



รายงาน

เรื่อง An Automated Tool for Software Testing

จัดทำโดย

B6400965	นายกิตติภพ	สระแกทอง
B6400989	นายพงศกร	ล้านใจดี
B6425203	นายชัยวัฒน์	พุนดี
B6417369	นางสาวสุกัศสรา	ไวยสุณี

เสนอ

อาจารย์ ดร.ธรรมศักดิ์ เขียวนิเวศน์

รายงานนี้เป็นส่วนหนึ่งของรายวิชา 1101160 Software Engineering Module

ภาคเรียนที่ 2 ปีการศึกษา 2565

มหาวิทยาลัยเทคโนโลยีสุรนารี

คำนำ

โครงงานเล่มนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของรายวิชา 1101162 Software Testing เพื่อให้ได้ศึกษาหาความรู้ในเรื่องการทดสอบระบบ โดยได้ศึกษาผ่านแหล่งความรู้ต่าง เช่น เว็บไซต์อินเทอร์เน็ต โดยรายงานเล่มนี้ต้องมีเนื้อหาเกี่ยวกับ ที่มาและความสำคัญของการทดสอบระบบ

ผู้จัดทำคาดหวังเป็นอย่างยิ่งว่าการจัดทำเอกสารฉบับนี้จะมีข้อมูลที่เป็นประโยชน์ต่อผู้ที่สนใจศึกษาการทดสอบระบบเป็นอย่างดี

สารบัญ

บทที่

หน้าที่

บทที่ 1 ที่มาและความสำคัญ

บทที่ 2 ความหมายของ Blaze Meter

บทที่ 3 Blaze Meter ทำงานอย่างไร ?

บทที่ 4 การทำงานเบื้องต้นของ Blaze Meter

บทที่ 1

ที่มาและความสำคัญ

BlazeMeter ถูกก่อตั้งขึ้นเมื่อปี 2011 โดยทีมผู้สร้างของ Apache JMeter ซึ่งเป็นเครื่องมือทดสอบเว็บแอปพลิเคชันแบบโอเพนซอร์ซ คือ Alon Girmonsky, Andrey Pokhilko, Denis Goodwin และ Peter Yared และในปี 2016 BlazeMeter ได้รับการเข้าซื้อโดย CA Technologies และต่อมาถูกซื้อโดย Broadcom Inc. เป็นเครื่องมือทดสอบแบบคลาวด์ (cloud-based testing) ที่มีความสำคัญสำหรับทีมพัฒนาซอฟต์แวร์และผู้ทดสอบซอฟต์แวร์ เนื่องจากสามารถช่วยประหยัดเวลาและค่าใช้จ่ายในการสร้างสภาพแวดล้อมทดสอบและทดสอบโค้ดของแอปพลิเคชัน ทั้งนี้เพราะ BlazeMeter มีคุณสมบัติที่สามารถทำการทดสอบแบบอัตโนมัติในแต่ละครั้งที่ต้องการ และสามารถจำลองปริมาณผู้ใช้ที่มากพอที่จะทำให้ทดสอบมีประสิทธิภาพและแม่นยำมากขึ้น ทั้งนี้ BlazeMeter ยังสามารถช่วยให้ทีมพัฒนาซอฟต์แวร์ได้ปรับปรุงแอปพลิเคชันของพวกเขาให้ดียิ่งขึ้นก่อนที่จะปล่อยสู่ตลาดโดยมีการตรวจสอบปัญหาและแก้ไขโค้ดอย่างรวดเร็ว

ดังนั้นทางผู้จัดทำจึงตัดสินใจเลือกใช้เว็บไซต์ BlazeMeter ในการทดสอบระบบ

บทที่ 2

ความหมายของ Blaze Meter

BlazeMeter เป็นเครื่องมือทดสอบเว็บแอปพลิเคชันและบริการคลาวด์ (cloud service) ที่ให้ผู้พัฒนาและทีมทดสอบซอฟต์แวร์สามารถทดสอบประสิทธิภาพและประสิทธิผลของแอปพลิเคชันของพวกเขาได้อย่างมีประสิทธิภาพ ด้วย BlazeMeter, คุณสามารถสร้างการทดสอบแบบอัตโนมัติหรือแบบมืออาชีพเพื่อวัดประสิทธิภาพการทำงานของแอปพลิเคชันของคุณในสถานการณ์การใช้งานจริง โดยการทดสอบอย่างเหมาะสมและมีประสิทธิภาพจะช่วยให้คุณตรวจสอบว่าแอปพลิเคชันของคุณสามารถรองรับการใช้งานจริงได้หรือไม่ และแก้ไขปัญหาก่อนที่จะเกิดขึ้นในสถานะการใช้งานจริง นอกจากนี้ BlazeMeter ยังมีคุณสมบัติอื่น ๆ เช่นการเปลี่ยนแปลงปริมาณของผู้ใช้และการดูรายงานผลการทดสอบอย่างละเอียด เพื่อช่วยให้ทีมของคุณมีการทดสอบและการปรับปรุงการทำงานของแอปพลิเคชันที่มีประสิทธิภาพมากขึ้น

ภาษาที่ใช้ใน BlazeMeter

1. JAVA

การทดสอบที่ใช้ JMeter

1. Performance Test

ตัวชี้วัดถึงประสิทธิภาพของซอฟต์แวร์เราได้ว่าซอฟต์แวร์ที่เราพัฒนาขึ้นมานั้นมีการตอบสนองเร็วหรือช้าหรือหากมีจำนวนผู้ใช้งานเข้ามามากๆแล้วระบบเราจะยังสามารถทำงานได้หรือไม่ ซึ่งการทำ Performance Testing นั้นจะทำให้เราทราบถึงข้อจำกัดและข้อผิดพลาดเชิงประสิทธิภาพได้ก่อนที่จะถูกปล่อยไป ตัวอย่างหนึ่งที่เกิดขึ้นบ่อยคือ การประกาศผลหรือคะแนนสอบผ่านเว็บไซต์ ลักษณะของกิจกรรมที่เกิดขึ้นนั้นพบว่าผู้ใช้งานเป็นจำนวนมากเข้าใช้งานบนเว็บไซต์พร้อมๆกันทำให้ระบบล่ม ซึ่งหากเราได้ศึกษาเกี่ยวกับ Performance ของระบบแล้วเราอาจจะเข้าใจมากขึ้นว่าทำไมระบบถึงล่มและจะหาทางป้องกันได้อย่างไรซึ่งบทความนี้จะมีคำตอบให้อย่างแน่นอน

การทดสอบแบบ Performance Test นั้นเป็นการทดสอบอยู่ในกลุ่มของ Non-Functional testing ซึ่งมีวัตถุประสงค์เพื่อให้ซอฟต์แวร์สามารถใช้งานได้อย่างมีประสิทธิภาพ ง่ายต่อการบำรุงรักษา ลดความเสี่ยงต่างๆที่อาจจะเกิดขึ้น โดยในกลุ่มของ Non-Functional testing สามารถแบ่งประเภทที่จะทดสอบออกเป็นอีกหลายประเภทและ Performance testing ก็เป็นหนึ่งในประเภทที่จะต้องสนใจในการทดสอบของ Non-Functional testing

ตัวอย่างการทำงานของ Performance Test ในรูปแบบต่างๆ

Smoke testing : คือการ test run load แบบน้อย ๆ เพื่อ prove Application ก่อนว่ามีความพร้อมก่อนที่จะสามารถรองรับ load เบื้องต้นได้ก่อนไหม ก่อนที่จะทำ load test ของจริง

Load Test : คือการ test ตาม scenario ที่คิดว่าจะเกิดขึ้นจริงใน Production อธิบายให้เข้าใจง่ายขึ้นก็คือการจำลองสถานการณ์ของการใช้ระบบที่น่าจะเกิดขึ้นจริงใน Case ต่างๆ ของการใช้ตัว Application

Stress Test : คือการ test หาจุด Break point ของ Application เพื่อให้รู้ว่าเมื่อ Volume ที่เข้ามาถึง ณ จุดหนึ่งระบบเราเริ่มจะรองรับไม่ไหวแล้ว จะได้ทราบและคาดคะเนจำนวนผู้ใช้ได้ เพื่อรองรับการใช้งานจริงที่จะเกิดขึ้น

Availability Test : คือการ test ดูความ stable ของระบบอย่างต่อเนื่อง เช่น เราอาจจะจำลองการ test ของระบบของเราทำงานไปเรื่อย ๆ ประมาณ 4-8 ชั่วโมง แล้วดูว่า Application ของเราจะยังคง response ดีตามปกติอยู่หรือป่าว มีการกิน resource จาก server

บทที่ 3

BlazeMeter ทำงานอย่างไร ?

1.การกำหนดแผนการทดสอบ: ผู้ใช้ต้องกำหนดว่าจะทดสอบแอปพลิเคชันในสถานการณ์ใด โดยใช้บล็อกการทดสอบที่มีอยู่ใน BlazeMeter เช่น จำลองการใช้งานของผู้ใช้จำนวนมาก หรือจำลองการทำงานในสภาพแวดล้อมที่เปลี่ยนแปลงได้ เพื่อตรวจสอบว่าแอปพลิเคชันทำงานได้อย่างมีประสิทธิภาพในสถานการณ์ที่ต่างกัน

2.การรวบรวมข้อมูลทดสอบ: BlazeMeter จะทำการรวบรวมข้อมูลการทดสอบเช่น เวลาที่เริ่มและสิ้นสุดของการทดสอบ จำนวนผู้ใช้ที่ทดสอบแอปพลิเคชันในแต่ละครั้ง และเวลาที่ใช้ในการโหลดแต่ละหน้าของแอปพลิเคชัน เป็นต้น

3.การวิเคราะห์ผลการทดสอบ: BlazeMeter จะวิเคราะห์ข้อมูลการทดสอบที่ได้รับและสร้างรายงานผลการทดสอบขึ้นมา เพื่อให้ผู้ใช้สามารถเข้าใจผลการทดสอบได้อย่างง่ายดาย รายงานผลการทดสอบจะแสดงผลการทดสอบที่สมบูรณ์และสามารถแสดงกราฟและผลการทดสอบในแต่ละตัวแปรเช่น ความเร็วโหลด ปริมาณของผู้ใช้ และเวลาที่ใช้ในการโหลดหน้าของแอปพลิเคชัน เป็นต้น

4.การเปรียบเทียบผลการทดสอบ: BlazeMeter ยังสามารถเปรียบเทียบผลการทดสอบระหว่างการทดสอบที่ต่างกัน เพื่อให้ผู้ใช้สามารถรู้ว่าแอปพลิเคชันทำงานได้อย่างมีประสิทธิภาพกับสถานการณ์การใช้งานที่แตกต่างกันอย่างไร

5.การจัดการการทดสอบ: BlazeMeter ยังมีเครื่องมือในการจัดการการทดสอบ เช่น การกำหนดวันเวลาในการทดสอบ การกำหนดค่าของการทดสอบ และการตั้งค่าการแจ้งเตือนเมื่อเกิดข้อผิดพลาดในการทดสอบ

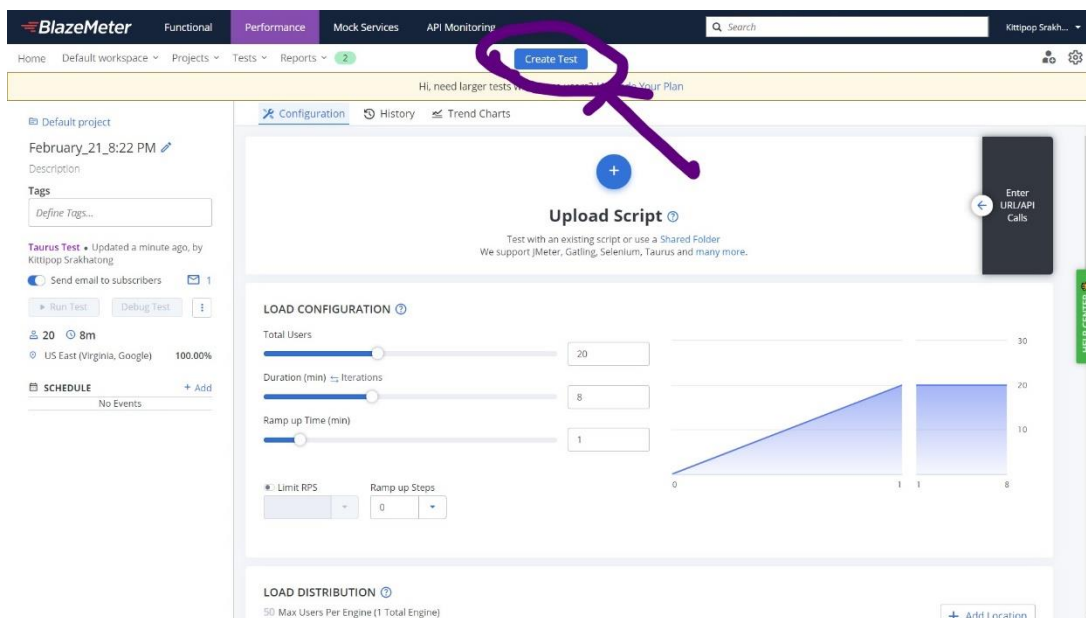
6.การทดสอบอัตโนมัติ: BlazeMeter ยังมีเครื่องมือในการทดสอบอัตโนมัติ เช่น JMeter ซึ่งช่วยให้ผู้ใช้สามารถกำหนดการทดสอบอัตโนมัติและปรับปรุงความแม่นยำของการทดสอบได้อย่างมีประสิทธิภาพ

7.การเชื่อมต่อกับเครื่องมืออื่น ๆ: BlazeMeter สามารถเชื่อมต่อกับเครื่องมืออื่น ๆ เช่น Jira, Jenkins, AWS และ Google Cloud Platform เพื่อให้ผู้ใช้สามารถใช้งานร่วมกับเครื่องมืออื่น ๆ ได้อย่างต่อเนื่อง

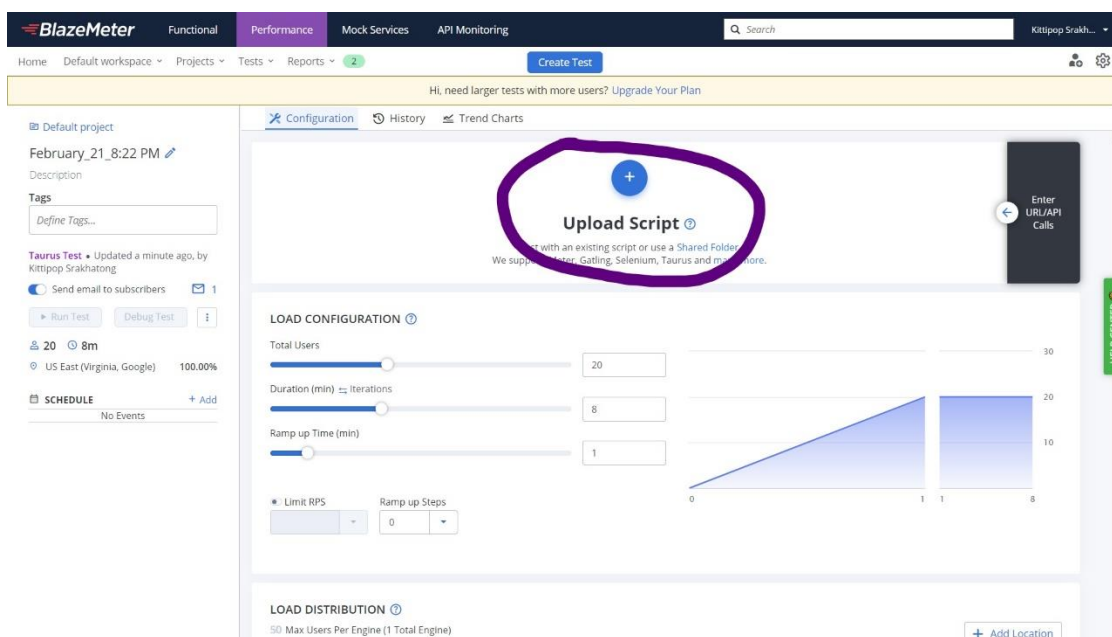
สรุปแล้ว BlazeMeter เป็นเครื่องมือที่ช่วยให้ผู้พัฒนาแอปพลิเคชันสามารถทดสอบประสิทธิภาพแอปพลิเคชันได้อย่างง่ายดาย โดย BlazeMeter สามารถทำงานได้หลากหลาย

การใช้งานเบื้องต้นของ BlazeMeter

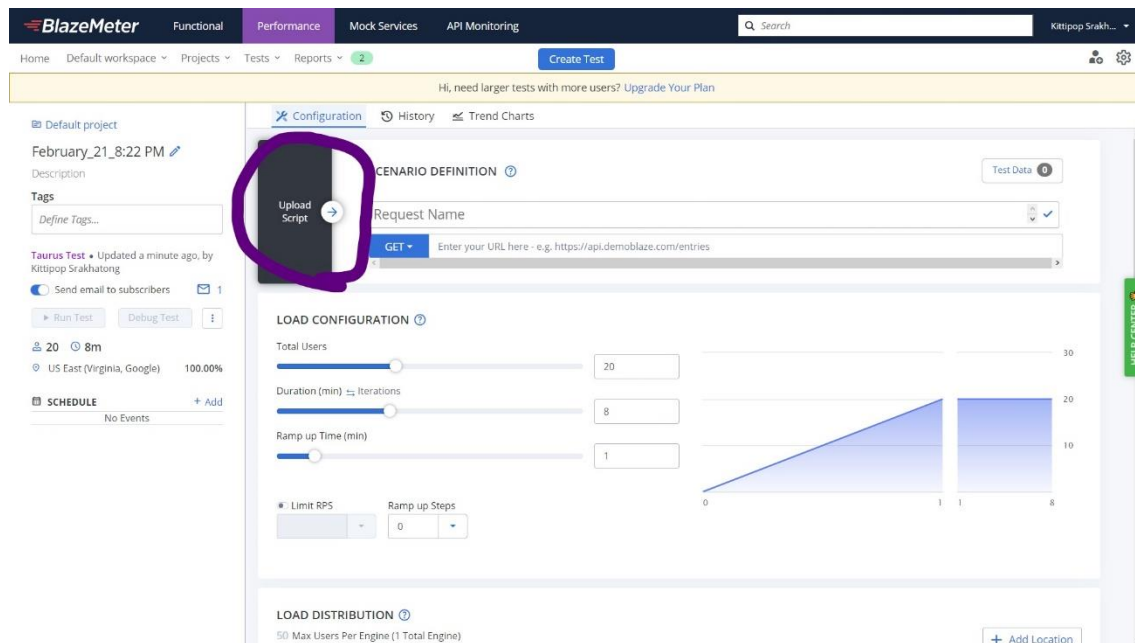
1. เริ่มต้นการสร้างเพื่อทดสอบระบบ



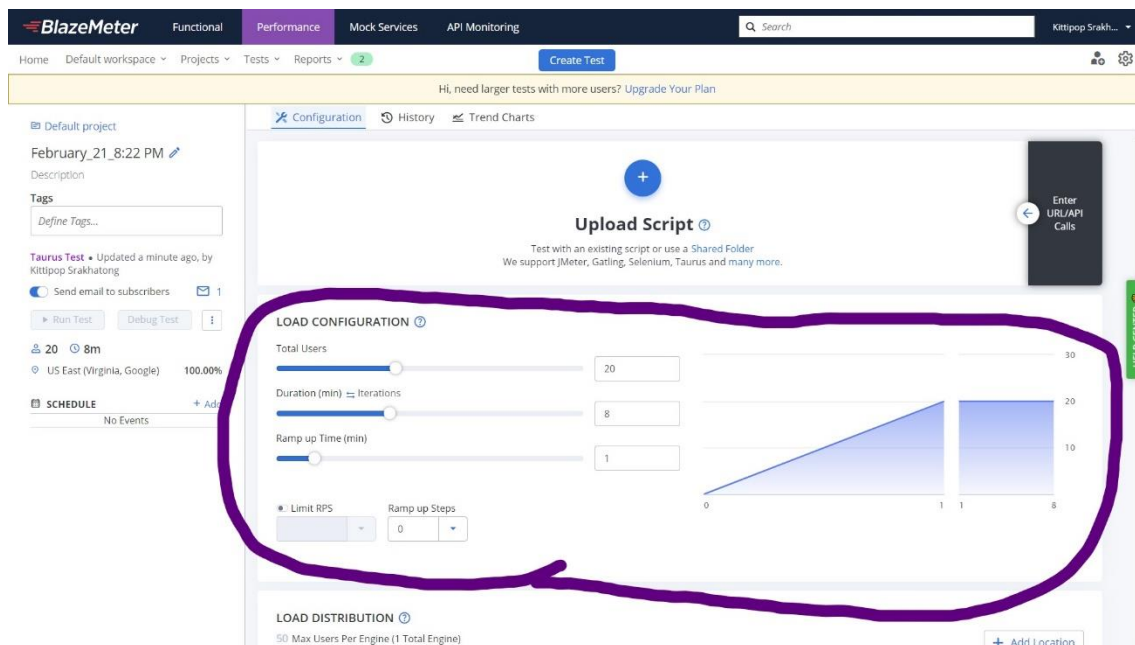
2. เลือกไฟล์เพื่ออัปโหลด



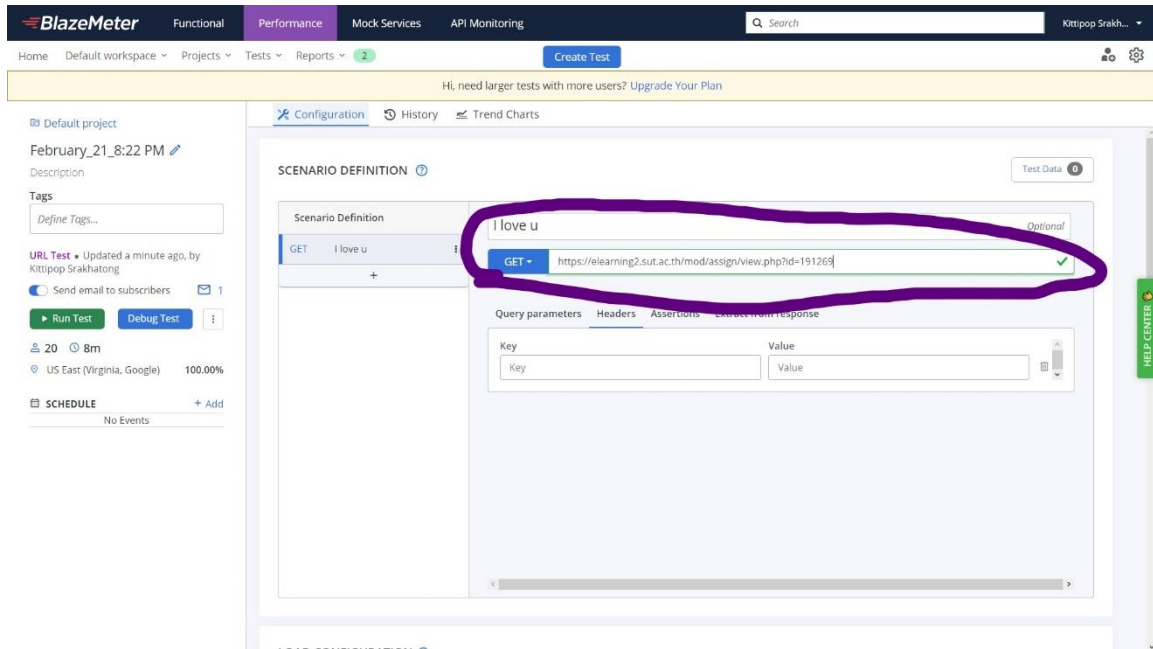
3. เปลี่ยนเป็นการทดสอบให้เป็นแบบ URL



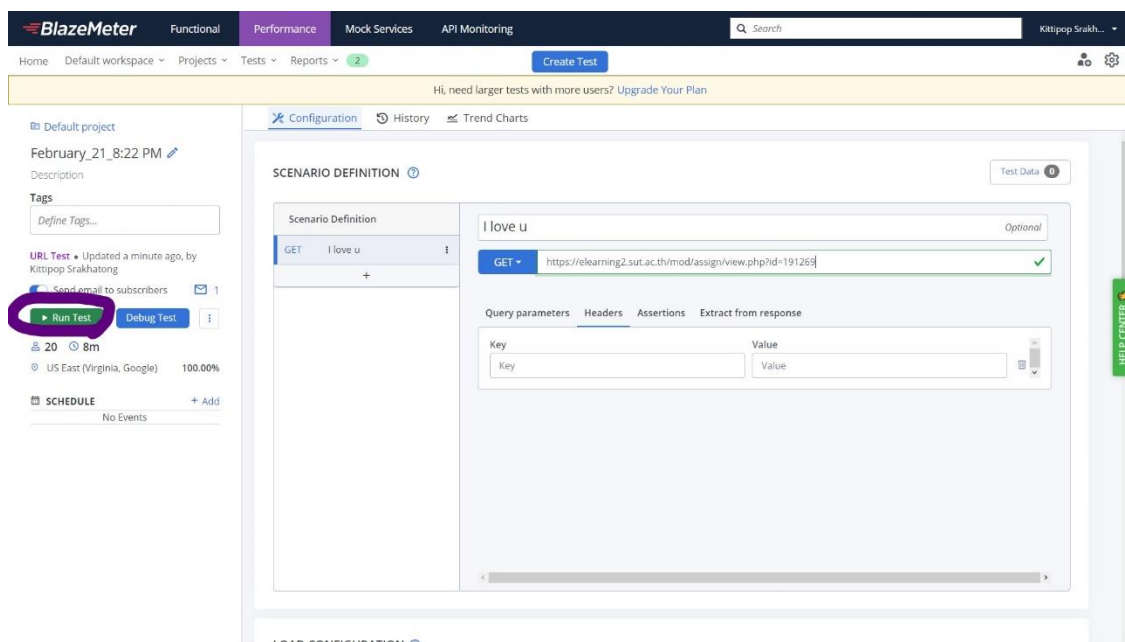
4. กรอกข้อมูลที่ต้องการทดสอบ



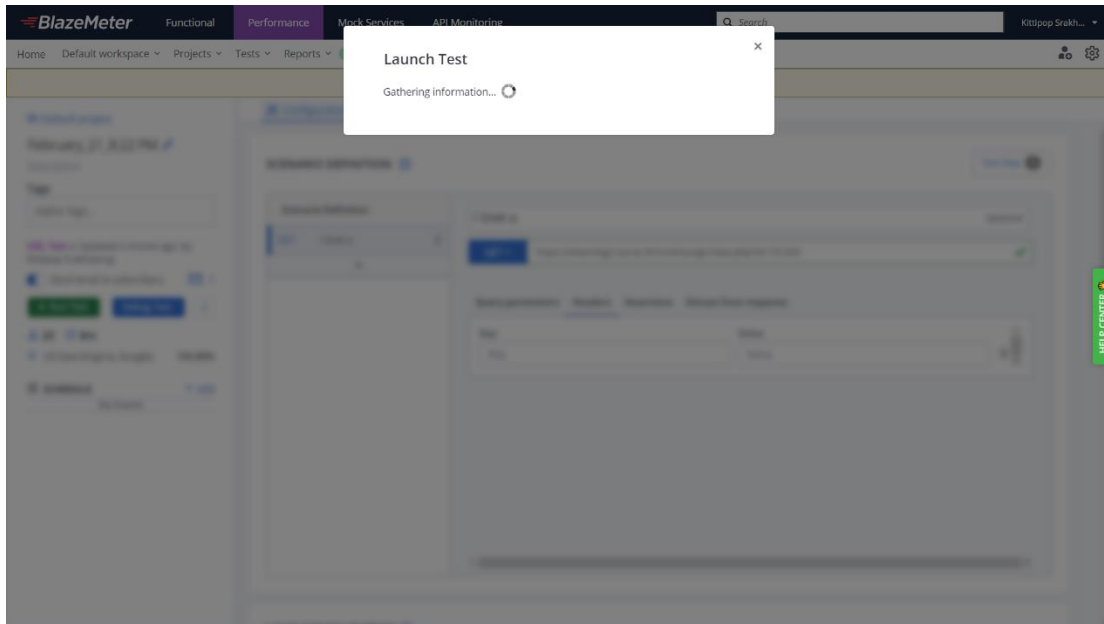
5. ใส่ URL ที่ต้องการทดสอบ



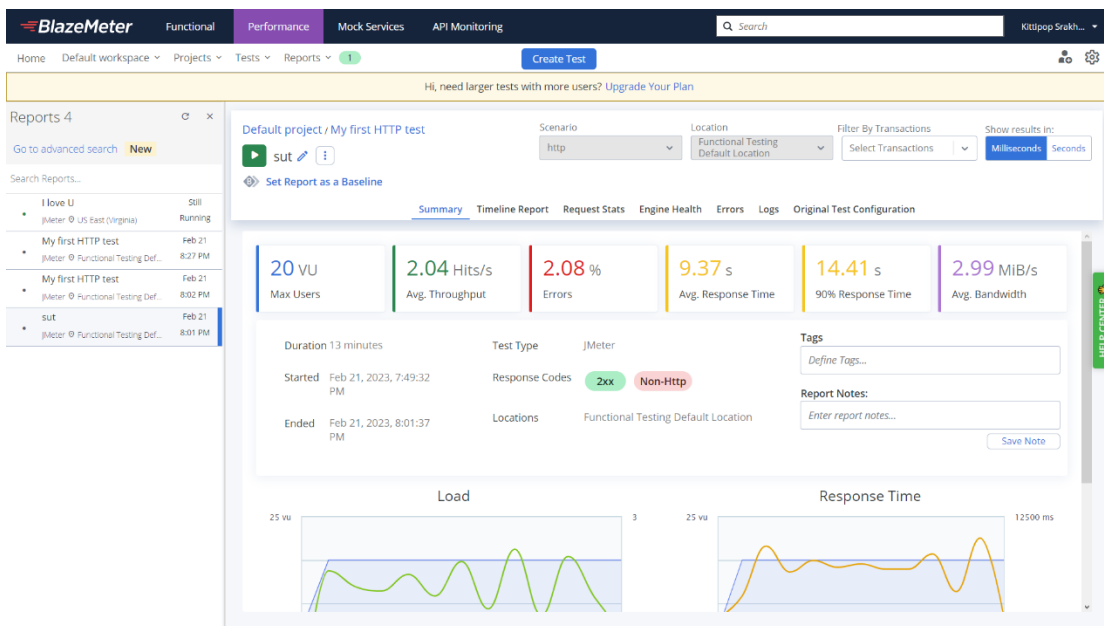
6. กด RUN เพื่อเริ่มการทดสอบ



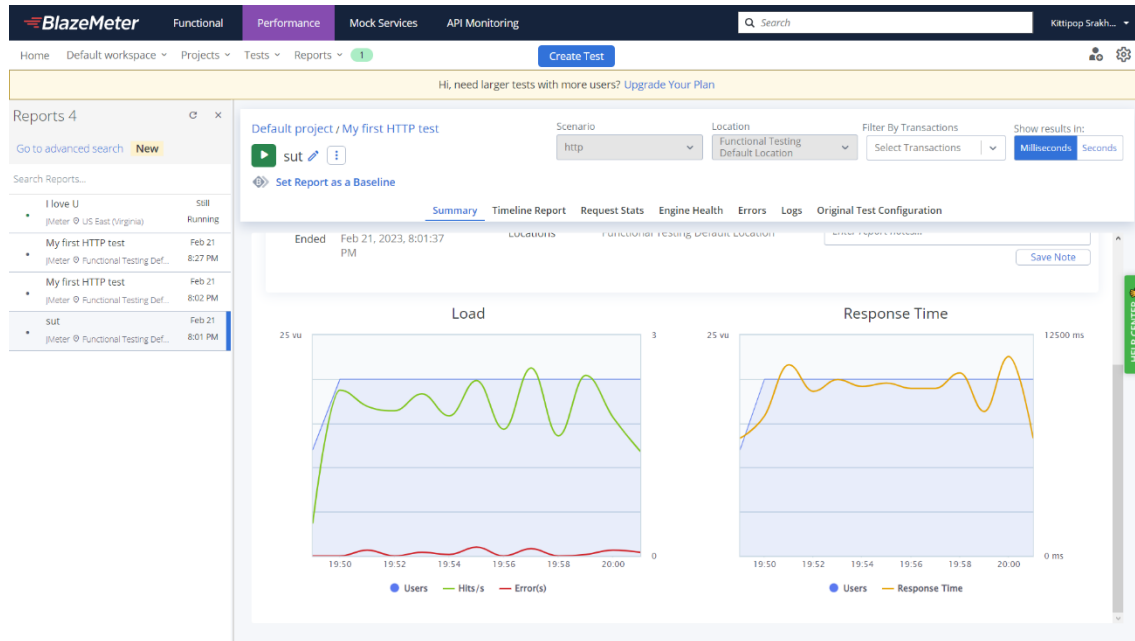
7. เว็บไซต์กำลังประมวลผล



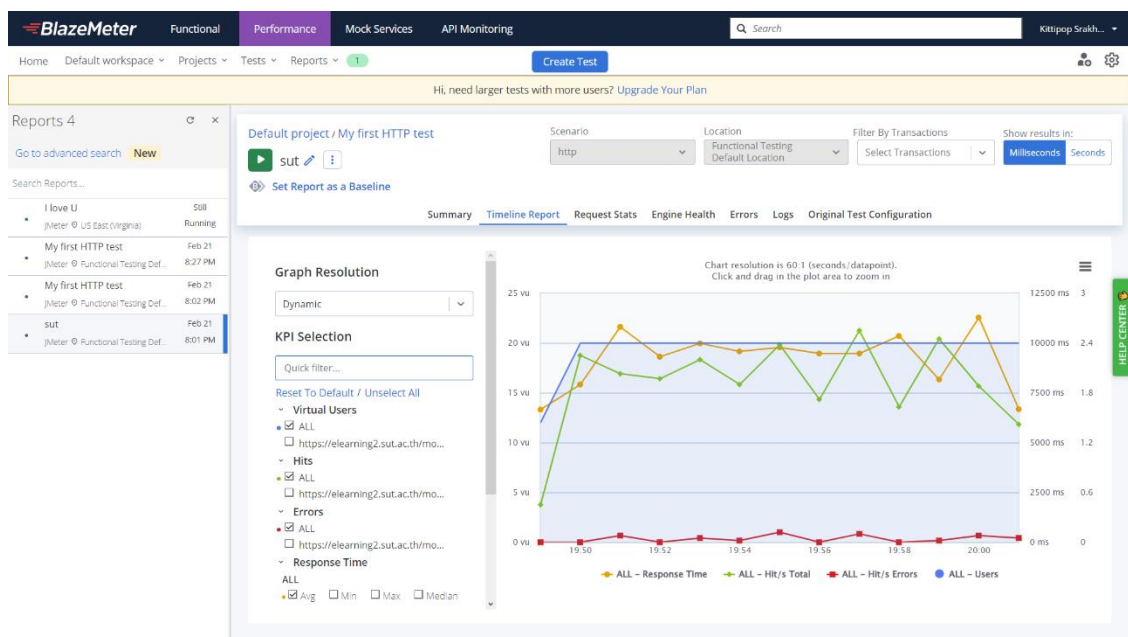
8. ผลสรุปการทดสอบ



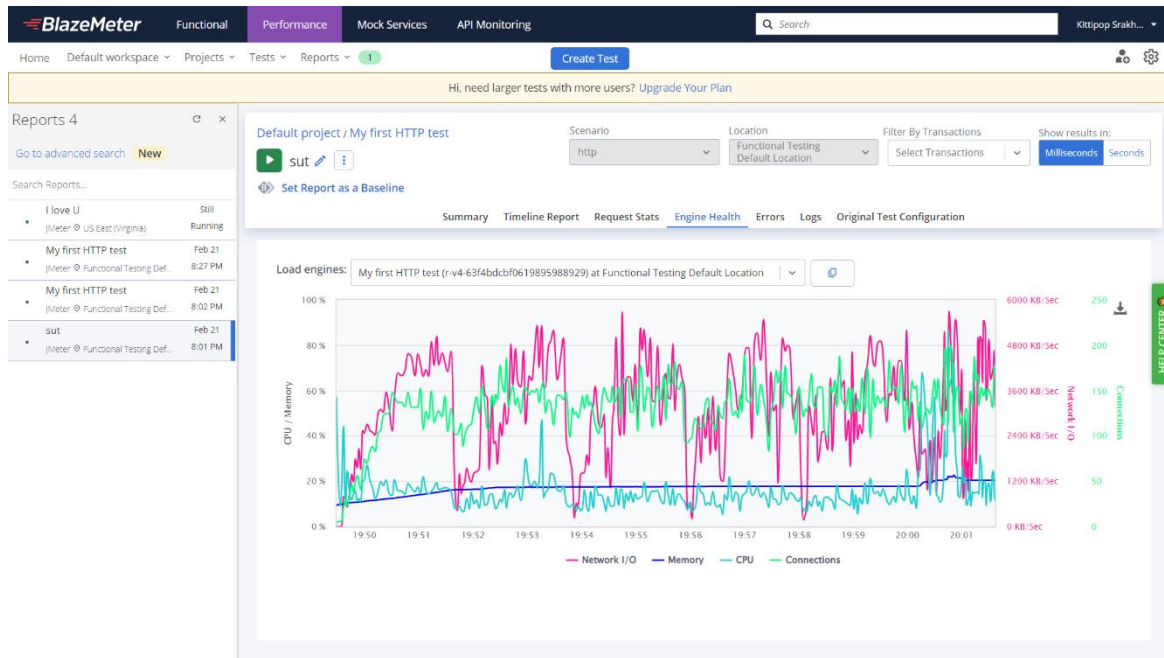
9. ผลสรุปการทดสอบ



10. ผลสรุปการทดสอบ



11. ผลสรุปการทดสอบ



12. ค่าที่เรากำหนดไว้

