

Силабус курсу

AQA Playwright JS/TS

Автоматизація тестування вебзастосунків на JavaScript та TypeScript

Курс допоможе перейти від Manual QA до automation testing: писати автотести на Playwright, працювати з JavaScript і TypeScript, тестувати UI, API та платіжні сценарії, налаштовувати reports і CI/CD.

Формат	Online
Рівень	після QA Manual / базовий JavaScript
Тривалість	32 заняття
Графік	2 заняття на тиждень
Результат	GitHub-портфоліо + automation framework
Фокус	Playwright, TypeScript, UI/API automation, CI/CD, reports, payment scenarios

Фокус курсу

Playwright	TypeScript	JavaScript
Node.js / npm	Git / GitHub	GitHub Actions
API / Postman	SQL / Data	Allure reports
Docker basics	Payment Testing	Mobile basics

Про курс

Курс допоможе перейти від Manual QA до automation testing: писати автотести на Playwright, працювати з JavaScript і TypeScript, тестувати UI, API та платіжні сценарії, налаштовувати reports і CI/CD.

Вивчаємо інструменти та технології

Playwright UI automation, browser contexts	TypeScript Типізація тестів і даних
JavaScript Логіка тестів і helpers	Node.js / npm Екосистема проєкту
Git / GitHub Branches, pull requests, README	GitHub Actions CI запуск автотестів
API / Postman Запити, колекції, аналіз API	SQL / Data Перевірки даних і test data
HTML / Allure reports Звіти для команди	Docker basics Локальне середовище
Payment Testing Checkout, validation, negative flows	Mobile basics Viewport і responsive checks

Кому підійде курс

- Manual QA, які хочуть перейти в automation
- Junior QA з базовими знаннями вебтестування
- студентам після QA Manual або JS basics
- тим, хто хоче GitHub-проєкт для портфоліо

Що буде на виході

- Playwright + TypeScript automation framework
- UI та API автотести
- reports, traces, screenshots, videos
- CI/CD запуск через GitHub Actions
- підготовка до першої ролі Junior AQA

Програма курсу. Частина 1

№	Тема заняття	Практика / результат
1	Вступ в AQA. Manual QA vs Automation QA vs SDET.	Розуміння ролі AQA в команді та очікувань роботодавців.
2	Як виглядає automation project: тести, репорти, CI/CD.	Розбір структури реального automation framework.
3	Налаштування середовища: Node.js, npm, VS Code, Playwright.	Створення першого Playwright project.
4	Git і GitHub для AQA: repository, branches, pull request.	Публікація проєкту в GitHub.
5	JavaScript basics: variables, types, operators, conditions.	Практичні задачі для QA automation.
6	Functions, arrays, objects.	Робота з тестовими даними та helper-функціями.
7	Async JavaScript: Promise, async/await.	Розуміння асинхронності в автотестах.
8	JSON, modules, imports / exports. Модульна робота 1.	Створення helper-файлів і структури test data.
9	TypeScript basics: types, interfaces, annotations.	Типізація тестових даних і функцій.
10	TypeScript у Playwright project.	Перенесення тестів на TypeScript.
11	Playwright Test: структура тесту, test, expect, hooks.	Перші стабільні UI-тести.
12	Locators: role, text, label, CSS, XPath.	Підбір надійних локаторів для вебелементів.
13	Actions: click, fill, select, upload, keyboard, mouse.	Автоматизація форм і user flows.
14	Assertions і auto-waiting.	Надійні перевірки без зайвих waits.
15	Browser, context, page.	Розуміння ізоляції тестів і стану браузера.
16	Debugging: inspector, trace viewer, screenshots, videos.	Аналіз падіння тестів. Модульна робота 2.
17	Config files: baseURL, projects, browsers, environments.	Запуск тестів у різних браузерах і середовищах.
18	Page Object Model.	Створення LoginPage, DashboardPage, CheckoutPage.

Навчання побудоване на практичних завданнях після кожної теми.

Програма курсу. Частина 2

№	Тема заняття	Практика / результат
19	Fixtures y Playwright.	Reusable setup для авторизації, даних і сторінок.
20	Test data management: JSON, env variables, faker basics.	Організація тестових даних.
21	Tags, grep, test suites, smoke і regression structure.	Запуск окремих груп тестів.
22	Reports: HTML report, Allure basics.	Генерація звітів для команди.
23	API testing basics: REST, JSON, status codes, headers.	Повторення API через практичні приклади.
24	Playwright APIRequestContext: GET / POST / PUT / DELETE.	Автоматизація базових API-запитів.
25	API + UI combined tests.	Підготовка даних через API і перевірка в UI.
26	Auth, tokens, headers, negative API tests.	Перевірка авторизації та помилкових сценаріїв.
27	Payment testing: checkout, validation, success і failed payment.	Автоматизація платіжного сценарію.
28	Mobile і responsive basics y Playwright.	Тести для mobile viewport і responsive layout.
29	CI/CD: GitHub Actions для Playwright.	Запуск автотестів у pipeline.
30	Docker basics для AQA.	Локальний запуск тестового середовища.
31	Framework cleanup: ESLint, Prettier, README, scripts.	Підготовка проєкту до портфолію.
32	Фінальний проєкт, презентація і mock interview.	Захист automation framework і співбесіда.

Навчання побудоване на практичних завданнях після кожної теми.

Підсумок курсу

Фінальний проєкт

Студент створює Playwright + TypeScript automation framework для вебзастосунку: UI-tests, API-tests, Page Object Model, fixtures, test data, reports, GitHub Actions, payment flow, mobile/responsive checks і README для портфоліо.

Що студент умітиме після курсу

- писати автотести на Playwright + TypeScript
- будувати Page Object Model та fixtures
- автоматизувати UI, API і payment scenarios
- працювати з reports, traces, screenshots і videos
- запускати тести в GitHub Actions
- пояснювати framework architecture на співбесіді

Кар'єрні перспективи

Після курсу студент може претендувати на позиції Junior QA Automation Engineer, Junior AQA Engineer, Automation QA Intern, Playwright QA Engineer або Junior SDET.

Переваги курсу

- практичний перехід з Manual QA в Automation
- фокус на web, API і payment scenarios
- реальний GitHub-проєкт для портфоліо
- підготовка до Junior AQA співбесіди
- структура курсу під роботу в команді