

ПОДХОДЫ К ТЕСТИРОВАНИЮ ПРОГРАММНОГО КОДА

Лукашевич Д.А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Качалов И.Л. – канд.ист. наук

В статье производится анализ видов и особенностей тестирования программного кода, производится описание стандартной ситуации тестирования программного модуля.

В техническом жаргоне тестирование означает проверку того, что наш код отвечает некоторым ожиданиям. Например: функция с именем «Concatenate» должна возвращать ожидаемый результат при некотором входе.

Существует много видов тестирования, но вот три основные категории [1]:

- модульное тестирование - тестирование отдельных модулей, таких как функции или классы, путем ввода и проверки того, что результат соответствует ожидаемому;
- интеграционное тестирование - тестирование процессов в нескольких модулях для достижения их целей, включая и побочные результаты
- End-to-end тестирование (также известные как тесты e2e или функциональные тесты) - тестирование функционирования сценариев на самом продукте с помощью управления браузером или веб-сайтом. Эти тесты обычно игнорируют внутреннюю структуру приложения и смотрят на них глазами пользователя, как на черный ящик

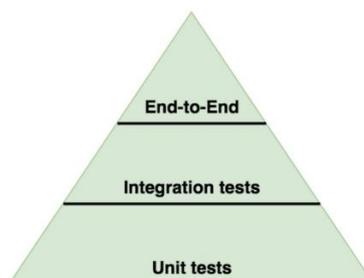


Рисунок 1 – Иерархия видов тестирования

Если вы пишете веб-приложение, хорошей отправной точкой будет тестирование каждой страницы приложения и каждого взаимодействия с пользователем. Но веб-приложения также состоят из блоков кода, таких как функции и модули, которые также должны быть протестированы.

В большинстве случаев есть два сценария:

- вы наследуете устаревший код, который поставляется без тестов
- вы должны реализовать новую функциональность из ничего

В обоих случаях рассматривать тесты за независимые части, которые проверяют, дает ли данная функция ожидаемый результат. Вот как выглядит типичный тестовый поток [2]:

- импортировать функцию для тестирования
- дать вход в функцию
- определить, что ожидать в качестве выхода
- проверьте, выдает ли функция ожидаемый результат

Jest — это JavaScript-фреймворк для тестирования, то есть библиотека JavaScript для создания, запуска и структурирования тестов. Jest поставляется в виде пакета NPM, имеется возможность установить его в любом проекте JavaScript. Jest - один из самых популярных фреймворков для тестирования в наши дни, и выбор по умолчанию для проектов на React.

Автоматические тесты дают уверенность, что ваша программа работает как задумано. Такие тесты можно запускать многократно. Успешное выполнение тестов покажет разработчику, что его изменения не сломали ничего, что ломать не планировалось.

Провалившийся тест позволит обнаружить, что в коде сделаны изменения, которые меняют или ломают его поведение. Исследование ошибки, которую выдает провалившийся тест, и сравнение ожидаемого результата с полученным даст возможность понять, где возникла ошибка, будь она в коде или в требованиях.

Список использованных источников:

1. Glenford J. Myers, *The Art of Software Testing*, 3-th edition 2012.
2. Cem Kaner, James Bach, Bret Pettichord *Lessons Learned in Software Testing* 2002.