Overview

· **Loadster là gì**: Loadster là một công cụ kiểm tra tải và hiệu suất mạnh mẽ, hoạt động trên nền tảng đám mây và máy tính để bàn, được thiết kế để mô phỏng lưu lượng thực tế và đo lường hiệu suất của các ứng dụng web và API dưới tải nặng. Nó hỗ trợ các giao thức HTTP, WebSocket và các API REST/SOAP.

· **Tại sao dùng Loadster**: Đảm bảo rằng các ứng dụng web, trang web hoặc API có thể xử lý được lượng người dùng lớn là rất quan trọng để mang lại trải nghiệm người dùng tốt và tránh tình trạng quá tải hệ thống. Loadster giúp các nhóm phát triển, kiểm thử và DevOps xác định các nút thắt hiệu suất trước khi đưa vào hoạt động, cung cấp dữ liệu để tối ưu hóa cấu hình máy chủ và mã nguồn ứng dụng.

· **Ai đã tạo ra Loadster**: Loadster được phát triển bởi Loadster, Inc., một công ty chuyên về các công cụ kiểm tra hiệu suất, nhằm cung cấp các giải pháp đáng tin cậy cho việc kiểm thử hiệu năng và giám sát ứng dụng.

· **Khách hàng tiêu biểu**: Loadster được sử dụng bởi nhiều công ty từ các startup nhỏ đến các tập đoàn lớn, thuộc nhiều lĩnh vực như thương mại điện tử, SaaS, fintech và y tế. Một số khách hàng nổi bật bao gồm **Spotify**, **Cisco**, và **Squarespace**, họ sử dụng Loadster để kiểm tra hiệu suất các dịch vụ web của mình.

Architecture

Loadster sử dụng kiến trúc **kết hợp**, kết hợp giữa **tạo bài kiểm tra trên máy tính để bàn** và **thực thi tải trên nền tảng đám mây**, mang lại sự linh hoạt trong cách thiết kế và thực thi các bài kiểm tra hiệu năng. Kiến trúc này được chia thành hai thành phần chính:

**Loadster Workbench (Ứng dụng máy tính để bàn)**:

* 1. **Loadster Workbench** là một ứng dụng máy tính cục bộ, chạy trên máy tính của người dùng. Đây là môi trường nơi người dùng tạo, sửa đổi và quản lý các kịch bản kiểm tra tải.
  2. Chức năng chính:
     1. **Tạo kịch bản (**Script Creation**)**: Người dùng có thể ghi lại các yêu cầu HTTP khi tương tác với ứng dụng web của mình, tạo ra các kịch bản kiểm tra chi tiết mà không cần viết mã thủ công. Các kịch bản này có thể được tùy chỉnh để phù hợp với những luồng công việc kiểm tra phức tạp hơn.
     2. **Quản lý kịch bản(**Scenario Management**)**: Người dùng có thể xây dựng các kịch bản kiểm tra mô phỏng các hành vi thực tế của người dùng, thiết lập các điều kiện tùy chỉnh và thêm các phiên người dùng song song để tái hiện các tải trọng thực tế.
     3. **Thực thi kiểm tra ban đầu(**Initial Test Execution**)**: Người dùng có thể chạy các bài kiểm tra quy mô nhỏ trên máy tính cục bộ để kiểm tra tính đúng đắn của các kịch bản trước khi thực hiện trên quy mô lớn trên đám mây.

**Loadster Cloud (Các engine tải phân tán)**:

* 1. **Các engine tải trên nền tảng đám mây** (Cloud-based Load Engines)được triển khai tại nhiều trung tâm dữ liệu toàn cầu, cho phép người dùng tạo lưu lượng truy cập lớn từ nhiều khu vực địa lý khác nhau.
  2. Chức năng chính:
     1. **Tạo tải phân tán(**Distributed Load Generation**)**: Khi bài kiểm tra được thực thi, các engine tải ở các khu vực khác nhau sẽ tạo ra lưu lượng truy cập dựa trên các kịch bản được tạo trong Loadster Workbench. Điều này cho phép mô phỏng lưu lượng toàn cầu, kiểm tra xem ứng dụng xử lý như thế nào với người dùng từ nhiều khu vực cùng lúc.
     2. **Giám sát và tổng hợp dữ liệu thời gian thực(**Real-Time Monitoring & Data Aggregation**)**: Trong khi kiểm tra diễn ra, Loadster thu thập dữ liệu hiệu năng từ tất cả các engine tải trên đám mây. Dữ liệu này được gửi lại Workbench trong thời gian thực, cho phép người dùng theo dõi các số liệu như thời gian phản hồi, thông lượng, và tỷ lệ lỗi ngay lập tức.
     3. **Mở rộng tải linh hoạt(**Load Scaling**)**: Hạ tầng đám mây cho phép mở rộng quy mô bài kiểm tra, giúp chạy các bài kiểm tra với hàng chục nghìn người dùng ảo từ khắp nơi trên thế giới.

### Highlight

· **Tạo kịch bản**: Loadster cho phép ghi và phát lại để tạo các kịch bản kiểm tra, giúp người dùng dễ dàng định nghĩa các luồng công việc phức tạp mà không cần chuyên môn về mã hóa.

· **Kiểm tra tải toàn cầu**: Các bài kiểm tra có thể được thực hiện từ nhiều địa điểm địa lý khác nhau thông qua các engine tải trên đám mây.

· **Giám sát theo thời gian thực**: Trong suốt quá trình kiểm tra, các số liệu như thời gian phản hồi, thông lượng và tỷ lệ lỗi được hiển thị theo thời gian thực.

· **Kiểm tra API**: Loadster hỗ trợ kiểm tra các API REST và SOAP, giúp xác nhận hiệu suất của backend.

· **Tích hợp dễ dàng**: Loadster có thể tích hợp với các pipeline CI/CD để tự động hóa các bài kiểm tra hiệu năng như một phần của quy trình phát triển liên tục.

### ****Chi Tiết Tính Năng****

**Kiểm tra tải từ nhiều địa điểm toàn cầu**:  
Loadster cho phép người dùng tạo lưu lượng truy cập từ nhiều địa điểm trên khắp thế giới, đảm bảo rằng các ứng dụng web được kiểm tra trong các điều kiện thực tế. Tính năng này giúp xác định các vấn đề hiệu suất đặc thù theo khu vực.

**Ghi lại và tạo kịch bản**:  
Người dùng có thể ghi lại các yêu cầu HTTP khi họ truy cập ứng dụng web của mình, và Loadster sẽ tự động tạo các kịch bản dựa trên các hành động này. Điều này giúp giảm thiểu việc tạo kịch bản thủ công, và người dùng nâng cao có thể tùy chỉnh kịch bản để thực hiện các tình huống phức tạp hơn.

**Kiểm tra API**:  
Loadster hỗ trợ kiểm tra API REST và SOAP, cho phép các nhóm đảm bảo rằng dịch vụ backend của họ có thể xử lý lưu lượng lớn. Loadster hỗ trợ cả các yêu cầu GET/POST đơn giản và các tương tác API phức tạp hơn.

**Kết quả thời gian thực**:  
Trong khi kiểm tra tải diễn ra, Loadster hiển thị các số liệu thời gian thực như thời gian phản hồi, độ trễ, thông lượng và số lượng yêu cầu thành công hay thất bại. Dữ liệu này được hiển thị qua các biểu đồ, giúp người dùng dễ dàng nhận ra các điểm nghẽn hiệu suất.

**Kiểm tra tải phân tán trên đám mây**:  
Các engine tải đám mây của Loadster cho phép kiểm tra phân tán từ nhiều trung tâm dữ liệu, mô phỏng lưu lượng truy cập từ các khu vực địa lý khác nhau. Điều này rất phù hợp để kiểm tra các ứng dụng web có cơ sở người dùng toàn cầu.

**Tích hợp với CI/CD**:  
Loadster có thể tích hợp vào các pipeline DevOps (như Jenkins, CircleCI, v.v.) để tự động hóa kiểm tra hiệu suất như một phần của quy trình phát triển liên tục. Điều này đảm bảo rằng các vấn đề hiệu suất được phát hiện sớm trong quá trình phát triển.

**Số liệu và xác nhận tùy chỉnh**:  
Người dùng có thể thiết lập các tiêu chí tùy chỉnh cho sự thành công của bài kiểm tra và ngưỡng hiệu suất. Nếu thời gian phản hồi vượt quá giới hạn được đặt trước, Loadster sẽ đánh dấu các vấn đề này, giúp dễ dàng xác định các vấn đề cần khắc phục.

