Науковий вісник НЛТУ України
Scientific Bulletin of UNFU
https://doi.org/10.36930/40350321
Article received 09.04.2025
Article accepted 29.05.2025
UDC 004.415.2
Correspondence author: Yu. S. Nazar kordunovayulia@gmail.com
ISSN 1994-7836 (print)
ISSN 2519-2477 (online)

О. М. Шопський[1], М. Б. Лінник[1], М. Б. Лінник[2], Ю. С. Назар[1]
[1] Львівський державний університет безпеки життєдіяльності, м. Львів, Україна
[2] SoftServe, м. Львів, Україна

РОЗРОБЛЕННЯ ТА ВПРОВАДЖЕННЯ ПЛАТФОРМИ ДЛЯ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ СИСТЕМ ОПОВІЩЕННЯ ТА МОНІТОРИНГУ НАДЗВИЧАЙНИХ СИТУАЦІЙ

Збільшення природних і техногенних надзвичайних ситуацій за останні роки свідчить про критичну важливість надійних систем раннього оповіщення населення для їхнього захисту та збереження інфраструктури. У роботі виявлено потребу автоматизованого підходу до тестування таких систем для забезпечення їхньої ефективної роботи та надання достовірної інформації. Встановлено переваги використання цифрових систем оповіщення (вебсервіси, мобільні додатки, SMS) над традиційними методами (сирени, телебачення, тощо) завдяки розробленій математичній моделі, яка нормалізує критерії швидкості доставки повідомлення, охоплення та сприйняття повідомлень. З'ясовано, що традиційні методи тестування програмного забезпечення недостатньо ефективні для складних систем оповіщення, що обґрунтовує потребу розроблення модульної архітектури автоматизованого фреймворку (інфраструктури програмних рішень, що полегшує розроблення складних систем), розробленого мовою програмування Python із використанням бібліотек pandas і matplotlib. Розроблено інтегровану платформу для функціонального тестування систем оповіщення, яка містить серверну, клієнтську та сервісну частини з інтеграцією в CI/CD-процеси, що забезпечує швидке виявлення збоїв у програмному забезпеченні й адаптивність до критичних умов. Оцінено вплив запропонованого підходу на продуктивність системи "Мапа тривог України", де середню тривалість відповіді покращено на 15-20 %, а стійкість до пікових навантажень і DDoS-атак підтверджена результатами її тестування. Охарактеризовано закономірності взаємодії компонентів фреймворку, де клієнтська частина ініціює тести через API, серверна частина обробляє дані, а сервісна – забезпечує безперервність, що усуває затримки під час оброблення. Удосконалено метод тестування програмного забезпечення, орієнтованого на системи раннього оповіщення населення та моніторингу надзвичайних ситуацій, а також здійснено практичне тестування розробленої системи на базі сервісу "Мапа тривог України". З'ясовано перспективи удосконалення досліджуваної платформи, зокрема – через додавання модулів симуляції для гібридних систем і впровадження штучного інтелекту для прогнозування збоїв, що може підвищити надійність критичної інфраструктури.

Ключові слова: інформаційна система оповіщення; тестування програмного забезпечення; програмна система; цифрові канали оповіщення; архітектури тестового фреймворку.

Вступ / Introduction

У сучасному світі збільшується кількість надзвичайних ситуацій (далі – НС) природного та техногенного характеру, що створює серйозні загрози для життя та здоров'я населення, а також для критичної інфраструктури [14]. Системи оповіщення населення відіграють ключову роль у мінімізації негативних наслідків таких ситуацій, даючи змогу оперативно інформувати громадян про можливі загрози та надавати потрібні рекомендації з подальших дій.

Проте для сучасних систем оповіщення характерні недостатня надійність, обмежена адаптивність до змінних умов та складність масштабування. Додатковою проблемою є відсутність ефективних методів їхнього комплексного тестування, що дало б змогу оцінювати працездатність систем в умовах реального навантаження та виявляти критичні точки збоїв. Наявні традиційні методики тестування здебільшого орієнтовані на ручний підхід або не враховують специфіки багатоканального цифрового інформування.

З огляду на це постає нагальна потреба розроблення автоматизованого функціонального тестування систем раннього оповіщення та моніторингу надзвичайних ситуацій, здатного забезпечити перевірку їхньої продуктивності, точності та стійкості до різних типів наванта-

Інформація про авторів:
Шопський Орест Михайлович, ад'юнкт, кафедра інформаційних технологій та систем електронних комунікацій.
Email: orestshopsky@gmail.com; https://orcid.org/0009-0000-1761-6643
Лінник Максим Борисович, студент, кафедра інформаційних технологій та систем електронних комунікацій.
Email: m.linnyk@ldubgd.edu.ua
Лінник Марта Борисівна, тестувальник. **Email:** martalinnyk@gmail.com
Назар Юлія Сергіївна, д-р філософії, ст. викладач, кафедра інформаційних технологій та систем електронних комунікацій.
Email: kordunovayulia@gmail.com; https://orcid.org/0000-0003-0151-8285
Цитування за ДСТУ: Шопський О. М., Лінник М. Б., Лінник М. Б., Назар Ю. С. Розроблення та впровадження платформи для автоматизованого тестування систем оповіщення та моніторингу надзвичайних ситуацій. Науковий вісник НЛТУ України. 2025, т. 35, № 3. С. 193–199.
Citation APA: Shopskiy, O. M., Linnyk, M. B., Linnyk, M. B., & Nazar, Yu. S. (2025). Development and implementation of an automated testing platform for emergency notification and monitoring systems. *Scientific Bulletin of UNFU*, 35(3), 193–199. https://doi.org/10.36930/40350321

жень у динамічних умовах експлуатації.

Об'єкт дослідження – тестування систем раннього оповіщення населення та моніторингу надзвичайних ситуацій.

Предмет дослідження – методи та засоби проєктування інтегрованої платформи для функціонального тестування систем раннього оповіщення, реалізація якої дасть змогу моделювати реальні умови експлуатації таких систем, оцінювати їхню продуктивність та стійкість до навантажень.

Мета роботи – спроєктувати архітектуру та розробити інтегровану платформу для автоматизованого функціонального тестування систем раннього оповіщення населення та моніторингу надзвичайних ситуацій, спрямованої на підвищення надійності, швидкості реагування та адаптивності цих сервісів до критичних умов їхнього застосування.

Для досягнення зазначеної мети визначено такі основні *завдання дослідження*:

- проаналізувати наявні системи раннього оповіщення та моніторингу НС, що дасть змогу виявити їхні переваги та недоліки, а також обґрунтувати потребу автоматизованого підходу до тестування цифрових систем оповіщення для підвищення ефективності процесу їх розроблення;
- спроєктувати архітектуру інтегрованої платформи автоматизованого функціонального тестування систем раннього оповіщення та моніторингу надзвичайних ситуацій із модульним розподілом компонентів, визначити вимоги до реалізації інтегрованої платформи та вибрати відповідні інструменти для її розроблення;
- розробити автоматизовані тестові сценарії з використанням мови Python, що дасть змогу моделювати реальні умови експлуатації систем оповіщення, оцінювати їхню продуктивність та стійкість до навантажень;
- протестувати розроблену платформу на реальних кейсах та оцінити її ефективність, що дасть можливість підтвердити практичну цінність розробленої платформи, виявити переваги та недоліки систем оповіщення, а також сформулювати рекомендації з їхнього удосконалення.

Аналіз останніх досліджень та публікацій. Проблематику надійності роботи системи оповіщення та моніторингу НС порушено у багатьох працях як українських, так і зарубіжних учених. Зокрема, у роботі [13] розглянуто питання вивчення та аналізу інформації про природні і техногенні загрози в еколого-геофізичних ситуаціях з використанням методів математичного комп'ютерного моделювання.

У роботі [4] акцентовано увагу на важливості ефективного використання мобільних системи оповіщення про надзвичайні ситуації. Вони вказують на те, що сучасні технології, такі як смартфони, можуть слугувати важливим інструментом для комунікації під час надзвичайної ситуації.

У роботі [9] досліджено вплив автоматизованої системи сповіщення на активацію медичних екстрених команд (МЕТ). Результати показали, що впровадження системи скоротило час активації медичних екстрених команд та знизило смертність серед госпіталізованих пацієнтів, що свідчить про важливість автоматизації в покращенні результатів. У дослідженні [12] наголошено на важливості систем публічних сповіщень у забезпеченні безпеки громадян під час надзвичайних ситуацій, виявлено ключові особливості, які можна використати для вдосконалення практик і політик у цій сфері.

У роботах [1, 3, 11] описано практики реалізації систем оповіщення та моніторингу надзвичайних ситуацій у різних країнах світу. Зокрема, у роботі [1] встановлено, що відсутність диференційованих підходів до інформування є критичною проблемою для системи оповіщення у вищих навчальних закладах США, зокрема в умовах надзвичайних ситуацій, таких як урагани. Аналіз комунікаційної системи Університету Флориди після урагану Доріан дав змогу виявити основні бар'єри: неадаптованість каналів зв'язку, надлишкове інформування, мовні труднощі та недовіра до джерел інформації.

Аналіз результатів опитування в роботі [11] серед фахівців із надзвичайних ситуацій у США показав значну варіативність у використанні систем зовнішнього звукового оповіщення, зокрема щодо критеріїв активації сирен (торнадо, сильний вітер, град) та моменту їх спрацювання. Здебільшого рішення про запуск сирен приймаються на місцевому рівні, що призводить до відсутності єдиного стандарту реагування. Така нестабільність та суб'єктивність в управлінні аналоговими системами створює ризики у критичних ситуаціях, що вказує на необхідність переходу до більш гнучких і керованих цифрових систем раннього оповіщення, функціональне тестування яких і є предметом цього дослідження.

У дослідженні [3], в якому наведено досвід Пуерто-Рико під час урагану "Марія", розглядають використання соціальних мереж у системах раннього оповіщення. Автори звертають увагу на конфлікт між традиційними централізованими каналами комунікації та децентралізованими інформаційними потоками у соцмережах. Незважаючи на зростання популярності цифрових платформ, їхнє широке впровадження стримується через проблеми довіри та достовірності інформації. Це свідчить про нагальну потребу створення сучасних цифрових систем раннього оповіщення, які поєднували б гнучкість нових технологій із перевіреними механізмами комунікації, з акцентом на їх тестування та адаптацію до реальних умов.

У дослідженні [8] наголошено на важливості доповнення систем раннього попередження прогнозами впливу, що дає змогу приймати обґрунтованіші рішення під час локалізації та ліквідації надзвичайних ситуацій. Автори обговорюють переваги впливових попереджень порівняно з традиційним прогнозуванням небезпеки, вказуючи на виклики та можливості для удосконалення систем раннього попередження, особливо в контексті європейських країн.

Отже, ефективність систем раннього оповіщення неможливо гарантувати без регулярного тестування та оцінювання їхньої продуктивності, зокрема, перевірки точності прогнозів, швидкості реагування та здатності до адаптації в умовах постійно змінюваних ризиків. Без належної верифікації, системи можуть не відповідати реальним потребам громади, що може призвести до серйозних наслідків під час надзвичайних ситуацій. Тому подальші дослідження потрібно зосередити не тільки на розробленні нових технологій для раннього попередження, але й на розробленні стандартів для тестування та верифікації наявних систем, що забезпечить їхню надійність та ефективність у критичних ситуаціях.

Матеріали та методи дослідження. Для проєктування інтегрованої платформи для автоматизованого функціонального тестування систем раннього оповіщення населення та моніторингу надзвичайних ситу-

ацій використано системний підхід для структурування завдань, математичне моделювання у вигляді багатокритеріального оцінювання ефективності видів оповіщення та емпіричний метод повного перебору (exhaustive search) для вибору оптимального варіанта. Як критерії оцінювання ефективності систем оповіщення обрано швидкість доставки повідомлення, охоплення аудиторії та рівень його сприйняття, що узгоджується з вимогами до оперативності, доступності й зрозумілості комунікації в кризових умовах. Вагові коефіцієнти встановлювали експертно з урахуванням пріоритетності зазначених критеріїв. Реалізацію автоматизованих тестів здійснено мовою Python із використанням бібліотек pandas – для оброблення даних, matplotlib – для побудови графіків і idxmax() – для визначення максимального значення ефективності. Нормалізацію критеріїв проведено згідно з визначеними математичними залежностями, що дало змогу забезпечити коректне порівняння показників.

Результати дослідження та їх обговорення / Research results and their discussion

Аналіз ефективності наявних видів оповіщення. Для обґрунтування важливості процесу тестування саме цифрових систем раннього оповіщення населення розроблено математичну модель, що дає змогу визначити найефективніший метод донесення інформації до населення у разі надзвичайних ситуацій. Аналізу підлягали такі види оповіщення, як мобільні пристрої (SMS, push-сповіщення, мобільні додатки), телебачення, радіо, соціальні мережі, альтернативні засоби оповіщення (гучномовці, стаціонарні сигнали).

Як критерії для визначення ефективності систем оповіщення населення обрано:

- швидкість доставки i-го повідомлення (у секундах) – T_i;
- частка населення, яка отримує повідомлення через i-те оповіщення (охоплення) – C_i;
- рівень сприйняття i-го повідомлення (ймовірність, що людина правильно розуміє повідомлення) – R_i;
- кількість видів оповіщень – m.

На підставі цих критеріїв було сформульовано функцію ефективності систем раннього оповіщення:

$$E_i = w_T \cdot \frac{1}{T_i} + w_C \cdot C_i + w_R \cdot R_i,\ i = \overline{1,m}, \quad (1)$$

де: w_T, w_C, w_R – вагові коефіцієнти (визначають важливість кожного критерію) та в сумі становлять одиницю; E_i – ефективність i-го виду оповіщення.

Оскільки абсолютні значення критеріїв мають різні діапазони, для коректного розрахунку їх потрібно нормалізувати. Для критерію швидкості доставки повідомлення нормалізацію здійснюють за формулою:

$$T_i^{'} = \frac{\min(T_i)}{T_i},\ i = \overline{1,m}, \quad (2)$$

де $\min(T)$ – найменший час доставки повідомлення серед усіх видів оповіщення. Отже, чим менше значення T_i, тим вищий нормалізований показник ефективності.

Критерії охоплення населення (C_i) інформацією та рівня сприйняття повідомлення (R_i) вже подано у вигляді відносних значень у межах [0;1] (наприклад, 85 % = 0,85), тому вони не потребують додаткової нормалізації.

Щоб обчислити ефективність i-го виду оповіщення НС, використаємо таку формулу:

$$E_i = w_T \cdot T^{'} + w_C \cdot C_i + w_R \cdot R_i,\ i = \overline{1,m}. \quad (3)$$

Отже, найефективніший вид оповіщення визначимо за такою формулою:

$$i^* = \arg\max(E_i). \quad (4)$$

Щоб обчислити значення цього виразу, а також подати результат у вигляді діаграми використано бібліотеки Python – pandas та matplotlib та отримано такий результат (рис. 1):

```
(  Канал              Швидкість доставки (T), с  Охоплення (C), %  \
0  Мобільні пристрої                         10                80
1        Телебачення                         30                95
2              Радіо                         60                70
3   Соціальні мережі                         15                85
4         Гучномовці                         90                50

Рівень сприйняття (R), %  T_norm  C_norm  R_norm  Ефективність (E)
0                        90  1.000000    0.80    0.90          0.900000
1                        80  0.333333    0.95    0.80          0.673333
2                        75  0.166667    0.70    0.75          0.496667
3                        85  0.666667    0.85    0.85          0.776667
4                        60  0.111111    0.50    0.60          0.364444  ,

Канал                          Мобільні пристрої
Швидкість доставки (T), с                 10
Охоплення (C), %                          80
Рівень сприйняття (R), %                  90
T_norm                                   1.0
C_norm                                   0.8
R_norm                                   0.9
Ефективність (E)                         0.9
Name: 0, dtype: object)
```

Рис. 1. Результат розв'язку рівняння / Result of the equation solving

Параметри для оцінювання ефективності каналів оповіщення (швидкість доставки повідомлення, охоплення населення, рівень сприйняття) сформовано на основі узагальнених даних відкритих джерел Державної служби України з надзвичайних ситуацій (ДСНС), практичного аналізу цифрових платформ і моделювання типових сценаріїв інформування населення в умовах надзвичайних ситуацій.

Діаграма на рис. 2 наочно підтверджує, що мобільні додатки та вебверсії систем моніторингу НС є найефективнішими засобами інформування населення у кризових ситуаціях.

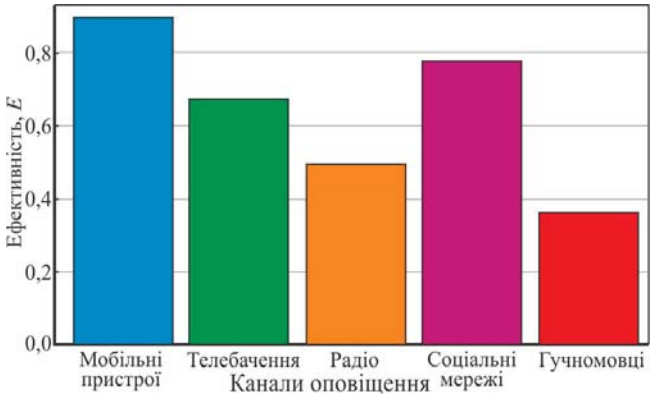


Рис. 2. Залежність ефективності каналів оповіщення / Impact of notification channel on efficiency

Отже, тестування саме цифрових систем оповіщення є пріоритетним завданням, оскільки їхня коректна робота безпосередньо впливає на швидкість та точність інформування громадян, а розроблення платформи для автоматизованого тестування таких систем зробить їх надійним джерелом оповіщення та моніторингу надзвичайних ситуацій.

Проєктування архітектури тестового фреймворку. Структурне проєктування тестового фреймворку є невід'ємною частиною процесу розроблення сучасних систем автоматизованого тестування. На рис. 3 зображено концептуальну схему інтегрованої системи тестування, яка дає змогу чітко розподілити обов'язки між різними її компонентами та забезпечити модульність, перевірку змін і легкість їх упровадження.

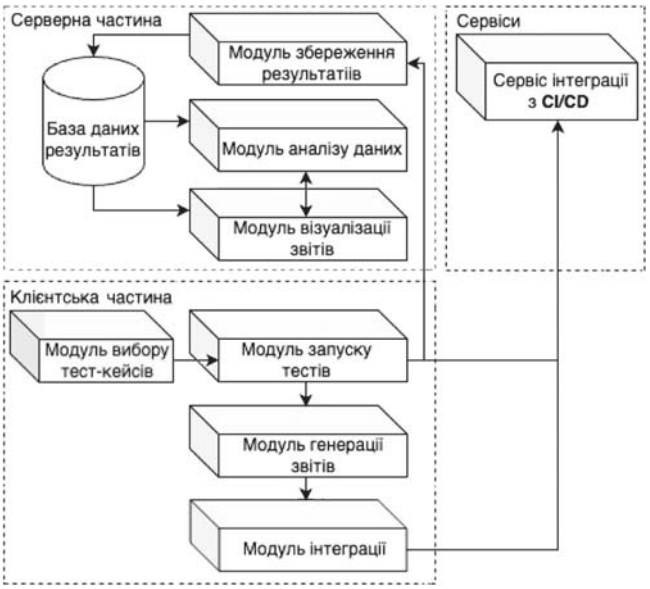


Рис. 3. Структурна схема тестового фреймворку / Structural diagram of the test framework

Пропоновану тестову платформу поділяють на серверну, клієнтську та сервісну частини.

Серверна частина відіграє центральну роль у збереженні, аналізі та візуалізації тестових даних. Основні її модулі:

- модуль збереження результатів – забезпечує надійне зберігання результатів тестів у базі даних, їхню організацію та захист від втрати;
- модуль аналізу даних – формує метрики успішності, визначає слабкі позиції системи та може містити алгоритми пошуку аномалій або генерацію рекомендацій;
- модуль візуалізації звітів – перетворює дані на інтерактивні діаграми, гістограми та звіти для зручного аналізу;
- база даних – забезпечує швидкий доступ до великих обсягів даних, підтримує транзакції та реплікацію для підвищення надійності.

Клієнтська частина відповідає за взаємодію користувача із фреймворком, починаючи від вибору сценаріїв тестування до формування фінальних звітів. Вона містить такі модулі:

- модуль вибору сценаріїв – надає інтуїтивний інтерфейс для налаштування параметрів тестів;
- модуль запуску тестів – відповідає за виконання обраних тестових сценаріїв і збирання метрик успішності;
- модуль генерації звітів – формує результати тестування у різних форматах (PDF, Excel тощо);
- модуль інтеграції – дає змогу взаємодіяти зі зовнішніми системами, такими як Jira, Slack, CI/CD, забезпечуючи безперервність робочих процесів.

Сервісна частина розширює функціональність фреймворку та забезпечує його інтеграцію з іншими системами. Вона містить сервіс інтеграції із CI/CD, що дає змогу автоматизувати процес тестування в межах конвеєра розроблення програмного забезпечення, автоматично запускаючи тести в разі змін у коді, а також підтримку Jenkins, GitHub Actions тощо, що забезпечує безперервний зворотний зв'язок для розробників та ефективність процесу тестування.

Взаємодія між компонентами фреймворку відбувається у такий спосіб: клієнтська частина через модуль запуску тестів надсилає запити до серверної частини для виконання тестових сценаріїв, використовуючи API-інтерфейс. Серверна частина обробляє ці запити, зберігає результати в базі даних і повертає оброблені метрики та візуалізації назад до клієнтської частини для відображення користувачу. Сервісна частина діє як посередник, забезпечуючи інтеграцію зі зовнішніми системами: наприклад, у разі виявлення змін у коді через CI/CD (Jenkins) автоматично ініціюється запуск тестів, а результати передаються до модуля звітності. Такий підхід гарантує безперебійний обмін даними між модулями та швидке виявлення проблем у системі оповіщення.

На підставі запропонованої схеми розроблено тестовий фреймворк мовою програмування Python, який забезпечує автоматизоване тестування ПЗ із чіткою структурою компонентів. Завдяки модульному підходу тестовий фреймворк дає змогу ефективно керувати тестовими сценаріями, зберігати та аналізувати отримані результати, а також інтегруватися із CI/CD-сервісами для безперервного його тестування. Це забезпечує гнучкість, масштабованість та зручність використання тестового фреймворку у процесах розроблення програмного забезпечення.

Практичне впровадження. У процесі розроблення тестового фреймворку реалізовано автоматизований тест для оцінювання продуктивності системи моніторингу "Мапа тривог України" [15]. Враховуючи критичну важливість стабільної роботи цієї системи, тестування ПЗ орієнтоване на моделювання умов пікового навантаження, що виникає під час масового доступу користувачів, а також стійкості до DDoS-атак.

Для тестування сервісу визначено ключові параметри навантажувального тесту (рис. 4). У цьому коді параметри BASE_URL – адреса тестованого ресурсу, CONCURRENT_USERS – одночасна кількість запитів (2000 користувачів), TOTAL_REQUESTS – загальна кількість виконаних запитів (10000), TIMEOUT – граничний час виконання запиту (5 с).

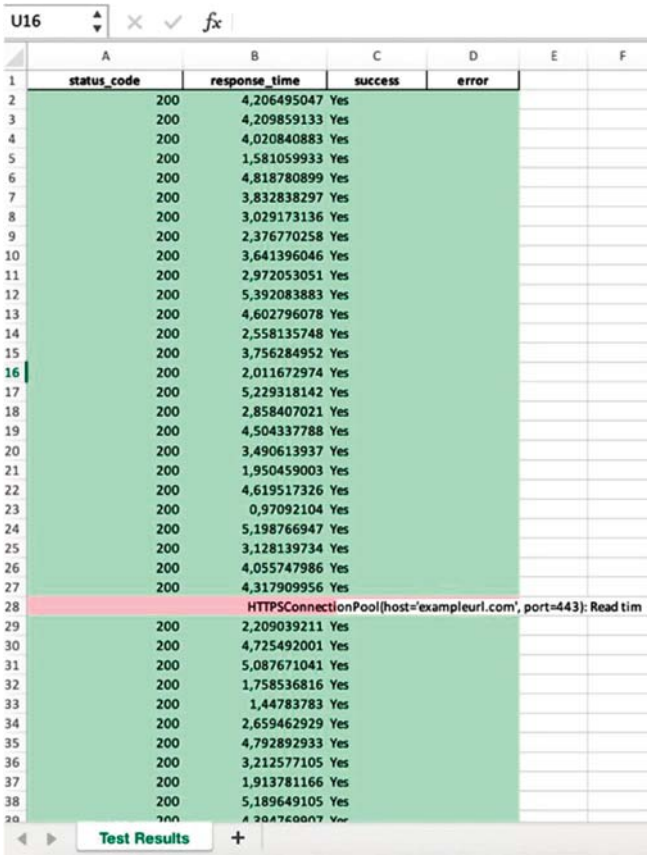
```
BASE_URL = "https://alerts.in.ua/"
NUM_REQUESTS = 10000
CONCURRENT_USERS = 2000
TIMEOUT = 5
```

Рис. 4. Початкові дані для тесту / Initial data for the test

Тестування ресурсу "Мапа тривог України" дало змогу оцінити стійкість системи до навантажень та виявити критичні точки продуктивності. Основні метрики продуктивності містили:

- середня тривалість відповіді – покращена на 15-20 % завдяки впровадженню автоматизованих тестів та модульної структури фреймворку, що забезпечило ефективніший розподіл запитів;
- частка невдалих запитів – залишалася в межах прийнятних значень навіть за умов пікового навантаження;
- стійкість до атак – система зберігала працездатність, незважаючи на імітовані DDoS-атаки.

Для подальшого аналізу спроєктованої системи реалізовано збереження даних у трьох форматах: JSON, CSV та Excel. На рис. 5 наведено результати тестування системи у середовищі Excel.



	A	B	C	D
1	status_code	response_time	success	error
2	200	4,206495047	Yes	
3	200	4,209859133	Yes	
4	200	4,020840883	Yes	
5	200	1,581059933	Yes	
6	200	4,818780899	Yes	
7	200	3,832838297	Yes	
8	200	3,029173136	Yes	
9	200	2,376770258	Yes	
10	200	3,641396046	Yes	
11	200	2,972053051	Yes	
12	200	5,392083883	Yes	
13	200	4,602796078	Yes	
14	200	2,558135748	Yes	
15	200	3,756284952	Yes	
16	200	2,011672974	Yes	
17	200	5,229318142	Yes	
18	200	2,858407021	Yes	
19	200	4,504337788	Yes	
20	200	3,490613937	Yes	
21	200	1,950459003	Yes	
22	200	4,619517326	Yes	
23	200	0,97092104	Yes	
24	200	5,198766947	Yes	
25	200	3,128139734	Yes	
26	200	4,055747986	Yes	
27	200	4,317909956	Yes	
28	HTTPSConnectionPool(host='exampleurl.com', port=443): Read tim			
29	200	2,209039211	Yes	
30	200	4,725492001	Yes	
31	200	5,087671041	Yes	
32	200	1,758536816	Yes	
33	200	1,44783783	Yes	
34	200	2,659462929	Yes	
35	200	4,792892933	Yes	
36	200	3,212577105	Yes	
37	200	1,913781166	Yes	
38	200	5,189649105	Yes	

Рис. 5. Результат тестування вебсервісу "Мапа тривог України", наведені у середовищі Excel / Result of the web-service test presented in.exel format

Для візуалізації результатів згенеровано графіки у форматі зображень. На рис. 6 наведено графік тривалості відповіді на кожен запит із виділенням невдалих запитів (з нульовим часом відповіді). На рис. 7 зображено графік співвідношення успішних і невдалих запитів, що дає змогу оцінити загальну стабільність роботи системи.

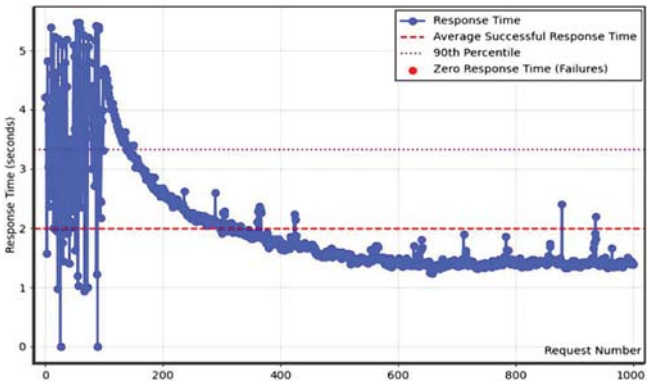


Рис. 6. Тривалість відповіді системи / System response time graph

Отримані результати дослідження підтвердили ефективність запропонованого підходу до автоматизованого тестування систем раннього оповіщення та дали можливість сформувати рекомендації з удосконалення продуктивності системи.

Обговорення результатів дослідження. Унаслідок виконання дослідження проаналізовано велику кількість праць стосовно тестування програмних систем. Отримані результати підтверджують, що розроблені попередниками інтегровані платформи для тестування ПЗ здатні ефективно вирішувати ключові проблеми, виявлені у процесі аналізу літератури та практичного впровадження.

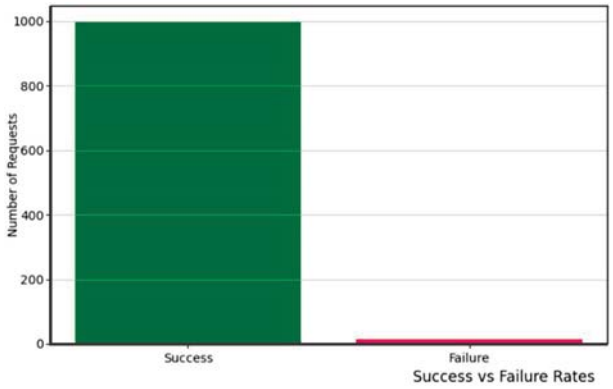


Рис. 7. Співвідношення успішних і невдалих запитів / Graph of the ratio of successful and unsuccessful requests

Зокрема, розроблений у роботі [2] підхід перевершує традиційні методи тестування ПЗ за рахунок використання автоматизованих сценаріїв мовою Python, які забезпечують швидшу ідентифікацію слабких позицій системи порівняно з ручним тестуванням. Це особливо важливо за умов пікового навантаження, де традиційні системи часто не справляються.

Розроблена платформа вирізняється широкою сферою застосування та адаптивністю до критичних умов порівняно з дослідженням [5], яке використовує Selenium Grid для розподіленого тестування в Agile-середовищах. У цій роботі автори досягли пришвидшення тестування вебдодатків способом паралельного виконання тестів, що скоротило час виведення продукту на ринок і покращило покриття тестів, однак їхній підхід обмежений контекстом Agile, який не завжди можна інтегрувати під час розроблення безпеко-орієнтованих сервісів [7].

Дослідження [10] стосується систем оповіщення, пропонуючи SMS-орієнтовану систему геолокації для екстрених медичних служб, з акцентом на швидкість доставки SMS. Автори досягли надійності роботи системи в межах 62,7-70,0 %, однак їхній підхід обмежується тільки SMS-повідомленнями й не враховує інших каналів оповіщення. На відміну від цього, запропонована нами платформа охоплює ширший спектр каналів (мобільні додатки, SMS, вебсервіси) завдяки математичному моделюванню визначення ефективності системи оповіщення населення, що забезпечує універсальність і практичну цінність для різних сценаріїв надзвичайних ситуацій.

У дослідженні [6] розглянуто використання AI-чатботів для автоматизації моніторингу та звітності у тестуванні програмного забезпечення, зосереджуючись на інтеграції із CI/CD. Автори вказують на підвищення ефективності та масштабованості завдяки AI, зокрема – в автоматизації аналізу збоїв і звітності, однак їхній підхід не торкається критичних систем, таких як системи оповіщення. У цьому дослідженні, окрім інтеграції з CI/CD (Jenkins), запропонована платформа забезпечує оцінювання продуктивності й стійкості систем оповіщення в реальних умовах їхнього функціонування, а також пропонує їхню модульну архітектуру, що робить її більш універсальною для застосування у разі надзвичайних ситуацій.

Отже, результати обговорення підтверджують доцільність і своєчасність проведеного дослідження, оскільки серед проаналізованих наукових джерел немає

комплексних підходів до автоматизованого функціонального тестування саме систем раннього оповіщення та моніторингу надзвичайних ситуацій. Запропонована в роботі платформа вирізняється тим, що поєднує математичне моделювання ефективності різних каналів оповіщення, автоматизоване тестування за допомогою сучасних інструментів і підтримку інтеграції із CI/CD-процесами. Такий підхід забезпечує не тільки високу адаптивність до різних умов, а й дає змогу оцінити реальну стійкість і продуктивність систем оповіщення, що є критично важливим для їхнього застосування в умовах надзвичайних ситуацій.

Отже, внаслідок виконаної роботи можна сформулювати такі наукову новизну та практичну значущість результатів дослідження.

Наукова новизна отриманих результатів дослідження – удосконалено методику тестування програмного забезпечення, орієнтованого на системи раннього оповіщення населення та моніторингу надзвичайних ситуацій, способом реалізації модульної архітектури тестового середовища та інтеграції із CI/CD-процесами, що забезпечує покращений аналіз продуктивності, її точності та стійкості за умов змінного навантаження, а також підвищує їхню гнучкість і адаптивність до критичних сценаріїв використання.

Практична значущість результатів дослідження – результати дослідження використано під час розроблення фреймворку для автоматизованого функціонального тестування систем оповіщення та моніторингу надзвичайних ситуацій, який можна застосувати для оцінювання продуктивності, точності та стійкості таких систем в умовах реального навантаження.

Надалі це дослідження може бути розширене способом тестування не тільки цифрових, але й гібридних систем оповіщення (наприклад, із використанням гучномовців чи сирен), якщо додати відповідні модулі симуляції. Це відкриває перспективи для майбутніх досліджень, зокрема – щодо інтеграції штучного інтелекту [16] для прогнозування збоїв і автоматичного коригування тестових сценаріїв у реальному часі.

Висновки / Conclusion

Спроєктовано архітектуру та розроблено інтегровану платформу для автоматизованого функціонального тестування систем раннього оповіщення населення та моніторингу надзвичайних ситуацій, спрямовану на підвищення надійності, швидкості реагування та адаптивності цих сервісів до критичних умов їхнього застосування. За результатами проведеного дослідження можна зробити такі основні висновки:

1. Проаналізовано сучасні методи та підходи до раннього оповіщення і моніторингу надзвичайних ситуацій, способом математичного моделювання, на основі чого прийнято рішення про удосконалення методу тестування цифрових систем оповіщення (вебсервіси, SMS-повідомлення, мобільні додатки) для підвищення ефективності інформування та реагування на надзвичайні ситуації.
2. Спроєктовано архітектуру та розроблено інтегровану платформу для автоматизованого функціонального тестування систем раннього оповіщення з використанням модульного підходу, що передбачає незалежну роботу клієнтської, серверної та сервісної частин фреймворку.
3. Реалізовано автоматизовані сценарії функціонального тестування систем раннього оповіщення з використанням мови програмування Python та відповідних бібліотек (pandas, matplotlib), що дало змогу виконати оброблення даних, візуалізацію отриманих результатів і порівняльне оцінювання продуктивності, точності і стійкості систем в умовах змінного навантаження.
4. Проведено експериментальне тестування вебсервісу "Мапа тривог України", внаслідок чого встановлено здатність системи до витримування пікових навантажень і стійкість до DDoS-атак. Обґрунтовано практичну доцільність використання розробленого фреймворку у процесах перевірки систем оповіщення та моніторингу НС, а також визначено перспективи його подальшого розвитку, зокрема через впровадження інтелектуальних модулів прогнозування збоїв та симуляцію гібридних систем.

References

1. Abukhalaf, A. H. I., & Von Meding, J. (2020). Communication challenges in campus emergency planning: The case of hurricane Dorian in Florida. *Natural Hazards*, *104*(2), 1535–1565. https://doi.org/10.1007/s11069-020-04231-1
2. Adeyemo, H. (2021). Comparative analysis of modelling and software testing tools: A survey. *Saudi Journal of Engineering and Technology*, 6(10), 371–377. URL: https://saudijournals.com/media/articles/SJEAT_610_371-377_L8hRMPy.pdf
3. Bui, L. (2019). Social media, rumors, and hurricane warning systems in Puerto Rico. *Proceedings of the* 52nd *Hawaii International Conference on System Sciences*, 1–10. https://doi.org/10.24251/HICSS.2019.321
4. Fischer-Pressler, D., Bonaretti, D., & Fischbach, K. (2020). Effective use of mobile-enabled emergency warning systems. *Research Papers*, 130. URL: https://aisel.aisnet.org/ecis2020_rp/130
5. Kodanda, R. R. M. (2024). Enhancing Test Automation Coverage and Efficiency with Selenium Grid: A Study on Distributed Testing in Agile Environments, *International Journal of Advanced Research in Engineering and Technology* (*IJARET*), *15*(3), 119–127. https://doi.org/10.17605/OSF.IO/NY4ZU
6. Koppanati, P. K. (2024). Leveraging AI-powered chatbots for real-time test monitoring and reporting. *IEEE Transactions on Systems, 11*(4), 117–123. URL: https://www.researchgate.net/publication/387794718_Leveraging_AI-Powered_Chatbots_for_Real-Time_Test_Monitoring_and_Reporting
7. Kordunova, Y., Prydatko, O., Smotr, O., & Golovatyi, R. (2023). Expert Decision Support System Modeling in Lifecycle Management of Specialized Software. *Lecture Notes in Data Engineering, Computational Intelligence, and Decision Making. ISDMCI 2022. Lecture Notes on Data Engineering and Communications Technologies,* 149, 367–383. https://doi.org/10.1007/978-3-031-16203-9_22
8. Merz, B., Kuhlicke, C., Kunz, M., Pittore, M., Babeyko, A., Bresch, D. N., et al. (2020). Impact forecasting to support emergency management of natural hazards. *Reviews of Geophysics*, 58, article ID e2020RG000704. https://doi.org/10.1029/2020RG000704
9. Na, S. J., Ko, R. E., Ko, M. G., et al. (2021). Automated alert and activation of medical emergency team using early warning score. *Journal of Intensive Care*, 9, 73 p. https://doi.org/10.1186/s40560-021-00588-y
10. Osebor, I., Misra, S., Omoregbe, N., Adewumi, A., & Fernandez-Sanz, L. (2017). Experimental simulation-based performance evaluation of an SMS-based emergency geolocation notification system. *Journal of Healthcare Engineering, 1*, article ID 7695045. https://doi.org/10.1155/2017/7695045
11. Rotzge, J. A., & Donner, W. R. (2015). General policy for activating outdoor warning siren systems for severe weather: Survey of emergency managers. *Natural Hazards Review*, *16*(2), 123–132. https://doi.org/10.1061/(ASCE)NH.1527-6996.0000155
12. Sadiq, A. A., Dougherty, R. B., Tyler, J., et al. (2023). Public alert and warning system literature review in the USA: Identifying research gaps and lessons for practice. *Natural Hazards*, 117, 1711–1744. https://doi.org/10.1007/s11069-023-05926-x

13. Starodub, Y. P., Kupliovsky, B. E., Brych, T. B., Havrys, A. P., & Yemelyanenko, S. O. (2022). Computer modelling of natural and man-made threats and ecological and geophysical situations. Lviv: Lviv State University of Life Safety URL: https://sci.ldubgd.edu.ua/jspui/handle/123456789/12398
14. State Emergency Service of Ukraine. (2024). Information and analytical reference on emergencies that occurred in Ukraine in 2024. URL: https://dsns.gov.ua/upload/2/2/9/2/0/9/7/12-2024.pdf
15. Ukraines alarm map. (2025). URL: https://alerts.in.ua/
16. Vorochek, O., & Solovei, I. (2024). Research on artificial intelligence tools for automating the software testing process. *Bulletin of National Technical University "KhPI". Series: System Analysis, Control and Information Technologies*, *1*(11), 58–64. https://doi.org/10.20998/2079-0023.2024.01.09

O. M. Shopskiy[1], M. B. Linnyk[1], M. B. Linnyk[2], Yu. S. Nazar[1]
[1] Lviv State University of Life Safety, Lviv, Ukraine
[2] SoftServe, Lviv, Ukraine

DEVELOPMENT AND IMPLEMENTATION OF AN AUTOMATED TESTING PLATFORM FOR EMERGENCY NOTIFICATION AND MONITORING SYSTEMS

The increasing frequency of natural and technogenic emergencies in recent years underscores the critical importance of reliable early warning systems for safeguarding population and infrastructure. The study identified the necessity of automated approaches to testing such systems in order to ensure their efficiency and integrity. The study established the superiority of digital notification platforms, such as mobile applications, over traditional methods through a mathematical model that normalizes delivery speed, coverage, and message comprehension criteria. The study determined that conventional testing methods lack efficiency for complex notification systems, justifying the need for a modular automated framework developed using Python with Pandas and Matplotlib libraries. In the course of research, we designed an integrated functional testing platform featuring server-side, client-side, and service components, integrated with CI/CD processes, enabling rapid fault detection and adaptability to real-world conditions. The study evaluated the proposed approach impact on the System of the Map of Air Alerts of Ukraine, improving average response time by 15–20 % and confirming resilience to peak loads and DDoS attacks through testing results. The study characterized the interaction patterns of the framework components as follows: the client-side initiates tests via API, the server-side processes data, and the service layer ensures seamless integration with Jenkins, eliminating processing delays noted in prior studies. The study demonstrated scientific novelty by combining automated testing with mathematical modeling and criteria normalization, alongside practical significance in creating a versatile tool for optimizing notification systems. Furthermore, the research outlined prospects for enhancing the platform through simulation modules for hybrid systems and the integration of artificial intelligence to predict failures, potentially enhancing critical infrastructure reliability.

Keywords: information alert system; software testing; software system; digital notification channels; test framework architecture.