Modernizing POS Testing

WHITEPAPER

Gen AI powered automation and robotic validation with UST QE360 and SGBI Sastra

ust.com

Table of contents

Executive summary

Modern Point-of-Sale (POS) systems are mission-critical to retail, requiring advanced testing strategies that address both complex software workflows and specialized hardware integrations. By combining UST QE360's GenAI-powered automation with SGBI Sastra's Robots and RAF framework, organizations gain a comprehensive and scalable approach to validating POS ecosystems across digital transactions and physical touchpoints. This integrated solution accelerates testing cycles by up to 60–80%, reduces costs by 35%, and ensures reliable, end-to-end transaction processing across diverse retail environments.

The modern POS testing challenge

The evolution of POS Landscape

Point-of-Sale (POS) systems have evolved from simple cash registers to complex ecosystems integrating multiple hardware & software components. They now form the backbone of modern retail and banking operations connecting inventory, customer experience, payments, loyalty programs, and analytics into a single integrated environment. As global commerce shifts toward omnichannel experiences, the role of the POS has expanded from a transactional endpoint to a mission-critical digital touchpoint.

Recent years have seen rapid innovation across the POS ecosystem. Touchscreen-based human-machine interfaces (HMIs), mobile POS devices, self-checkout kiosks, smart card readers, biometric authentication, AI-driven recommendations, and cloud-connected payment services have transformed how businesses interact with customers. At the same time, rising expectations for seamless, zero-downtime transactions have increased the pressure on POS systems to be faster, more secure, and more reliable than ever. This evolution has introduced new layers of complexity. Today's POS solutions integrate a diverse mix of hardware components such as scale scanners, barcode scanners, touch screen displays, payment terminals, printers, NFC modules, cash drawers, and a growing stack of software elements including operating systems, payment applications, inventory management systems, third-party plugins, security layers, and cloud APIs. These components must operate flawlessly together in environments that vary significantly by store format, geography, network infrastructure, and regulatory requirements.

The industry's accelerated release cycles exacerbate this complexity. Frequent updates to payment software, security patches, UI changes, firmware upgrades, and new integrations are now routine. Retailers and solution providers face the challenge of verifying every update across multiple POS models and configurations, ensuring performance, compliance, accuracy, and a consistent customer experience across thousands of deployment points. This complexity introduces unique testing challenges that traditional automation approaches struggle to address effectively.

In this landscape, quality assurance has become not just a technical necessity but a business-critical function. A small UI glitch, a delayed tap-to-pay response, or a printer malfunction can directly impact revenue, customer satisfaction, and brand trust. Traditional manual testing methodologies, however, struggle to keep pace with the scale, diversity, and speed required by today's retail and payment ecosystems.

Current challenges in POS testing

Despite major advancements in retail technology, POS systems remain one of the most difficult platforms to validate. Their hybrid nature combining hardware, software, peripherals, cloud services, and multiple types of human interaction. That mix creates a testing landscape that is significantly more complex than typical enterprise applications. The following challenges highlight why POS testing demands new approaches that go beyond traditional manual or software-only testing tools.

1. **Fragmented hardware ecosystem**
   The POS ecosystem is highly diverse, with thousands of hardware variations deployed across global retail, and banking environments. This fragmentation includes:

   - Registers with touchscreen displays with different sizes, resolutions, and sensitivity levels running on Windows, Android, custom operating system/firmware.
   - Payment terminals from different makes, models, firmware versions.
   - Receipt printers from different makes, models & technology.
   - Barcode scanners from different make, models serving different purpose.
   - Cash drawers and customer displays.
   - Peripheral devices such as NFC readers, MSR/EMV modules, biometric sensors.
   - Country-specific devices mandated by local regulations or payment networks.

   Every retailer may use a unique combination of these components, and even the same brand may operate a different configuration across regions. As a result:

   - Test cases must be repeated on numerous hardware setups.
   - Hardware variations introduce unpredictable behavioral differences.
   - Maintaining test labs becomes expensive and operationally difficult.

   This fragmentation directly increases the complexity, cost, and time required to achieve adequate test coverage.

2. **Touchscreen and HMI challenges**
   Modern POS systems rely heavily on touch-driven human–machine interfaces operating in various operating system such as Windows, Linux, Android or custom firmware. Ensuring accurate and consistent HMI performance across devices introduces several challenges:

   - Software simulation tools may be able to simulate touches but they fail to produce simulated touches on touch screens in payment terminals or device that operate on custom operating systems.
   - Inconsistent Touch Sensitivity: Touch response varies by model, touch technology (capacitive, resistive), screen wear, and environmental conditions so even if we use conventional robotic arms the finger module need to be equipped with force sensing technology.
   - Latency Under Load: During peak hours or under heavy transaction loads, UI responsiveness may degrade, measuring this latency accurately requires repeatable, timed inputs something humans cannot reproduce with precision.
   - Multi-Touch and Gesture Validation: Many POS systems support multi-finger swipe, drag, long-press, pinch, and zoom gestures. Validating these interactions manually is inconsistent and difficult to automate without physical testing tools without multi fingers.

   These HMI challenges make human-repeatable validation nearly impossible without robotic assistance.

3. **Regression testing bottlenecks**
POS systems undergo frequent updates, security patches, feature additions, UI changes, compliance updates, and new payment flows. This results in:

Massive regression test suites: Enterprises often maintain thousands of scenarios covering:

- Login and billing flows
- Loyalty and coupon handling
- Payment flows and discounting methods
- Partial tendering with multiple payment methods
- Printer/display validations
- Localization
- Error and recovery states
- Peripheral compatibility
- Return and refund

Manual testing is slow and labor-intensive: Each release cycle requires executing these tests across multiple hardware configurations. Manual execution consumes significant time and resources, most of the time achieving this on a stringent timeline with a small number of testing team is nearly impossible.

Human fatigue and inconsistency: Humans cannot maintain consistent timing, accuracy, or repetition over large test cycles. This leads to:

- Inconsistent results
- Missed defects
- Difficulty reproducing issues
- Longer debugging cycles

Regression testing bottlenecks ultimately slow down release cycle and increase operational risk, a small unattended bug can create a multi-millions dollar loss in a few hours. Even with conventional software testing tools, we still need a human in the loop, which prevents us from running regression tests overnight or on weekends.

4. **OS dependency and limitations of software-only automation**
While POS systems increasingly rely on modern operating platforms such as Windows, Linux, and Android, many retail environments still operate a hybrid stack that includes iOS-based mobile POS devices and payment terminals running proprietary or custom operating systems. This diversity creates significant constraints for traditional software automation frameworks. Emulators don't replicate real hardware behaviors.

Tools such as Selenium, Appium, and other UI automation libraries are effective only when the underlying OS exposes accessibility hooks, debuggers, or automation interfaces. However, a large portion of POS hardware, especially payment terminals and secure transaction devices does not expose these interfaces for security and certification reasons. As a result, even the most advanced software-only automation approaches struggle to deliver complete coverage. Several key limitations emerge:

Time-consuming script creation: Developing automated scripts for moderately complex test cases (such as a 2-minute transaction flow) can take 30–45 minutes even for an expert engineer. For large enterprises with thousands of scenarios, script development becomes a significant bottleneck.

High knowledge dependency: Test script creation in conventional frameworks demands engineers with specialized skills in:

OS-level automation	UI element recognition	XPath structures
Payment device integration	Debugging complex UI flows	

This reliance on senior engineers increases cost and makes scaling the automation program difficult.

Poor maintainability: POS systems receive frequent updates, UI changes, OS patches, firmware upgrades, API changes, and security enhancements. Each update requires script revalidation and often script rewriting, creating recurring overhead that delays regression cycles and increases long-term maintenance cost.

Limited automation coverage: Even with extensive engineering effort, traditional software automation can cover only 30-40% of the full end-to-end POS workflow due to:

- OS fragmentation (Windows, Android, Linux, iOS, embedded OS)
- Proprietary payment terminal interfaces
- Unexposed secure-access layers
- Physical interactions such as card insertion, tap-to-pay, printing, scanning, touchscreen input

As a result, critical flows like EMV transactions, printer validations, barcode scanning, and real-time touch interactions remain unautomated using software-only methods.

BUSINESS IMPACT OF POS FAILURES

POS system reliability is not just a technical concern, it directly affects revenue, customer satisfaction, and brand reputation. Even minor disruptions in checkout processes can translate into measurable business losses. Studies show that 86% of consumers have left a store due to long checkout lines, and 77% are less likely to return to a store where they experienced checkout problems. Those numbers translate to real revenue impact, making POS reliability a bottom-line issue, not just an operational inconvenience. If your POS system crashes, you can't process transactions. Every minute your registers are down is a minute you're losing potential revenue.

Revenue loss & downtime impact: POS downtime has immediate financial consequences for retailers of all sizes:

- **Large retailers:** For every hour of POS downtime, sales losses can reach approximately **$5 million**, particularly during peak hours, promotional campaigns, or special events.
- **Small and mid-sized retailers:** Losses can range from **hundreds to thousands of dollars/ hour**, compounding quickly during busy periods such as holiday shopping or weekend peaks.
- **Transactional impact:** When registers fail, all in-store transactions pause, leading to abandoned carts, long queues, and frustrated customers who may leave without completing purchases.

Failure rates and industry trends: While detailed, industry-wide failure rates are not always publicly disclosed, research and case studies provide insight:

- **Estimated downtime impact:** POS failures or significant slowdowns are estimated to affect 1-5% of annual transaction time, depending on system age, infrastructure, and maintenance practices.
- **High-impact outages:**
  Large Global Coffee & QSR Chain (Publicly reported, 2015) - A large-scale POS outage disrupted operations across thousands of stores in multiple regions, forcing temporary closures. The incident resulted in significant revenue loss and demonstrated the financial exposure created by centralized POS architectures in high-volume retail.

  Global POS Platform – Multi-Region Payment Gateway Failure (Publicly reported, 2024) A global POS and payments platform experienced a multi-hour outage that prevented card transactions across regions. Merchants were forced into cash or limited offline modes, resulting in lost revenue, reconciliation challenges, & elevated operational risk.
- **Cumulative minor issues:** Frequent minor failures slow processing, app crashes, or peripheral errors can silently reduce sales performance by up to 10% per year when accounting for downtime, lost customers, and operational inefficiencies. Even 1-4 hours of card-processing downtime across a large chain means thousands to millions in lost sales, plus labor inefficiency and manual rework.

Additional business consequences: Beyond immediate revenue loss, POS failures carry broader business risks:

- Operational costs: Additional labor may be required to manage queues, troubleshoot systems, or manually process transactions.
- Customer churn and brand reputation: Repeated issues can drive long-term customer attrition and damage brand trust.
- Delayed business growth: Persistent downtime limits store throughput and slows overall operational scalability.

Implications for retailers: Beyond immediate revenue loss, these outages erode customer trust and can push shoppers to competitors, especially when they coincide with peak times (morning commute, lunch rush, evenings).

Compliance risks: Payment data security vulnerabilities expose businesses to regulatory penalties

Payment data security vulnerabilities around POS systems create serious compliance risk because they sit at the intersection of PCI DSS, data protection laws (like GDPR/CCPA), and card-brand rules

- **PCI DSS penalties:** If a business is not PCI compliant, banks and card networks can impose monthly fines, often ranging from thousands to tens of thousands of dollars, until compliance is restored. After a breach, additional per card penalties may apply, which can quickly add up when large volumes of card data are exposed.
- **Data protection fines:** Regulators can impose significant fines for failure to protect personal and payment data. Under GDPR, penalties can reach up to 4% of global annual revenue, while US regulations such as CCPA allow fines for each violation, adding further financial exposure.
- **Compounding impact:** These penalties are additive, not alternative. A single POS related card data breach can trigger PCI fines, card network charges, regulatory penalties, legal costs, and remediation expenses simultaneously.

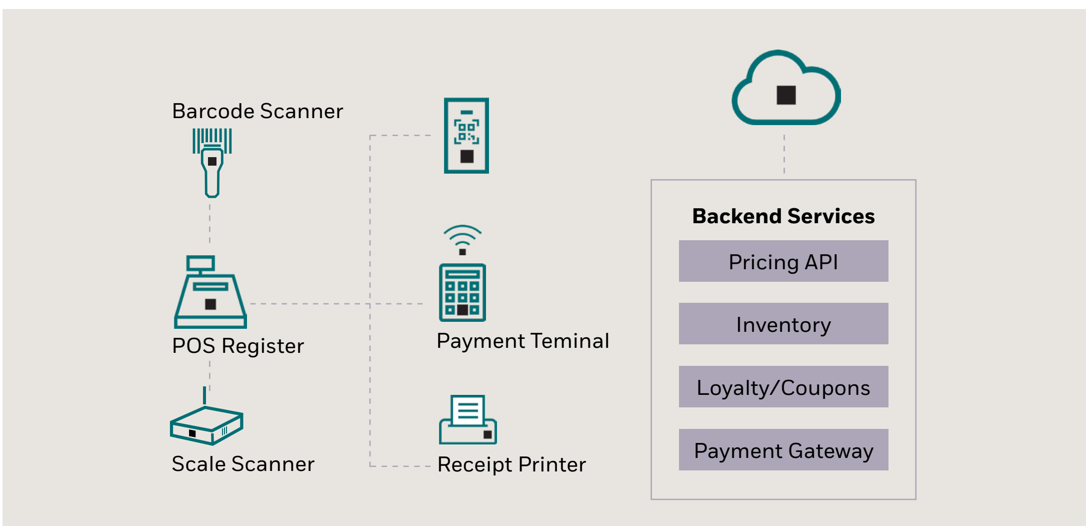
Problem areas and key challenges

HARDWARE + SOFTWARE INTEGRATION

Challenge: POS environments require seamless coordination between software applications and numerous physical devices (barcode scanners, printers, card readers, cash drawers). Traditional automation tools struggle to validate these end-to-end hardware-software interactions, creating gaps in test coverage that can lead to production failures.

Point-of-Sale (POS) systems operate with a combination of software applications and a large ecosystem of physical retail devices. A typical checkout experience depends on seamless coordination between multiple components, barcode scanners, scale scanners, receipt printers, card payment terminals, cash drawers, customer-facing displays, and back-end services. Ensuring that all these elements work together reliably is one of the most complex challenges in retail technology quality assurance. Most traditional software automation tools are designed to test only on-screen digital workflows. They cannot validate end-to-end interactions that involve physical devices, signal exchanges, sensor inputs, and human-machine interface (HMI) behavior. As a result, significant gaps exist in test coverage, leading to real defects appearing only in production, where the business impact is immediate and costly.

Example: self-checkout system: A Highly Complex Test Environment A modern self-checkout (SCO) station is an ideal example of how complex a POS testing environment can be. A typical SCO system includes multiple hardware components tightly integrated with a POS application running on a retail-grade terminal.



Scanners and scale-scanners

- Barcode/QR Scanners: These devices require physical stimuli such as barcodes, QR codes, membership codes, coupons, or loyalty cards. Simulating each input consistently requires robotic or programmable hardware.
- Scale Scanners (Bioptic Scanners): These require not only barcode/QR input but also weight simulation, matching expected product profiles to detect theft or mis-scans. Examples: NCR, Toshiba, Diebold Nixdorf bioptic scanners

POS register / retail terminal
The scanner is connected to a POS register, where the main POS software runs. These registers typically run:

- Windows (most common in retail)
- Linux variants (Ubuntu, Yocto-based distributions, SUSE)
- Android (in newer lightweight POS terminals)

The user interface may be:

- Touchscreen, requiring precise touch simulation and gesture accuracy, or
- Non-touch display, requiring keyboard, keypad, or physical button input.

Testing this layer requires:

- Robotic touch inputs
- Simulated keyboard entry
- Vision-based validation of UI states, animations, element placement, item lists, price changes, discount application, and error prompts.

Payment terminal integration
Once items are scanned and totals calculated, the customer interacts with a payment terminal. This introduces additional complexities:

Common payment terminal types:

- Ingenico (Tetra OS, Android)
- Verifone (V/OS or Verifone Secure OS, now Android-based in new models)
- PAX (Android)
- Castles (Linux or Android)

Payment methods supported:

- Magstripe swipe
- Chip insert (EMV)
- NFC tap
- Digital wallets (Apple Pay, Google Pay, Samsung Pay)
- QR-code based payments (e.g., Paytm, Alipay, WeChat, UPI apps)

Validating partial tenders (e.g., split cash + card) or failure scenarios requires physical simulation of card actions, NFC triggers, or interactions with smartphone/wearable devices:

- Android devices
- iOS devices
- Smartwatches (Apple Watch, WearOS)

Each of these interactions includes timing, retries, card-reader states, PIN entry, and edge-case conditions that traditional automation cannot reliably simulate.

Receipt printing
After payment is completed, the system generates a receipt through one of many printer types:

- Thermal printers (Epson, Star Micronics, Zebra, Custom)
- Different widths (58mm, 80mm)
- Various print speeds and fonts

Validating printed output requires:

- External camera vision
- Optical character recognition (OCR) to validate characters, alignment, barcode correctness
- Physical inspection for cut length, clarity, and receipt branding

Backend service integration
Behind the POS terminal lies an extensive ecosystem:

- Pricing APIs
- Inventory services
- Loyalty/cloud coupon systems
- Payment gateways
- Tax calculation engines
- Settlement & reconciliation services
- Fraud-detection systems

Testing these requires:

- Real-time API validation
- Network condition simulations
- Data consistency checks across multiple systems

When combined with front-end physical interactions, this makes POS QA one of the most layered testing environments in enterprise technology.

Why this environment is exceptionally hard to test
To validate a complete checkout flow, the test system must coordinate: Physical device actions Human-machine touch inputs, vision analysis, payment terminal behaviors, Network and backend API responses, etc. Conventional automation is limited to the UI layer and cannot:

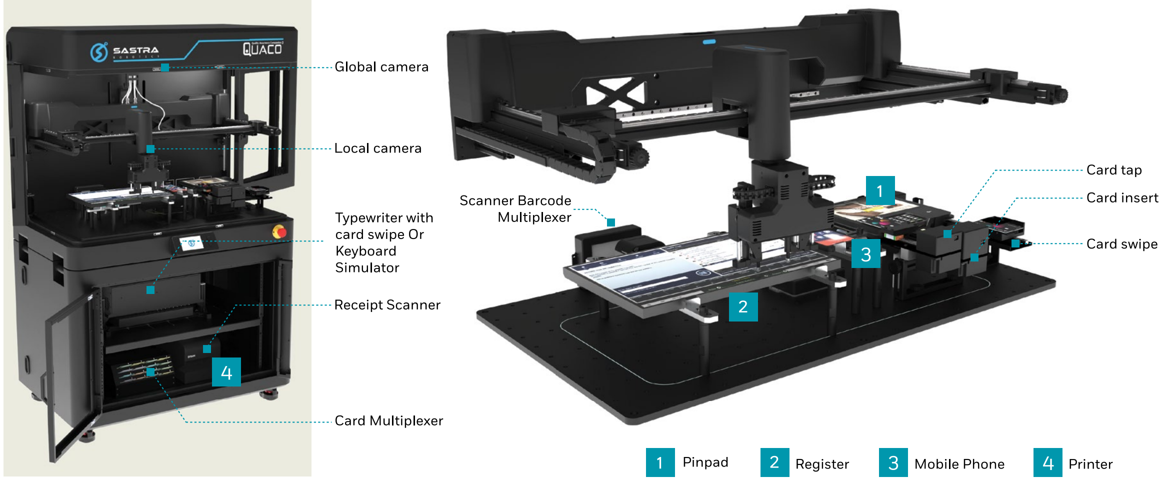
- Generate physical scanner inputs
- Simulate EMV card actions
- Validate printed receipts
- Recognize real-world visual states
- Orchestrate hardware-device timing

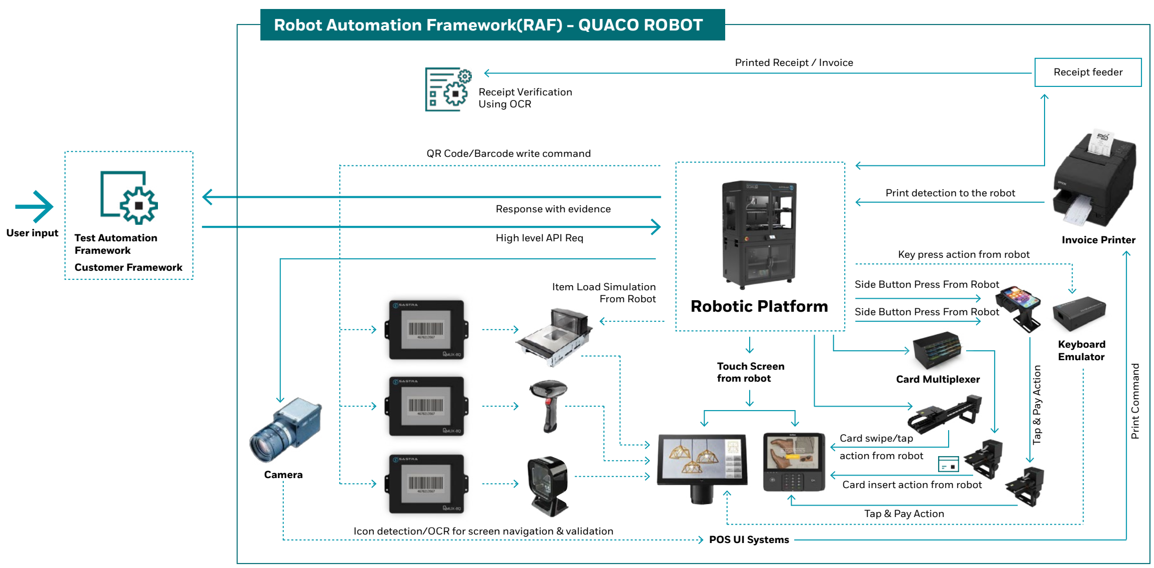
This is why retail enterprises spend millions annually resolving failures that could have been prevented with proper robot-aided, AI-driven end-to-end automation.

Solution approach (SGBI Robots + UST QE360 GenAI): SGBI Sastra Robots physically simulate hardware actions(Keypad touch, ... scanning, printing, card swiping) while UST QE360 orchestrates comprehensive software automation flows. This integrated approach enables true end-to-end validation where robotic systems execute physical device interactions while UST QE360 validates software responses, data flow, and business logic simultaneously.

POS testing

Device Arrangement





Sastra Robotics (SGBI)'s QUACO AI Studio Robotic Platform + UST QE360 GenAI platform together is capable of simulating every real-world action required in a modern POS or Self-Checkout environment, including:

- Barcode and QR-code scans in front of standard scanners and bioptic scale-scanners
- Weight simulation for scale-scanner
- Touch inputs for the registers and other device under consideration (physical & emulated)
- Vision-based validation of UI states
- Physical simulation of card actions(tap, insert & swipe) & PIN entry
- NFC triggers using card, or interactions with smartphone/wearable devices
- Physical button press action in smartphones for digital wallet trigger
- QR-based mobile payments using various smart device running different OS
- Validating printed output (receipts/coupons etc) using external camera vision system

- OCR to validate the content/message on screen or physical prints
- Real time API validation
- Network condition simulation
- Data consistency checks across multiple systems

CI/CD INTEGRATION

Challenge: Integrating POS devices into continuous integration pipelines presents significant obstacles. Physical hardware provisioning is complex and expensive, feedback cycles are slow, and maintaining compliance with payment security standards (PCI-DSS) while handling test data creates additional complexity.

Solution approach (SGBI Robots + UST QE360 GenAI): UST QE360's CI/CD plugins The platform integrates hardware simulators for payment processing and inventory management while enforcing PCI-compliant test data handling. GenAI capabilities automatically generate secure test scenarios and maintain

REPORTING AND INSIGHTS

Challenge: POS testing generates scattered logs across multiple devices and systems, making failure triage difficult and obscuring business impact analysis. Traditional reporting lacks correlation between hardware events, software responses, and transaction outcomes.

Solution approach (SGBI Robots + UST QE360 GenAI): UST QE360's Unified Dashboard aggregates logs from both hardware robots and software components, providing comprehensive visibility into test execution. The platform includes video playback capabilities for physical robot actions and leverages GenAI for automated root cause analysis. Business impact assessment links technical failures to revenue implications and customer experience metrics.

PARALLEL EXECUTION

Challenge: Limited availability of physical POS devices constrains test scalability, while parallel execution creates database conflicts and test data management issues. This bottleneck significantly impacts testing efficiency and release velocity.

Solution approach (SGBI Robots + UST QE360 GenAI): UST QE360's parallel orchestrator implements hybrid execution strategies combining real devices, simulators, and intelligent mocks. Data virtualization capabilities prevent backend conflicts by creating isolated test environments. The platform dynamically allocates resources between physical robots and virtual components based on test requirements and device availability.

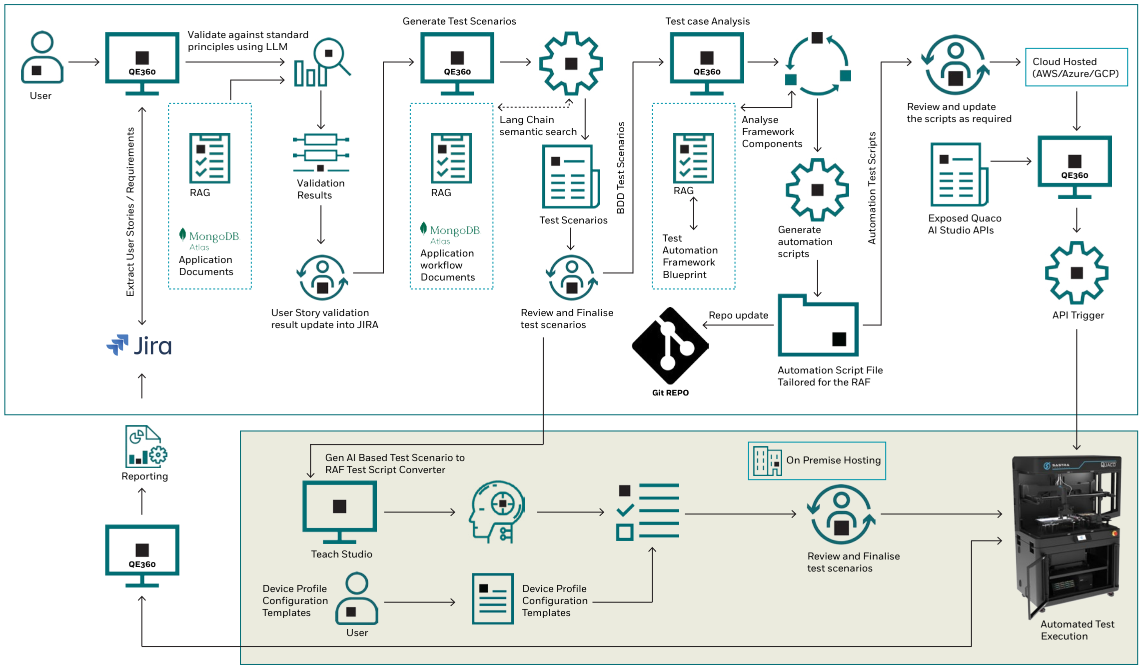
SCALABILITY

Challenge: Adding new POS hardware types or business workflows traditionally requires extensive re-engineering of test automation frameworks. This creates maintenance overhead and slows adoption of new technologies or business processes.

Solution approach (UST QE360 +GenAI + SGBI Robots): UST QE360's model-based approach treats new workflows as reusable components rather than custom scripts. The platform's extensible architecture seamlessly accommodates new hardware types through standardized integration patterns. SGBI robots support diverse.

Accelerating intelligent test automation

QE360 – QUACO Integrated Workflow



UST's QE360 and SGBI's QUACO RAF (Robot Automation Framework) leverage advanced GenAI capabilities to transform the way functional and hardware validation testing is created, executed, and scaled. UST QE360 automatically generates test cases and user stories directly from user instructions or from existing customer assets such as manual test scripts, SOP documents, user manuals, release notes, and change logs. UST QE360 can also integrate with the customer's existing tools to fetch structured or unstructured data, convert them into coherent user stories, and pass these as standardized inputs to the QUACO system. Even when no prior documentation exists, users can simply describe scenarios in plain English, such as:

"Write a test case to verify one sales transaction using partial tendering. Log in to the POS system with <credentials> as a cashier, add three items—scan a barcode for the first, use the lookup table for the second, manually enter the item number for the third—apply a 30% discount, then use partial tendering: insert a credit card for 50% of the bill and tap a connected digital wallet for the remaining amount. Validate the receipt."

From such natural-language instructions, UST QE360 dynamically generates:

- A complete user story
- A structured automation workflow
- Pre-test conditions
- Validation criteria

This output is written in simple, step-by-step human-readable form, allowing users to review or edit the logic like an expert test architect. With a single click, the system converts the workflow into executable test scripts using GenAI and automatically creates the APIs required to drive QUACO robots and connected accessories.

During execution, QUACO RAF performs actions exactly like a human operator. It uses a multi-camera vision system to observe the devices, understand the current screen state, and make autonomous decisions about where to touch, how long to wait, what to validate, and which evidence or screenshots to capture. Every action and observation is presented back to the user for proofreading. Once approved, the workflow is saved as a reusable regression test.

Because the robot "witnesses" the system behavior and understands expected output references, it can immediately detect abnormalities or deviations during future regression cycles.

In addition, RAF interprets POS wireframes and validates UI elements across physical touchpoints while simulating hardware actions with high precision. Human-in-the-loop review ensures accuracy, business alignment, and transparent decision-making. Once the dry-run is completed and validated, the finalized test scripts are automatically pushed into a centralized repository, enabling seamless CI/CD integration and highly scalable automation.

TEST ORCHESTRATION ENGINE

- **UST QE360/RAF GenAI script generation:** Automatically converts business requirements into executable test scenarios covering both software workflows and hardware interactions
- **Intelligent object recognition:** AI-powered element identification adapts to interface changes and maintains test stability
- **Multi-channel validation:** Simultaneous testing across web, mobile, and API interfaces with hardware integration points
- **RAF physical device simulation:** Precise robotic manipulation of barcode scanners, card readers, printers, and touchscreen interfaces.
- **AI-powered vision system:** Computer vision validates device responses and captures evidence of physical interactions.
- **Hardware abstraction layer:** Standardized interfaces enable testing across diverse POS hardware configurations.

UNIFIED EXECUTION PLATFORM

- **API-driven integration:** UST QE360 triggers SGBI robot actions through standardized APIs, enabling seamless workflow orchestration
- **Real-time synchronization**: Coordinated execution ensures software and hardware actions occur in proper sequence with appropriate timing
- **Comprehensive evidence collection:** Combined logging captures software events, robot actions, and device responses for complete traceability

TEST ARCHITECTURE AND FLOW

Step	Description	Key Technologies
1	GenAI designs and scripts POS test flows	UST QE360 GenAI Engine
2	Scripts integrated into Test Management	JIRA, Zephyr, ADO, Test Rail
3	SGBI RAF executes physical device actions	SGBI Platform, Robot arms, Multiplexers, OCR Vision camera, Emulators
4	Results logged and reported centrally	UST QE360 Analytics Dashboard

Measurable business benefits

Efficiency and cost

60%
Faster test cycles
Automated and parallel execution

30%
Lower QA costs
Reduced manual effort, better defect detection

40%
Quicker feedback
Early integration and continuous monitoring

Quality and compliance

80%
Less test optimization effort
AI-driven generation and maintenance

100%
Hardware coverage
Robotic validation of device interactions

Automated compliance
PCI-DSS validation & secure data handling

Conclusion

The convergence of UST QE360's GenAI-powered automation platform with SGBI Sastra's robotic testing framework represents a paradigm shift in POS quality assurance. This integrated solution addresses the fundamental challenge of comprehensive hardware-software validation while delivering unprecedented efficiency gains and quality improvements.

By unifying intelligent test automation with physical device validation, organizations can achieve:

Complete test coverage:
End-to-end validation spanning software logic, hardware integration, and real-world usage scenarios

Enhanced reliability:
Comprehensive validation ensures consistent POS performance across diverse operational environments

Accelerated release cycles:
Automated test generation and parallel execution dramatically reduce time-to-market

Future readiness:
Extensible architecture accommodates evolving payment technologies and business requirements

The retail and payment processing industries face increasing pressure to deliver flawless customer experiences while adopting emerging technologies. Traditional testing approaches cannot adequately address this complexity, creating risks that impact revenue, compliance, and customer satisfaction.

UST QE360 and SGBI Sastra's integrated solution provides the comprehensive, scalable, and intelligent testing platform necessary to navigate this complexity successfully. Organizations implementing this approach will gain significant competitive advantages through superior product quality, faster innovation cycles, and reduced operational risks.

As POS systems continue evolving toward greater complexity and integration, the need for sophisticated testing approaches will only intensify. The UST QE360-SGBI partnership establishes a foundation for sustained quality excellence that will serve organizations well into the future of retail and payment processing.

Authors

With 20+ years of experience in enterprise Quality Engineering transformation. He leads the vision and execution of UST QE360, an AI- and Agentic QE-driven platform enabling intelligent, scalable test automation across complex systems.

Prabhu has driven QE and delivery transformations across the US, UK, and Australia for banking, technology, consumer, and telecom organizations, with deep expertise in aligning AI-powered quality strategies to business outcomes.

Prabhu Rajendra Babu
Group Product Owner - QE Services
UST

A technology innovator and entrepreneur driven by a lifelong passion for building intelligent systems that augment human capabilities. With a background in Electronics and Communication Engineering, he specializes in robotics and human machine interaction.

He is a co - founder of Sastra Robotics India Pvt. Ltd., where he focuses on developing practical robotic solutions that replicate human touch and precision to solve real world testing and automation challenges

Aronin Ponaappan
CEO, Co-Founder
SGBI

Together, we build for boundless impact

Since 1999, UST has partnered with the world's leading companies to create a powerful impact through transformation. Powered by technology, inspired by people, and guided by its purpose, UST collaborates with clients from design to operation. The company's digital solutions, proprietary platforms, engineering, R&D, products, and innovation ecosystem transform core challenges into disruptive, impactful solutions. With deep industry expertise and a future-ready mindset, UST infuses innovation and agility into its clients' organizations, delivering measurable value and lasting positive change for them, their customers, and communities worldwide. Together with 30,000+ employees in more than 30 countries, UST builds for boundless impact, touching billions of lives in the process.

Digital Solutions I Platforms I Engineering, R&D, and Products

ust.com

Version 0103-20260330

U · S T