

Sample Resumes: <https://www.h2kinfosys.com/sample-resumes/>

Forums: <https://forums.h2kinfosys.com/>

WhatsApp Community: <https://whatsapp.com/channel/0029VaEaWUCLo4hafm4drj3F>

Customized ATS & AI Résumé Tips — QA, ETL, API, Mobile, and Automation (Student/Early-Career)

Goal: Make your résumé parse cleanly in ATS and clearly prove your skills in Manual + Automation Testing, including ETL, API, Mobile, and Performance, with AI-assisted test generation.

Keep it 1 page when possible; 2 pages only if you have strong, recent, and relevant experience.

Core Formatting (ATS-safe)

- Single-column layout; standard section headers (Summary, Skills, Projects, Experience, Education, Certifications).
- Fonts: Arial/Calibri/Times New Roman 10.5–12 pt; 1.0–1.15 line spacing.
- Avoid tables/text boxes/graphics; keep essential info out of headers/footers.
- Save as .docx (and PDF only if allowed; ensure text-based PDF).

Keyword Strategy

- Mirror exact terms from the job post truthfully (e.g., “Selenium WebDriver,” “Cucumber BDD,” “SQL joins,” “ETL testing”).
- Put priority keywords in both SKILLS and relevant PROJECT/EXPERIENCE bullets (with context and outcomes).
- Use synonyms when relevant: “defects/bugs/issues,” “test plan/strategy,” “test cases/scenarios,” etc.

Manual + Automation Blend (how to show it)

- Start your SUMMARY with “Manual + Automation QA” and specify your strongest stacks (Selenium-Java, TestNG, Cucumber).
- In PROJECTS, include both: (1) manual test design/execution and (2) automation coverage + CI integration metrics.
- Quantify: % test coverage automated, regression time saved, defects caught pre-release, performance baselines, etc.

AI-Assisted Testing (what to highlight)

- Tools: ChatGPT, GitHub Copilot, and AI agents for:
 - Generating boundary/equivalence test ideas, edge cases, and negative scenarios.
 - Drafting Selenium step definitions/TestNG boilerplate; refactoring Page Objects.
 - Auto-generating synthetic test data and SQL joins for ETL validations.
 - Drafting API test cases and assertions for REST/GraphQL; fuzz ideas.
- Always review and validate AI-generated output; cite improvements (e.g., “reduced authoring time by 40%”).

QA/ETL/API/Mobile/Performance: Bullet Examples

- Web UI: “Automated 120+ smoke/regression tests in Selenium-Java (POM, TestNG); failures surfaced in CI per PR; cut regression cycle from 2 days to 6 hours.”

- BDD: “Defined Cucumber feature files and step defs; aligned scenarios with acceptance criteria; traceability in Zephyr/qTest.”
- API: “Built REST API test suite with [tool of choice]; validated status codes, response schemas, and DB effects; caught 12 Sev-2 defects pre-release.”
- ETL/Data: “Created SQL join checks to reconcile source→warehouse; implemented row-count, null, and referential integrity validations; 0 data drift at launch.”
- Unix/Linux: “Used shell (grep, awk) to parse logs, verify cron ETL runs, and triage failures; wrote reusable scripts.”
- Mobile: “Designed smoke/regression on Android/iOS; automated app flows (if using Appium) and tracked defects in Jira.”
- Performance: “Modeled user journeys; executed JMeter tests (thread groups, timers); established baseline TPS + latency; flagged 95th-percentile regressions.”
- STLC/SDLC: “Participated in Agile ceremonies; authored test plans/cases; ensured traceability from user stories to test results and defects.”

Tooling & Platforms (include only what you can discuss)

- Test Types: Functional QA, Regression, Smoke, UAT, Sanity, ETL Testing, API Testing, Mobile App Testing, Performance Testing.
- Automation: Selenium (Java), TestNG, Cucumber (BDD), Page Object Model, Maven/Gradle.
- Test Management: Jira, Zephyr, Tricentis qTest; Defect tracking and triage workflows.
- Model-Based/Enterprise: Tricentis Tosca (if applicable).
- Performance: JMeter (HTTP/REST, correlation, assertions).
- Databases/SQL: SQL queries, SQL joins, stored procedures; data quality checks.
- OS/Scripting: Unix/Linux, bash; log analysis; basic networking.
- Process: Agile/Scrum, SDLC, STLC; test plans, test cases, defects; traceability.
- AI: ChatGPT, GitHub Copilot; AI agents for test idea generation, data generation, and code scaffolding.

ATS Keyword Bank (mix honestly; do not stuff)

QA testing; Manual testing; Automation testing; Selenium; Java; TestNG; Cucumber; BDD; Page Object Model; Jira; Zephyr; qTest; Tosca; JMeter; API testing; REST; GraphQL; ETL testing; SQL; SQL joins; Unix; Linux; STLC; SDLC; test plan; test case; defects; Agile; Mobile app testing; Appium (if used); Git; CI/CD; Jenkins; ChatGPT; GitHub Copilot; AI test generation; synthetic test data; test automation with AI; boundary testing; negative testing.

Pre-Apply Checklist (QA-focused)

- ☐ Contact + LinkedIn + GitHub clickable ☐ Single column ☐ Standard headings
- ☐ “Manual + Automation QA” in Summary ☐ Keywords from job post reflected truthfully
- ☐ Projects show Web/UI + API + ETL or Mobile (at least two) ☐ At least 3 quantified outcomes
- ☐ Test management + defect tracking (Jira + Zephyr/qTest) mentioned ☐ AI usage value-add stated
- ☐ Saved as .docx; professional file name: First_Last_Resume.docx

Sample Resumes: <https://www.h2kinfosys.com/sample-resumes/>

Forums: <https://forums.h2kinfosys.com/>

WhatsApp Community: <https://whatsapp.com/channel/0029VaEaWUCLo4hafm4drj3F>