

Load Testing Tool — Project Documentation

This tool simulates HTTP load on a given website by sending multiple requests, inspired by legacy tools like **LOIC**. It is developed strictly for **educational purposes only**, with a **100-request cap** to prevent misuse.

Technologies Used

- **Frontend:** React, Tailwind CSS
 - **Backend:** Java 17, Spring Boot
 - **Deployment:** AWS EC2 (backend), Vercel (frontend), Cloudflare Tunnel
 - **Version Control:** Git & GitHub
-

Key Concepts Practiced

- Java multithreading (ExecutorService)
 - Exception handling and logging
 - Cross-origin API request handling (CORS)
 - Secure API tunneling using Cloudflare Tunnel
 - Smooth React → Spring Boot integration
 - Auto-deployment with Vercel via GitHub
-




Security Considerations

- Request count limited to 100 to avoid abuse
 - CORS configured for frontend-origin access only
 - Backend access secured via Cloudflare Tunnel instead of exposing EC2 public IP
-

How It Works

1. User enters a **URL** and **request count** (1–100) in the React UI.
2. React sends a POST request to `/api/send-requests`.
3. Spring Boot backend launches multiple threads to hit the target URL.
4. Each thread sends a GET request using Java's `HttpClient`.
5. Backend logs success/failure for each request.

Project Links

-  **Live App:** website-down-tester.vercel.app
 -  **Frontend Repo:** github.com/Appash047/website-down-tester
 -  **Backend Repo:** github.com/Appash047/website-down-tester-backend
-

How to Run Locally

Backend (Spring Boot)

```
git clone https://github.com/Appash047/website-down-tester-backend.git
cd website-down-tester-backend
./mvnw spring-boot:run
```

Frontend (React)

```
git clone https://github.com/Appash047/website-down-tester.git
cd website-down-tester
npm install
npm start
```

Disclaimer

This tool is for **educational purposes only**.

Do **not** use it to target or harm real websites. Doing so may be **illegal** and unethical.