

## **РОЗРОБКА ТА ДОСЛІДЖЕННЯ СИСТЕМИ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ ДЛЯ МЕРЕЖЕВИХ СЕРВІСІВ**

**Підлісник М.В., Подорожняк А.О.**  
*Національний технічний університет  
«Харківський політехнічний інститут»,  
м. Харків*

При розробці мережесервісів багато розробників стикаються з тим що їх зміни до кодової бази програмного додатку можуть призвести до програмних недоліків, більш поширена назва яких – «баги». Традиційно, перевіркою усіх аспектів роботи програмного додатку займалася окрема людина (або група людей), що монотонно десятки, сотні й тисячі разів виконувала одні й ті ж сценарії роботи з програмним додатком, аби звести до мінімуму вірогідність того, що будь-який недолік дійде до кінцевої версії програмного забезпечення.

Автоматизація тестування передбачає, що машина буде виконувати те, що перш робила людина, при цьому набагато швидше, з меншою ймовірністю помилки і безперервно. Це можна порівняти з появою конвеєрної стрічки в епоху промислової революції, коли машини стали виконувати монотонні й нудні операції за людини. Люди схильні швидко втрачати концентрацію, роблячи таку роботу, і упускати наявні помилки і недоліки. У цьому відношенні машини діють набагато швидше і надійніше.

Слід зазначити, що система автоматизованого тестування дозволяє автоматично генерувати й надсилати графічні звіти про хід виконання автоматизованих тестів, що значно поліпшує її використання усіма розробниками веб-сервісу.

У роботі пропонується розробка веб-сервісу з продажу ігор, використовуючи технології HTML, CSS, JavaScript, Java. Веб-сервіс побудований з використанням архітектури REST. Розробка бази даних буде виконана за допомогою об'єктно-реляційної системи керування базами даних «PostgreSQL».

Для веб-сервісу також пропонується розробка системи автоматизованого тестування, що базується на використанні системи неперервної інтеграції «Travis CI». Така система автоматизованого тестування може використовувати технології Jasmine, Selenide, Cucumber. Пропонуються до розробки такі рівні тестування:

- 1) юніт-тести – тестування одного модуля коду (одна функція або один клас в разі ООП-коду) в ізольованому оточенні;
- 2) інтеграційні тести – тестування якогось компонента системи, який зазвичай складається з багатьох модулів (класів або функцій);
- 3) тестування GUI – тестування графічного інтерфейсу, тобто того що користувач бачить на екрані;
- 4) тестування API – тестування прикладного програмного інтерфейсу.

Метою подальших досліджень є розробка запропонованого веб-сервісу з продажу ігор та системи для його автоматизованого тестування.