Quality Assurance Portfolio

Our QA engineers possess technical skills and experience in the following:

Virtualization

- VMware Workstation
- VMware vSphere
- VMware ESXi
- Oracle VM VirtualBox
- Hyper-V/SCVMM
- Citrix
- QEMU/KVM
- Proxmox

OS installation and configuration

- Windows (XP and higher)
- Windows Server (2003 and higher)
- Linux (Debian, Ubuntu, Mint, RedHat, Fedora, CentOS, openSUSE, SLED distributions)
- macOS (macOS 10.9 Mavericks and higher)
- Solaris 11
- AIX 7.1

Server and network configuration

- Active Directory
- DHCP servers
- DNS
- Failover cluster configuration

Network tools and utilities

- Wireshark
- Charles
- Fiddler
- TCPView
- NMap
- WANem
- Packet Sender
- iPerf
- NetLimiter
- Browser debugging tools
- Burp Suite

Debugging tools	❖ WinDbg ❖ Driver Verifier ❖ Application Verifier ❖ Sysinternals Suite
Databases	❖ MS SQL Server ❖ Firebird ❖ PostgreSQL ❖ Oracle
Web servers	❖ IIS ❖ Apache ❖ nginx
Performance and load testing tools	❖ JMeter ❖ Tsung ❖ nGrinder ❖ Windows Performance Monitor ❖ iostat (Linux)
Security and penetration testing	❖ Kali Linux toolkit
Test automation tools and frameworks	❖ Selenium WebDriver ❖ Cypress framework ❖ NUnit ❖ TestNG ❖ Telerik Test Studio ❖ CodedUI ❖ AutoIT ❖ Postman ❖ SoapUI ❖ MS Visual Studio ❖ IntelliJ IDEA ❖ REST APIs ❖ SVN/Git
Behaviour Driven and Keyword Driven Development frameworks	❖ Robot Framework ❖ Gauge

Programming languages	❖ C# ❖ Java ❖ JavaScript ❖ Python ❖ C/C++ ❖ Gherkin
Scripting	❖ Bash ❖ PowerShell ❖ Batch files
CI/CD tools	❖ Jenkins ❖ Bamboo ❖ GitLab CI/CD ❖ Azure DevOps
Containers and orchestration	❖ Docker ❖ Docker Compose ❖ Docker Swarm ❖ Kubernetes
Configuration and management of cloud services	❖ Amazon Web Services ❖ Microsoft Azure ❖ Google Cloud ❖ OpenStack
Configuration and work with monitoring tools	❖ Nagios ❖ Zabbix ❖ Grafana ❖ Telegraf ❖ ELK stack (Elasticsearch, Logstash, Kibana)
Mobile testing tools	❖ TestFlight ❖ Android Studio ❖ ADB
Cross-browser testing	❖ BrowserStack

Test management, bug tracking, and project management systems	❖ TestRail ❖ Zephyr ❖ Jira ❖ Confluence ❖ Bitbucket ❖ Fisheye
Static analysis tools	❖ SonarQube for working with code quality metrics and setting up quality gates

Our QA engineers are ISTQB-certified specialists with experience in:

- Functional testing for desktop applications, Windows drivers, Linux kernel modules, and SaaS, web, and mobile applications
- Non-functional testing (performance, load, stress) for desktop and web applications
- Penetration and security testing for web applications
- Test automation and CI systems implementation
- Test environment configuration
- Test documentation creation
- Estimation of and reporting on testing activities
- Technical and marketing research
- Requirements testing
- Test process improvements with TPI Next and STEP
- Test process implementation and integration of testing processes into existing projects
- Communication with end users and provision of first- and second-level customer support

Our QA process is based on the ISTQB fundamental test process and includes these steps:

- Analyze requirements, acceptance criteria, and product risks, discuss additional questions, and create a design testing strategy and estimates based on this data
- Design product test cases and have a second Apriorit QA review them to verify coverage of requirements and risks
- Set up the required testing environment
- Perform testing at different levels: component, integration, and system
- Perform functional and non-functional testing
- Perform confirmation and regression testing for all affected features after bug fixes
- Perform acceptance testing according to the acceptance test suite to ensure all business requirements are met

- Perform test closure activities: prepare a report on testing results, archive tools and test environments, etc.

Our QA team provides a fully managed test automation process that includes these steps:

- Prepare an automation testing strategy from scratch depending on the specifics of the system under test
- Research and select the necessary toolsets and frameworks for test automation
- Design test cases for test automation
- Implementing automation code
- Perform code review of test automation code
- Measure automation coverage and other metrics using integrations with TestRail or the BDD/KDD framework to provide transparency

We base our product testing strategy on the ISTQB methodology, combining several of the listed strategy types depending on the specifics of the system under test:

1. Analytical test strategy (requirements-based, risk-based)
2. Model-based test strategy
3. Consultative test strategy
4. Reactive test strategy
5. Standard-compliant test strategy (We have experience with HIPAA and NIST standards.)
6. Methodical test strategy
7. Automation (or regression-averse) test strategy

We provide our clients with reports throughout the whole testing process. Reporting process is discussed at the start of a project, and reports can contain the following items:

- All developed test documentation: test strategy, estimates, test suites, test plans, test cases, and bug reports
- Weekly or/and daily status reports
- Time tracking in Jira
- Knowledge base items (wiki, instructions), environment and infrastructure descriptions, etc.