



Egecan Şen

canegecansen@gmail.com

LinkedIn

GitHub

Email

Download

## About

Experienced in;

- **Test Automation Frameworks:** Designing and implementing frameworks using Page Object Model and BDD.
- **Cross-Platform Testing:** Automation tools such as Selenium, Appium for mobile testing, and API testing tools to ensure integration across web, mobile, and backend systems.
- **QA Processes & Strategy:** Optimizing SDLC models, establishing test automation stages, and supporting scrum activities to streamline quality assurance processes.
- **Release Management:** Played a key role in release approval processes, coordinating manual regression testing, validating automation results, and tracking deployment pipelines with real-time updates.

## Experience

**QA Automation Engineer @citizenM Hotels** (19.06.2023 - Present)

citizenM is a Netherlands-based, global hotel developer, investor, and hotel chain.

- Handled website QA process during a database migration project called Sunrise.
- Handled the UI part of the **Performance Tests** for the backend migration project.
- Handled the QA process for the website **Nuxt migration** project, creating and automating key flow scripts.
- Proposed and **established a test automation stage for website** deployment pipelines, setting up a Slack channel to track results.
- **Co-led QA Scrum processes**, organizing meetings and scrum activities.
- **Supported QA process of the mobile app** during periods of QA absence.
- **Adjusted new SDLC models** for the website product and the content management system based on the main model, defining testing stages and workflows. Created SDLC documentation for both the website team and the Contentful product.
- Increased focus on **API automation** for future cycles.
- Proposed the **X-ray Test Management Tool** to improve QA processes, handled the integration process for the website team.
- **Became one of the release approvers** for the website.
- Handled the QA process of a content migration project called Headless Marketing content.
- **Led Black Friday QA coordination**, overseeing campaign changes and flow tracking.

### Key Projects

- **Sunrise Database Migration Project:** Handled website QA processes and performed UI performance tests for backend migration.
- **Nuxt Migration Project:** Automated key flows and ensured smooth migration to Nuxt framework.
- **Headless Marketing Content Migration:** Oversaw QA testing for migrating content to the new content management system.
- **Xray Test Management Tool Setup:** Proposed for QA processes and configured for the website, enhancing test tracking, execution, and reporting processes.
- **Black Friday Campaign QA Coordination:** Led QA efforts to ensure successful campaign implementation and flow tracking.
- **Hotel Service Project:** Managed the end-to-end QA processes for the Hotel Service website, a platform designed for downloading invoices, managing bookings, and completing payments.

**Test Automation Engineer @Smart Nearshore** (19.06.2023 - Present)

Smart is a Portugal-based company that provides information technology and consulting.

- Providing outsourced services to key clients.
- Assisted in project decision-making and requirements gathering for various projects.
- Supported the execution of QA strategies and automation processes for multiple projects.
- Provided technical consultation on automation and testing best practices for critical projects.

**Automation Engineer** (08.08.2022 - 09.04.2025)

- Developed comprehensive automation frameworks for diverse use cases, including UI testing and API interactions.
- Contributed to open-source libraries, improving their functionality and expanding Java-based automation capabilities.
- Created reusable project templates and utilities to standardize common methods and interactions, boosting efficiency for future projects.
- Co-developed the [GPT-Utilities](#) library, introducing AI-powered features to enhance automation workflows.
- Built an AI-supported chat integration for the [Pickleib](#) library, enabling user interaction and query handling.
- Continuously integrated innovative technologies into automation processes to enhance productivity and scalability.

## Skills

- **Selenium** - Web UI Automation
- **Appium** - Mobile UI Automation
- **Retrofit & OkHttp** - API Test Automation
- **Springboot** - Dependency Injection
- **Cucumber** - BDD Framework
- **Git** - Version control
- **OctoPerf** - Performance Testing
- **Jenkins & Azure DevOps** - Continuous Integration
- **Postman** - Manual API Testing
- **X-ray** Test Management Tool
- Intermediate level **Java**
- Intermediate level **Maven**

## Education

- Music Technology - BA, Department of Music Sciences
  - Dokuz Eylul University
- Protohistory and Near Eastern Archaeology - BA, Faculty of Letters
  - Ege University

## Certificates

- [ISTQB Foundation Level \(CTFL\)](#)

# Projects

### Unified-Automation-Template

- Unified Automation Template is a modular automation framework built on top of the [Pickleib](#) library.
- This template is designed to provide a unified solution for Web UI, Mobile UI, and API automation.
- Multi-Platform Testing: Supports Web UI, Mobile UI, and API automation.
- Scenario Tagging: Enables running specific test scenarios using tags.
- Customizable Driver Selection: Defaults to Chrome for Web UI, with the option to choose other drivers.
- Headless Execution: Supports headless mode via configuration.
- Project page — [Unified-Automation-Template](#)

### Simplified-Automation-Template

- This project is a template project that includes UI & API interactions.
- The Web-UI part of the project is based on Selenium & Cucumber and the API part is based on Okhttp & Retrofit.
- The CommonSteps class stores the generic UI steps and saves us from the repeated code.
- The ObjectRepository class is used for storing the page object instances; the page objects are automatically instantiated when the element of that page is interacted with.
- The PageObject class is for overwriting Selenium's default field decorator to be able to use WebComponents.
- You can find the cucumber test reports under the reports directory in resources folder. After the run is completed you can find the generated results in the target directory under the reports folder.
- Project page — [Simplified-Automation-Template](#)

### Appium-Springboot-Template

- This project is a automation testing framework built with JUnit 5, Appium, and Spring Boot, designed to support mobile automation.
- Project is designed for CI/CD integration using Jenkins and Docker. All dependencies are installed inside the Jenkins container automatically.
- UI automation follows the Page Object Model for clarity and maintainability.
- API tests use a custom library layered on Retrofit & OkHttp for seamless integration and request handling.
- The project uses a Maven multi-module layout, where tests and post-report are sibling modules under a common parent.
- Reports are generated via Maven Surefire. The HTML report can be emailed by providing credentials and setting the send-report-email property to true.
- Project page — [Appium-Springboot-Template](#)

## Open-Source Project Contributions

- [Pickleib](#)
  - Picklelib is a quickstart library for test automation projects capable of webUI, mobileUI, API, database, data layer & email tests.
  - Improved documentation, added new methods, property configurations and scalability for future upgrades.
  - Created unit tests and StatusWatcher feature for monitoring the test statuses.
  - A MongoDB database was created containing methods and documentation of the library.
  - Added chat support functionality. This feature leverages the OpenAI API and MongoDB Atlas API services along with the GPT-Utilities library.

[Pickleib v2.0.1](#)

- [Wasapi](#)
  - Wasapi is a lightweight, Java-based API utility library that simplifies API calls using Retrofit & OkHttp, making it easier to automate API calls in Java projects.
  - Dockerized a MongoDB backend for unit testing, making it ideal for modern CI pipelines.
  - Created unit tests based on the Food Planner application and a local MongoDB database.

[Wasapi v0.0.1](#)

- [GPT-Utilities](#)
  - This library utilizes the OpenAI API service to provide inner GPT functionality for Java projects.
  - It offers various features such as chat support, data generation, and more, which can be seamlessly integrated into any Java project.
  - Established the initial framework and set up the foundation for API usage. Additionally, provided support throughout the development process.

[gpt-utilities v0.1.5](#)

- [Java-Utilities](#)
  - Created documentation, specify additional descriptions and ensure that it is comprehensive and effortless for users to understand.
  - Added new methods to the library. These enhancements aimed to extend the functionality and practicality of the Java Utilities library and to provide end users with more alternatives and efficiencies when working with Java applications.

[Utilities v1.7.5](#)