



НАВАНТАЖУВАЛЬНЕ ТЕСТУВАННЯ ТА АНАЛІЗ ПОКАЗНИКІВ ЙОГО ПРОДУКТИВНОСТІ

Хламов С.В., доцент, кафедра МСТ, ХНУРЕ
Менделєва М.В., асистент, кафедра МСТ, ХНУРЕ
Трунова Т.О., асистент, кафедра МСТ, ХНУРЕ

Abstract. *The theses are devoted to methods of load testing of modern software systems and analysis of their performance indicators. The principles of assessing the stability, scalability and efficiency of systems under different load levels are considered, as well as the issues of planning test scenarios and interpreting results to optimize the operation of digital resources.*

Keywords: *load testing, performance, metrics, stress testing, system resources, optimization.*

Сучасні інформаційні системи функціонують у середовищі, що характеризується високою динамікою користувацьких запитів та великим обсягом оброблюваних даних. У таких умовах продуктивність програмних рішень стає критичною, адже навіть короточасне перевантаження може призвести до втрати даних, зниження якості обслуговування або повного збоїв системи. Навантажувальне тестування виступає ключовим інструментом оцінки здатності системи ефективно працювати під інтенсивним навантаженням. Воно дозволяє не лише виявляти вузькі місця, але й прогнозувати поведінку системи у пікові моменти, що є важливою умовою забезпечення стабільності та надійності цифрового продукту [1].

Навантажувальне тестування передбачає створення контрольованих умов, у яких система працює під різними рівнями користувацьких запитів. Наприклад, один зі сценаріїв може моделювати одночасне підключення 500 користувачів, що виконують типові дії – заповнення форм, завантаження файлів, перегляд сторінок. Інший сценарій передбачає поступове збільшення навантаження від 100 до 2000 запитів на хвилину для оцінки меж стабільної роботи системи. Такі моделювання дозволяють визначати як час відгуку сервера, так і середній обсяг оброблюваних даних у секунду, а також виявляти можливі точки деградації.

У процесі тестування важливе значення має аналіз поведінки системи у пікові моменти. Наприклад, при різкому збільшенні запитів час відгуку може зростати з 0,5 секунд до понад 5 секунд, що свідчить про перевантаження серверних ресурсів або неефективну обробку запитів. На підставі таких спостережень розробляються рекомендації щодо оптимізації алгоритмів, балансування навантаження та масштабування інфраструктури.

Особливе значення в навантажувальному тестуванні має моделювання *реалістичних сценаріїв користувацької поведінки*. Для цього важливо враховувати як кількість одночасних користувачів, так і різноманіття їх дій у системі. Наприклад, частина користувачів може виконувати прості запити, такі як перегляд сторінок або отримання звітів, тоді як інші одночасно створюють і редагують великі обсяги даних або завантажують мультимедійні файли. Таке



розділення дозволяє оцінити, як система справляється з комбінованим навантаженням і чи виникають вузькі місця на різних рівнях архітектури.

Також значущим є *посттестовий аналіз логів і метрик*, який дозволяє не лише оцінити ефективність роботи системи, але й ідентифікувати конкретні компоненти, що потребують оптимізації. Це включає аналіз часу виконання запитів, використання ресурсів сервера та бази даних, частоти помилок, а також інцидентів блокування або очікування. Завдяки такому комплексному підходу розробники отримують конкретні рекомендації щодо покращення продуктивності, що дозволяє системі витримувати високі навантаження без втрати стабільності та якості обслуговування користувачів. Продуктивність системи оцінюється через комплекс метрик. Найпоширеніші з них включають середній, мінімальний та максимальний час відгуку, кількість оброблених запитів на одиницю часу (Throughput), відсоток помилок та використання ресурсів серверу, таких як CPU та оперативна пам'ять. Наприклад, тестування веб-додатку показало середній час відгуку 1,2 секунди при навантаженні 800 одночасних користувачів, тоді як максимальний час відгуку досягав 4,5 секунд, а рівень помилок не перевищував 0,8%. Такі дані дозволяють зрозуміти, на яких компонентах системи слід зосередити оптимізаційні зусилля.

Іншим важливим аспектом є *аналіз динаміки змін показників у часі*. Наприклад, при тривалому навантаженні на веб-сервіс спостерігалось поступове зростання використання пам'яті до 85%, що сигналізує про можливий витік ресурсів та необхідність оптимізації коду. Такі детальні спостереження допомагають прогнозувати поведінку системи у довгостроковій перспективі та забезпечують надійність її роботи навіть у пікові моменти.

Результати навантажувального тестування та аналізу продуктивності дозволяють приймати обґрунтовані рішення щодо розширення серверних потужностей, зміни конфігурації баз даних та застосування методів кешування. Використання інструментів моніторингу у реальному часі, таких як JMeter або LoadRunner, забезпечує можливість оперативно реагувати на зміни навантаження та підтримувати стабільну роботу системи. Важливо також враховувати вплив навантаження на користувацький досвід: навіть технічно стабільна система може втрачати ефективність, якщо час очікування користувача перевищує прийнятні межі.

Навантажувальне тестування є необхідним етапом забезпечення якості сучасних програмних систем. Комплексний підхід до оцінки продуктивності – включаючи планування сценаріїв, аналіз ключових метрик та оцінку поведінки системи під піковим навантаженням – дозволяє виявляти вузькі місця та приймати ефективні рішення щодо оптимізації. Така методологія забезпечує стабільність роботи системи, покращує користувацький досвід та гарантує готовність до змінних та екстремальних умов експлуатації.

Список літератури

1. Mendieliyeva, M., Khlamov, S.V., Vovk, O., & Trunova, T.O. (2025). Analysis of percentiles for performance of load testing tools. Information Control Systems and Technologies (ICST-ODESA-2025). (p. 247-249).