🧠 End-to-End Performance Testing – RACI Breakdown (High Level)

🗂️ PHASE 1: Requirement & NFR Gathering

Activity	Perf. Tester	Perf. Architect	Developer / Tech Lead	Infra / Ops	QA Lead	Product Owner	Notes
Stakeholder interviews	C	A	I	I	A	R	Capture use cases, business flows
Capture SLAs, SLOs, performance KPIs	R	A	C	C	A	C	e.g., response times, throughput
Risk identification & assumptions	R	A	C	C	A	I	Define test boundaries

---

🛠️ PHASE 2: Planning & Strategy

Activity	Perf. Tester	Perf. Architect	Developer / Tech Lead	Infra / Ops	QA Lead	Product Owner	Notes
Workload modeling (concurrent users, pacing)	R	A	C	C	A	C	Align with production usage
Test type planning (Load, Soak, Spike, etc.)	R	A	C	I	A	I	Define objectives per test type

Tools selection (JMeter, Gatling, k6, etc.)	R	A	I	I	A	I	Align with test goals and tech stack
Data modeling for test scenarios	R	A	C	C	A	I	Include think time, randomness

---

💻 PHASE 3: Script Design & Test Assets

Activity	Perf. Tester	Perf. Architect	Developer / Tech Lead	Infra / Ops	QA Lead	Product Owner	Notes
Script creation (JMeter, Gatling, LoadRunner)	R	A	C	I	I	I	Use correlation, parameterization, assertions
Dynamic token handling (SSO, OAuth, JWT)	R	A	C	I	I	I	Script advanced auth flows
Error handling logic & test validations	R	A	C	I	I	I	Think time, retries, custom JS/Groovy
Version control & integration (Git, CI/CD)	R	A	C	I	A	I	CI/CD-ready, versioned scripts

---

🧪 PHASE 4: Test Environment Setup

Activity	Perf. Tester	Perf. Architect	Developer / Tech Lead	Infra / Ops	QA Lead	Product Owner	Notes
Env. provisioning (EC2, K8s, VM, SAP, etc.)	I	C	I	R	A	I	Ensure parity with prod
APM agents installation (Dynatrace, NR, etc.)	I	A	I	R	I	I	Validate metrics visibility
GC log enabling, JVM flags tuning	C	A	R	C	I	I	Enable -Xloggc, GC viewer flags
Infra tuning (ulimits, network, threads)	I	C	C	R	A	I	Readiness checks

---

🚀 PHASE 5: Execution

Activity	Perf. Tester	Perf. Architect	Developer / Tech Lead	Infra / Ops	QA Lead	Product Owner	Notes
Dry run / sanity test	R	A	I	R	A	I	Baseline verification
Distributed test orchestration (Master/Slave)	R	A	I	R	I	I	Use JMeter with EC2 or K8s
Monitoring dashboards (Grafana, Prometheus)	C	A	I	R	I	I	Ensure system metrics visibility

Full load / spike / stress test	R	A	C	R	A	I	Controlled ramp-up, threshold gating
Custom timers, pacing, rendezvous points	R	A	I	I	I	I	Simulate realistic user concurrency

---

📉 PHASE 6: Analysis & Bottleneck Diagnosis

Activity	**Perf. Tester**	**Perf. Architect**	**Developer / Tech Lead**	**Infra / Ops**	**QA Lead**	**Product Owner**	**Notes**
Response time vs throughput vs error trend	R	A	I	I	A	I	Check latency trends
GC log analysis (Pause time, Allocation rate)	C	A	R	I	I	I	Use GCViewer / GCEasy
Heap dump, thread dump correlation	C	A	R	I	I	I	For stuck threads, memory leaks
DB bottlenecks (AWR, Top SQL, waits)	C	A	C	R	I	I	Analyze with AWR, query plans
Infra & container metrics (CPU, memory, iops)	C	A	I	R	I	I	Use CloudWatch / Node exporter
RCA documentation & heatmaps	R	A	C	C	A	I	Include GC, CPU, memory,

							DB latency overlays

📝 PHASE 7: Tuning, Retest & Sign-Off

Activity	Perf. Tester	Perf. Architect	Developer / Tech Lead	Infra / Ops	QA Lead	Product Owner	Notes
Tune JVM params, DB, thread pools	C	A	R	C	A	I	GC tuning, thread pool configs
Fix infra configs (disk, CPU, cache tuning)	I	C	I	R	A	I	Max open files, memlock, swappiness
Test re-run and comparison	R	A	I	R	A	I	Validate fixes against baseline
Final summary report (graphs + analysis)	R	A	I	I	A	I	Executive dashboard + technical appendix
Business sign-off	I	A	I	I	A	R	Align KPIs with product goals

📜 Final Deliverables Checklist

Deliverable	Owner	Reviewed By
NFRs Document	QA Lead	Perf Architect
Workload Modelling Sheet	Perf Tester	Perf Architect
Test Strategy + Tooling Plan	Perf Architect	QA Lead
JMeter / Gatling Scripts (Git-linked)	Perf Tester	Peer Reviewer

Baseline + Comparison Test Results	Perf Tester	Perf Architect
Root Cause Analysis with Logs + Dumps	Perf Architect	Dev Lead
Performance Tuning Recommendations	Perf Architect	Dev Lead
Final Report & Executive Summary	Perf Tester	QA Lead

🔑 **Roles Explained**

- **R – Responsible:** Executes the task
- **A – Accountable:** Ultimately answerable for the correct completion
- **C – Consulted:** Has critical input based on domain or expertise
- **I – Informed:** Needs to be kept updated, but not involved directly

---

✅ **Quick Notes:**

- **Performance Tester** handles scripting, execution, monitoring, and first-level analysis.
- **Performance Architect** governs strategy, workload models, GC tuning, and RCA.
- **Dev / App Team** is crucial during fix cycles, code tuning, and JVM optimizations.
- **Ops / Infra Team** ensures resource availability, OS tuning, and server instrumentation.
- **QA Manager** ensures timely execution, coordination, and overall test governance.
- **Business Owner** is looped in during requirement gathering and final sign-off.