

ENTERPRISE PROGRAM BRIEF

AI Evaluation and Quality Engineering

Create a repeatable quality evaluation process for retrieval-augmented and tool-using AI applications. Participants define task-specific measures, build representative datasets, implement checks, calibrate human and model-assisted reviews, and produce a release recommendation.

VNode ITes

VAI-203

Coming soon

DURATION
24 hours

LEVEL
Intermediate

FORMAT
Private cohort

AUDIENCE PROFILE

Built for these roles

QA engineers, AI engineers, SDETs, data scientists, technical leads, and engineers responsible for AI application quality.

OVERVIEW

Executive overview

Create a repeatable quality evaluation process for retrieval-augmented and tool-using AI applications. Participants define task-specific measures, build representative datasets, implement checks, calibrate human and model-assisted reviews, and produce a release recommendation.

READINESS

Prerequisites

- Working Python, JSON/CSV, basic statistics and software testing, plus familiarity with LLM applications. The programme assumes a supplied working application; it does not build a full agent from scratch.

PROGRAM OUTCOMES

Capabilities your teams will gain

- Distinguish retrieval, generation, tool, system, and business measures.
- Construct development and holdout datasets with provenance.
- Implement deterministic checks and calibrated review rubrics.
- Compare candidate versions without hiding trade-offs.
- Create an auditable release recommendation.

ENTERPRISE CUSTOMIZATION

Enterprise customization

Tailor this program to your organization's priorities: Build evaluation datasets and evidence-based release decisions for AI applications

- 3 learning days of 8 hours, or 6 sessions of 4 hours. Learning hours exclude breaks and include module exercises and capstone time. Setup is completed before the course.

CURRICULUM

Curriculum roadmap

1 MODULE 1 · 2 PROPOSED HOURS Quality models and failure taxonomy

An AI system can fail at several layers even when the final text looks plausible. Define what the user task requires and classify failures before selecting a metric. Business outcomes need separate measurement from model response quality.

- Retrieval versus response versus task quality
- Deterministic and probabilistic tests
- Failure categories and acceptance criteria

Planned lab: Review ten seeded failures and map each to the component and evidence needed.

Expected output: Quality model and failure taxonomy.

2 MODULE 2 · 4 PROPOSED HOURS Dataset design and provenance

Examples should reflect the task population and important exceptions. Preserve source provenance and labels, and keep a holdout set separate from examples used for tuning. A small curated set is useful for a lab but does not establish broad reliability.

- Representative cases and expected evidence
- Labelling guidance and reviewer disagreement
- Dataset versioning and holdout separation

Planned lab: Build a small versioned dataset with normal, unsupported, restricted, ambiguous, and failure cases.

Expected output: Dataset, label guide, and coverage inventory.

3 MODULE 3 · 4 PROPOSED HOURS Deterministic checks and retrieval evaluation

Use code to check behaviours that have explicit expected outcomes. Separate candidate-source checks from the question of whether a generated claim is supported. Explain the denominator and dataset composition for every reported rate.

- Schema and exact boundary checks
- Expected source retrieval and negative cases
- Tool arguments and action-state checks

Planned lab: Implement checks for required and forbidden source IDs, valid schemas, tool arguments, and approved state changes.

Expected output: Repeatable deterministic evaluation suite.

4 MODULE 4 · 4 PROPOSED HOURS Human and model-assisted review

A review rubric needs clear anchors and examples. A model-assisted judge can help organise review, but it can also be biased or inconsistent. Compare its judgements with human labels before treating the scores as reliable evidence.

- Support, relevance, and task-completion rubrics
- Judge calibration and disagreement review
- Bias, leakage, and scorer limitations

Planned lab: Score sampled answers manually, run a model-assisted judge where authorised, and investigate disagreements.

Expected output: Calibrated rubric and disagreement analysis.

5 MODULE 5 · 3 PROPOSED HOURS Experiments and release gates

Compare changes against a fixed baseline and record prompt, model, data, and configuration versions. Repeated trials may reveal unstable behaviour. A higher average score can hide an important failing subgroup or a higher cost per useful task.

- Baseline and candidate comparison
- Repeated runs and confidence limits
- Quality cost and latency trade-offs

Planned lab: Compare two candidate configurations and document the cases that improve, regress, or remain uncertain.

Expected output: Experiment report and decision thresholds.

6

MODULE 6 · 3 PROPOSED HOURS

Automation and production feedback

Evaluation should run when relevant application assets change and should learn from operational feedback. Avoid letting new production examples contaminate a holdout set without version control. Define who investigates a failed release gate.

- CI integration and reports
- Feedback triage and new regression cases
- Sensitive-data handling in evidence

Planned lab: Add a release check and convert one observed failure into a versioned regression case.

Expected output: Pipeline specification and feedback process.

7

MODULE 7 · 4 PROPOSED HOURS

Evaluation capstone

Review a candidate AI application as a quality owner. Present evidence, unresolved risks, and a clear release decision. A failed critical boundary should remain visible even if the aggregate quality score appears acceptable.

- Independent holdout run
- Error analysis and manual sampling
- Release recommendation and evidence index

Planned lab: Evaluate a supplied support agent candidate, reject or accept it with conditions, and defend the decision.

Expected output: Evaluation pack and release memo.

CAPSTONE

Capstone

An evaluation and release evidence pack for a supplied knowledge-and-tool application, covering deterministic boundaries and generated-answer quality.

ASSESSMENT

Assessment

Dataset design 25%; implemented checks 25%; rubric and analysis 30%; release recommendation 20%. Completion requires correctly distinguishing measured results from assumptions.

DELIVERABLES

Participant deliverables

- Versioned dataset and label guide
- Evaluation harness and score definitions
- Candidate comparison and failure analysis
- Release decision with evidence index

SCOPE

Scope boundaries

Foundation-model pretraining research, distributed GPU benchmark operations, a universal model leaderboard, and claims that one metric proves safety. ****Delivery options:**** 3 learning days of 8 hours, or 6 sessions of 4 hours. Learning hours exclude breaks and include module exercises and capstone time. Setup is completed before the course.

DELIVERY

Delivery options

3 learning days of 8 hours, or 6 sessions of 4 hours. Learning hours exclude breaks and include module exercises and capstone time. Setup is completed before the course. Proposed hours exclude breaks and are subject to pilot validation.

PRIVATE DELIVERY**Plan a private cohort for your team**

Scope delivery format, role mix, and rollout timeline with our solutions team.

[Plan Private Delivery](#)[Book advisory call](#)