Mayur Prajapati
Sr Software Engineer (Automation)

LinkedIn: https://linkedin.com/in/mayurengineer
Medium: https://medium.com/@mayurengineer
Github: https://github.com/mayurprajapati (Major projects are private)

M: 635 133 4389 | |Email: mayurprajapati1042@gmail.com

OBJECTIVE: Looking for a dynamic role as a Software Developer specializing in Automation within an innovative organization, where I can apply my extensive technical skills and academic background to contribute to the advancement of the company while enhancing my expertise.

PROFESSIONAL SUMMARY

- Over 4 years of hands-on experience in **Automation Development**, specializing in **Java**, **Selenium, Appium**, and **various automation domains**, including **Windows Application Driver**, **Web**, **API**, **Excel**, **CSV**, **Playwright**.

- Proficient in **Java 8 & 11**, with expertise in **JDK Lambda Functions**, **Object-Oriented Programming** (OOPs), and **debugging**.

- Skilled in **configuring and managing Jenkins**, including plugin installation.

- Experience in **Java Multithreading** using modern classes such as *ExecutorService*, *Future*, and *ThreadLocal*.

- **Developed and worked on Core Automation frameworks** incorporating Selenium, Appium, PDF, and OCR, along with basic Automation frameworks.

- Strong understanding and practical application of **Java Functional Programming** within the *java.util.function* package.

- Extensive expertise in Selenium, including advanced knowledge of its internal workings, iFrames, waiting strategies, and Selenium wait strategies (Fluent, Implicit, Explicit).

- Advanced proficiency in **building complex and nested XPath** expressions, practical experience with **XPath Axes** and **XPath Functions**, and working with Browser iFrames.

- Comprehensive experience using **Eclipse IDE**, emphasizing its importance in becoming a proficient developer.

- Extensive practical experience with **Eclipse Debugging features**, utilizing features like **Hot Code Replace** and **Debug Shell** on a daily basis.

- Utilized Eclipse's Git Management (EGit) feature as a user-friendly alternative to Git CLI.

- Established and maintained **Code Templates in Eclipse**, sharing them with the team for improved productivity.

- Installed, managed, and **administered Jenkins, with expertise in Plugins Management**, overseeing plugins like **Pipeline Plugin**, **HTTP Request Plugin**, and **Pipeline Utility Steps**.
- **Configured and managed over 2000+ Jenkins jobs**, including the development of a script for programmatically updating Jenkins job Pipeline scripts.
- Developed and managed a Groovy project using **Jenkins Shared Library**, ensuring the execution of every job on Briq's Jenkins. The initial 100-line script has expanded to over 400 lines, incorporating advanced Jenkins features.
- **Applied PDF automations**, leveraging Regex features extensively and utilizing a **regex debugger** for enhanced efficiency.
- Utilized **Google Logs** and **Google Bucket** in an Automation framework.
- Demonstrated expertise in **Firebase's Firestore** database.
- Developed and maintained a diverse range of Automation scripts (over 2300 scripts) using Java and Open-Source technologies.
- Managed over 350 **automations in the Construction domain**, covering **Web**, **Desktop**, **PDF**, **API**, and **OCR automations**.
- **Automated renowned Construction Software applications**, including Sage 300, Vista, Quickbooks, Procore, AccuBuild, Foundation, Foundation Cloud (OCR), ComputerEase and CMiC.
- Engineered and maintained a comprehensive **Screen Recording feature** for Web and Desktop automations using FFmpeg, ensuring cross-platform functionality on both Windows and Linux/Docker.
- Implemented Dynamic iFrame switching in Selenium using JavaScript, streamlining navigation in scenarios with multiple iFrames.
- **Overcame challenges in reading large Excel files using XSSF SAX API** due to Apache POI limitations with large files.
- **Developed a Date Format Parser capable of handling 100+ date formats**, with the ability to detect date formats with **nearly 95% accuracy**.
- Integrated GPT-4 API into automation, allowing the generation of key-value text from human-readable input for Procore integration.
- Extensive experience with Sikuli & Sikuli IDE, as well as AWS Textract for building OCR automations.

<u>TECHNICAL SUMMARY</u>

Programming Languages: Java, Groovy, Python, Powershell, Bash

Known Java Libraries: Lombok, Apache (Commons Email, Commons Lang, Apache POI), Guava, OpenCSV

AI Tools: Chat GPT

CI/CD: Jenkins, Gitlab CI/CD

Query Languages: XPath, Json Path, Google Logs Query Language

Automation Tools: Selenium, Playwright, Appium, Windows Application Driver

REST Clients: RestAssured

Build Tools: Maven, Gradle

Operating Systems: Windows 10, Windows 11, Ubuntu

IDE: Eclipse, VS Code

Source Control: Git

Database: PostgreSQL, MongoDB, Firestore

Database Connection Tools: MongoDB Compass, DBeaver

Virtual Machine Connection Client: VNC, RDP, RDP RemoteApp

Operating Systems: Windows 11, Ubuntu

Utilities: Debugging, Lombok, Google Logs Query Language, Google Bucket, Regex, Power Toys

PROFESSIONAL EXPERIENCE:

Briq, Pune || Apr 2021 - till date

Sr. Software Engineer (Automation)

<u>Responsibilities</u>

- Developed diverse Web, Desktop, API, and Database automation bots tailored to unique client requirements.
- **Played a key role in decision-making** alongside Team Lead, VP, and senior team members for the automation team.
- Specialized in diverse automations: Web, Desktop, API, Database, OCR.
- Automated Multi-Factor Authentication systems and contributed to core modules of the Automation framework, leveraging various open-source technologies and Java.
- Reviewed peers' Merge Requests, provided constructive feedback, and engaged in mutual learning sessions with team members.
- Developed utility tools to enhance team productivity.
- Served as a mentor, assisting team members in problem-solving.
- Executed ETL automations across various domains: Web, Database, API, and Desktop.
- Automated ERP applications like Sage 300 & Procore, employing OCR technologies such as Sikuli and AWS Textract for [RemoteApps](#).
- Enhanced Selenium with Dynamic iFrame switching, enabling seamless navigation and interaction within multiple iFrames.
- Resolved challenges in handling large Excel files by implementing the XSSF SAX API for efficient reading, overcoming limitations with Apache POI.
- Developed a versatile Date Format parser capable of converting date formats from over 100 standards, detecting date formats with 95% accuracy.
- Automated [Java Code Generation](#) along with Git, and GitLab Merge Requests, based on available Error Codes and their Description in the Database
- Automated over 350 bots at Briq, covering:
  - o Web automation with Selenium.
  - o Desktop automation with Appium and WinAppDriver.
  - o PDF automation.
  - o API automation using RestAssured.
  - o OCR automation with Sikuli and Textract.

Accomplishments

- [Recognized by the VP](#) **for enhancing scalability in existing and new workflows**, **resulting in a 90% increase** in overall efficiency for artifact deployment.
- **Completed the migration of our primary Automation project**, **eliminating JUnit** dependency **in over 1200 bots** (Java files), and introduced BriqBotRunner for centralized bot execution using Java Reflection.
- Contributed to the optimization of Briq App's Audit Trail (Vue.js-based), **enhancing API calls and adding daily-used features**.
- **Significantly reduced compilation time by 90%** by implementing Gitlab artifacts and CI/CD across 1200+ Jenkins jobs.
- **Enhanced internal Python Micro Service projects and improved Automation-related tools**.
- Implemented Gitlab CI/CD for Maven dependency deployment in the core Automation project, utilized by briq-bot as a Maven dependency.
- **Introduced the briqbot-runner** ([a Jenkins Shared Library](#)) project with utility methods, including:
    - Checking Jenkins node status
    - Providing user information for job runs
    - Compiling source code selectively based on release branches
    - Identifying the operating system for appropriate command execution (e.g., Unix/Windows)
    - Reading environment variables to detect JDK 11 and running the Automation project accordingly.
- **The briqbot-runner streamlined the execution of each bot on a Jenkins node, consolidating code and simplifying configuration updates across multiple jobs**.
- **Implemented a platform-independent video recording feature using FFmpeg** for both web and desktop automations, **a highly praised functionality among clients**.
- Implemented cryptography middleware for file transfer between Java(automation framework) and Python (Briq backend) using Google Tink
- **Integrated GPT into automation**, allowing the conversion of human-readable text to machine-readable text, facilitating input into Procore and receiving positive feedback from clients.
- Introduced "Briq Spy," a Windows Applications inspection tool inspired by **[Accessibility Insights](#)**.

- Enhanced and expanded features in a clone of **mRemoteNG**, ensuring constant connectivity for RDP VMs to maintain viewport and enable smooth automation execution on Windows systems.
- Overhauled the entire Briq Setup Exe, a Python project responsible for configuring new client Windows VMs. The setup includes:
    - Optional installation of VNC server based on client preference (VNC or RDP).
    - Configuration of Jenkins agent, establishing a connection with our Jenkins server.
    - Activation of Developer options in Windows for seamless execution of Automations.
    - Installation of Maven dependencies to save time and streamline Automation project setup.
    - Installation of WinAppDriver, Inspect Tool, and other tools for spying and automating Windows applications.
- Introduced **Briq WebDriver** to address significant challenges:
    - Eliminated the need to install Eclipse and other tools on client systems, reducing setup complexity.
    - Mitigated potential source code breaches by enabling remote automation/debugging of Windows applications from the developer's system to the client's system.
    - Leveraged TCP Tunnelling with Websockets for secure and efficient communication.

Oracle, Gandhinagar || Jul 2019 – Apr 2021
Quality Assurance Analyst

<u>Responsibilities</u>

- Development and maintenance of test scripts provided by Manual QA.
- Proficient in Selenium, Git, and Oracle OATS.
- Automation of Oracle HCM products using OATS & KWiK framework in Java.
- Resolution of automation script bugs found on ALM.
- Developed & Maintained over 200 Automation scripts using OATS, Java & Selenium.

<u>Accomplishments</u>

- Constructed and enhanced automation tools/framework for Oracle HCM automation.
- Served as the full-stack developer for the HCM Data Search Utility, resulting in a 30% boost in automation team productivity.
- Revamped the HCM Simple Report Generator, reducing load time and introducing new features.
- Implemented a new feature in the HCM KWik recorder.

1. HCM Data Search Utility tool

- An efficient tool for searching HCM automation data sources, supporting various file formats (CSV, XLS, Zip, JSON). Provides quick search results within 3-4 seconds over 18000+ files, with advanced filters for relevance.
- Used binary search and indexing mechanism to reduce search time.
- Tech Stack: Kotlin, KTor, GSON (Backend) | VueJS with CLI, Vuex, Vuetify (Frontend).

2. HCM Simple Report Generator

- Transforms HTML reports generated by OpenScript into a new, well-structured format with additional information for improved debugging and detailed QA insights. Features a REST service for integration with other tools.
- Tech Stack: Kotlin, KTor, GSON (Backend) | VueJS with CLI, Bootstrap, JQuery (Frontend).

3. KWik Recorder

- Introduced a valuable video recording feature to extend the capabilities of the KWik Recorder extension. - The language used - JavaScript

With an outstanding experience of OpenSript & Selenium with TypeScript.

All In Motion IT Solutions, Vadodara || Jan 2019 – Apr 2019

Flutter Developer Internship

Responsibilities

Design and develop Flutter apps along with Firebase.

Moto Mates (Application)
- A cross platform app to unite Riders
- This is a social media like app which has specific functionality for Riders also to organize an event & schedule a ride.
- Riders from other cities can also get notifications & join.