ModelOps "Executive Visibility for AI" RFP Requirements

This document is an example RFP for addressing ModelOps "Executive Visibility for AI" functional requirements. It is the result of interviews with several industry experts and analysts.

Category	Requirement	Description
Visualization & Dashboarding suggested weighting: 30%		
	Executive Dashboard	The ModelOps capability should provide a user friendly, unified view of all "production" models used for business decisioning with a quick view (red/green/yellow stoplight) into the adherence to business, infrastructure/operational, data, statistical, and risk KPI's for each model. The dashboard should allow for groupings based on organization/model category and drill down into each model. Finally, the dashboard should include roll-up metrics such as aggregate business value across all models, aggregate usage (e.g. inference/scoring requests), aggregate number of outstanding issues, etc.
	Executive Dashboard Customization	The Executive Dashboard should be customizable to the specific KPI's of interest across the enterprise. The ModelOps capability must be flexible to leverage different metrics, which may be obtained from a combination of out-of-the-box calculations as well as custom calculations defined by the given enterprise.
	Visualization: BI tools integrations	Ability to integrate with BI tools to create custom reports to gain visibility of aggregated and detailed views on model performance against statistical, business, and risk thresholds.
Evergreen Model Inventory: 30%		
	Evergreen Model Inventory	Provide a centralized store for viewing, managing, and maintaining all models (post-development) across the enterprise, regardless of the model type/methodology, language, framework, execution platform, or environment.
	Standard Model Definition	Provide a consistent definition (and underlying persistence mechanism) of all the core elements that compose a model, regardless of the language/framework, Data Science workbench, underlying infrastructure, or data platform used.
	Custom Metadata	Allow for extension of the core Standard Model definition with custom metadata, which may come from various integrating systems or via user input. Custom metadata may be supplied for a model as a whole or for specific snapshots of that model. The custom metadata must be available to be used in a Model Life Cycle to allow an enterprise to define and enforce governance, technical, and business requirements that are specific to their processes.
	Model Registration	Allow users to onboard their developed model into the ModelOps system by collecting the key elements that compose a model. Model registration should be enabled via CLI, import from a git repository, or via model factory plugins, such as a Jupyter or Rstudio plugin.
	Model "Snapshots" ("Versions")	Provide the ability to systematically manage snapshots of any model, including all of the model's source code, artifacts, documentation, and other metadata. This snapshot must be immutable and maintained in perpituity for long-term auditability. Each Production model must have traceability to a specific immutable snapshot for Executive and Operations visibility. The snapshot — and all of the associate metadata for that model snapshot — can be exported for reporting purposes.
	Continuous Compliance Checking - Define Rules	Ability to define — via decision tables — specific conditions in which a model must operate to be within compliance. Must be able to define the rules on a per-model basis which are persisted, maintained, and orchestrated with each and every version of a given model. These rules must be able to incorporate custom metadata for a given model, which may have been pulled in automatically via an MRM/compliance/other governance system.
	Continuous Compliance Checking - Enforcing Rules	Automated enforcement of all compliance/governance rules that have been defined on a per model basis. Breach of any particular rule can trigger alerts/notifications as dictated by the Model Life Cycle.
	Ethical Fairness/Bias Integration	Integrate with leading frameworks (Aequitas) to measure and monitor a model's fairness in providing positive outcomes for all protected and/or sensitive classes.
	Test Results Management	Persist all instances of tests/metrics that have been executed on a per-model basis, regardless of the type of model or the platform upon which it executes. A "record" of a test/metrics job should be tied to an immutable snapshot of a model for history and auditability.

Orchestration & Monitoring suggested weighting: 30%		
	Automated Workflows for Dashboard Population	Manage and automate all processes required to populate the executive dashboard, including executing all out-of-the-box and custom calculations and tasks for all models, regardless of the model type/methodology, language, execution environment, data system, or environment. This includes tying into the various integrating Data, Risk, App/Infra, Security, and Operational systems to obtain the required data and/or metrics.
	Integrations	Ability to integrate the Automation workflows with existing application/infrastructure monitoring systems (e.g. Application Performance Monitoring systems), operational systems (e.g. Production Support CR/IM ticketing, alerting systems, etc.), existing Governance/Risk systems (e.g. MRM/Compliance systems), existing data platforms (RDBMS/DW, file/object storage, streaming). The integration must be systematic (via API's/other) to allow for automatically populating the executive dashboard. In addition to the actual data/metrics, must provide the ability to systematically collect information/metadata from the integrating systems and persist as metadata with the specific version of the model, as required.
	Unencoded Thresholds Definitions	Ability to define — via decision tables or other rules — specific thresholds within which a model should operate in production to satisfy business, operational, data, security, and risk KPI's. For example, thresholds may be set for matching compliance rules back to statistical performance to provide clear but comprehensive rules for operating the model within the bounds of the compliance and business requirements. For model drift metrics, this includes comparisons to baseline data (e.g. compare a current window of production inference requests to the training data input feature distribution). Thresholds definitions do not require programming skillset to be decoded.
	Alerting and Automated Notifications	Provide ability to automatically raise alerts and notifications when specific thresholds are breached (see "unencoded threshold definition"). These alerts are handled by the automated workflows and would be tied to the "unencoded threshold" to allow for clear, but comprehensive, orchestration of when alerts are triggered. The ModelOps capability should be flexible to integrate with existing operational alerting and/or production ticketing systems to enable L1 support to provide 24x7 production support for all models across the enterprise.
	Orchestrated Issue Resolution	Allow for defining and implementing automated workflows to orchestrate the various "remediation paths" that are required to address any issue or warning that arises from the aforementioned monitors. These remediation paths may differ based on the type of model, the risk classification, type of issue, severity of issue, etc. The ModelOps orchestration capability must be able to intelligently and automatically process each issue or warning that is surfaced based on the breach in thresholds, and ensure that the issue/warning is taken to resolution by following the prescribed remediation path.
	Customization of Automated Workflows	Ability to customize the automated workflows based on the varying business, technical, and compliance/governance requirements across different business units and teams. Must provide the ability to create different automations for different groups, classes of models, or even on a per model basis.
	Business Monitors	Business value tracking monitors with integration to automated thresholds and alerting. Must allow for defining and uploading custom business value monitors for different models.
	Model Specific Monitors	Enable the registration, orchestration, and tracking of comprehensive model monitors for all Production Models and/or Production candidates, including: - Data Drift Monitoring - Volumetric Monitoring - Model Concept Drift Monitoring - Statistical Performance Monitoring
	Compliance Monitors	Enable the registration, orchestration, and tracking of comprehensive compliance-focused monitors and controls for all Production Models and/or Production candidates, including: - Characteristic Stability - Population Stability Index - Rank Order Break - Ethical Fairness Drift
	Service Health Monitor Integration	Support for integrating with existing infrastructure monitoring and application monitoring systems (e.g. APM tools), as well as model execution platform systems (e.g. SageMaker, Spark) to collect specific service health information. This includes the ability to answer the following questions for all "production" models used for business decisioning: is the model running successfully? is the latency or job duration within SLA? are there any issues returning results to the calling application? The ModelOps capability should not replace the existing data pipeline monitoring tools, but rather pull the requisite information from the existing systems to obtain a holistic view of the health/status of the production model. The ModelOps capability must be able to evaluate the metrics obtained from the integrating system against model-specific thresholds and tie into the orchestration and alerting system listed above.
	Data Pipeline Monitor Integration	Support for integrating with existing data pipeline monitoring capabilities to collect model-specific pipeline information. The ModelOps capability should not replace the existing data pipeline monitoring tools, but rather pull the requisite information from the existing systems to obtain a holistic view of the health/status of the production model. The ModelOps capability must be able to evaluate the metrics obtained from the integrating system against model-specific thresholds and tie into the orchestration and alerting system listed above.
	Custom Metrics Monitoring	Allow a data scientist or ModelOps engineer to define custom statistics to be captured about the model.

Security suggested weighting: 5%		
	Authentication/Authorization	Integration with the enterprise's oauth2 and LDAP/AD systems for authentication, leveraging existing oauth/ldap systems, processes, and AD/LDAP group structure to for authentication
	Role-Based Access Control	Role-based access control to expose functionality according to the user's role (e.g. Admin vs. Non-Admin), which should be obtained based on group membership within the organization's AD/LDAP.
	Multi-Tenancy (Group-based isol	Group-based access control to isolate which groups/teams can see specific models, model assets, runtimes with specific deployed models, test results, integrating with the enterprise's oauth2 and LDAP/AD systems to obtain the associated group entity information.
	Model Privacy	All privacy aspects for the Model including "policy that the model was built, retrained, validated with data that adhered to the corp data privacy requirements" (this can be call out to Data Governance); assurance that the production scores adhere to data privacy policy (e.g. opaque IDs in the scores); when there is a one to one relationship of model to person (e.g. individual customer behavior model) that the appropriate policies are enforced.
Non-Functional Requirements suggested weighting: 5%		
	Product Architecture	ModelOps capability should be designed to be flexible and extensible, using a modern micro services architecture, to allow enterprises to customize and integrate the ModelOps services into existing systems, platforms, and processes.
	Infrastructure	Ability to run Software anywhere: on Cloud, on Prem, Hybrid.
	Systematic Access (API's)	Provide a comprehensive API to allow enterprises to create custom Model Life Cycles or custom integrations into consuming business applications. Expose core ModelOps services as RESTful API's allowing enterprises to easily customize the use of the ModelOps solution to their specific business requirements.
	Ownership	Allow for definition of ownership for model operationalization, by providing a console for 24x7 model management in production, and with visibility into the model path to/in production with pre-defined ownership of responsibilities, triage and remediation processes.
	Localization	Support for the linguistic, technical, and overall cultural requirements for all regions of a global enterprise.