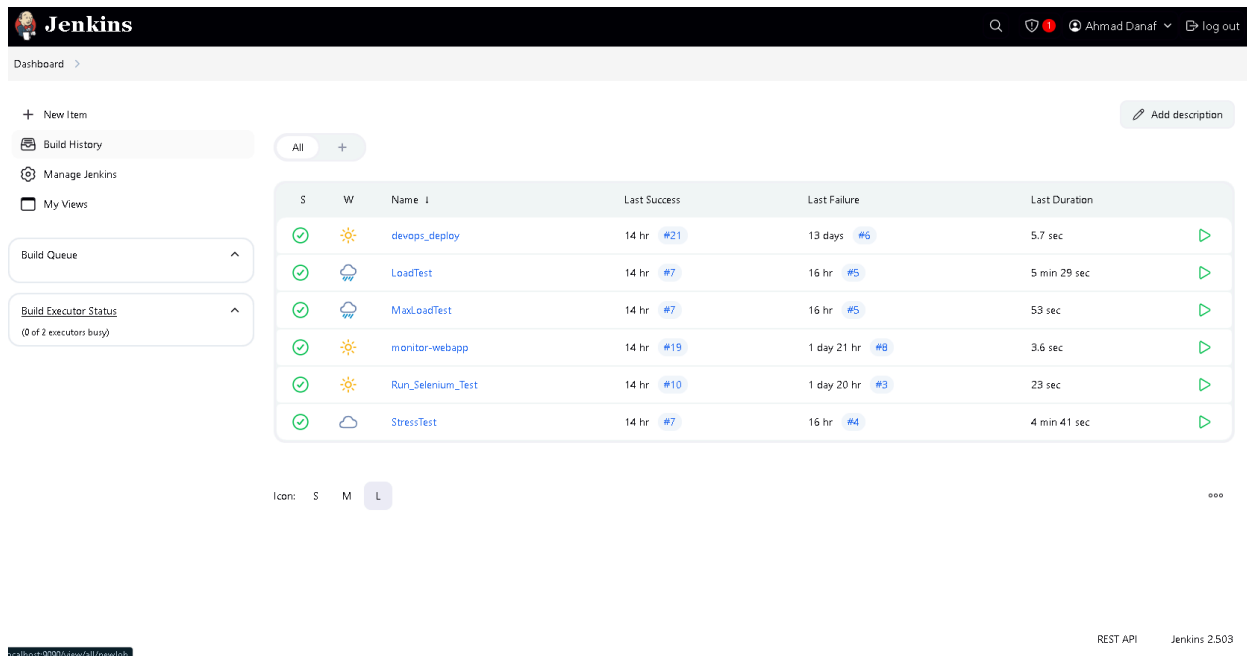


Ahmad Danaf 211787833  
Adam Taktury 322907247

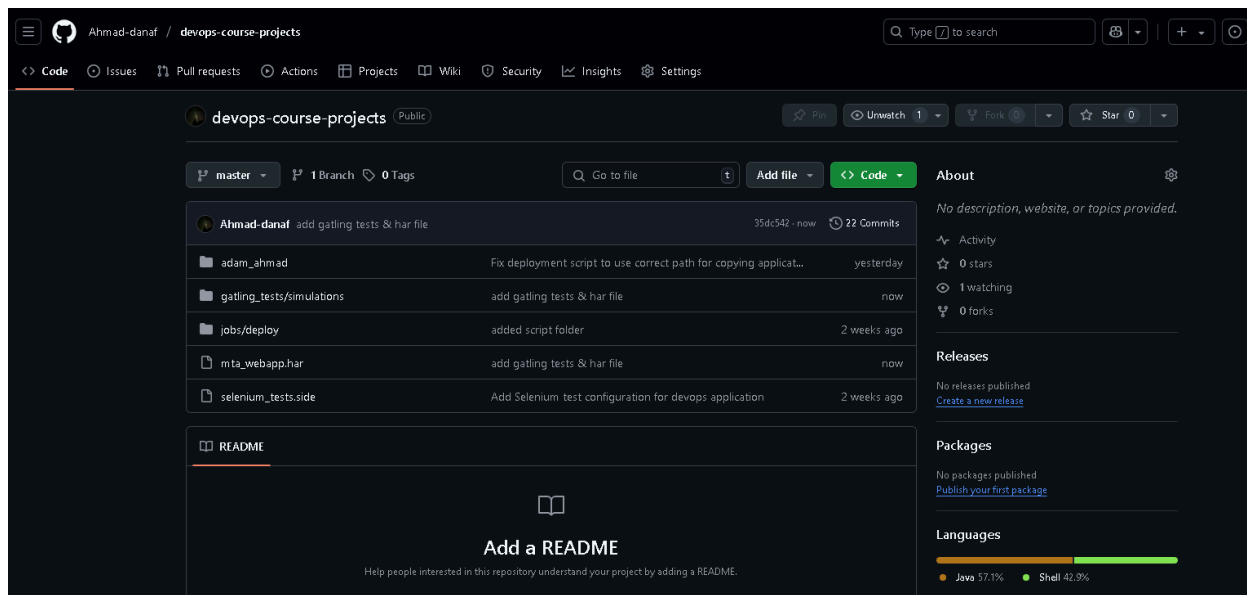
## JENKINS:



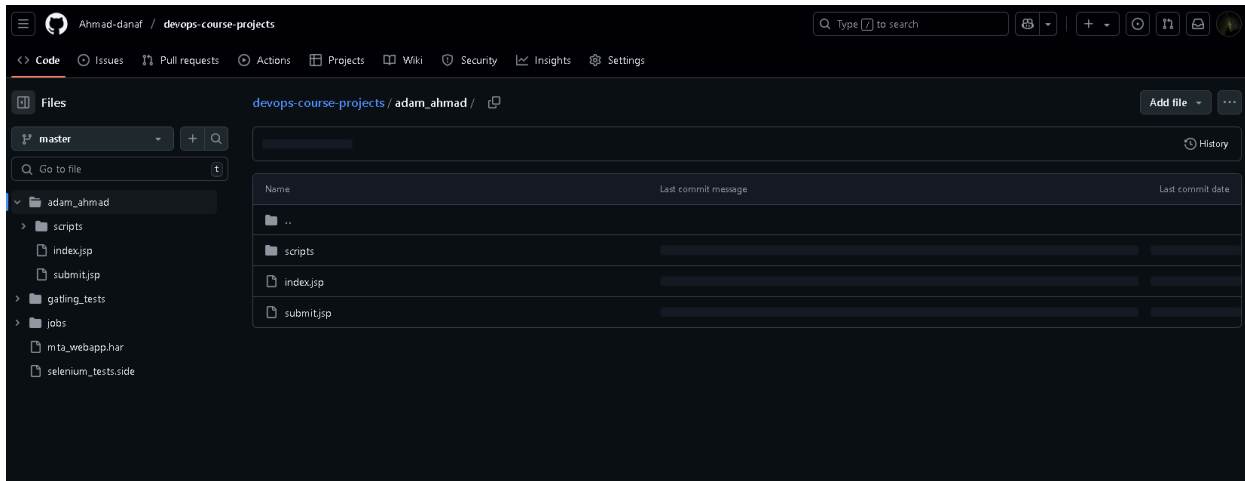
The screenshot shows the Jenkins Dashboard. The top navigation bar includes the Jenkins logo, a search icon, a shield icon with a red exclamation mark, the user name 'Ahmad Danaf', and a 'log out' button. The main content area features a sidebar on the left with links for 'New Item', 'Build History', 'Manage Jenkins', 'My Views', 'Build Queue', and 'Build Executor Status'. The central table displays a list of builds with columns for status, icon, name, last success, last failure, and last duration. The table is filtered by 'All' and shows 6 builds. The bottom right corner of the dashboard displays 'REST API' and 'Jenkins 2.503'.

S	W	Name	Last Success	Last Failure	Last Duration
✓	☀	devops_deploy	14 hr #21	13 days #6	5.7 sec
✓	☁	LoadTest	14 hr #7	16 hr #5	5 min 29 sec
✓	☁	MaxLoadTest	14 hr #7	16 hr #5	53 sec
✓	☀	monitor-webapp	14 hr #19	1 day 21 hr #8	3.6 sec
✓	☀	Run_Selenium_Test	14 hr #10	1 day 20 hr #3	23 sec
✓	☁	StressTest	14 hr #7	16 hr #4	4 min 41 sec

## B

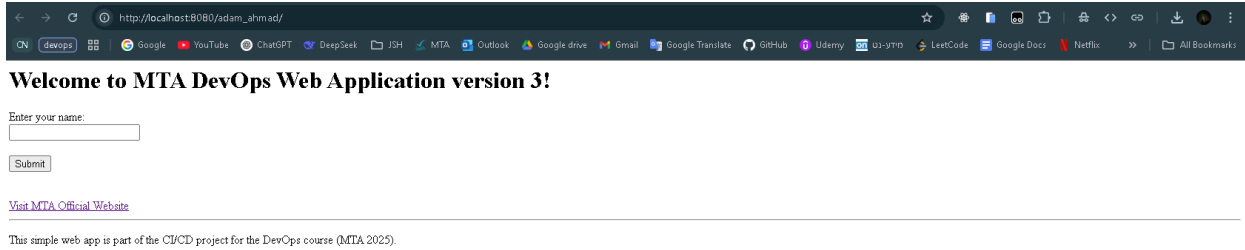


The screenshot shows the GitHub repository page for 'devops-course-projects'. The repository is public and has 354,542 commits and 22 commits. The repository is managed by 'Ahmad-danaf'. The repository contains several files and folders, including 'adam\_ahmad', 'gating\_tests/simulations', 'jobs/deploy', 'mta\_webapp.har', and 'selenium\_tests.side'. The repository also has a README file. The repository is currently on the 'master' branch. The repository has 0 stars, 1 watching, and 0 forks. The repository is currently on the 'master' branch. The repository has 0 stars, 1 watching, and 0 forks. The repository is currently on the 'master' branch. The repository has 0 stars, 1 watching, and 0 forks.



```
index.jsp
learning-notes > deleterme > experiments > devops-course > adam_ahmad > index.jsp > ? > html > body > form > br
You, 2 days ago | 1 author (You)
1 <%@ page contentType="text/html; charset=UTF-8" language="java" %>
2 <html>
3 <head>
4   <title>MTA DevOps WebApp</title>
5 </head>
6 <body>
7   <h1>Welcome to MTA DevOps Web Application version 3!</h1>
8
9   <!-- Text Input -->
10  <form action="submit.jsp" method="post">
11    <label for="username">Enter your name:</label><br>
12    <input type="text" id="username" name="username" required>
13    <br><br>
14
15    <!-- Submit Button -->
16    <input type="submit" value="Submit">
17  </form>
18
19  <br>
20
21  <!-- Clickable Link -->
22  <a href="https://www.mta.ac.il" target="_blank">Visit MTA Official Website</a>
23
24  <hr>
25  <p>This simple web app is part of the CI/CD project for the DevOps course (MTA 2025).</p>
26 </body>
27 </html>
28
```

C:



:D

We used **UptimeRobot** to monitor the availability of our web app. Since the app runs locally on `localhost:8080/adam_ahmad`, we used **ngrok** to expose it to the internet with a public URL. The monitor checks the URL every 5 minutes.

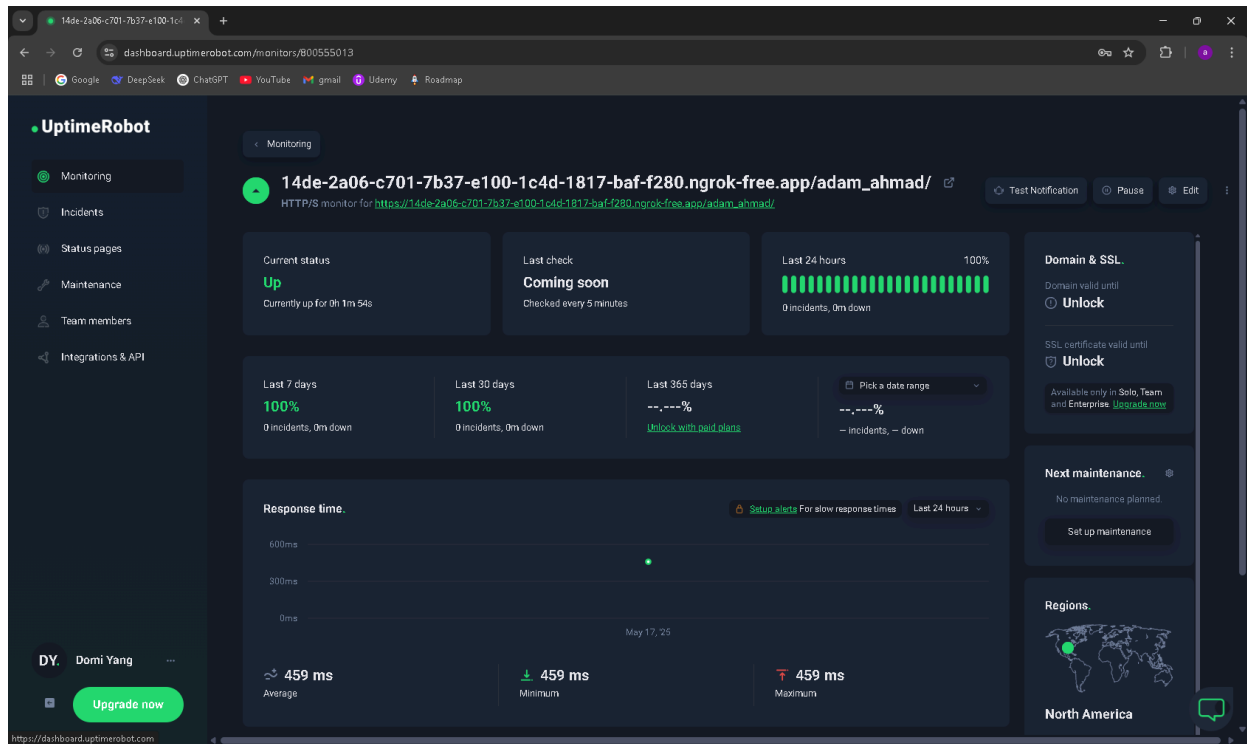
For local and Jenkins-based availability checks, we used:

bash

CopyEdit

```
curl http://localhost:8080/adam_ahmad/
```

Both monitoring methods confirmed the app is available (HTTP 200 OK).



שלב F:

Project: devops\_selenium\_tests

Executing tests.side http://localhost:8080/adam\_ahmad/

	Command	Target	Value
1	open	http://localhost:8080/adam_ahmad/	
2	set window size	1660x990	
3	click	id=username	
4	type	id=username	Adam
5	click	css=input:nth-child(6)	
6	click	linkText=Go Back	
7	click	id=username	
8	type	id=username	Adam
9	click	css=form	
10	click	linkText=Visit MTA Official Website	
11	store window handle	root	
12	select window	handle=\${win8827}	
13	run script	window.scrollTo(0,100)	
14	select window	handle=\${root}	
15	select window	handle=\${win8827}	

Command:  #

Target:

Value:

Description:

Runs: 1 Failures: 0

Log	Reference
10. click on link with text 'Visit MTA Official Website' OK	11:18:17
11. storeWindowHandle on root OK	11:18:17
12. selectWindow on handle=\${win8827} OK	11:18:17
13. runScript on window.scrollTo(0,100) OK	11:18:17
14. selectWindow on handle=\${root} OK	11:18:22
15. selectWindow on handle=\${win8827} OK	11:18:22
*tests.side* completed successfully	

```
C:\Users\ahmad\OneDrive\Desktop\FS_study\before_django\learing-notes\deleteme\experiments\devops-course>selenium-side-runner se
lenium_tests.side
info: Running selenium_tests.side

RUNS ./DefaultSuite.test.js

RUNS ./DefaultSuite.test.js
WARNING: All log messages before absl::InitializeLog() is called are written to STDERR
RUNS ./DefaultSuite.test.js
PASS ./DefaultSuite.test.js (17.199s)
  Default Suite
    ✓ tests.side (10871ms)

Test Suites: 1 passed, 1 total
Tests: 1 passed, 1 total
Snapshots: 0 total
Time: 18.444s
Ran all test suites.
```

שלב (g) – תיאור תרחיש HAR:

The HAR scenario performs the following steps:

1. The user visits the home page at `/adam_ahmad/index.jsp`.
2. They enter their name into the input text box (e.g., "Ahmad").
3. They click the **Submit** button, which sends a POST request to `/adam_ahmad/submit.jsp`.

- The response page displays a personalized greeting with the user's name and a **"Go Back"** link.
- The user clicks **"Go Back"**, which returns them to the home page.
- Finally, the user clicks the link to the **MTA Official Website**, navigating to <https://www.mta.ac.il>.

שלב (j)

תמונה זו מציגה את שלב **Load Test** כחלק מ-j Deliverable. התמונה מציגה את סיכום הרצת בדיקת העומס, עם 3000 בקשות שבוצעו בהצלחה וללא כשלים (OK 100%).

```

Stress Test
=====
[#####] 91.91%
waiting: 900 / active: 1 / done: 11,139

2025-06-01 11:50:11 GMT
-----
Requests -----
> Global | 11,625 | 11,625 | 0
> Stress Request | 11,625 | 11,625 | 0

Stress Test
=====
[#####] 95.92%
waiting: 493 / active: 2 / done: 11,625

2025-06-01 11:50:16 GMT
-----
Requests -----
> Global | 12,120 | 12,120 | 0
> Stress Request | 12,120 | 12,120 | 0

Stress Test
=====
[#####] 100%
waiting: 0 / active: 0 / done: 12,120

Parsing log file(s)...
Parsing log file(s) done in 0s.
Generating reports...

Global Information
-----
> request count | 12,120 | 12,120 | -
> min response time (ms) | 1 | 1 | -
> max response time (ms) | 42 | 42 | -
> mean response time (ms) | 5 | 5 | -
> response time std deviation (ms) | 2 | 2 | -
> response time 50th percentile (ms) | 5 | 5 | -
> response time 75th percentile (ms) | 6 | 6 | -
> response time 95th percentile (ms) | 8 | 8 | -
> response time 99th percentile (ms) | 9 | 9 | -
> mean throughput (rps) | 50.5 | 50.5 | -

Response Time Distribution
-----
> OK: t < 800 ms | 12,120 | (100%)
> OK: 800 ms <= t < 1200 ms | 0 | (0%)
> OK: t >= 1200 ms | 0 | (0%)
> KO | 0 | (0%)

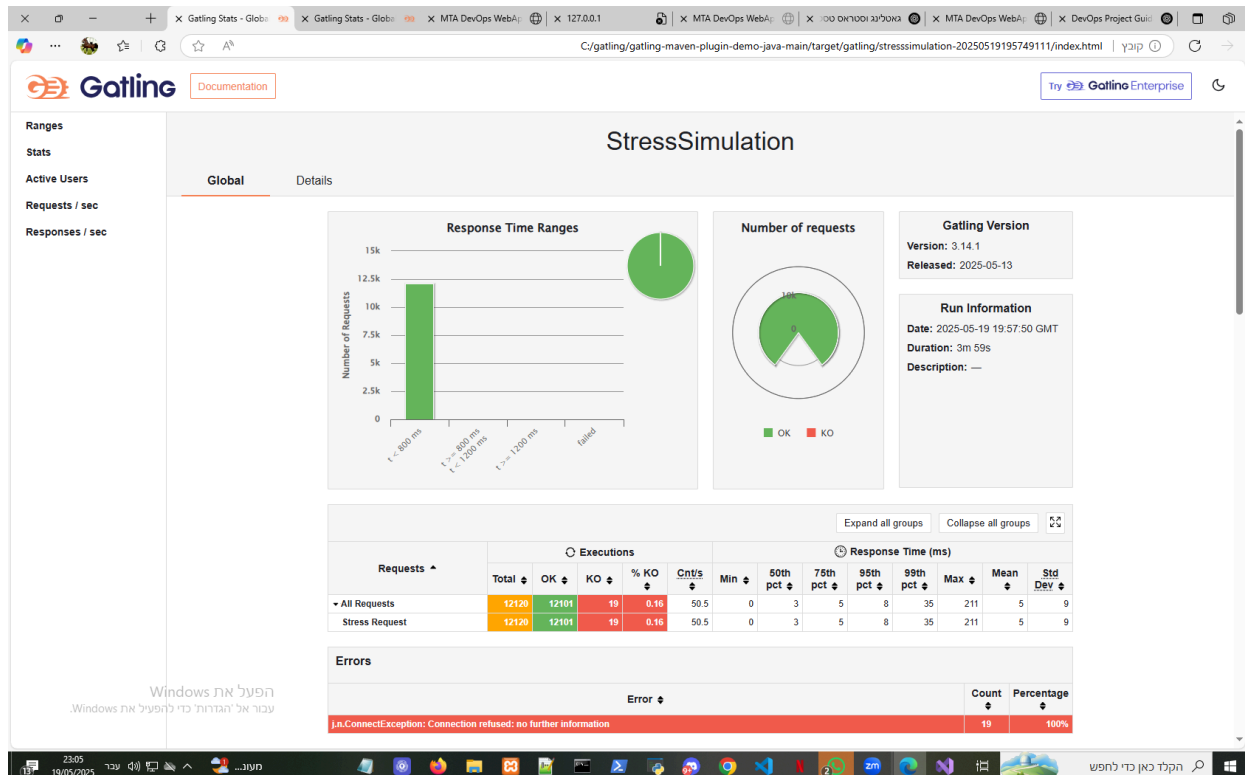
Reports generated, please open the following file: file:///C:/gatling/gatling-maven-plugin-demo-java-main/target/gatling/stresssimulation-20250601114614191/index.html
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 04:19 Windows
[INFO] Finished at: 2025-06-01T14:50:17+03:00
[INFO] -----
c:\gatling\gatling-maven-plugin-demo-java-main>

```

## Stress Test

תמונה זו כוללת סיכום הרצת בדיקת העומס הקיצונית עם 12120 בקשות מתוך 19 נכשלו, המייצגות 0.16% כשלים בלבד.

תמונה זו נועדה להמחיש את ביצוע המערכת בעומס מרבי שנמשך כ-4 דקות, בהתאם לדרישות הבדיקה שהוגדרו במטלה.



## (f) Selenium IDE Validations Explanation

I used the following validations in my Selenium IDE .side test:

### Text Field Validation

Command: type

Target: id=username

Purpose: To verify that the input field correctly receives and displays user input (Adam).

- This validates user interaction and form readiness.

### Button Click Validation

Command: click

Target: css=input:nth-child(6) (Submit button)

Purpose: To test the form submission behavior.

-This confirms that clicking a button triggers the expected action.

### Link Navigation Validation

Command: click

Target: linkText=Visit MTA Official Website

Purpose: To verify correct navigation to an external page in a new window.  
-This checks for proper link handling and multi-window behavior.

These validations ensure that the basic user interactions input, submission, and redirection are functioning correctly.

(h) Attached HAR file – `devops\_test.har`

Note: the HAR file was recorded using port 8081 (older local run) but reflects the same interaction flow used in the Gatling and Selenium simulations:

1. User types a name in the text field
2. Clicks the Submit button
3. Clicks the "Go Back" link
4. Clicks the "Visit MTA Official Website" link
5. Switches between windows

The flow represents real user behavior and matches the tests executed in my current simulations.

(.har .מצורף גם קובץ)



```
שורת הפקודה
2025-06-01 11:35:25 GMT
--- Requests ---
> Global | Total | OK | KO |
> Load Request | 2,500 | 2,500 | 0 |

--- Load Test ---
[=====] 83.33%
waiting: 5 / active: 0 / done: 25

2025-06-01 11:35:29 GMT
--- Requests ---
> Global | Total | OK | KO |
> Load Request | 3,000 | 3,000 | 0 |

--- Load Test ---
[=====] 100%
waiting: 0 / active: 0 / done: 30

Parsing log file(s)...
Parsing log file(s) done in 0s.
Generating reports...

--- Global Information ---
> request count | 3,000 | 3,000 | - |
> min response time (ms) | 2 | 2 | - |
> max response time (ms) | 152 | 152 | - |
> mean response time (ms) | 3 | 3 | - |
> response time std deviation (ms) | 3 | 3 | - |
> response time 50th percentile (ms) | 3 | 3 | - |
> response time 75th percentile (ms) | 3 | 3 | - |
> response time 95th percentile (ms) | 6 | 6 | - |
> response time 99th percentile (ms) | 9 | 9 | - |
> mean throughput (rps) | 100 | 100 | - |

--- Response Time Distribution ---
> OK: t < 800 ms | 3,000 | (100%) |
> OK: 800 ms <= t < 1200 ms | 0 | (0%) |
> OK: t >= 1200 ms | 0 | (0%) |
> KO | 0 | (0%) |

Reports generated, please open the following file: file:///C:/gatling/gatling-maven-plugin-demo-java-main/target/gatling/loadsimulation-20250601113459100/index.html
[INFO] BUILD SUCCESS
[INFO] Total time: 45.573 s
[INFO] Finished at: 2025-06-01T14:35:30+03:00
[INFO]

c:\gatling\gatling-maven-plugin-demo-java-main>

הפעל את Windows
עבור אל הגדרות כדי להפעיל את Windows
```

הקובץ המצורף (CMD) דרך שורת הפקודה Gatling של **Load Test** תמונה זו מתעדת את הרצת עם כל (Summary) הוא מציג את סיכום הריצה Person B. בתפקיד (j) Deliverable הוא חלק מ- (KO) ואחוזי כישלון (OK) הפרמטרים החשובים: מספר בקשות, זמני תגובה, מספר אחוזי הצלחה בפורמט המצופה. Gatling זה נדרש לצורך המטלה של מציאת הגבול המקסימלי והצגת תוצאות

```
Parsing log file(s)...
Parsing log file(s) done in 0s.
Generating reports...

--- Global Information ---
> request count | 400 | 400 | - |
> min response time (ms) | 2 | 2 | - |
> max response time (ms) | 358 | 358 | - |
> mean response time (ms) | 11 | 11 | - |
> response time std deviation (ms) | 35 | 35 | - |
> response time 50th percentile (ms) | 7 | 7 | - |
> response time 75th percentile (ms) | 7 | 7 | - |
> response time 95th percentile (ms) | 10 | 10 | - |
> response time 99th percentile (ms) | 282 | 282 | - |
> mean throughput (rps) | 13.33 | 13.33 | - |

--- Response Time Distribution ---
> OK: t < 800 ms | 400 | (100%) |
> OK: 800 ms <= t < 1200 ms | 0 | (0%) |
> OK: t >= 1200 ms | 0 | (0%) |
> KO | 0 | (0%) |

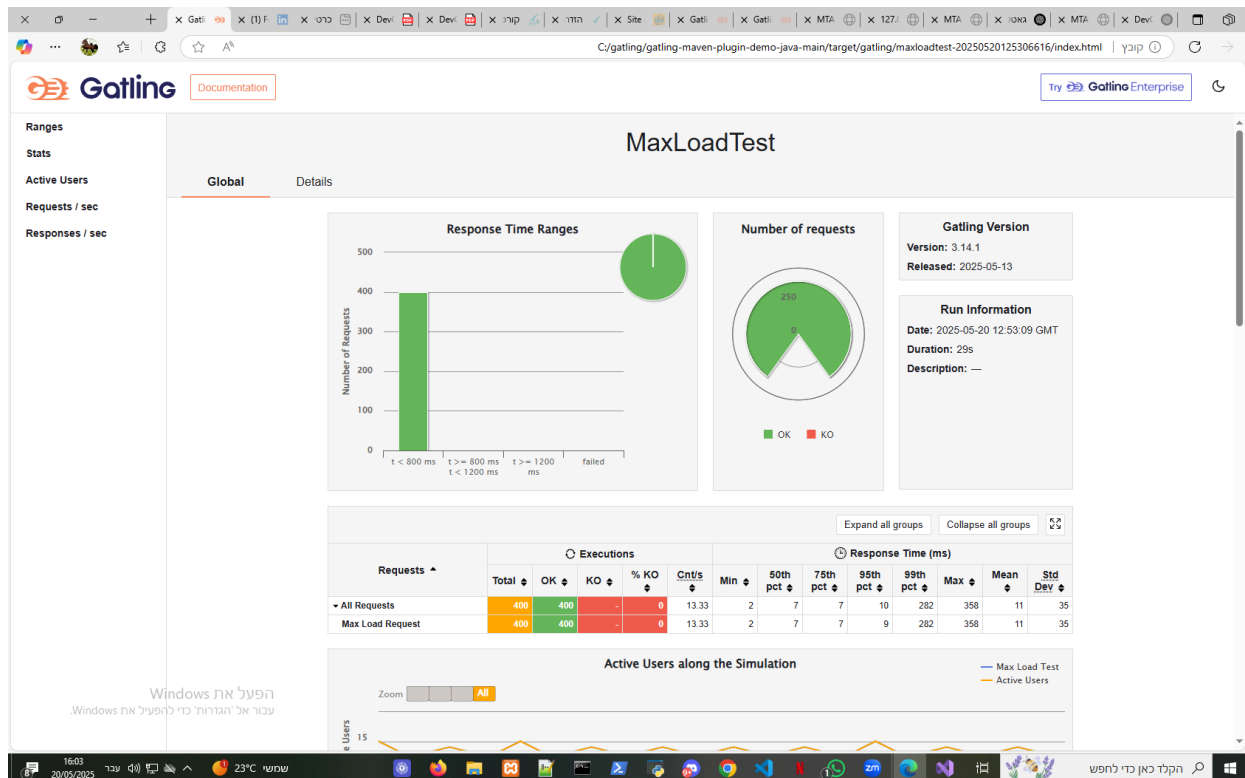
Reports generated, please open the following file: file:///C:/gatling/gatling-maven-plugin-demo-java-main/target/gatling/maxloadtest-20250520125306616/index.html
[INFO] BUILD SUCCESS
[INFO] Total time: 47.078 s
[INFO] Finished at: 2025-05-20T15:53:39+03:00
[INFO]

הפעל את Windows
עבור אל הגדרות כדי להפעיל את Windows
```

Gatling מתוך כלי **Max Load Test** בתמונה זו מוצג סיכום של

( $KO=0$ ) ללא שגיאות (BUILD SUCCESS) הבדיקה בוצעה בהצלחה.  
 (ps) כמות הבקשות שבוצעה הייתה 400, כולן הסתיימו בהצלחה עם ממוצע זמן תגובה נמוך (13.33)  
 בלבד. ממוצע תגובה מרבי של 35  
 זוהי בדיקה מתוך הדרישות של המטלה, במסגרתה נבדק הגבול המקסימלי של המערכת - כמה  
 בקשות בו זמנית היא מסוגלת לעבד לפני קריסה או האטה.

✓ הסבר מהמטלה: בדיקה זו מהווה את השלב הראשון בגטלינג (Max Limit Test), לפני המעבר  
 לבדיקה הבאה (Load Test).



התמונה הזו מציגה את תוצאות **Max Load Test** בגטלינג.  
 כפי שנדרש במטלה, זהו החלק שבו בוצעה בדיקה למציאת נקודת השבירה (**max limit**) של  
 המערכת.  
 הגרף מראה כי בוצעו 400 בקשות – כולן התקבלו בהצלחה ( $OK = 400$ ,  $KO = 0$ ).  
 מדדים חשובים נוספים בתמונה:  
 -זמן תגובה מרבי – 358ms  
 - ממוצע – 11ms  
 - אין כשלים

\* Deliverable: התמונה הזו מיועדת להיכלל בתיק ההגשות תחת הקטגוריה "Max Load Test – Gatling Run Summary Screenshot".

\*\* מצורפים קבצי טקסט והסברים לשלבים J+K+I בגיט ומצורף למייל.