Міністерство освіти і науки України

Запорізький національний технічний університет

кафедра програмних засобів

РЕФЕРАТ

з дисципліни: «Якість програмного забезпечення»

на тему: «Тестування навантаження. Fiddler»

Виконав:

студент групи КНТ-415 Д. О. Тарасенко

Перевірила:

к.т.н., професор Г. В. Табунщик

Запоріжжя

2018ЗМІСТ

[1. ТЕСТУВАННЯ НАВАНТАЖЕННЯ 3](#_Toc531550716)

[2. FIDDLER 4](#_Toc531550717)

[СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 7](#_Toc531550718)

# ТЕСТУВАННЯ НАВАНТАЖЕННЯ

Тестування частини веб-додатки, розміщеної на веб-сервері, можна провести і минаючи графічний (клієнтський) інтерфейс, проте це вимагає від фахівця певного рівня знань і навичок технічного характеру, а також застосування додаткових інструментів. Розглянемо веб-сервер з точки зору навантажувального тестування.

Тестування навантаження імітує роботу з додатком певної кількості користувачів. Цей вид тестування здійснюється за допомогою спеціальних інструментів (наприклад, Fiddler та jMeter), головна мета яких - визначити профілі навантаження і штучно створити для них навантаження, що виявляє граничні можливості програми (або сервера) в умовах роботи з ним тієї чи іншої кількості користувачів.

Отримана інформація піддається ретельному аналізу з подальшим виявленням «вузьких шляхів» і граничних програмних і апаратних можливостей, які в подальшому використовуються для забезпечення стабільності веб-сервера і самого додатка, що працює на ньому.

# FIDDLER

Fiddler - ПО для відстеження всього вашого трафіку та генерації запитів. Використовуючи Fiddler, можна з легкістю відстежувати всі запити від клієнта і відповіді, переглядати їх деталі, а також вносити свої зміни і відправляти модифіковані запити на сервер, оцінюючи поведінку системи в такому випадку.

На практиці, використовуючи при тестуванні Fiddler, ви без зусиль зможете впорядкувати свої запити і відповіді за кодом стану і відібрати, наприклад, все 400-і і 500-і коди з подальшим їх аналізом. Таким чином дуже швидко «відловлюються» дефекти з стилями, скриптами, файлами, функціями програми, що не працюють.

Почавши роботу з ним, ми тим самим розробляємо віртуальний web proxy, а значить, кожен запит від вашого локального браузера спрямовуватиметься виключно через його структуру. Всередині програми ви зможете розглядати всі до єдиного HTTP запити, наприклад, картинки, структуру CSS коду, JS скрипти і інші складові елементи класичного веб-ресурсу. Будь-який запит і відповідь має свої персональні параметри, за допомогою яких можна судити про оптимальність або проблемах функціонування розроблюваного веб-сайту.

Наприклад, ви можете детально проаналізувати такі ситуації, як віддача веб-порталом невеликих ресурсів - невеликих картинок або іконок. Логічно, що їх запросто можна об'єднати в один спрайт, щоб виконувати лише один запит, що кардинальним чином знизить ступінь і час завантаження медіаресурсів.

Вас також можуть насторожити деякі формати і розміри веб-ресурсів. Дуже великі зображення найкраще оптимізувати, наприклад, за допомогою конвертації в інший формат. Для текстових даних можна використовувати поширену конвертацію в Gzip-формат.

До слова, саме Fiddler демонструє, включено стиснення для вхідного трафіку чи ні. Даний параметр можна запросто простежити за спеціальним заголовку Content-Encoding: gzip.

Кожен цікавить запит можна проаналізувати в формі спеціальної часової стрічки, по якій максимально чітко видно, який проміжок часу необхідний, щоб вся сторінка повністю завантажилася, а також час, необхідний для оптимального відображення кожного її структурного елементу.

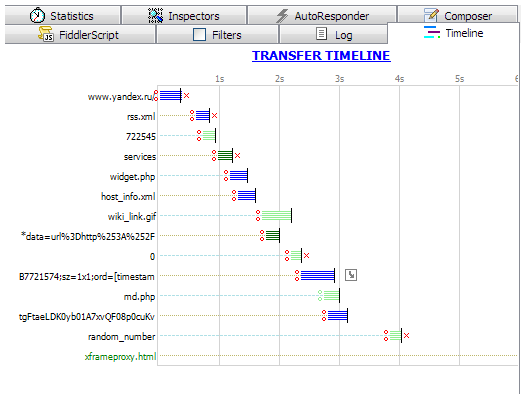


Рисунок 3.1 — Часова стрічка

У додатку є достатня кількість зручних і корисних функцій, за допомогою яких, наприклад, можна виставити параметри повільного з'єднання. Так тестувальники можуть побачити весь процес завантаження кожного окремого елемента сайту за умови, що у клієнтів буде дуже повільний інтернет.

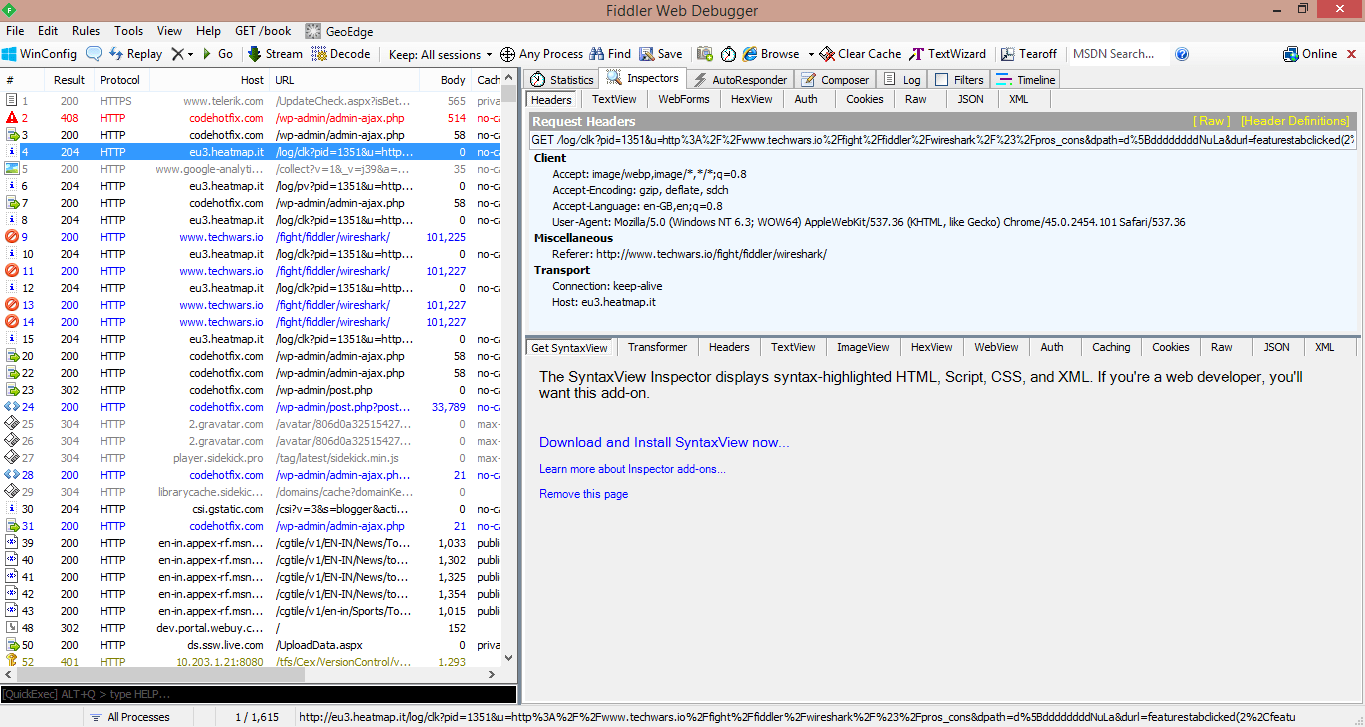


Рисунок 3.2 — Вікно запитів

До речі, дуже корисно, що можна наочно переглянути, які елементи завантажуються першими, а які - в останню чергу. Тобто можна оптимізувати порядок вивантаження всіх веб-елементів.

Також можна відзначити наявність спеціального редактора для HTTP запитів. У користувача з'явиться реальна можливість підміняти будь-який обраний HTTP заголовок за допомогою спеціального скриптового редактора.

Також можна використовувати Firebug, а не Fiddler, але з точки зору оптимізації параметрів саме Fiddler можна вважати найбільш багатофункціональним і оптимізованим додатком.

# Також можна відзначити, що він абсолютно безкоштовний, постійно розвивається, а значить, дозволяє оперувати виключно свіжими і працездатними версіями.

**СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ**

1. Отладка с помощью fiddler и charles [Електронний ресурс]. – Режим доступу: <https://testmatick.com/ru/otladka-s-pomoshhyu-fiddler-i-charles/>
2. Інженерія якості програмного забезпечення:навчальний посібник / Г.В Табунщик, Р.К. Кудерметов, Т.І. Брагіна. – Запоріжжя: ЗНТУ, 2013. – 180 с.
3. Особенности тестирования веб-приложений [Електронний ресурс]. – Режим доступу: http://quality-lab.ru/key-principles-of-web-testing/;
4. Fiddler – помощник в отладке Javascript [Електронний ресурс]. – Режим доступу: <https://habr.com/post/140147>;
5. Использование Fiddler для эмуляции различных сетевых условий в автотестах[Електронний ресурс]. – Режим доступу: <https://www.slideshare.net/VLDCORP/fiddler-69866720>