

**МЕТОДИ ЗАБЕЗПЕЧЕННЯ ТА КОНТРОЛЮ ЯКОСТІ  
WEB-ЗАСТОСУВАНЬ НА СТАДІЯХ ЖИТТЄВОГО ЦИКЛУ**

Актуальність побудови моделей якості WEB-застосувань обумовлена тим, що створення якісного програмного продукту – основна мета будь-якого проекту з розробки ПЗ. Забезпечення якості ПЗ при проектуванні у відповідності з вимогами користувача повинно базуватись на реалізації процесу «управління якістю».

Процес «управління якістю» повинен реалізуватись шляхом моніторингу якості проміжних продуктів на всіх етапах життєвого циклу (ЖЦ) ПЗ, таким чином, щоб кінцевий програмний продукт відповідав користувацьким вимогам. Для цього необхідно визначити вимоги якості до проміжних продуктів, в тому числі і до архітектури ПЗ.

За результатами дослідження та аналізу міжнародних стандартів по оцінюванні якості ПЗ ISO/IEC TR 9126-2. Softwareengineering – Productquality – Part 2: Externalmetrics, 2003 – 86 p.ISO/IEC 25010. Software Engineering – product quality. Quality model, зовнішня модель якості була перероблена в контексті застосування її до web-додатків (із урахуванням критеріїв функціональності, надійності, зручності використання, ефективності, супроводжуваності, переносимості).

Внутрішня якість визначається потребами розробників і адміністраторів застосування. Розробники в першу чергу фокусуються на читаності коду, який легко зрозуміти, адаптувати й розширювати. Якщо вони не зроблять цього, реалізація наступних вимог клієнта стає більш складною й, отже, більш дорогою з часом. Це підвищує ризик того, що навіть невеликі зміни в програмному забезпеченні можуть привести до несподіваних побічних ефектів.

Для дослідження внутрішньої якості було використано аналіз метрик коду та аналіз продуктивності за допомогою JetBrains' dotTrace Performance v4.5.950.32.

Метрики коду – це кількісні показники, що дозволяють оцінити якість вихідного коду програми. Так як для розробки web-застосування використовувалася VisualStudio, а також так як вона дозволяє вимірювати п'ять основних метрик (MaintainabilityIndex) – комплексний показник якості коду, Пов'язаність (coupling), Число рядків коду (LinesofCode), Цикломатична складність (Cyclomaticcomplexity), Глибина спадкування (DepthofInheritance)), то для їх виміру була також використана вона.

Для виміру продуктивності, було використано dotTracePerformance – це профайлер для додатків, розроблених для платформи Microsoft.Net. Він дозволяє вести профілювання настільних (desktop) додатків, веб-додатків, під IIS або ASP