

**Non-functional testing (Нефункціональне тестування)** включає в собі тестування вимог, які напрямую не зв'язані з функціональністю, таких як продуктивність, безпека, масштабованість, зручність використання, надійність тощо.

**Завдання:** Відкрийте веб-застосунок <https://finance.ua>. Опишіть, які види нефункціонального тестування можуть бути застосовані для цього веб-застосунку, поясніть, чому саме ці види тестування були б актуальними і як би ви їх застосовували?

[Finance.ua](https://finance.ua) - це український фінісовий портал, яким щодня користуються тисячі організацій та десятки тисяч користувачів. Тому на перший план для такого ресурсу виходять такі критерії, як надійність, захищеність, масштабованість, зручність у користуванні. Виходячи з цих вимог я запропонував би для веб-застосунку <https://finance.ua> наступні види нефункціонального тестування:

1. **Тестування навантаження (Load Testing)** — визначення масштабованості додатку під навантаженням, при цьому відбувається: вимір часу виконання вибраних операцій за певних інтенсивностей виконання цих операцій; визначення кількості користувачів, що одночасно працюють з додатком; визначення меж прийнятної продуктивності при збільшенні навантаження (при збільшенні інтенсивності виконання цих операцій); дослідження продуктивності при високих, граничних, стресових навантаженнях.
2. **Стресове тестування (Stress Testing)** дозволяє перевірити, наскільки додаток і система в цілому працездатні в умовах стресу, і також оцінити здатність системи до регенерації, тобто до повернення до нормального стану після припинення впливу стресу. Стресом у даному контексті може бути підвищення інтенсивності виконання операцій до дуже високих значень або аварійна зміна конфігурації сервера. Також одним із завдань при стресовому тестуванні може бути оцінка деградації продуктивності, таким чином цілі стресового тестування можуть перетинатися з цілями тестування продуктивності.
3. **Тестування на відмову і відновлення (Failover and Recovery Testing)** перевіряє додаток з точки зору здатності протистояти й успішно відновлюватися після можливих збоїв, що виникли у зв'язку з помилками програмного забезпечення, відмовами обладнання або проблемами зв'язку (наприклад, відмова мережі). Метою даного виду тестування є перевірка систем відновлення (або дублюючих основний функціонал систем), які, у

разі виникнення збоїв, забезпечують збереження і цілісність даних тестованого продукту.

4. **Конфігураційне тестування (Configuration Testing or Portability Testing)** — ще один вид традиційного тестування продуктивності. У цьому випадку замість того, щоб тестувати продуктивність системи з точки зору навантаження, тестується ефект впливу на продуктивність змін у конфігурації. Прикладом такого тестування можуть бути експерименти з різними методами балансування навантаження. Конфігураційне тестування також може бути поєднане з навантажувальним, стрес- або тестуванням стабільності.
5. **Тестування локалізації (Localization Testing)** — перевірка правильності перекладу елементів інтерфейсу користувача, перевірка правильності перекладу системних повідомлень і помилок, перевірка перекладу розділу «Допомога», « Довідка» і супровідної документації. Мета тесту локалізації — переконатися, що додаток підтримує багатомовний інтерфейс і функції. А також проблеми, пов'язані з локалізацією (переклад на іншу мову, формат дат і чисел, поштові адреси, порядок імені та прізвища, валюти і т. д.). Орфографія і граматики зазвичай не тестуються.
6. **Тестування установки (Installation testing)** спрямоване на перевірку успішної інсталяції та настройки, а також оновлення або видалення програмного забезпечення. На даний момент найбільш поширена установка ПЗ за допомогою інсталяторів (спеціальних програм, які самі по собі так само потребують належного тестування). У реальних умовах інсталяторів може не бути. У цьому випадку доведеться самостійно виконувати установку програмного забезпечення, використовуючи документацію у вигляді інструкцій або *readme* файлів, де крок за кроком описано всі необхідні дії та перевірки;
7. **Тестування зручності користування (Usability Testing)** — це метод тестування, спрямований на встановлення ступеня зручності використання, навченості, зрозумілості та привабливості для користувачів розроблюваного продукту в контексті заданих умов.