

*Одарущенко О. Б. к.т.н.,
Шпак С. О. магістрант,
Полтавський національний технічний університет
імені Юрія Кондратюка*

РОЗРОБКА ВЕБ-СЕРВІСУ ДЛЯ ПОЧАТКОВОГО АНАЛІЗУ НАДІЙНОСТІ ПРОГРАМНИХ ЗАСОБІВ ЗА ДОПОМОГОЮ МЕТОДУ ВИЗНАЧЕННЯ МОДЕЛЕЙ НАДІЙНОСТІ ПРОГРАМНИХ ЗАСОБІВ

Вступ

Надійність програмного забезпечення можна вважати однією з найважливіших його характеристик, що, в деяких випадках, дозволяє знехтувати швидкістю його виконання та візуальною красою. Але досі, незважаючи на швидкий розвиток і вдосконалення технологій розробки, абсолютна надійність програмного забезпечення є недосяжною.

Щоб досягти більшої надійності програмного забезпечення запропоновано досить немало підходів, серед яких: організаційні методи розробки, вбудовані механізми тестування, що передбачають попереднє написання тестів програмістом, стандартизація написання коду програми, використання послуг тестерів та інше. Все це потребує додаткового часу, грошей та персоналу.

На даний час, теорія надійності знаходиться на досить високому рівні, що дозволяє розглядати детально вивчені теоретичні і практичні питання оцінки апаратних засобів. Та цього не вистачає для реального застосування кількісної оцінки програмних засобів, хоча й доступна велика кількість ймовірнісних моделей [8]. Це пов'язано з тим, що в процесі моделювання використовуються достатньо грубі припущення, які не враховують розробку і підтримку програмних засобів. Та все ж, це дозволяє, з високою точністю, досліджувати показники надійності програмного забезпечення, що вже було написано і планується до використання.

Метод вибору моделі надійності програмного забезпечення є актуальним при тестуванні не великих програм та проектів, що мають вбудований збір статистики. Запропонований веб-сервіс дозволить розробникам не втрачати час та при необхідності провести попередній аналіз надійності програмного забезпечення у ручному режимі.

Метою роботи є дослідження наявних класифікацій та моделей надійності програмних засобів для використання їх у алгоритмі роботи веб-сервісу визначення надійності програмних засобів, а також написання самого додатку, з можливістю встановлення на виділений сервер мережі чи будь-який користувацький комп'ютер.

В даній статті наведено базові класифікації моделей надійності, а також пропонується метод вибору моделі надійності програмних засобів за, наданою користувачем, статистикою роботи програми. Даний метод дозволяє швидко визначити найбільш точну модель розрахунку надійності програми за повнотою наданих даних та точністю самої моделі. За допомогою даного методу реалізований алгоритм роботи веб-сервісу.

1.1. Класифікація моделей надійності програмних засобів

Велике різноманіття моделей надійності потребує їх класифікації.

Класифікація, запропонована **Благодатських В.О.** в роботі [1] – одна з найбільш простих. Вона розділяє методи надійності програмних засобів на аналітичні і емпіричні.

Класифікація Полоннікова Р.І. більш повна. Автор [3] пропонує розділити всі моделі на класи по наступним ознакам: структура часу, складність програмного забезпечення, розмітка помилок, структура простору вхідних даних, структура тексту програми.

В **класифікації Хетча** [7] припущено розділення моделей надійності програмного забезпечення на прогнозуючі, вимірювальні і оціночні.

Класифікація Гоела [6] розділює моделі надійності на 4 класи – без підрахунку помилок, з підрахунком помилок, з додаванням помилок, з вибором областей вхідних значень.

По класифікації Харченко В.С. [2], методи надійності програмних засобів можна розділити на наступні типи: параметричні, статистичні, структурні і ймовірнісні.

З усіх згаданих схем класифікації можна вибрати моделі надійності програмного забезпечення, які можливо використати в сервісі, що призначений для кінцевого користувача. Такими моделями є: модель перехідного процесу, Желінського – Моранди, Шика – Вольвертона, проста експоненційна, експоненційна Муси, геометрична Моранди, Вейбула, Дюена, логарифмічна Муси – Окумото, S – образного росту надійності, S – образного росту надійності з перегинами.

1.2. Метод визначення моделей надійності

За основу методу було використано дані результатів оцінки кількості програмних помилок, що були отримані за допомогою різних моделей надійності програмних засобів (Таблиці 1 – 2) [4].

Зважаючи на те, що визначення моделі надійності потрібно для роботи веб-сервісу, а отримання даних для тестування виконується за допомогою файлу формату «*.csv» (текстовий файл, з розділенням даних за допомогою коми), що можливо створити в редакторі таблиць Excel.

Сервер створений за допомогою мови Python, з фреймворком Django [5], шаблонами на мові розмітки HTML та таблицею стилів CSS. Після завантаження файлу на сервер, починається його аналіз та перевірка на наявність потрібних даних. Якщо файл виявився не того формату, або ж формат правильний але відсутні стовпці з відповідними назвами (наперед відомими і описаними в інструкції для користування) то: сервер повертає помилку і пропонує завантажити новий файл. В випадку, коли файл відповідає всім вимогам, починається підбір моделей надійності за таким алгоритмом:

1. Перевірка наявності даних для застосування моделі перехідного процесу, якщо всі розрахунки можливо виконати – відносимо її до розділу рекомендованих та пропонуємо користувачеві застосувати її. Якщо ж

розрахунки зробити не можливо – відносимо модель до розділу «не можливо використати».

2. Далі робимо перевірку для всіх інших моделей керуючись даними про відсоток відхилення по сумарній кількості помилок (Таблиці 1, 2). Тобто, пріоритет на використання моделей буде такий: модель перехідного процесу, S-образного росту надійності, Джелінського-Моранди, експоненційна, експоненційна Муси, Шика-Волвертона, геометрична Моранди, Дюена, Вейбула, логарифмічна Муси-Окумото, S-образного росту надійності з перегинами. Аналіз кожної моделі виконується аналогічно до першого пункту цього списку. Після розподілення їх за групами (рекомендовано до використання та неможливо використати) виконуємо додаткову перевірку вибірки рекомендованих до застосування на предмет високого відсотку помилок. Якщо моделей у вибірці рекомендованих більше трьох, відносимо надлишок до групи доступних, за умови, що кожна перенесена модель має похибку розрахунків більшу ніж 5%. В випадку, коли доступно 3 чи менше – залишаємо результат розподілу без змін.

3. Далі генеруємо сторінку з групами моделей і надаємо користувачеві можливість вибрати модель (або декілька моделей) для тестування надійності і підтвердити свій вибір, що запустить алгоритм розрахунку надійності. Зважаючи на те, що дані для роботи надані користувачем, перед початком основних розрахунків, перевіряємо все на наявність помилок (невірний тип даних, невірне значення, відсутність значення). Після перевірки виконуються розрахунки і генерується файл «*.csv», куди й зберігаються дані розрахунків за всіма моделями. На кінцевому етапі, сервер пропонує завантажити файл користувачеві.

Таблиця 1

Результати оцінки кількості програмних помилок, отриманих за допомогою моделей перехідного процесу, Джелінського-Моранди, Шика-Волвертона, експоненційної та експоненційної моделі Муси

	Фактична кількість помилок	Моделі				
		Перехідного процесу	Джелінського-Моранди	Шика-Волвертона	Експоненційна	Експоненційна Муси
Результат: сумарна кількість помилок	42	42,00	41,33	38,50	40,07	39,96
відхилення по сумарній кількості помилок, %		0	-1,6	-8,3	-4,6	-4,86

Таблиця 2

Результати оцінки кількості програмних помилок, отриманих за допомогою геометричної моделі Моранди, Вейбула, Дюена, логарифмічної Муси-Окумото, S-образного росту надійності, S-образного росту надійності з перегинами

	Фактична кількість помилок	Моделі					
		Геометрична Моранди	Вейбула	Дюена	Логарифмічна Муси-Окумото	S-образного росту надійності	S-образного росту надійності з перегинами
Результат: сумарна кількість помилок	42	46,01	47,03	46,04	51,85	41,99	30,30
відхилення по сумарній кількості помилок, %		+9,54	+11,98	+9,62	+23,45	-0,02	-27,85

Висновки

Результатом даної роботи є веб-додаток, написаний на фреймворку Django, що являється надбудовою мови програмування Python. Він використовує описаний в статті метод вибору моделей надійності програмних засобів та проводить розрахунок показника надійності програмних засобів за наведеними моделями. Розроблений веб-сервіс може використовуватися лише для тестування готових програм. В майбутньому планується реалізація тестування за допомогою моделей надійності, що призначені для програм, котрі знаходяться на етапах розробки та тестування.

Список використаної літератури

1. *Благодатских В.А. Стандартизация разработки программных средств /Благодатских В.А., Волюнин В.А., Посакалов К.Ф.: Уч. Пособие под ред. О.С. Разумова // Финансы и статистика. – М.: - 2003, 284 с.*
2. *Оценка качества и экспертиза программного обеспечения /Харченко В.С., Андрашов А.А., Гордеев А.А., Лобачева Е.И. // Харьков: ХАИ. 2008, - 99 с.*
3. *Полонников Р.И. Методы оценки показателей надежности программного обеспечения /Полонников Р.И., Никандров А.В. // СПб.: Политехника. 1992, - 78 с.*
4. *Яремчук С.А. Анализ моделей надежности программного обеспечения гарантоспособных информационных систем /С.А. Яремчук // Електромашинобудування та обладнання. – 2010. № 76. – С. 68 – 79.*
5. *Django – Вікіпедія [Електронний ресурс]. – Режим доступу: URL : <https://uk.wikipedia.org/wiki/Django/>. – Django – Вікіпедія.*
6. *Goel A.L. Software reliability models & Assumptions, Limitations and Applicability // IEEE Trans. – N 2. – P.1411-1423*
7. *Hagg S., Raja H.K., Sekade L.L. Quality Function Deployment. Usage in Software Development //Comm. Of ACM. – 1998. – 39. – N1.*
8. *Software Reliability Models by Neeta Singh, Poonam Verma, Ashok Kumar :: SSRN: [Електронний ресурс]. – Режим доступу: URL : <https://ssrn.com/abstract=2836387/>. - Software Reliability Models by Neeta Singh, Poonam Verma, Ashok Kumar :: SSRN:*