết bị di động, máy tính bảng, máy tính.

# Kiểm thử phần mềm adcbook.net.vn

## Thực hiện kiểm thử

### Kiểm thử tải- Load Testing (Nguyễn Mỹ Hạnh)

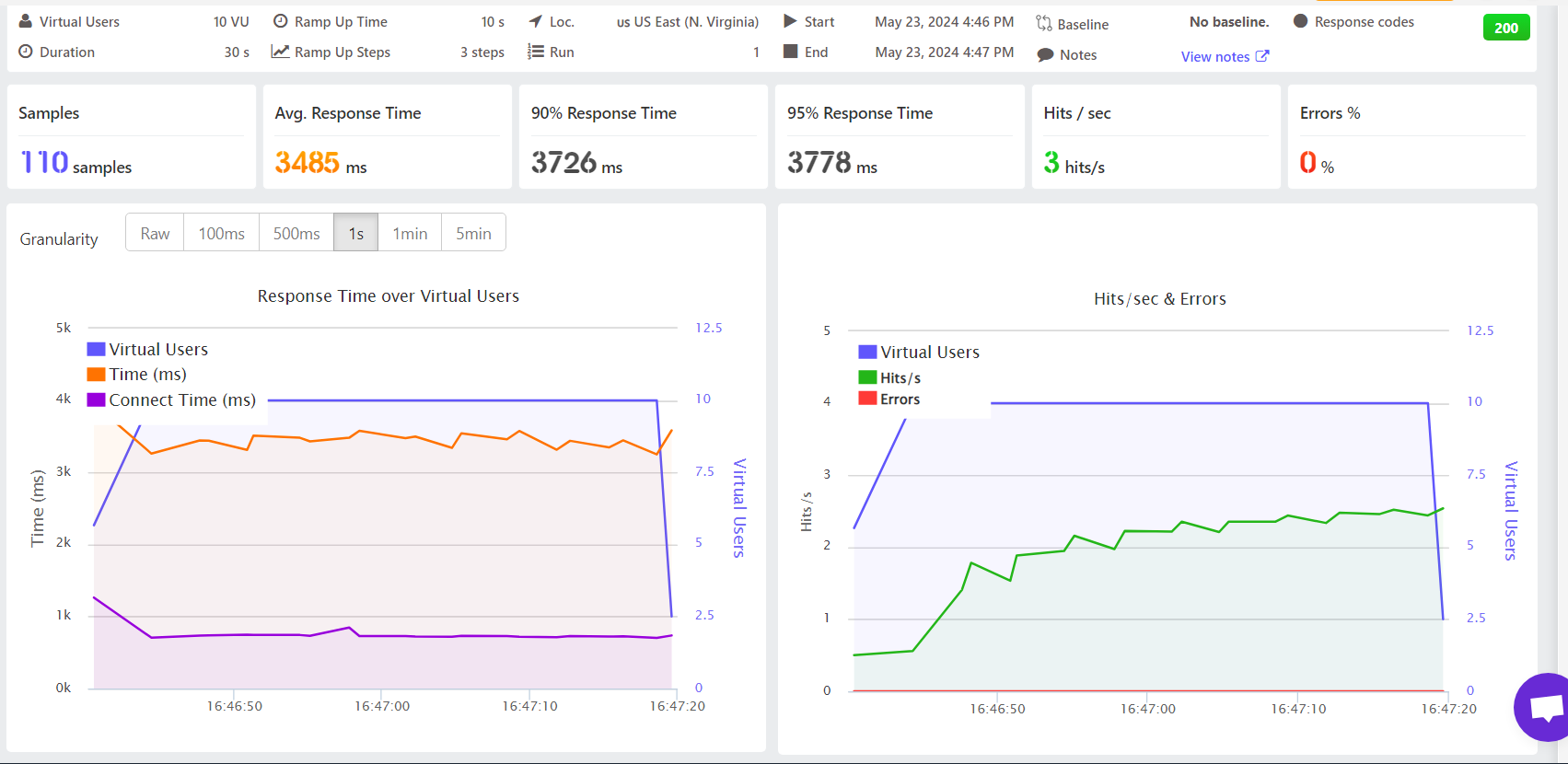
#### Kế hoạch kiểm thử

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TC | Function | Test case Name | Method | Step | Expected Output | Actual Output |
| TC01 | Kiểm tra hiệu suất trang web | Kiểm thử tải để tìm kiếm các sự cố cấu hình | Bằng cách chạy thử nghiệm với một số lượng nhỏ người dùng ảo. (phương thức Get) | 1. 1.Tại màn hình của trang web <https://loadfocus.com/newtest> chọn chức năng “INITIAL LOAD”. 2. 2.Cài đặt : 3. + Virtual Users :10 + Duration(s) :30 4. + Ramp up time:10 5. + Ramp up steps: 3 6. 3. Web Page URL: [https://adcbook.net.vn/](https://www.haui.edu.vn/vn) 7. vào mục URL. 8. 4.Chọn EXECUTE TEST | Hiển thị kết quả sau khi chạy | Hiển thị kết quả sau khi chạy |
| TC02 | Kiểm tra hiệu suất trang web | Tìm ra các vấn đề về thời gian tải và phản hồi cũng như các lỗi tiềm ẩn | Bằng cách kiểm tra các mã phản hồi. (phương thức Get) | 1.Tại màn hình của trang web <https://loadfocus.com/newtest> chọn chức năng “WARM-UP LOAD”.  2.Cài đặt :  + Virtual Users :10  + Duration(s) :30  + Ramp up time:10  + Ramp up steps: 2  3. Web Page URL: [https://adcbook.net.vn/](https://www.haui.edu.vn/vn)  4.Chọn EXECUTE TEST | Hiển thị kết quả sau khi chạy | Hiển thị kết quả sau khi chạy |
| TC03 | Kiểm tra hiệu suất trang web | Tìm ra các vấn đề với một thiết đặt vừa phải cho điểm cuối đang thử nghiệm. | Bằng cách theo dõi thời gian phản hòi và kiểm tra lỗi/ thất bại(Phương thức Get) | 1.Tại màn hình của trang web <https://loadfocus.com/newtest> chọn chức năng “MODERATE LOAD”.  2.Cài đặt :  + Virtual Users :10 + Duration(s) :30  + Ramp up time:10  + Ramp up steps: 3  3. Web Page URL: [https://adcbook.net.vn/](https://www.haui.edu.vn/vn)  4.Chọn EXECUTE TEST | Hiển thị kết quả sau khi chạy | Hiển thị kết quả sau khi chạy |

#### Báo cáo kết quả kiểm thử

**TC01: Kiểm thử tải để tìm kiếm các sự cố cấu hình**

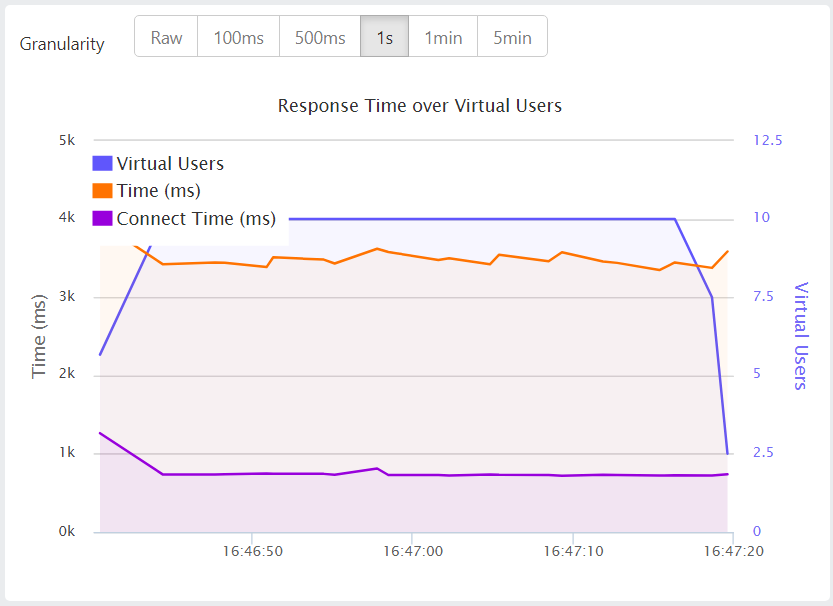
* Kết quả kiểm thử tổng quan



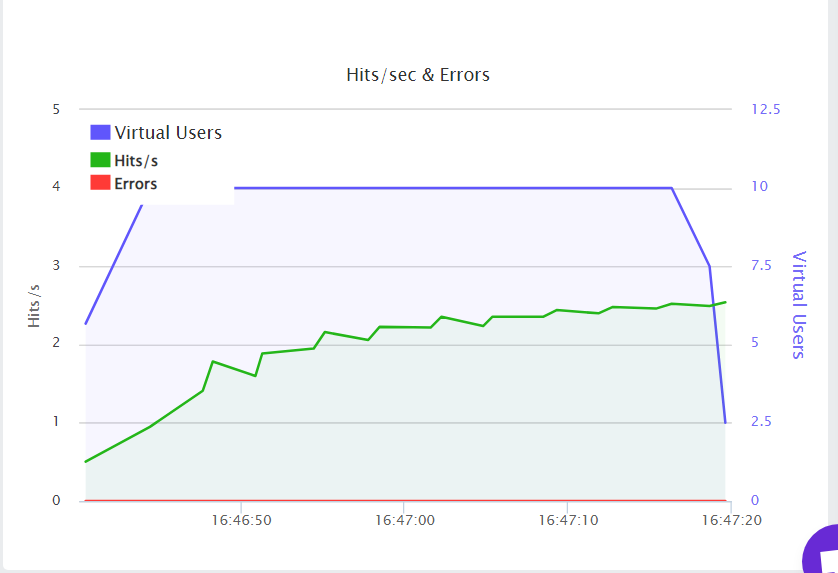
* Chú thích các thông số có trong kết quả kiểm thử

|  |  |  |
| --- | --- | --- |
| STT | Thông số | Chú thích |
| 1 | Virtual users | Số lượng người dùng ảo được khởi tạo |
| 2 | Duration | Tổng thời gian chạy, thời gian mà người dùng sẽ truy cập vào trang web. |
| 3 | Ramp Up Time | Độ trễ giữa lúc bắt đầu kiểm tra cho đến khi tất cả người dùng ảo đang chạy |
| 4 | Ramp up steps | Số bước cho thời gian tăng tốc |
| 5 | Location | Nơi server chạy |
| 6 | Run | Lần bắt đầu |
| 7 | Start time | Thời gian bắt đầu ca kiểm thử |
| 8 | End time | Thời gian kết thức ca kiểm thử |
| 9 | Baseline | Đường cơ sở |
| 10 | Notes | Chú thích |
| 11 | Response codes | Mã phản hồi |
| 12 | Samples | Tổng số lượng requests đã gửi đến server trong thời gian gian chạy ca kiểm thử |
| 13 | Avg. Response | Thời gian trung bình, khoảng thời gian tính từ thời điểm người dùng gửi yêu cầu cho đến thời điểm wungs dụng báo rằng yêu cầu đã hoàn thành. Giá trị trung bình số học cho tất cả các phản hồi (tổng của tất cả thời gian/số lượng phản hồi) |
| 14 | 90% Respone Time | 90% thời gian hỏi hồi hoặc phân vị thứ 90 là giá trị mà 90% phản hồi (điểm dữ liệu) nhỏ hơn và 10% lớn hơn |
| 15 | 95% Respone Time | 95% thời gian hỏi hồi hoặc phân vị thứ 90 là giá trị mà 90% phản hồi (điểm dữ liệu) nhỏ hơn và 5% lớn hơn |
| 16 | Hits/ sec | Lượt truy cập / giây cho biết số lượt truy cập mà ứng dụng phải chịu mỗi giây. giá trị trung bình sẽ được hiển thị |
| 17 | Errrors | Tổng số lỗi gặp phải khi chạy kiểm tra đối với URL đã nhập (mã phản hồi 4xx và 5xx hoặc các lỗi kết nối khác). Tổng gái trị hiển thị |
| 18 | Granularity | Độ chi tiết, để hiển thị trên biểu đồ tất cả các thử nghiệm. Nếu không, các giá trị trung bình sẽ được hiển thị trên biểu đồ, được tính trung bình cứ sau 100 mili giây, 500 mili giây, 1 giây, 1 phút hoặc 5 phút. |

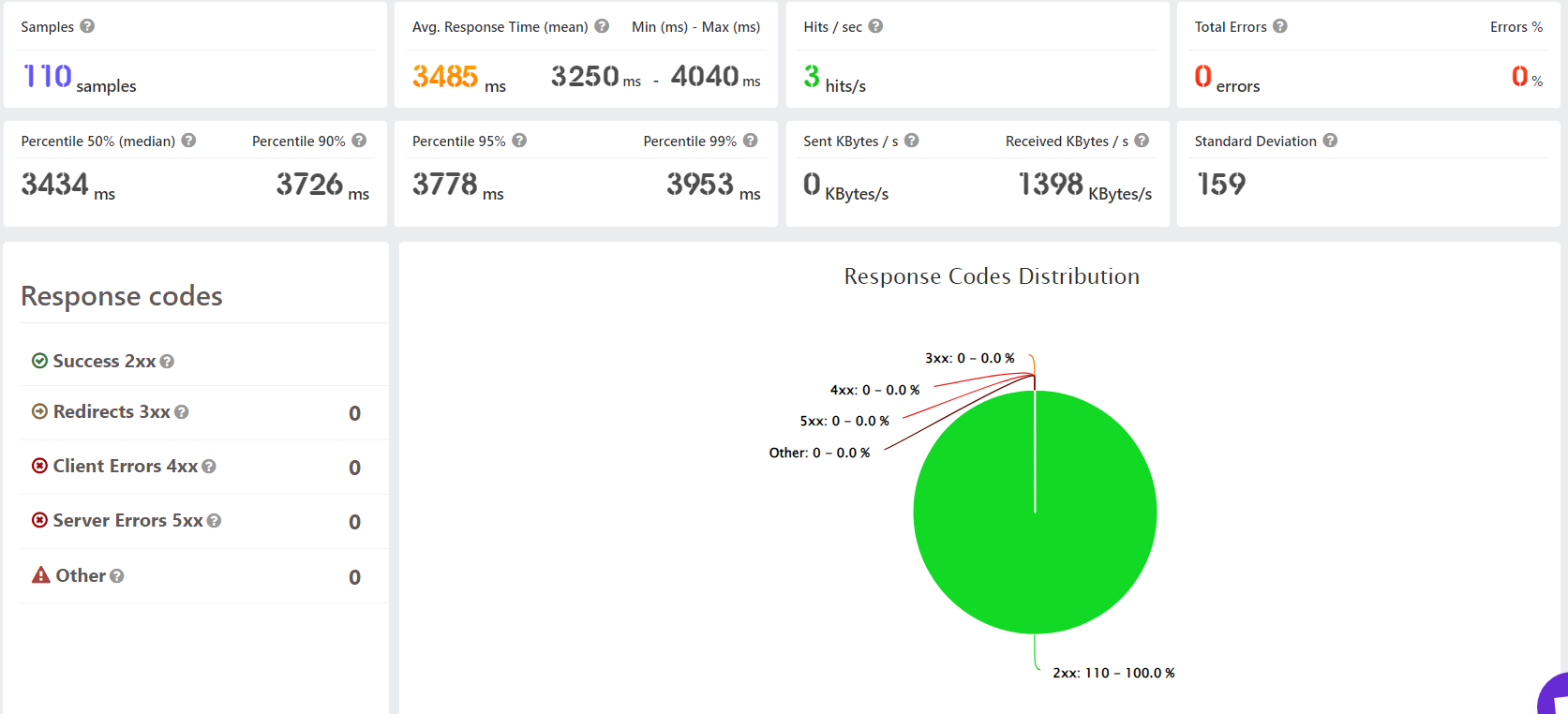
* Đồ thị thể hiện mức độ thời gian trung bình phản hồi với số lượng users truy cập:

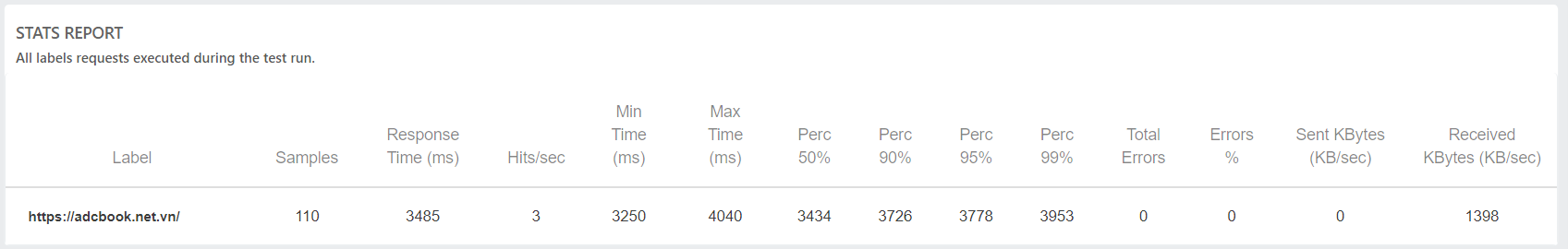


* Đồ thị thể hiện mức độ lượt truy cập của user và mức độ lỗi theo thời gian:



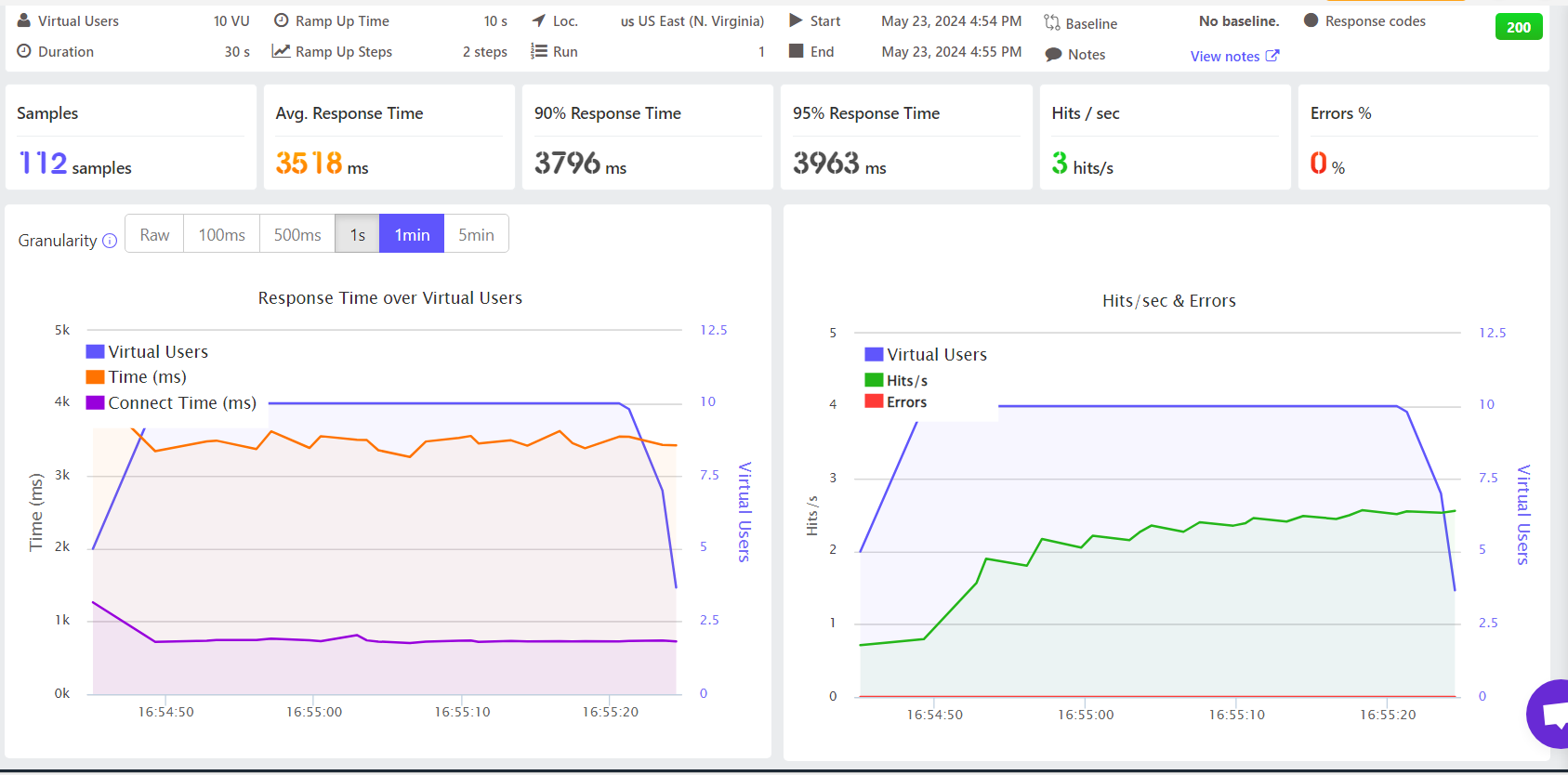
* Chi tiết các thông số cụ thể trong TC01



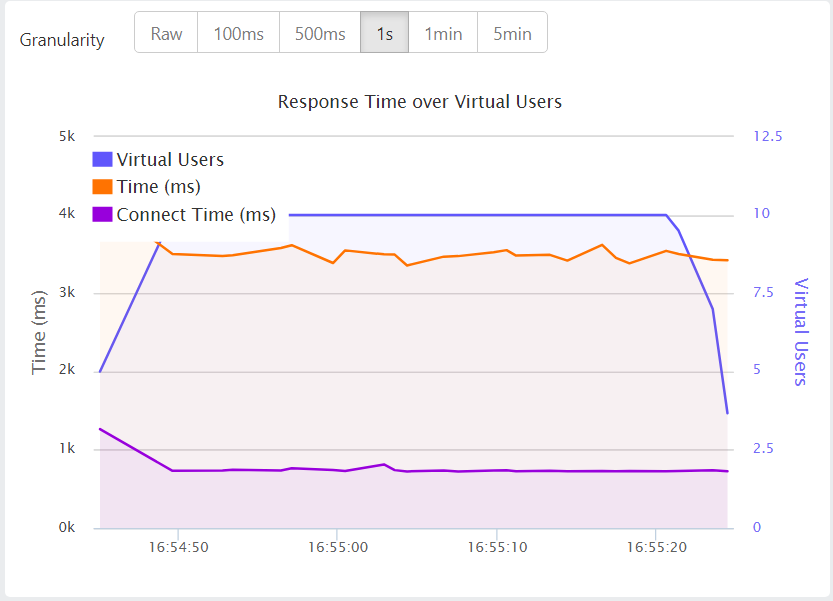


**TC02: Kiểm thử tìm ra các vấn đề về thời gian tải và phản hồi cũng như các lỗi tiềm ẩn**

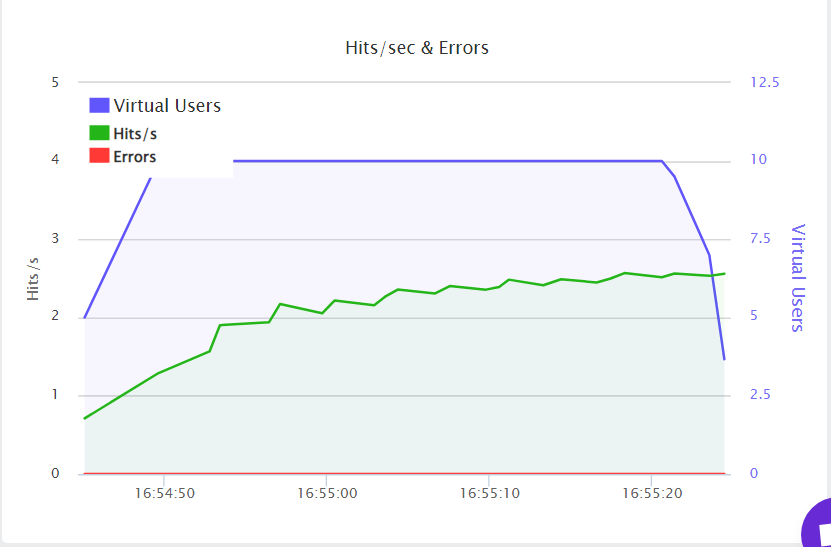
* Kết quả kiểm thử tổng quan



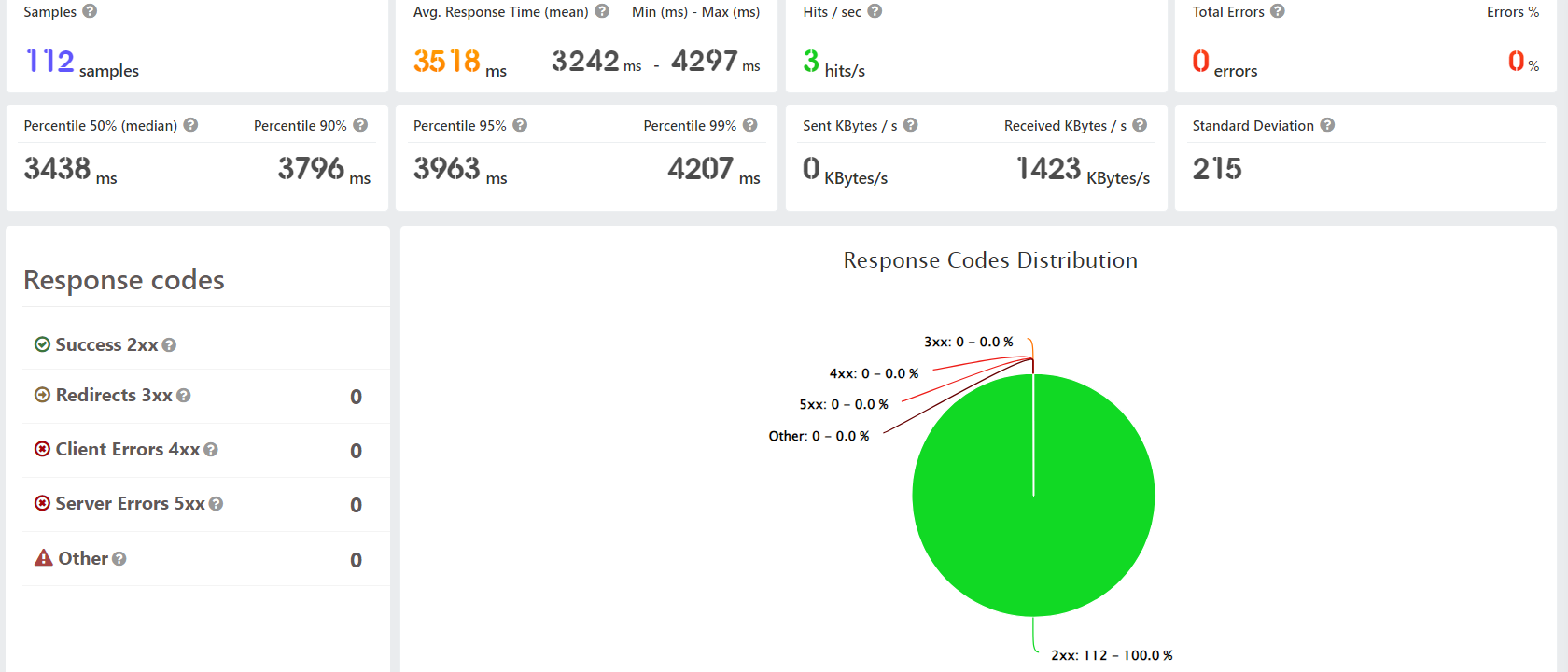
* Đồ thị thể hiện mức độ thời gian trung bình phản hồi với số lượng users truy cập:

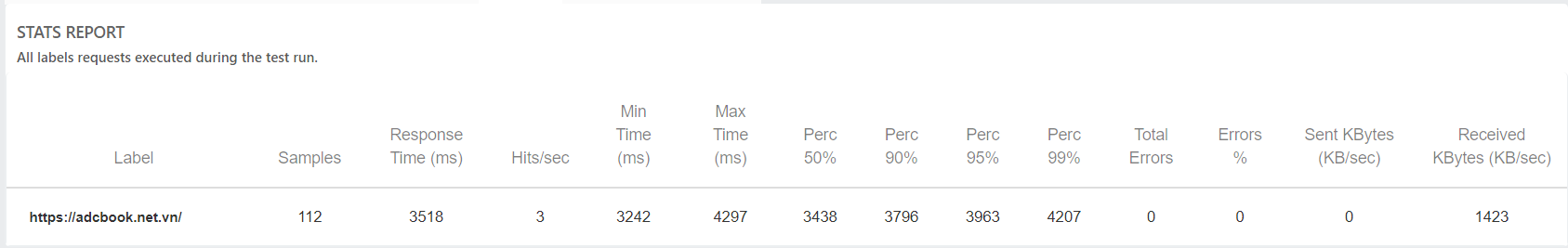


* Đồ thị thể hiện mức độ lượt truy cập của user và mức độ lỗi theo thời gian:

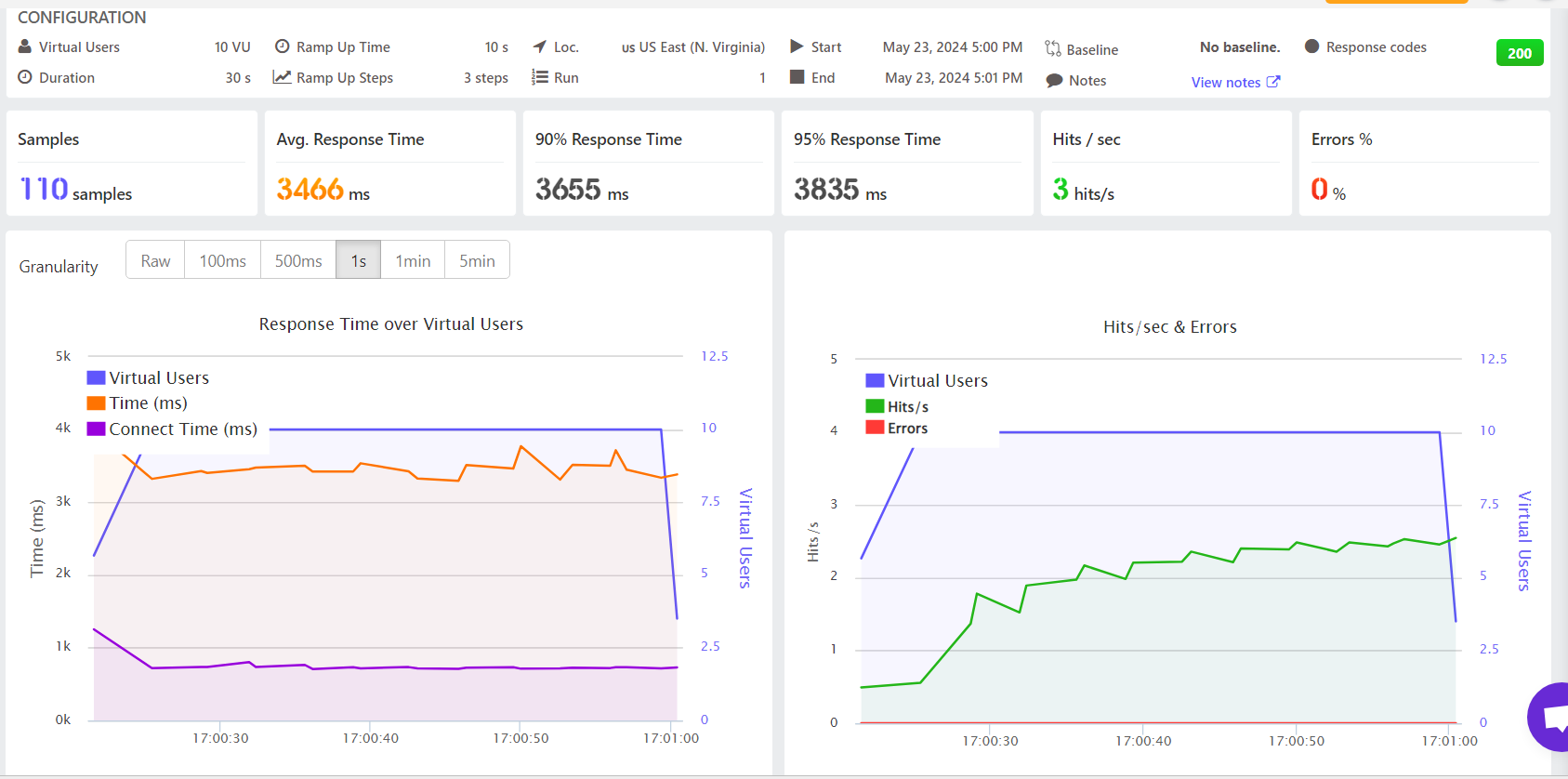


* Chi tiết các thông số cụ thể trong TC02

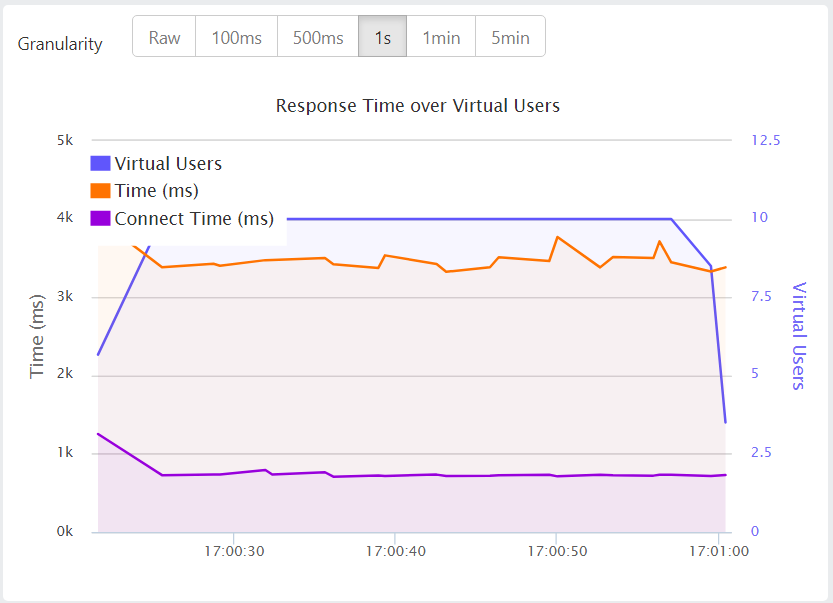


**TC03: Kiểm thử tìm ra các vấn đề với một thiết đặt vừa phải cho điểm cuối đang thử nghiệm.**

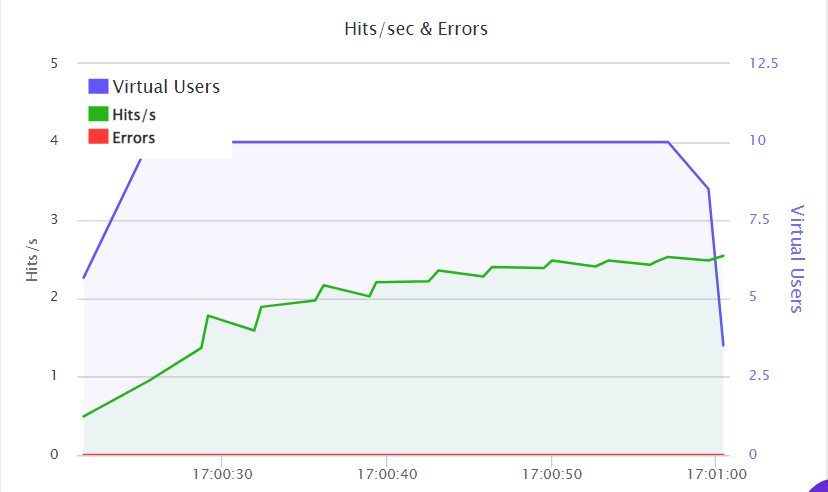
* Kết quả kiểm thử tổng quan



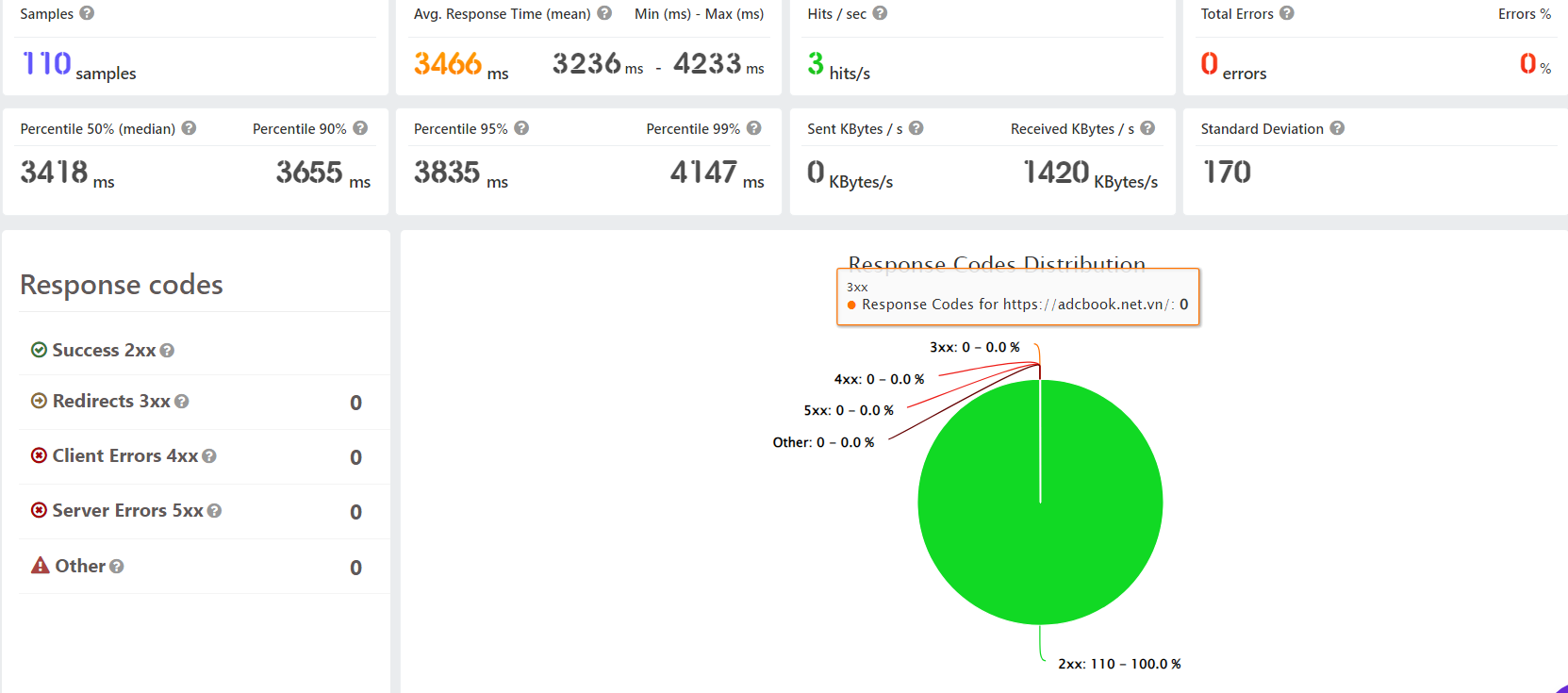
* Đồ thị thể hiện mức độ thời gian trung bình phản hồi với số lượng users truy cập:

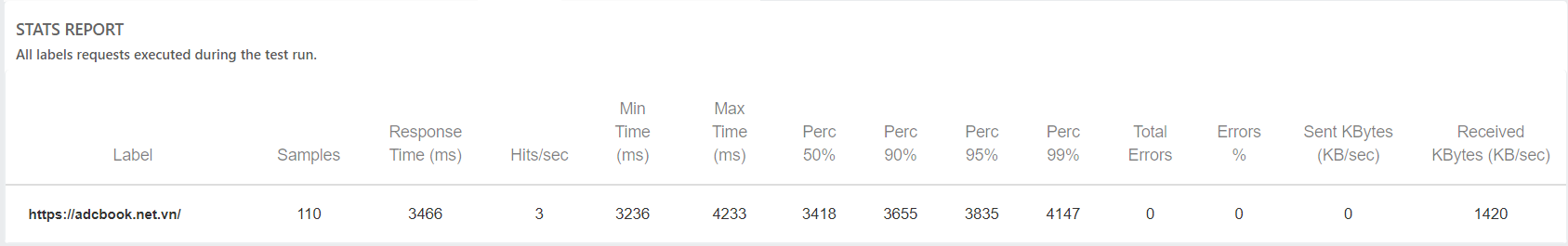


* Đồ thị thể hiện mức độ lượt truy cập của user và mức độ lỗi theo thời gian:



* Chi tiết các thông số cụ thể trong TC03:





### Kiểm thử giám sát tốc độ trang web- Page speed Monitoring (Nguyễn Trọng Trường)

#### Kế hoạch kiểm thử

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TC | Function | Test case Name | Method | Step | Expected Output | Actual Output |
| TC01 | Giám sát tốc độ trang web | Kiểm thử trên nền tảng các thiết bị Desktop | Sử dụng chức năng Page speed Monitoring trên loadfocus | 1. 1.Tại màn hình chức năng của trang web <https://loadfocus.com/newtest> chọn “New Test” 2. 2.Cài đặt:   +Monitor Name: May\_25 2024 2:55 PM  + Location: us US East(N.Virginia)  + Web Page URL: [https://adcbook.net.vn/](https://www.haui.edu.vn/vn)  + **Device**: Desktop  + **Schedule:None**   1. 3. Chọn EXECUTE TEST | Hiển thị kết quả sau khi chạy | Hiển thị kết quả sau khi chạy |
| TC02 | Kiểm tra hiệu suất trang web | Kiểm thử trên nền tảng các thiết bị Di động | Sử dụng chức năng Page speed Monitoring trên loadfocus | 1.Tại màn hình chức năng của trang web <https://loadfocus.com/newtest> chọn “New Test”  2.Cài đặt :  +Monitor Name: May\_25 2024 3:00 PM  + Location: us US East(N.Virginia)  + Web Page URL: [https://adcbook.net.vn/](https://www.haui.edu.vn/vn)  + **Device**: mobie  + **Schedule:None**   1. 3. Chọn EXECUTE TEST | Hiển thị kết quả sau khi chạy | Hiển thị kết quả sau khi chạy |

#### Báo cáo kết quả kiểm thử

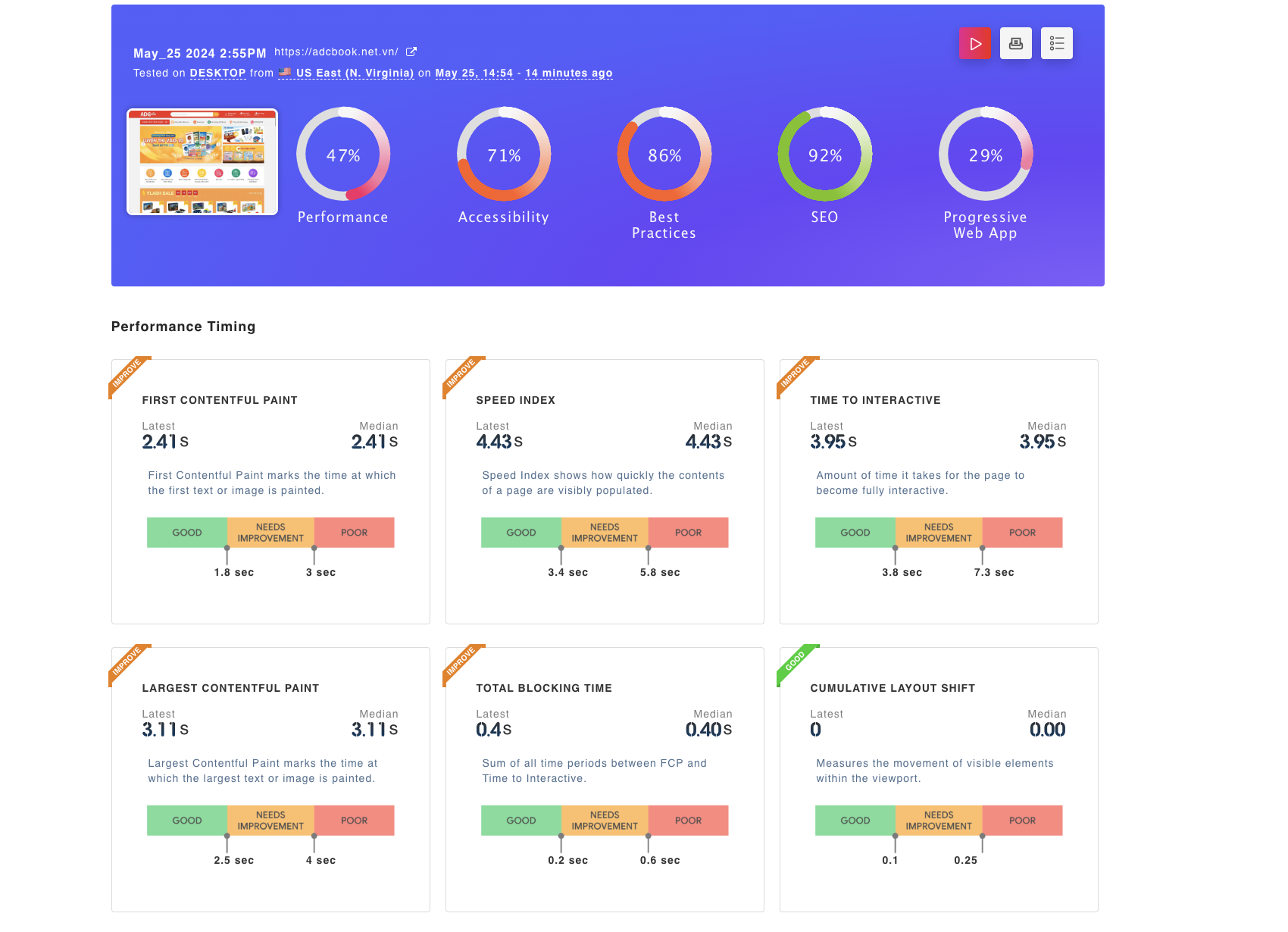
* Nội dung ca kiểm thử: Kiểm tra tốc độ hiển thị nội dung trang web trên những thiết bị khác nhau (Moblie và Desktop), xem tốc độ phản hồi và hiệu suất của trang web.
* Dưới đây là màn hình tổng quan sau khi chạy xong ca kiểm thử:

Chú thích

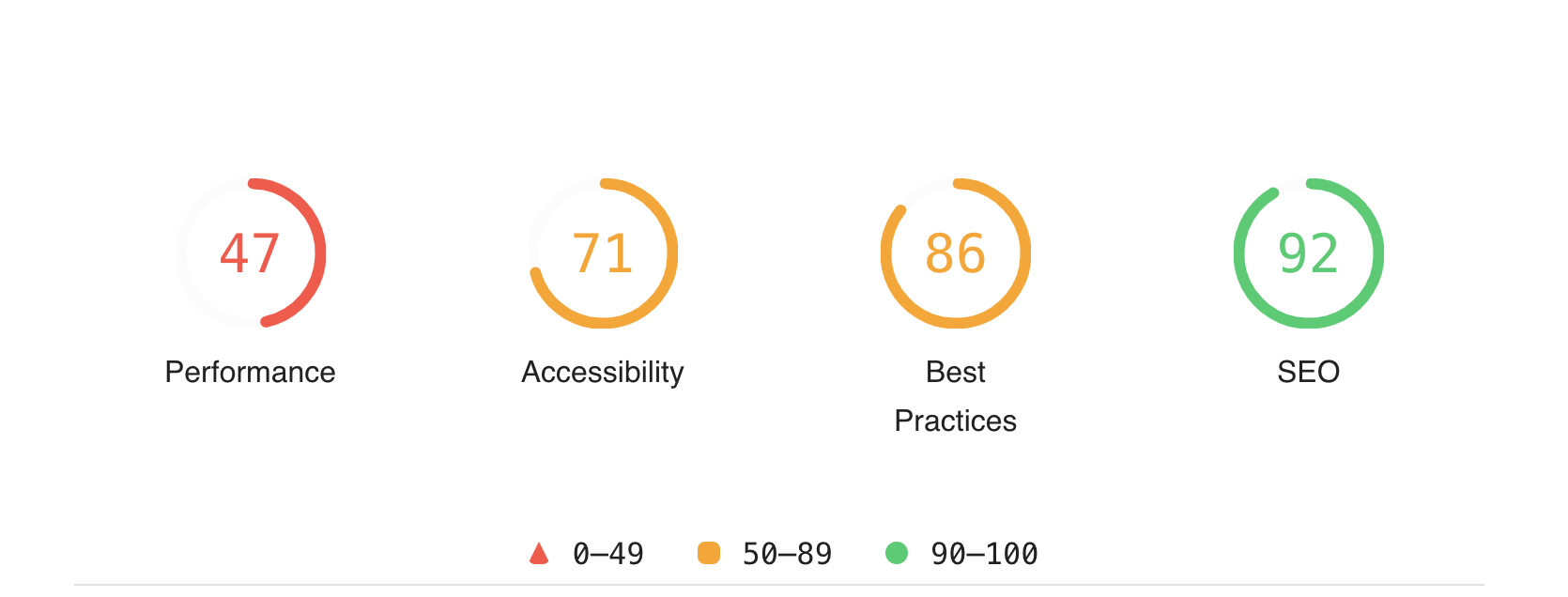
|  |  |  |
| --- | --- | --- |
| STT | Thông số | Chú thích |
| 1 | Performance | Hiệu suất trang web (tối đa là 100) |
| 2 | Accessibility | Khả năng tiếp cận |
| 3 | Best practices | Thực hành tốt nhất |
| 4 | SEO | Tối ưu hóa trên công cụ tìm kiếm |
| 5 | Progressive Web App | Các tính năng khác của web |
| 6 | Performance Timing | Thời gian thực hiện |
| 7 | First Contentful Paint | Thời gian tải xong nội dung đầu tiên lên trang web |
| 8 | Speed Index | Tốc độ hiển thị nội dung của trang |
| 9 | Time To Interactive | Thời gian cần thiết để trang có thể tương tác được |
| 10 | Largest Contentful Paint | Thời giản tải xong những nội dung có dung lượng lớn  lên trang web |
| 11 | Total Blocking Time | Tổng thời gian chặn |
| 12 | Cumulative Layout Shift | Thay đổi của bố cục khung nhìn |
| 13 | Last Run Timeline | Dòng thời gian chạy lần cuối của trang (các giá trị có thể thay đổi) |

**TC01: DESKTOP**

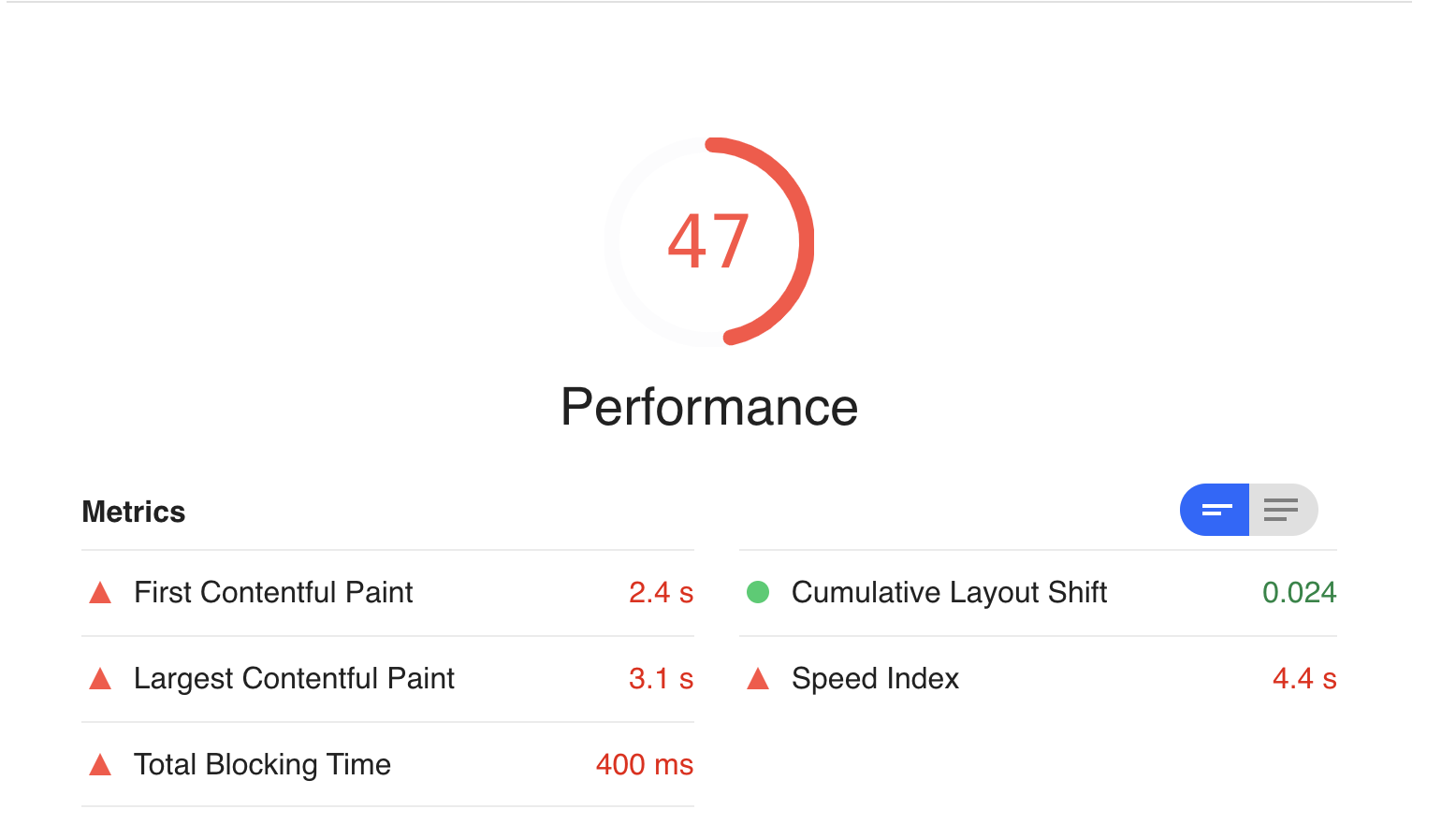
Giao diện chi tiết khi chạy xong ca kiểm thử:



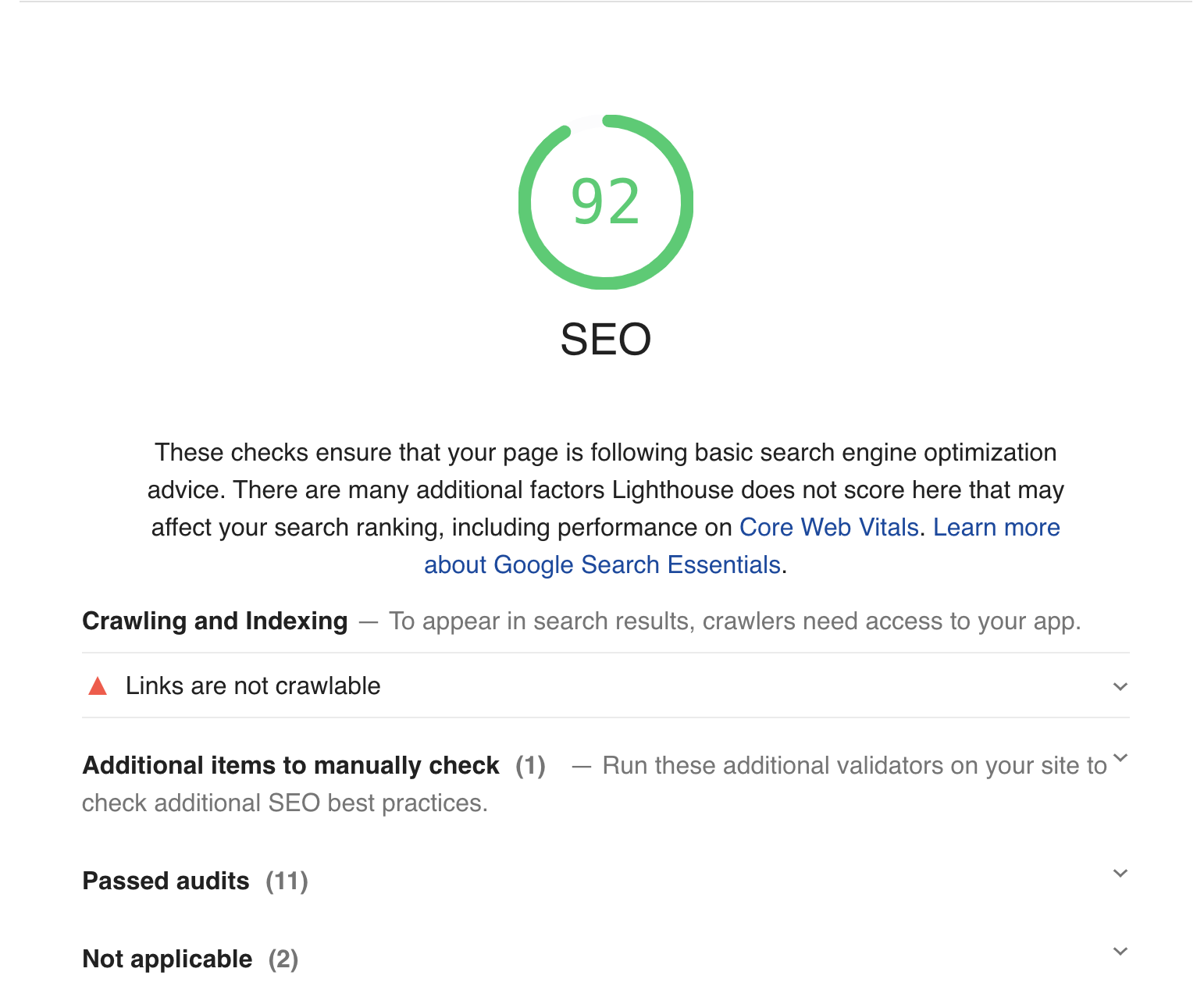
Các mức hiệu suất:



* 0 – 49: Hiệu suất thấp
* 50 – 89: Hiệu suất trung bình
* 90 – 100: Hiệu suất cao
* Đây là giao diện thể hiện các thông số và phương án giả thiết về hiệu suất: Từ những thông số trên có thể thấy rằng hiệu năng của trang web đang ở mức thấp khi chỉ đạt 47/100 điểm. Về mức độ SEO của trang web ở mức tốt 92/100đ



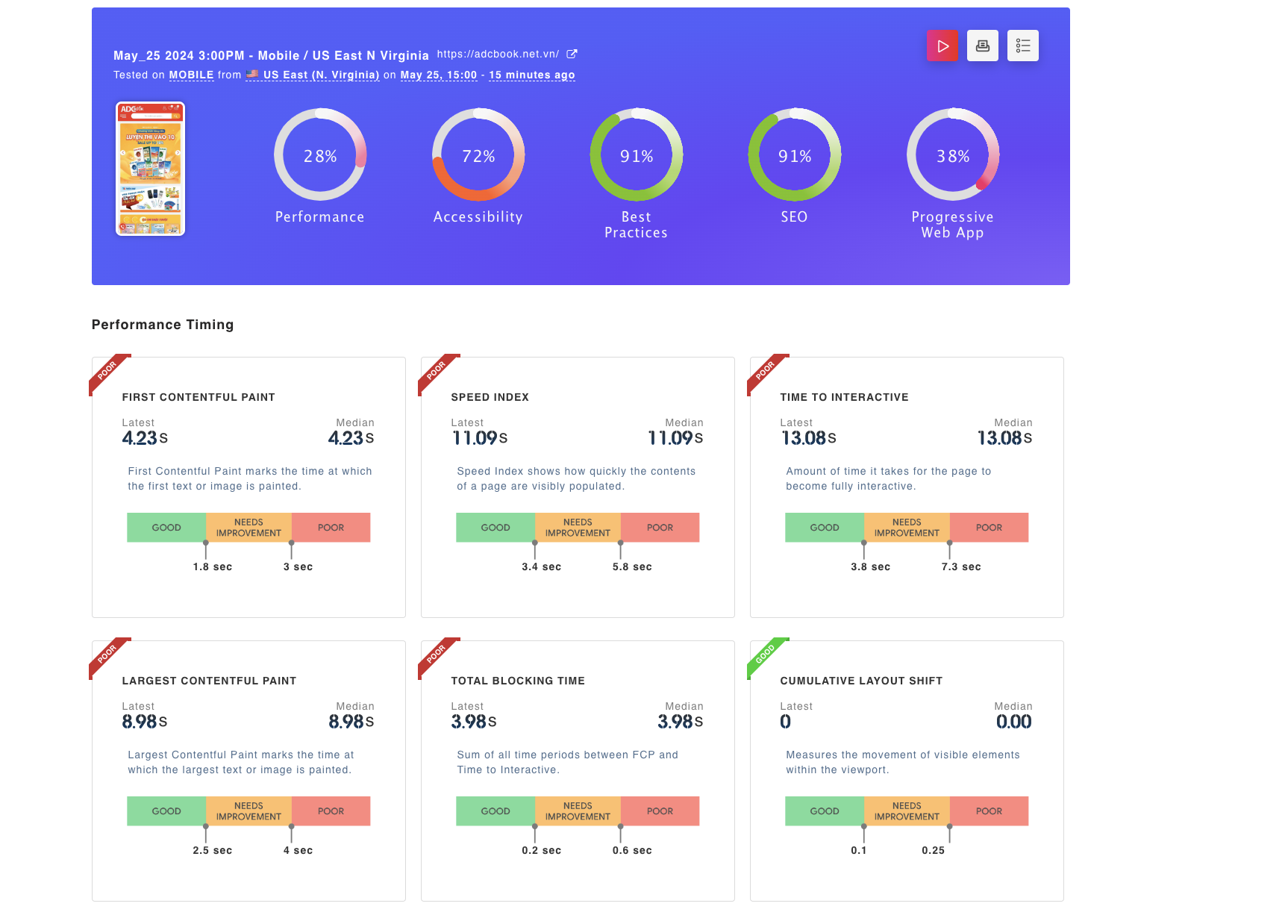
Ở đây có thể thấy được điểm hiệu suất là kém. Khi mất tới 3.1s để có thể tải được 1 nội dung lớn Chỉ số tốc độ của trang web cũng ở mức thấp khi mất khoảng 4.4 giây



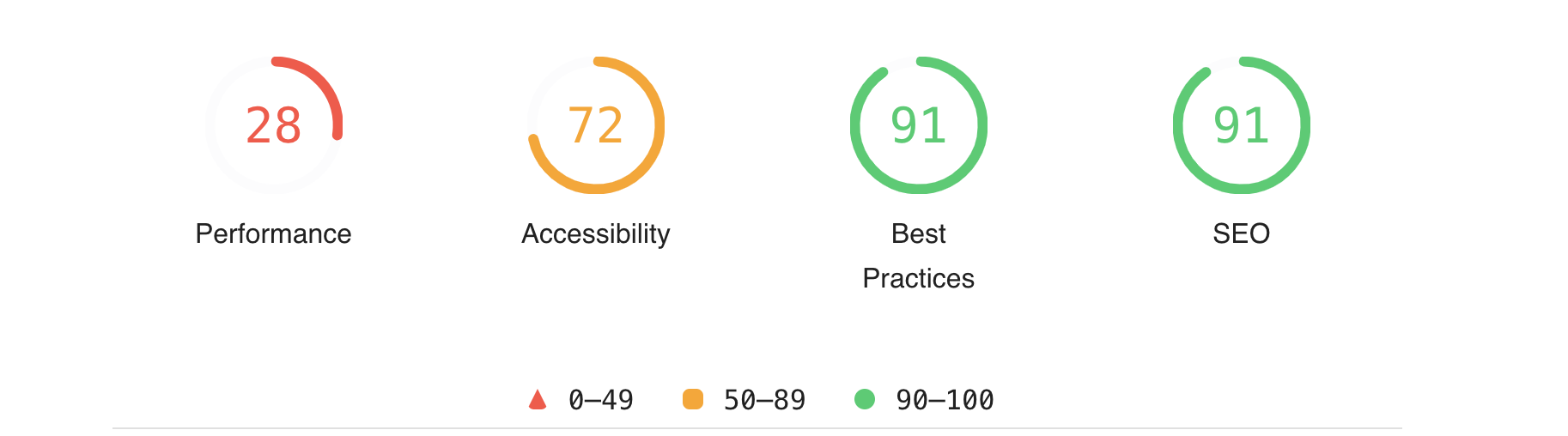
* Về điểm SEO của trang web cho kết quả ở mức tốt. Điều này cho thấy trang thấy trang web có thể dễ dàng được tìm thấy khi người dùng tìm trên các công cụ tìm kiếm.

**TC02: Mobile**

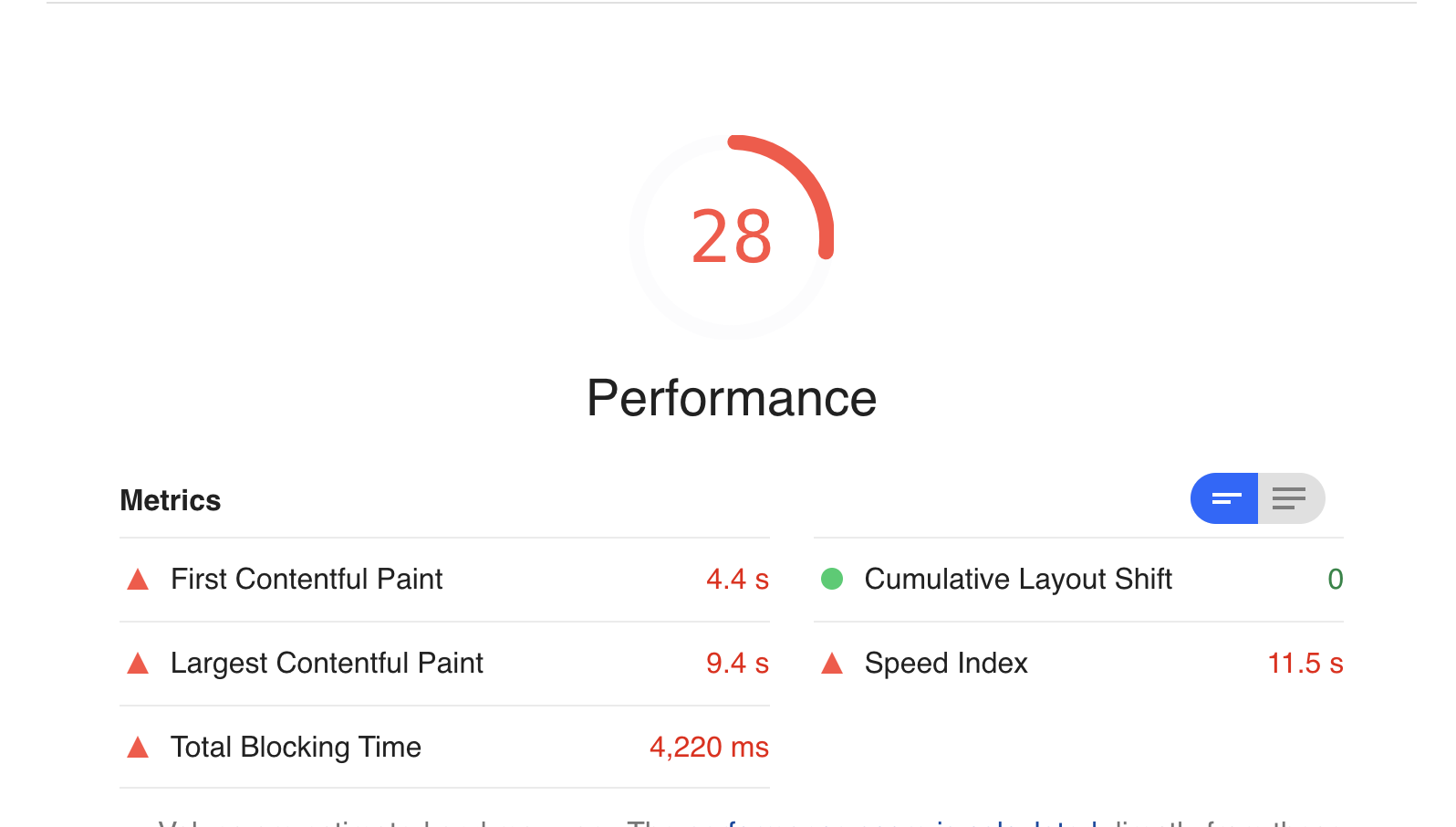
*Giao diện chi tiết khi chạy xong ca kiểm thử:*



*Các mức hiệu suất:*



* Đây là giao diện thể hiện các thông số và phương án giả thiết về hiệu suất: Từ những thông số trên có thể thấy rằng hiệu năng của trang web đang ở mức thấp khi chỉ đạt 28/100 điểm. Về mức độ SEO của trang web ở mức tốt 91/100đ



Điểm hiệu suất là kém. Khi mất tới 9.4s để có thể tải được 1 nội dung lớn. Chỉ số tốc độ của trang web cũng ở mức thấp khi mất khoảng 11.5 giây

### Kiểm thử API với jmeter bằng LoadFocus (Nguyễn Quang Vũ)

#### Kế hoạch kiểm thử

B1: Thực hiện setup cấu hình cho tệp Jmeter, Số lượng người dùng ảo, Vòng lặp, Ramp-up

A screenshot of a computer

Description automatically generated

B2: Tạo một HTTPS Request với những thông số đơn giản

A screenshot of a computer

Description automatically generated

B3: Truy cập vào trang web Loadfocus.com và chọn JMeter Load Testing

A screenshot of a test

Description automatically generated

B4: Kích vào nút “New Test” và tiến hành tải tập lệnh Jmeter lên

A close-up of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

B5: Bấm nút run test và chờ kết quả

* Tại mục Insight có thể xem thông tin về các mã lỗi phản hồi. và tại mục Error có thể xem thông tin về các lỗi ( Nếu có ). Ngoài ra tại mục Log. Người dùng có thể xem thông tin về tệp Jmx( Jmeter ) được tạo ra bởi Load Forcus.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TC | Function | Test case Name | Method | Step | Expected Output | Actual Output |
| TC01 | Kiểm tra hiệu suất với API với file Jmeter được tải lên (HTTP Request) | Kiểm thử tải để tìm kiếm các sự cố cấu hình | Bằng cách chạy thử nghiệm với một số lượng nhỏ người dùng ảo (Phương thức GET) | 1. 1.Tại màn hình của trang web <https://loadfocus.com/newtest> chọn chức năng “INITIAL LOAD”. 2. 2.Cài đặt : 3. + Virtual Users :10 + Duration(s) :30 4. + Ramp up time:10 5. + Ramp up steps: 3 6. 3.Tải lên file jmeter thread đã cấu hình sẵn API. 7. 4.Chọn EXECUTE TEST | Hiển thị kết quả sau khi chạy | Hiển thị kết quả sau khi chạy |
| TC02 | Kiểm tra hiệu suất API với file Jmeter được tải lên (HTTP Request) | Tìm ra các vấn đề về thời gian tải và phản hồi cũng như các lỗi tiềm ẩn | Bằng cách kiểm tra các mã phản hồi (Phương thức GET) | 1.Tại màn hình của trang web <https://loadfocus.com/newtest> chọn chức năng “WARM-UP LOAD”.  2.Cài đặt :  + Virtual Users :10  + Duration(s) :30  + Ramp up time:10  + Ramp up steps: 2  3.Tải lên file jmeter thread đã cấu hình sẵn API.  4.Chọn EXECUTE TEST | Hiển thị kết quả sau khi chạy | Hiển thị kết quả sau khi chạy |
| TC03 | Kiểm tra hiệu suất API với file Jmeter được tải lên (HTTP Request) | Tìm ra các vấn đề với một thiết đặt vừa phải cho điểm cuối đang thử nghiệm. | KIểm tra các mã phản hồi (Phương thức GET) | 1.Tại màn hình của trang web <https://loadfocus.com/newtest> chọn chức năng “MODERATE LOAD”.  2.Cài đặt :  + Virtual Users :10 + Duration(s) :30  + Ramp up time:10  + Ramp up steps: 3  3.Tải lên file jmeter thread đã cấu hình sẵn API.  4.Chọn EXECUTE TEST | Hiển thị kết quả sau khi chạy | Hiển thị kết quả sau khi chạy |

#### Báo cáo kết quả kiểm thử

**TC01: Kiểm thử tải để tìm kiếm các sự cố cấu hình**

- Tổng quan:

A screenshot of a graph

Description automatically generated

* Đồ thị thể hiện mức độ thời gian trung bình phản hồi với số lượng users truy cập:

A graph of a graph

Description automatically generated with medium confidence

* Đồ thị thể hiện mức độ lượt truy cập của user và mức độ lỗi theo thời gian:

A graph with a line going up

Description automatically generated

* Chi tiết các thông số cụ thể trong TC01:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**TC02: Kiểm thử tìm ra các vấn đề về thời gian tải và phản hồi cũng như các lỗi tiềm ẩn**

- Tổng quan:

A screenshot of a graph

Description automatically generated

* Đồ thị thể hiện mức độ thời gian trung bình phản hồi với số lượng users truy cập:

A graph showing a line

Description automatically generated with medium confidence

* Đồ thị thể hiện mức độ lượt truy cập của user và mức độ lỗi theo thời gian:

A graph with a green line

Description automatically generated

* Chi tiết các thông số cụ thể trong TC02:

A screenshot of a computer

Description automatically generated

A group of letters on a white background

Description automatically generated

**TC03: Kiểm thử tìm ra các vấn đề với một thiết đặt vừa phải cho điểm cuối đang thử nghiệm.**

* Kết quả kiểm thử tổng quan

A screenshot of a graph

Description automatically generated

* Đồ thị thể hiện mức độ thời gian trung bình phản hồi với số lượng users truy cập:

A graph showing a number of users

Description automatically generated

* Đồ thị thể hiện mức độ lượt truy cập của user và mức độ lỗi theo thời gian:

A graph with a line

Description automatically generated

* Chi tiết các thông số cụ thể trong TC03:

A screenshot of a computer

Description automatically generated

# Phụ lục

## Phụ lục hình ảnh

## Phụ lục bảng biểu

# Tài liệu tham khảo

[1] *Tên tài liệu*  
Tên tác giả - Tên nhà xuất bản, tạp chí, hội nghị, năm xuất bản

[2] *Tên tài liệu*  
Tên tác giả - Tên nhà xuất bản, tạp chí, hội nghị, năm xuất bản