

Тестування API

№ уроку: 34 **Курс:** Manual QA

Засоби навчання: Браузер, Microsoft Office, Postman, SoapUI

Огляд, мета та призначення уроку

Метою даного уроку є огляд теорії, необхідної для API тестування та можливості сучасних інструментів для швидкого розгортання та підтримки API тестів.

Вивчивши матеріал даного заняття, учень зможе:

- Розбиратися в базовій теорії API
- Проводити тестування API.
- Оцінювати та швидко налаштовувати автоматизоване API тестування

Зміст уроку

- Основна теорія
 - GET, POST, PUT, DELETE etc.
 - REST & SOAP
 - JSON & XML
 - Корисна тест-документація для API
- Swagger
- Postman
 - Приклад базових функцій (Одинковий запит; Колекції; Глобальні змінні та змінні Оточення; JS перевірки)
- SoapUI
 - Приклад базових функцій (Одинковий запит; Тест кейси)

Резюме

- **API (Application Programming Interface)** — це інтерфейс програмування програми.
- **API** — це специфікація можливих взаємодій із компонентом ПЗ. Наприклад, якби машина була компонентом ПЗ, її API містила б інформація про те, що машина може прискорюватися, гальмувати і включати радіо. Її API також включало б інформацію про те, як прискорюватися.
- **Веб-сервіси (або веб-служби)** — це технологія, що дозволяє системам обмінюватися даними один з одним через підключення до мережі. Зазвичай веб-сервіси працюють поверх протоколу HTTP або протоколу вищого рівня. Веб-сервіс - просто адреса, посилання, звернення до якої дозволяє отримати дані або виконати дію.
- Існують два основні підходи до створення програмного інтерфейсу веб-додатків: REST (RESTful) API та SOAP API.
- REST архітектурний стиль 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, інформацію від сервера – у тілі відповіді (яка може бути, наприклад, JSON-об'єктом або XML-документом). REST є архітектурним стилем, а не стандартом.
- **SOAP** є протоколом і має специфікацію, що характеризується використанням HTTP(S)-протоколу лише як транспорту (найчастіше, методом POST). Усі деталі повідомлень (в обидві сторони – від клієнта до сервера та назад) передаються у стандартизованому

XML-документі. SOAP може працювати з іншими протоколами прикладного рівня (SMTP, FTP), але найчастіше він застосовується поверх HTTP(S). WSDL (Web Services Description Language) — мова опису веб-сервісів та доступу до них, заснована на мові XML.

- **SOAP** Найбільш застосовний у складних архітектурах. Довгий час саме він був протоколом обміну повідомленнями, який використовували майже всі веб-служби. Оскільки в наші дні розробникам необхідно створювати легкі веб-додатки та мобільні програми, гнучкіша архітектура REST швидко завоювала популярність через свою простоту і прозорість. Дійсно, якщо будь-яким об'єктам вашого сервісу не потрібні складніші взаємини, крім: "Створити", "Прочитати", "Змінити", "Видалити" (як правило - в 99% випадків цього достатньо), можливо, саме REST стане правильним вибором. Крім того, REST в порівнянні з SOAP, може виявитися і більш продуктивним, так як не вимагає витрат на розбір складних команд XML на сервері (виконуються звичайні HTTP запити - PUT, GET, POST, DELETE). Хоча SOAP, у свою чергу, більш надійний та безпечний.
- **Swagger** — це фреймворк та специфікація для визначення REST APIs у форматі, дружньому до користувача та комп'ютера
- **Postman** – це інструмент для роботи з API, який дозволяє тестувальнику надсилати запити до сервісів та працювати з їхніми відповідями. З його допомогою можна протестувати бекенд та переконатися, що він коректно працює.
- **Переваги:**
 - інтуїтивно-зрозумілий і простий у використанні, не вимагає якогось складного налаштування або знання мов програмування;
 - безкоштовний;
 - підтримує різні API (REST, SOAP, GraphQL);
 - розширюється під будь-які потреби за допомогою Postman API;
 - легко інтегрується в CI/CD за допомогою Newman - консольної утиліти для запуску тестів;
 - запускається на будь-яких ОС;
 - підтримує ручне та автоматизоване тестування;
 - зібрав навколо себе велике ком'юніті, де можна знайти відповіді на будь-які питання.
- Тестувальнику цей інструмент дозволяє:
 - надсилати запити та отримувати відповіді;
 - зберігати запити до папок та колекцій;
 - змінювати параметри запитів;
 - змінювати оточення (dev, test, production);
 - виконувати автотести, використовуючи Collections runner, у тому числі за розкладом;
 - імпортувати та експортувати колекції запитів та набори тестів, щоб обмінюватися даними з колегами.
- **SoapUI** підтримує сервіси REST та SOAP. Спеціалістам для автоматизації тестування API на вибір доступні у версіях open-source та pro. Pro версія має зручний інтерфейс та кілька додаткових можливостей, такі як майстер створення перевірок, редактор форм та конструктор SQL запитів. SoapUI – входить до готового набору інструментів Ready API suite, що пропонує компанія SmartBear.
- **Переваги:**
 - Легка генерація тестів за допомогою перетягування та point-and-click
 - Потужне тестування з підстановкою даних з фалів або БД
 - Повторне використання сценаріїв
 - Mocks – сервіси з використанням RESTful mocking
 - Асинхронне тестування

Закріплення матеріалу

- Чи можна використовувати формат RESTfull API XML для передачі повідомлень?
- Яке завдання у API?
- Коротко опишіть різницю між REST та SOAP

Самостійна діяльність учня

Завдання 1

Створіть 5 запитів у Postman. Додайте кожному запиту тест, об'єднайте їх в одну колекцію і запускайте в Monitor. Усі тести мають проходити успішно. Використовуйте змінні оточення та глобальні змінні. Ви можете використовувати будь-який відкритий REST API.

Завдання 2

Створити 3 тест кейси в SoapUI/ReadyAPI. Додайте кожному тесту хоча б один свій assert. Усі тести мають проходити. Ви можете використовувати будь-який відкритий SOAP API.

Рекомендовані ресурси

Що таке API і як це працює

<https://vps.ua/blog/how-api-works/>

Відмінності REST та SOAP

<https://habr.com/ru/post/483204/>

REST / SOAP

<https://qastart.by/class-2/23-rest-soap>

Swagger Demo

<https://petstore.swagger.io/>

Postman Learning Center

<https://learning.postman.com/docs/getting-started/introduction/>

SoapUI Docs

<https://www.soapui.org/docs/soapui-projects/>

Public SOAP APIs

<https://documenter.getpostman.com/view/8854915/Szf26WHn?version=latest>