

І.В. Богач
І.В. Вернік

ДОСЛІДЖЕННЯ ІНСТРУМЕНТІВ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ СЕРВІСІВ АРІ СЕРВЕРА

Вінницький національний технічний університет

Анотація

У даній роботі проаналізовано різні підходи до автоматизованого тестування сервісів АРІ сервера та обрано найефективніші інструменти для розробки автоматизованих тестів.

Ключові слова: АРІ, HTTP-методи, REST АРІ, SOAP АРІ, автоматизоване тестування.

Abstract

Different approaches of ARI server services automated testing are analyzed and the most effective tools are chosen for development of automated test API in this work.

Keywords: API, HTTP- methods, REST API, SOAP API, automated testing.

Вступ

У наш час більшість сучасних додатків працюють через АРІ. Для збільшення популярності свого продукту компанії пропонують відкриті АРІ, якими можуть скористатися сторонні розробники і створити щось власне як для простого сайту так і для високонавантажених розподілених систем та програм. Зацікавлення до тестування нестримно зростає протягом кількох останніх років, згідно з дослідженнями Google Trends [1]. Опитування, проведене компанією Smartbear в 2017 році, серед 5000 професіоналів в області розробки програмного забезпечення, показало, що більше 50% опитаних респондентів використовують автоматичні засоби тестування АРІ, і протягом наступних двох років очікується зростання їх кількості на 30% (з 59% до 77%), причому 80% учасників опитування вказали, що відповідають за тестування АРІ.

Метою даної роботи є дослідження підходів до автоматизованого тестування АРІ та обрання найкращого інструменту для автоматизованого тестування сервісів АРІ серверу.

Результати дослідження

АРІ (Application Programming Interface) – набір опису способів, якими одна комп'ютерна програма може взаємодіяти з іншою програмою; зазвичай входить в опис будь-якого інтернет-протоколу (наприклад, RFC), програмного каркаса (фреймворка) або стандартів викликів функцій операційної системи. АРІ часто реалізується окремою програмною бібліотекою або сервісом операційної системи та використовується програмістами при написанні програмних додатків. [1].

Тестування АРІ повністю відрізняється від тестування графічного інтерфейсу і в основному концентрується на шарі бізнес-логіки архітектури програмного забезпечення. GUI (graphical user interface) в АРІ тестуванні практично не потрібен.

Замість стандартних видів введення, призначених для користувача, даних (заповнення форм) для передачі даних тут використовується програмне забезпечення. Іншими словами, АРІ є посередником між розробником додатків і будь-яким середовищем, з яким цей додаток повинен взаємодіяти. АРІ спрощує створення коду, оскільки надає набір готових класів, функцій або структур для роботи з наявними даними.

Сучасні АРІ часто приймають форму веб-сервісів, які надають користувачам (як людям, так і іншим веб-сервісам) якусь інформацію. Зазвичай процедура обміну інформацією і формат передачі даних структуровані так, щоб обидві сторони знали, як взаємодіяти між собою.

Загально прийнятим стилем при зверненні до веб АРІ є використання запитів HTTP. Існують стандартні методи, які можуть міститися в HTTP запиті. Ці методи так і називають – HTTP-методи:

- GET – згідно з різними джерелами, найбільш популярний тип запиту; використовується для отримання або читання даних;

- PUT – частіше всього використовується для оновлення ресурсу;
- POST – використовується найчастіше для створення нового ресурсу;
- DELETE - видаляє дані. Та інші.

На сьогоднішній день є два основні підходи до побудови програмного інтерфейсу веб-додатків: REST (RESTful) API і SOAP API:

- REST (від англ. Representational State Transfer – «передача стану уявлення») забезпечує спілкування між клієнтом (як правило, це браузер) і сервером за допомогою звичайних HTTP-запитів (GET, POST, PUT, DELETE і т.д.), передаючи інформацію від клієнта – в параметрах самих запитів або інформацію від сервера – в тілі відповіді (може бути, наприклад, JSON-об'єктом або XML-документом). REST є архітектурним стилем, а не стандартом;

- SOAP (від англ. Simple Object Access Protocol – простий протокол доступу до об'єктів, аж до специфікації 1.2) характеризується використанням HTTP(S) протоколом лише як транспорту (найчастіше, методом POST). Всі деталі повідомлень (в обидві сторони – від клієнта до сервера і назад) передаються в стандартизованому XML-документі. SOAP може працювати і з іншими протоколами прикладного рівня (SMTP, FTP). SOAP є протоколом і має власну специфікацію [3].

Ми надамо перевагу інструментам для REST API, так як цей підхід є більш поширеним через свою відносну простоту і зручності для розробників. SOAP API переважно характерний для великих корпоративних (enterprise) систем.

Звичайно API можна перевіряти і вручну, за допомогою інструментів розробника в браузері, але автоматизоване тестування має ряд переваг [4]:

- точна локалізація. Уміння працювати з API дозволяє краще розуміти і точніше описувати виникли помилки;
- економія часу при підготовці тестових даних і ситуацій;
- можливість відтворювати тести на великих наборах вхідних даних;
- можливість брати участь в проектах, де робота з API є вимогою тест-дизайну.

Зважаючи на вище написане, ми можемо навести список інструментів які можуть допомогти у вирішенні задач автоматизованого тестування API [5]:

1. Postman. Будучи спочатку плагіном браузера Chrome, тепер Postman розширює свої технічні рішення разом з оригінальними версіями як для Mac, так і для Windows. Postman є відмінним вибором API тестування для тих, хто не бажає мати справи з кодуваннями в інтегрованому середовищі розробки, використовуючи ту саму мову програмування, що і розробник:

- легкий у використанні клієнт REST;
- може використовуватися як при автоматизованому, так і при дослідницькому тестуванні;
- володіє пакетом засобів інтеграції, таких як підтримка форматів Swagger і RAML;
- володіє функціями Run, Test, Document і Monitoring;
- не потребує вивчення нової мови програмування.

2. JMeter. Відкрите програмне забезпечення, широко використовується для функціонального тестування API, проте спочатку воно створювалось лише для навантажувального тестування, має ряд унікальних переваг:

- автоматично працює з файлами CSV, дозволяючи команді швидко створювати унікальні значення параметрів для тестування API;
- завдяки інтеграції між JMeter і Jenkins, користувачі можуть включати тести API в конвеєрні обробки CI;
- даний інструмент може використовуватися як для статичного, так і динамічного тестування продуктивності ресурсів.

3. Katalon Studio. Є безкоштовним інструментом автоматичного тестування, який надає загальну середу для створення і виконання UI функціоналу, служб API/Web і тестування мобільних платформ.

Здатність комбінувати рівні UI і Business (служби API/Web) для різних операційних середовищ (Windows, Mac OS, Linux) розцінюється як значна перевага Katalon Studio перед аналогічними продуктами. Katalon Studio підтримує запити SOAP і RESTful з різними типами команд (GET, POST, PUT, DELETE) з параметризованими можливостями. Має наступні переваги:

- сотні вбудованих ключів для створення тестових завдань;
- підтримка однієї з найпотужніших бібліотеки перевірки тверджень AssertJ для створення динамічних тверджень в BDD-стилі;
- підтримка підходу, керованого даними;

- може використовуватися як при автоматичному, так і при дослідницькому тестуванні;
- підходить як для професіоналів, так і для новачків.

4. Tricentis Tosca. Являє собою платформу безперервного тестування для Agile і DevOps. Серед переваг Tricentis Tosca слід зазначити:

- інтеграцію в цикли Agile і DevOps;
- максимізацію багаторазового використання і здатність до супроводу засобів автоматизації тестування на основі використання моделей;
- тести API можуть використовуватися як на мобільних, так на браузерних і пакетних додатках;
- досягнуто автоматизація, підтримувана новими технологіями;
- знижено час, необхідний на проведення регресивного тестування.

5. Arigee. Являє собою крос «хмарний» засіб тестування API, що дозволяє користувачам вимірювати і тестувати продуктивність API, забезпечувати технічну підтримку і розробку API за допомогою інших редакторів, таких як Swagger. Має наступні переваги:

- даний інструмент є багатокроковим і знаходиться під управлінням Javascript;
- він дозволяє розробляти, відстежувати, виконувати розгортання і масштабування API;
- ідентифікує проблеми шляхом відстеження трафіку API, рівня помилок і часу відповіді;
- легко створює проксі API з Open API Specification і виконує їх розгортку в «хмарі»;
- моделі розгортання в «хмарі», локального розгортання і гібридного розгортання працюють на основі одного коду;
- PCI, HIPAA, SOC2, і PII для додатків і API;
- Arigee розроблений спеціально для цифрового бізнесу і завдань з інтенсивною обробкою даних на під управлінням мобільних платформ API і додатків, які керують ним.

При розробці та тестуванні такого типу задач потрібно обирати необхідний метод опираючись на завдання, які потрібно вирішити розробникам.

Висновки

Тестування сервісів API сервера створило свій власний тренд в області автоматичного тестування, і чим далі, тим більше інструментів буде створюватися для задоволення зростаючих запитів від команд розробки програмного забезпечення. В даному дослідженні ми показали що інструмент для такого тестування потрібно обирати спираючись на задачу яка стоїть перед командою розробників.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. API testing [Електронний ресурс] // Google Trends – 2018. – Режим доступу до ресурсу: <https://trends.google.com/trends/explore?date=today%205-y&q=api%20testing> - Назва з екрану.
2. API [Електронний ресурс] //Вікіпедія: вільна енциклопедія – 2011. – Режим доступу до ресурсу: <https://ru.wikipedia.org/wiki/API> - Назва з екрану.
3. SOAP Version 1.2 Part 1: Messaging Framework (Second Edition) [Електронний ресурс] – 2007. – Режим доступу до ресурсу: <https://www.w3.org/TR/soap12/> - Назва з екрану.
4. Шальнев А. Освоение тестирования REST API [Електронний ресурс] / Андрей Шальнев // Лаборатории Качества. – 2017. – Режим доступу до ресурсу: <https://quality-lab.ru/rest-api-testing/>. - Назва з екрану.
5. TOP 5 API TESTING TOOLS FOR 2018 [Електронний ресурс] // Katalon Studio Team. –2018. – Режим доступу до ресурсу: <http://www.abodeqa.com/top-api-testing-tools-review/>.- Назва з екрану.

Богач Ілона Віталіївна – кандидат технічних наук, доцент факультету комп'ютерних систем і автоматики, Вінницький національний технічний університет, м. Вінниця. e-mail: ilona.bogach@gmail.com

Вернік Ірина Володимирівна – студентка групи І-15б, факультет комп'ютерних систем та автоматики, Вінницький національний технічний університет, м. Вінниця, e-mail: vernik.iryana@gmail.com.

Iлона Bogach – PhD, Associate Professor of the department of automation and intelligent information technologies, Vinnytsia National Technical University, Vinnytsia, e-mail: ilona.bogach@gmail.com

Vernik Iryna – Faculty of computer systems and automatics, Vinnytsia National Technical University, Vinnytsia, email: vernik.iryana@gmail.com.