Pragati Gupta

+1-514-754-9774 | guptapragati1990@gmail.com | github.com/pragatig25 | Montreal, QC

Experienced QA Engineer & AI Test Automation

PROFESSIONAL SUMMARY

QA engineer with 10+ years of catching the failures nobody else saw — across aerospace and defence simulation, enterprise software, and cloud platforms. I have spent that decade building structured test programs, automating mission-critical regression suites, and shipping CI/CD pipelines in Azure DevOps for environments where "good enough" was never enough.

My focus now is AI-native quality engineering. As teams build and ship faster with GenAI, the hard problem has shifted from writing tests to validating non-deterministic systems — and that is exactly where I am investing: evaluation harnesses, guardrails, and quality gates that let teams move fast without quietly shipping defects.

Recent open-source work — Qalibur (multi-agent QE pipeline), DriftGate (visual regression as a CI gate), Spectra (AI test generation from OpenAPI specs), TerraGuard (Terraform security regression gate), and a self-healing Playwright framework — is the proof point: the same rigour that kept safety-critical software shipping can absolutely move at AI speed.

SKILLS & TOOLS

- **Test Automation & Performance:** Playwright, pytest, Cypress, Selenium, Cucumber (BDD), JMeter, Self-Healing Selectors
- **Programming Languages:** Python, TypeScript, JavaScript, SQL, Java, Bash
- **API & Contract Testing:** FastAPI, REST, OpenAPI 3.x, Postman, JIRA
- **Cloud & DevOps:** AWS (basic), Azure DevOps, Docker, GitHub Actions, CI/CD
- **Quality & Compliance:** ISO 9001:2015, Risk-Based Testing, OWASP, CAR/PAR, Agile
- **GenAI & LLM Tooling:** Claude (Sonnet · Haiku · Vision), LangChain, LangSmith, Pydantic, Prompt Engineering
- **AI Quality & Evaluation:** LLM Evaluation, LLM-as-a-Judge, Multi-Agent Testing, AI Quality Gates & Guardrails

PORTFOLIO PROJECTS

DriftGate — Visual Regression as a CI Gate | *Playwright · Claude Vision · FastAPI* GitHub · Demo

- Replaced manual visual QA (~15–30 min per PR) with a sub-60-second automated gate — an agentic loop that renders every front-end change in a real browser, scores it against a design system, and proposes CSS fixes until the screen conforms; shipped as a GitHub Actions CI gate plus a published live demo.
- Designed a hybrid gate so a non-deterministic model can never wrongly fail a build: deterministic pixel diff and design-token assertions are the hard pass/fail, while the Claude-vision critic stays advisory — holding cost near ~$0.01 per run through prompt caching of the design system.

Spectra — AI Test Generation Platform | *LangChain · FastAPI · Pydantic · Docker · CI/CD* GitHub

- Reduced test authoring from hours to seconds by building an end-to-end AI platform that parses OpenAPI 3.x specs, scores each endpoint by risk tier (Low/Med/High/Critical) using a LangChain agent, and generates four test categories — happy path, boundary, auth bypass, malformed input — with Pydantic-enforced output schemas guaranteeing zero malformed test cases in CI.
- Cut redundant generated tests by 40%+ by implementing cosine similarity deduplication (threshold 0.92 via text embeddings), preventing test suite bloat before execution and keeping pipeline runtimes lean.
- Delivered full LLM observability — prompt, latency, and token cost per run — via LangSmith tracing, giving teams a direct line to AI spend and enabling prompt optimization across releases.

Self-Healing E2E Test Platform | *Playwright · Node.js · LLM · Next.js · CI/CD* GitHub

- Eliminated manual selector debugging entirely by building an autonomous framework that detects broken Playwright locators, queries an LLM to analyse the live DOM, generates a repaired selector, and files an auto-PR — reducing mean time to fix from hours of manual investigation to zero human intervention per selector failure.
- Shipped a real-time Next.js dashboard showing healed selector history, per-run metrics, and CI pipeline status across stable and breaking application versions simultaneously.

Qalibur — Ten-Agent QE Pipeline | *Claude Sonnet · Haiku · TypeScript · Node + Express · React · GitHub API* GitHub · Demo

- Built a 10-agent pipeline that owns the full QE lifecycle — repo walk, risk strategy, Gherkin authoring, Playwright code, PR creation, CI dispatch, failure triage — with a Gatekeeper agent scoring every handoff at ≥8.0/10 or retrying upstream up to 3× before escalating, so non-deterministic models never gate the merge.
- Cut model spend by routing comparison-heavy stages (Gatekeeper, Architect, Herald) to Claude Haiku 4.5 with a prompt-cached rubric, reserving Sonnet for synthesis stages — live-mode run cost ≈ $0.04 end-to-end including PR creation and CI dispatch.

- Made every artefact auditable by recording its agent, parent artefact, gate score, and attempt number, then rendered the full chain as a self-contained HTML report — turning an agentic pipeline into something a reviewer could actually inspect end-to-end.

TerraGuard — Terraform Security Regression Gate | *Checkov · tfsec · Trivy · pytest · Claude Haiku · GitHub Actions · Pydantic*
GitHub · Dashboard

- Cut Terraform security regressions from invisible to PR-blocking by building a CI pipeline that scans every plan with Checkov (CIS/NIST-mapped) + tfsec + Trivy, diffs findings against the main-branch baseline to isolate true regressions, and gates merge via pytest invariants per domain (networking, IAM, storage, compute, database, logging, secrets).
- Closed the feedback loop with a Claude-driven triage stage that scores exploitability and CIS/NIST mapping per regression and emits a minimal HCL auto-fix PR for remediable findings — turning a noisy scanner dump into an actionable PR comment with sanitised IPs, CIDRs, ARNs, and account IDs.
- Shipped a public GitHub Pages dashboard publishing the posture-score delta on every run (real LIVE data on main, synthetic in DEMO mode) — surfacing the regression-per-PR number platform teams actually react to, not the absolute finding count they had stopped reading.

PROFESSIONAL EXPERIENCE

Software Test Engineer — CAE Canada (Presagis)
Montreal, QC | March 2018 – April 2026

- Built and validated CI/CD test pipelines end-to-end in Azure DevOps — PR-gated smoke and regression stages on every commit, release pipelines orchestrating full automation across simulation modules — replacing manual test handoffs with continuous, on-every-build validation.
- Converted a fully manual regression process into a repeatable automated framework by building a Selenium WebDriver and Java suite targeting the mission-critical simulation software modules, then wiring it into Azure DevOps so the same automation ran identically on every product release.
- Kept automation pipeline reliability high across 5 application teams in Azure DevOps — diagnosing flaky tests, parallelising slow stages, and tuning agent pools — so build-and-test cycles stayed short enough that developers actually waited for green before merging.
- Strengthened the product security posture by integrating Nessus vulnerability scan testing into the CI pipeline, translating raw scan outputs into prioritised, developer-ready action items so security regressions surfaced on PR rather than at release.
- Kept ISO 9001:2015 and CMMC quality gates code-enforced rather than process-policed by embedding the required checkpoints directly into the Azure DevOps pipelines engineering teams already ran — and led CAR/PAR root-cause reviews on the failures that still slipped through, closing recurring defect patterns at source.

QA Analyst — Teleperformance Canada
Montreal, QC | May 2016 – January 2018

- Improved service quality consistency across a 50+ agent team, as tracked through monthly QA monitoring reports, by evaluating 200+ customer service interactions per month and delivering structured coaching feedback within 2-week turnaround cycles.
- Reduced go-live defect count on SAP CRM e-commerce deployments by overhauling QA integration gates before each production release, catching functional and CRM workflow issues that had previously reached customers.

QA Engineer — Infostretch Corporation Pvt. Ltd.
Ahmedabad, India | January 2015 – December 2015

- Prevented a live security incident, avoiding a potential third-party data exposure event, by identifying an undiscovered payment redirect flaw across 3 checkout pages during manual testing before the release went to production.
- Brought defect detection upstream by transitioning the engineering team to Test-Driven Development (TDD), shifting quality ownership to the front of the cycle and reducing the defect load entering the QA queue.

QA Tester — VISCAR Education LLP
Chandigarh, India | January 2014 – December 2014

- Delivered 100% planned test execution across 3 client education software releases by building the QA practice from scratch — designing and running system, regression, performance, and load test suites that gated each release before it reached the client.

EDUCATION

Masters in Automation Testing
SimpliLearn, Online | Oct 2021 – April 2023

Masters of Engineering, Quality Systems Engineering
Concordia University, Montreal, QC | Jan 2016 – May 2018

Masters in Computer Applications
HP Institute of Management Studies, Shimla, India | Aug 2011 – Dec 2014

Bachelors in Computer Applications
St. Bede's College, Shimla, India | July 2008 – June 2011